**Doctor of Philosophy**

# HLA-Compliant Interactive Disaster Simulation with Application to Multi-level Restoration Management

**February 2015**

Department of Architecture & Architectural Engineering
The Graduate School
Seoul National University

Sungjoo Hwang

# Abstract

## HLA-Compliant Interactive Disaster Simulation with Application to Multi-level Restoration Management



Sungjoo Hwang

Department of Architecture & Architectural Engineering

The Graduate School

Seoul National University



Although disaster-related simulations can be effectively applied in disaster response and recovery management in a chaotic situation, each single simulation for analyzing disaster intensities, damage patterns, and response and recovery efforts needs to be able to interact with others. In other words, different combinations of simulations need to be made in different situations, due to numerous types of disasters and their complex effects on various damage patterns as well as diverse response and recovery efforts. In the recovery stages, it is particularly essential that a disaster-related simulation is able to satisfy lots of analysis requirements for complex and multiple recovery efforts, including many types of recovery activities for numerous facilities,

performed by diverse agents at both macro and micro levels of management (e.g., the regional level and the individual facility level).

Therefore, this study develops an interactive simulation framework for disaster recovery management which has extendibility with regard to numerous disasters, various damage situations, and multiple recovery efforts. For the purpose of promoting extended applications of the recovery simulation in the future, an interactive simulation is designed through the use of the High Level Architecture (HLA) (IEEE 1516)—developed by the U.S. Department of Defense (DoD) to provide seamless interactions among simulations (i.e., federates) by its general rules for distributed simulation environments. To present the interoperability, reusability, and extendibility of the framework, this study further develops a prototype of the interactive simulation with a focus on a structural damage and recovery situation in the aftermath of a seismic event, in order to assist both regional- and facility-level (i.e., multi-level) recovery planning. A System Dynamics (SD)–Discrete Event Simulation (DES) integrated multi-level damage and recovery simulation federate interacts with both an USGS near real-time seismic data retrieval technique (i.e., USGS federate) and an OpenSees structural response simulator (i.e., OpenSees federate), by dynamic data exchanges among three federates in the developed prototype where different simulations can be synchronized. By the integrated uses with an USGS federate, in detail, an SD model provides a comprehensive understanding of multiple interdependent

recovery efforts at the regional level according to widespread devastation. On the other hand, a DES model enables a detailed examination of restoration operations as well as a project's performance and uncertainty at the facility level according to facility's structural damage and its surroundings through the integrated uses with both USGS and OpenSees federates.

Such different combinations of simulations in the interactive simulation can provide appropriate information to respectively assist multi-level recovery planning performed by different agents. With several future utilization scenarios into overall disaster management areas in various damage situations, this study further examines the reusability and extendibility of the developed framework through the interactions with other disaster-related simulations (e.g., fire simulator). The base technology and generic model structures of an interoperable HLA-compliant distributed simulation for disaster recovery management can potentially provide the simulation with further extendibility to numerous types of disasters (e.g., hurricanes), to various damage situations including non-structural damage (e.g., power blackout), and to the all-time disaster management (e.g., evacuation planning) when diverse disaster-related simulations are interacted.

**Keywords: Disaster Management; Disaster Responses; Distributed Simulation; Facility Management; High Level Architecture**

**Student Number: 2010-30174**

# Contents

# List of Tables

# List of Figures

# Chapter 1. Introduction

## 1.1 Research Background

Catastrophic disasters such as floods, hurricanes, and earthquakes can cause fatal damage on the core infrastructures and facilities, which can generate unusually large and immediate functional losses of the overall built environment (Pachakis and Kiremidjian 2004; Olshansky et al. 2012). The damage severity caused by major disasters in an urban area has been increasingly getting worse in recent days, due to a high population density and an increase of large-scale buildings or hazardous facilities. Since economic and social activities in urban areas depend on not only residential and commercial services but also various products and public services (Shoji and Toyota 2009), rapid and appropriate recovery planning is important to alleviate damage and to recover both the inconvenient and impoverished daily lives of populations and the interrupted operations of facilities/infrastructures to their pre-disaster conditions.

However, in an uncertain and complex disaster situation, insufficient information on damage patterns and recovery processes causes difficulties in implementing immediate and efficient recovery plans (Orabi et al. 2010; Olshansky et al. 2012). Computer simulation techniques have thus been widely applied in disaster management areas because of its ability to handle greater complexity and uncertainty (Harrison et al. 2007). Focusing on this capability, the computer simulation is particularly effective in the planning of complex and multiple recovery processes because the disaster recovery stages include many types of activities (e.g., loss estimation, debris disposal, and restoration) for numerous facilities and infrastructures (e.g., houses, plants, and transportation systems) performed by diverse agents (e.g., governments, local communities, and private sectors) at both macro and micro levels of management (e.g., the regional level and the individual facility level) (Hu et al. 2009; Pachakis and Kiremidjian 2004; Olshansky et al. 2012).

## 1.2 Problem Description

To assist disaster management, there exist numerous available technologies, tools, and systems for simulating and analyzing disaster intensities, damage patterns, and recovery efforts. Although each disaster-related simulation can be helpful for making effective recovery plans in a chaotic situation, there exist following limitations on developing the integrated and comprehensive disaster simulation for recovery management: (a) a disaster recovery simulation needs to consider various damage patterns according to different types of disasters; (b) the complex effects of disasters such as both structural and non-structural damage need to be considered; (c) a disaster recovery simulation may also require an analysis of diverse and complex recovery efforts performed by different agents at different levels; (d) numerous information of different facilities and regions may be required for recovery management; and (e) the needs for the analysis of a disaster and recovery situation change over time according to a disaster recovery management cycle (Pachakis and Kiremidjian 2004; Currion et al. 2007; Hu et al. 2009; Yotsukura and Takahashi 2009; Olshansky et al. 2012).

Therefore, each single simulation for analyzing disasters, damage, and recovery efforts needs to be able to interact with others because different combinations of simulations need to be made in different situations. Although many integrated disaster simulations in a single platform (i.e., centralized simulation) have been developed to overcome these problems, the centralized simulation, which may include all kinds of sub-simulations, have problems on not only performing rapid simulations due to its heavy size but also involving new types of disaster and diverse recovery situations (Dimakis et al. 2010). In recent times, people's requirements with regard to disaster-related simulations and technologies have changed, and different systems have been developed for different purposes (Yotsukura and Takahashi 2009).

In contrast, a distributed disaster simulation approach alleviates some of these problems via a new distributed simulation platform where different simulations interact with each other. It can make use of diverse systems that are used or have been developed for their own purposes at the same time (Yotsukura and Takahashi 2009). Therefore, distributed simulation can be effectively utilized to multiple levels and types of disaster recovery management in different situations by comprehensively and variously analyzing disaster intensities, damage patterns, and recovery efforts, through the incorporation of diverse disaster-related simulations (e.g., disaster data retrieval techniques, disaster simulations, and damage estimation techniques) into the multiple recovery simulations, especially when analytical requirements vary according to the purposes of analysis.

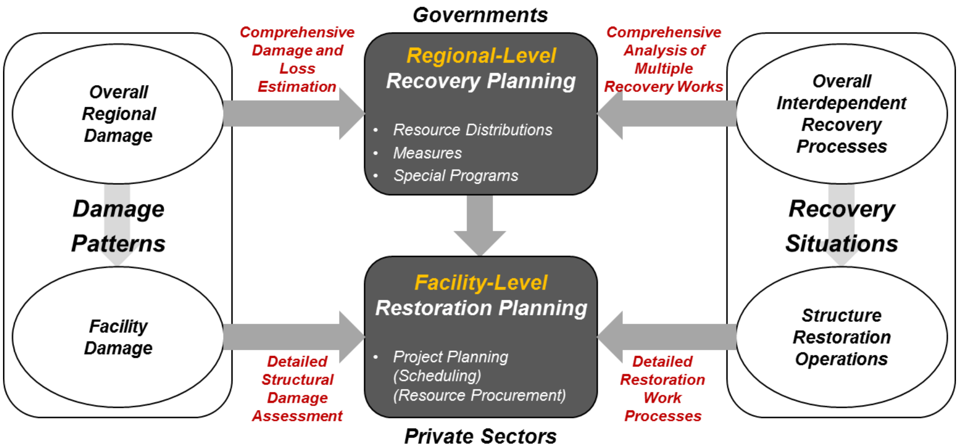



**Figure 1.1** Different Analytical Requirements in Multiple Levels of Disaster Recovery Management

As shown in Fig. 1.1, in particular, requirements for analyzing damage situations and recovery processes can be considerably different according to both regional- and facility-level recovery efforts for damaged facilities/infrastructures, [1] For example, regional-level recovery planning requires comprehensive understandings of overall damage patterns and whole recovery processes among diverse types of facilities/infrastructures with a

[1] In this study, both regional and facility levels of recovery efforts performed by different agents with different recovery management scopes are defined as “multi-level” recovery efforts.

consideration of recovery resources and the societal needs for a facility's functions. This understanding is helpful for implementing comprehensive recovery plans including resource (budget) distribution plans, measure, and special recovery programs (Shoji and Toyota 2009; Orabi et al. 2010; Holguin-Veras and Jaller 2012).[2] In this situation, disaster data retrieval techniques can be effectively used to estimate regional damage patterns by the rapid reflection of disaster intensities. Moreover, post-disaster recovery efforts at the regional level may vary with damage and recovery situations among sub-regions, and may significantly change over time (i.e., they are dynamic), according to different damage intensities, multiple recovery works, and ensuing recoveries of facilities' diverse functions (i.e., multiple feedback processes) (Orabi et al. 2010; Olshansky et al. 2012). Since the System Dynamics (SD) has a powerful ability in capturing multiple feedback processes and dynamic state changes in a complex system as a whole with a deeper understanding of policy effects (Sterman 2000; Williams 2002), SD can be effective in the comprehensive and strategic investigations of dynamic features in overall recovery systems that are significantly affected by the governmental recovery plans.

[2] Since the recovery of damaged built environments at the whole devastated region requires great amounts of public expenditures, both the central and local governments have generally main responsibilities for overall recovery planning (Karlaftis et al. 2007). This evidence can be found in actual disaster recovery frameworks in the U.S. and Japan after the catastrophic disasters such as the Hurricanes Katrina and Rita and the 2011 Earthquake of Tohoku (Pike 2007; FEMA 2011; RA 2011; RA 2013).

On the other hand, facility-level recovery planning requires detailed information of structural damage, as well as unfavorable post-disaster project conditions affected by surrounding damage and recovery situations (Pinelli et al. 2004; Ma et al. 2005; Orabi et al. 2010; Olshansky et al. 2012).[3] Detailed work processes also need to be captured in order to make valid facility restoration project plans. The integrated use of structural damage simulation techniques can thus be useful for analyzing detailed damage of facilities. In addition, to support rapid and reliable construction (including repair/reconstruction) process analysis, the Discrete Event Simulation (DES) can be successfully applied to construction operation analysis due to its powerful ability to handle the complex and uncertain nature of construction processes (Law and Kelton 2006). Further, for the purpose of implementing more reliable recovery plans in an uncertain and complex post-disaster situation, the SD can support the DES in understanding unfavorable project conditions over time caused by facility's surrounding damage and recovery situations, with a SD-DES interacted simulation.

According to different purposes of analysis for both regional- and facility-level recovery efforts respectively, as mentioned above, different combinations of simulations in a distributed simulation environment thus need to be made to meet each requirements of analysis in different levels of

[3] In general, restoration planning of the most of facilities except the major facilities such as nuclear power plants may be implemented by project managers in the municipalities, local communities, and private sectors under the governmental recovery plans and supports (Berke et al. 2014).

recovery efforts. However the lack of modeling framework prevents the interoperability, reusability, and future extendibility of simulations when each model in the interactive simulation environment is separately developed, and when interactions among models are determined. Therefore, the interactive simulation framework for disaster recovery management has to be able to clarify communication architectures among diverse disaster-related simulations by determining analytical requirements for diverse disaster recovery management.

## 1.3 Research Objectives and Scope

For the purpose of more accurately modeling and comprehensively analyzing complex and uncertain disaster recovery situations, this study develops an interactive disaster simulation framework for multi-level recovery management which has extendibility with regard to numerous disasters, different damage situations, and multiple recovery efforts in diverse facilities and regions. To promote reusability and extendibility of distributed recovery simulation in the future, the interactive recovery simulation in this study is developed based on the principles defined in the High Level Architecture (HLA) (IEEE 1516)—that is capable of synchronizing different simulation models or incoming data streams by its general rules for distributed simulation environments (Schulze et al. 1999; AbouRizk 2010; Menassa et al. 2014).

Based on the developed framework, this study further develops a prototype of the interactive recovery simulation with a focus on both regional and facility levels of damage situations in the aftermath of a seismic event, for the purpose of identifying the interoperability, reusability, and extendibility of the interactive simulation. In the prototype, instant seismic data retrieval techniques, structural damage simulation techniques, and SD and DES simulation models for multi-level recovery planning are differently interacted according to each purpose of analysis, based on the effectiveness of the interactive simulation in satisfying different analytical purposes with different combinations of simulations. Then, case simulations are conducted to understand both regional- and facility-level recovery situations over time as well as offer insights with regard to both regional recovery management and facility restoration project management. By providing experimental results and policy implications, the research outcome is expected to respectively support recovery managers of different agents (e.g., governments or private sectors) in implementing immediate and valid multi-level (i.e., both regional and facility levels, respectively) recovery plans in a chaotic situation.

According to the research scope, in detail, the prototype in this study pays attention to the recovery stages among an overall disaster management cycle.[4] Among numerous types of disasters, this study especially focuses on the post-earthquake situation after a series of catastrophic seismic events

---

[4] Overall disaster management cycle includes the disaster identification, prediction, preparedness and mitigation, emergency response and emergency restoration, and recovery (Pradhan et al. 2007).

because more increasing uncertainties of recovery efforts due to a widespread devastated area make our comprehensions more challenging. Furthermore, the urban area for regional-level recovery analysis and the most common ten- or fewer-story damaged building for facility-level recovery analysis are the main interests for the purpose of a wider range of future uses for this study's outcomes.

Although both structural and non-structural damage (e.g., power blackout and operating system shut-down) affect facilities and infrastructures after a disaster, non-structural damage caused by the interruptions of surrounding lifeline systems may generally be treated at an early recovery phase within several weeks (Shoji and Toyota 2009). Therefore, this study mainly focuses on restorations from structural damage which requires lengthy durations of recovery works while non-structural damage is beyond the scope of simulation. Although this study set bounds to the scope of the interactive recovery simulation as described above, the developed framework can be further applied beyond the research scope. For the purpose of presenting the reusability and extendibility of the research outcome, therefore, this study also provides detailed example scenarios for the extendable uses of the developed interactive disaster simulation in the future, with regard to numerous types of disasters and different damage situations as well as diverse response and recovery planning in an overall disaster management cycle.

## 1.4 Dissertation Outline

To overcome difficulties in immediate and valid recovery planning in a complex post-disaster recovery situation, this study aims to provide a comprehensive understanding of the regional damage and overall recovery processes as well as a detailed analysis of the facility damage and consequent repair/reconstruction operations, respectively, through the use of different combinations of simulations in the interactive simulation. Firstly, a regional-level recovery simulation utilizes an SD model through the interaction with a regional damage estimation model while a facility-level repair/reconstruction operation analysis makes use of a DES model through the interaction with a facility damage assessment model. In addition, the SD-DES interaction enables a comprehensive analysis of interdependencies among recovery efforts of regions and facilities. To incorporate immediate damage assessment into recovery management for more rapid and reliable planning, the existing incoming data stream (e.g., seismic data retrieval technique) and simulation software (e.g., structural response simulation) need to be utilized and interacted with recovery simulation models in the distributed simulation environment due to the differences in technical implementations and platforms among simulations (AbouRizk 2010). To solve these problems, in particular, this study applies HLA—that provides interoperability and reusability among each simulation in distributed simulation environments (Zhang et al. 2011)—with expectations of both wide-range of uses and extended applications of the developed prototype.

Fig. 1.2 describes an overall research process of the interactive simulations for post-disaster multiple recovery management. The dissertation begins with the introduction in Chapter 1, by briefly describing problems in analyzing different recovery processes for both regional- and facility-level recovery planning in a chaotic situation. Then, this study explains the need for the interactive simulation and consequent research objectives.

In Chapter 2, after introducing previous research efforts and major issues on recovery management as well as disaster-related simulations, this study provides investigations of the opportunities and hurdles of simulation approaches, and then explains why an interactive simulation is required for multi-level recovery planning. Firstly, the needs for SD and DES are described to comprehensively analyze both regional-level recovery processes and facility-level restoration operations. Secondly, the need for a distributed simulation is provided to overcome technical hurdles for incorporating prompt damage assessment of both regions and facilities into multiple recovery management. Then, the concept of the HLA is briefly introduced.

In Chapter 3, this study provides the interactive simulation framework based on the investigation of previous research efforts, useful modeling methodologies, and relevant techniques. By examining input and output information and purposes of each simulation model and technique, the framework is developed for the purpose of preventing the loss of the modeling scope and purpose when each model in the interactive recovery simulation is separately developed and when interactions among models are determined. Firstly, this study clarifies the framework for the SD and DES models for multi-level recovery simulations in a single simulation platform. Secondly, the distributed simulation framework that integrates a disaster data retrieval module, a damage assessment module, and a SD-DES recovery simulation module in a distributed simulation platform are further developed by conducting an in-depth analysis of useful existing techniques

In Chapter 4, model components for interactive recovery simulation are developed based on the framework presented in Ch.3. Model components for multi-level recovery planning include: (a) a regional-level damage and recovery simulation component that consists of both a regional damage estimation model and an overall recovery process analysis model using SD according to regional damage; (b) a facility-level damage and restoration simulation component that consists of both an immediate structural damage assessment model and a facility restoration operation analysis model using DES according to the facility's structural damage and facility's surroundings. In this section, model descriptions and test results using case examples are provided. Data exchanges among simulation model components in the recovery simulation are also described.

In Chapter 5, interactions among simulations, including the seismic data retrieval technique, the structural response simulation, and the recovery simulation in the HLA-compliant distributed simulation environment, are provided. In addition, this study offers detailed descriptions of the developed prototype of distributed recovery simulation.

In Chapter 6, the developed prototype of interactive recovery simulations is tested. Then, this study conducts case simulations both for a comprehensive understanding of regional recovery efforts and for a detailed analysis of facility restoration works to inspect the usefulness of interactive simulations, with regard to interoperability, reusability, and extendibility according to each purpose of analysis for multi-level recovery planning. This study also conducts both government- and project-level recovery policy experiments to assist with different recovery manager's decision-making in the recovery planning phase with both regional and facility levels. Based on these analyses and experiments, discussions and policy implications on regional recovery management and facility restoration management are respectively provided.

In Chapter 7, based on the extendibility of the developed prototype, the expected future uses to diverse damage and response situations are presented with several examples of detailed scenarios (e.g., facility evacuation planning).

Finally, in Chapter 8, the research results and this study's contributions to the body of knowledge in the field of construction and disaster management are described. Then, this study finally provides the limitations of this study and required future works for the purpose of enabling the research outcomes to be applied to the real world situations in the future.

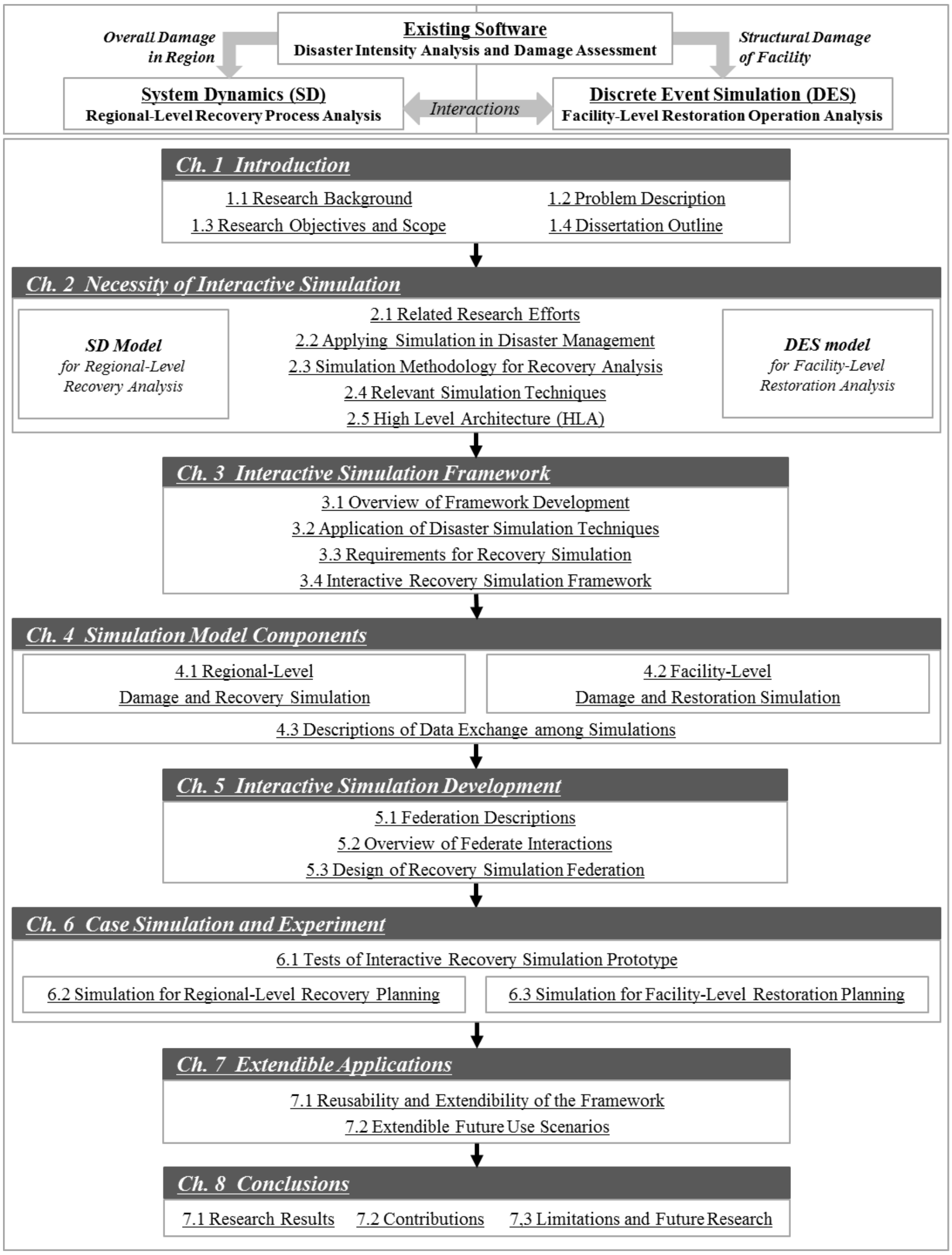



**Figure 1.2** Dissertation Outline

# Chapter 2. Necessity of Interactive Simulation

In this chapter, this study explains why the interactive simulation can be helpful for complex and multiple recovery management at both regional and facility levels in an uncertain post-disaster situation. This study first introduces research efforts and issues on recovery management. Then, the opportunities and hurdles of the simulation approach are provided with regard to relevant simulation methodologies (e.g., SD and DES) and available disaster simulation techniques and tools. With a particular focus on the SD-DES simulation from a methodological perspective, this study further explains the need for the interactive simulation for recovery management of both regional and facility levels, in order to not only comprehensively analyze overall damage patterns and ensuing recovery processes and plans in regions but also analyze facility's structural damage and consequent restoration operations in detail. Finally, to incorporate prompt damage assessments into multiple recovery management, the need for a distributed simulation is provided from a technical perspective with the description of the concept of the HLA for the distributed recovery simulation.

## 2.1 Related Research Efforts

Due to the severe damage to the facilities and infrastructures after a catastrophic disaster, it is essential to implement immediate and appropriate restoration plans for recovering the functions of facilities/infrastructures within a limited period of time. In this regard, considerable research efforts have been conducted on the disaster management cycle following a disastrous event, including damage assessments, emergency responses and recoveries. This body of research includes damage predictions and assessments (Ghobarah 1999; Pinelli et al. 2004; Ma et al. 2005; Hartmann et al. 2008; Menches et al. 2008; Anil et al. 2013), disaster-related loss estimations (Shiraki et al. 2007; Wald et al. 2011; Jaiswal and Wald 2013), emergency responses including damage mitigation and temporary planning (Kovel 2000; Pachakis and Kiremidjian 2004; El-Anwar et al. 2009; Peña-Mora et al. 2012), and recovery planning (Shoji and Toyota 2009; Olshansky et al. 2012).

Since the restorations of damaged facilities and civil infrastructure systems particularly need careful plans in order to alleviate the impact of disasters on local communities (Karlaftis et al. 2007), there have been diverse research efforts to analyze both the restoration operations and the functional recovery processes of facilities/infrastructures at both a regional level and a project level. These efforts include resource supply chain management (Masurier et al. 2006; Orabi et al. 2010), debris disposal management (Swan 2000; Shen et al. 2004), and recoveries of transport networks (Chen and

Tzeng 1999; Lambert and Patterson 2002; Orabi et al. 2009).

Despite their contributions toward improving recovery processes, recovery management should comprehensively consider complex and multiple recovery efforts performed by diverse agents (e.g., governments, local communities, and private sectors) at different levels (e.g., regional and facility levels). Therefore, it is necessary for recovery simulations to satisfy diverse analytical requirements among different recovery efforts. For example, a key to implementing better recovery plans at the whole damaged regions is to understand how recovery systems change over time from a holistic perspective, according to overall damage patterns in regions and the interdependencies among facilities as well as policy effects (Shoji and Toyota 2009; Holguin-Veras and Jaller 2012). On the other hand, the crucial requirement for implementing facility restoration plans within limited time and resources may include an analysis of how and to what extent the uncertain facility damage and external restoration conditions affect a facility restoration, from a detailed perspective (Ma et al. 2005; Holguin-Veras and Jaller 2012).

In this situation, prolonged damage assessments of facilities and infrastructures due to complicated damage judgment procedures may impede recovery planning (Pinelli et al. 2004; Anil et al. 2013). Therefore, the accurate and immediate assessment of damage in both regions and facilities is essential for approximately determining the amount and the type of required repair/reconstruction works prior to implementing recovery plans at both regional and facility levels (Stephens and Yao 1987; Ma et al. 2005).[5] However, when a wide range of area is excessively damaged, a lack of skilled engineers who conduct damage assessment causes more lengthy duration of damage judgments. If these assessments are performed by non-expert damage assessors who have a limited knowledge of earthquakes and facilities, inaccurate damage estimations can be made especially when a damage severity is difficult to be determined with an observation (e.g., observation of crack and deformation) (Menches et al. 2008; Anil et al. 2013), which causes invalid recovery planning at both regional and facility levels.

These understandings of damage situations and recovery processes are especially helpful after mega-disasters, as they help to cope with increasing uncertainty and complexity of the recovery situations when decisions need to be made before sufficient information and data for recovery planning are provided. In this context, the computer simulation can be effectively applied to solve these problems. This application is because a computer simulation is generally effective to manage complex problems, especially when problems are characterized by uncertainty, when problems are technically or methodically complex, and when an integrated solution and analysis is required (AbouRizk 2010).

---

[5] The amount and the type of required repair/reconstruction works according to different damage patterns can be defined as a "restoration work scope". Therefore, critical decisions on various degrees of damage must be made within a short period of time concerning which of these buildings should be demolished and restored using available strong motion earthquake records and structural analyses performed using available data (Stephens and Yao 1987).

## 2.2 Applying Simulation in Disaster Management

Due to the ability to articulate the complex behavior of interest (e.g., construction processes) over time by formalizing theories, processes, or factors as a set of computational rules as described previously (Harrison et al. 2007), computer simulation techniques have widely been applied to disaster management in practice. The areas of disaster management using computer simulations include the disaster predictions and disaster intensity analyses (e.g., earthquakes, hurricanes, and tsunami), damage estimations and analyses (e.g., fire, structural damage, and non-structural damage), and disaster responses (e.g., evacuation, rescue, and recovery efforts).

Table 2.1 shows an investigation of existing disaster-related simulation tools and research efforts including disaster predictions and disaster data retrieval techniques, damage simulations, and disaster response simulations. According to the scope of this study described in the previous chapter, Ch. 1.3, this investigation focuses on the earthquake, structural damage, and recovery simulation software. In detail, they include: the *USGS* Real-time data feeds and *Earthquake-Report* for seismic events (USGS 2014), the *Tunami N2* for earthquake simulation (Imamura 2006); *SAHANA* for earthquake simulation (Currion et al. 2007); *HAZUS-MH* for natural disaster simulation (Vickery et al. 2006); *OpenSees* for structural engineering and structural behavior analysis (McKenna et al. 2000); *LS-DYNA* for structural change simulation (HallQuist 2000); *ClientRunner* for assisting facility restoration management

(ClientRunner 2014); and *MEETSIM* for simulating transportation system restorations and operations (Song et al. 2010).

**Table 2.1** Existing Disaster-Related Simulators

| Categories | Techniques | Main Objectives | Features |
|---|---|---|---|
| Disaster Data Retrieval | *USGS* | Real-time data feed for seismic events | Query-based data feed using API |
| | *Earthquake-Report* | Real-time data feed for seismic events | - |
| Disaster Simulation | *Tunami N2* | Tsunami and earthquake simulation | Closed source code, Scenario database |
| | *SAHANA* | Tsunami and earthquake simulation | Open source code (Javascript) |
| | *HAZUS-MH* | Natural disaster simulation (hurricanes and earthquakes) | Geographic information system (GIS) |
| Damage Simulation | *OpenSees* | Earthquake engineering (structural behavior analysis) | Fully-open source code (C++ and Tcl), structure library, visualization |
| | *LS-DYNA* | Structural change simulation (seismic collapse) | Commercial, 3D module, material and element library, |
| Recovery Simulation | *ClientRunner* | Facility restoration management | Web-based, commercial |
| | *MEETSIM* | Transportation system simulation and resource allocation planning | Closed source code, 3D visualization |

To enhance the applicability of simulation techniques to the real-word disaster situation, diverse research efforts have focused on improving the usability and accuracy of simulations for estimating disaster intensity and damage as well as analyzing disaster responses, through the development of the technical platforms or a bunch of software. These efforts include the development of data input and output modules, visualization modules, web-based systems, and so forth. Some of those simulations have thus been widely used for disaster prediction, damage estimation, and disaster response planning in practice. Although these simulation approaches can be effective to assist diverse disaster management, a deeper understanding of complex response and recovery processes is essential for assisting more reliable response and recovery planning, by providing key insights into the overall response and recovery systems. Therefore, diverse simulation methodologies have been used to provide an in-depth analysis of an overall disaster response and recovery management cycle such as evacuation simulations (Dimakis et al. 2010; Chu et al. 2012), emergency restorations and recovery simulations (Pachakis and Kiremidjian 2004; Janssen and Ostrom 2006; Sanford Bernhardt and McNeil 2008), and causal analyses of man-made disasters (Cooke and Rohleder 2006), with a consideration of interrelationship among response and recovery work processes and among agents performing these works. The methodologies used in these previous studies include the DES for analyzing detailed sequential processes, the SD for capturing feedback processes to comprehensively analyze overall systems and policy effects, and the Agent-Based Model (ABM) for capturing emergent behaviors as a result of interactions among the agents and interdependencies among system networks. According to these evidences, these simulation methodologies—that have different goals, scopes, and detail levels of analyses—can be respectively utilized to satisfy different functional needs for multiple recovery management with their abilities to provide both comprehensive and detailed understandings of the multiple recovery systems, respectively.

On the other hand, numerous available technologies, tools, and a bunch of simulation software can strengthen reliabilities of the recovery analysis by rapid damage estimations in regions and facilities during the early recovery planning phase. Based on these backgrounds, the integration of both simulation methodologies and techniques can not only enable a more reliable and rapid estimations of damage patterns but also provide a comprehensive understanding of recovery systems and an insight into recovery planning.

## 2.3 Simulation Methodology for Recovery Analysis

### 2.3.1 Use of System Dynamics (SD)

Post-disaster recovery systems at a regional level include repairing and rebuilding the damaged built environments–such as buildings, core facilities and infrastructures—in an overall region, disposing excessive debris, and so forth (Olshansky et al. 2012). Due to the need for great amounts of public expenditures for rapid recoveries of devastated regions in the aftermath of a disaster, governmental recovery plans need to be carefully designed with an understanding of overall recovery processes.[6] However, there exist difficulties in comprehensively understanding overall recovery processes among diverse types of facilities/infrastructures and in implementing appropriate plans in post-disaster recovery situations for the following reasons: (a) the availability of resources (including materials, equipment, and workforces) is generally expected to be limited due to excessive damage (Pachakis and Kiremidjian 2004; Orabi et al. 2009); and (b) there exist not only differences in the extent of the damage but also the relative importance and associated interdependency among numerous facilities/infrastructures' functions (Shoji and Toyota 2009).

In the situation that restoration operations and functionality recoveries affect each other (i.e., feedback process), the degree of the diverse functional recoveries of the built environments can vary and change over time (i.e., they are dynamic) because limited restoration resources may be unevenly distributed and periodically adjusted to multiple recovery efforts according to governmental plans, the functionality loss levels, and the interdependencies (Shoji and Toyota 2009; Orabi et al. 2010).

In detail, Fig. 2.1 shows multiple feedback processes between diverse restoration operations and their numerous functionality recoveries of the

[6] Governmental recovery plans include recovery budget distributions, special measures (e.g., relaxed regulation and procedures) and special programs (e.g., temporary debris clearance, temporary housing, and land use policies). This evidence is derived from the investigation of the past disaster recovery situations.

overall built environment in the region. Since there exist numerous types of facilities and infrastructures that have different functions in the devastated region, multiple feedback processes for numerous facility/infrastructure restorations can be found (Olshansky et al. 2012).[7] Therefore, the comprehensive analysis of multiple complex recovery efforts at the regional level is required for better recovery planning.

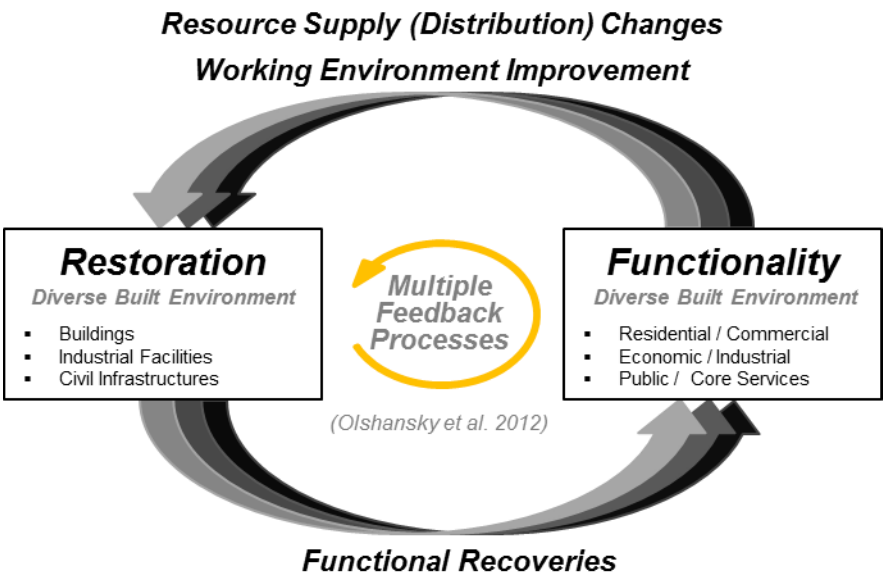


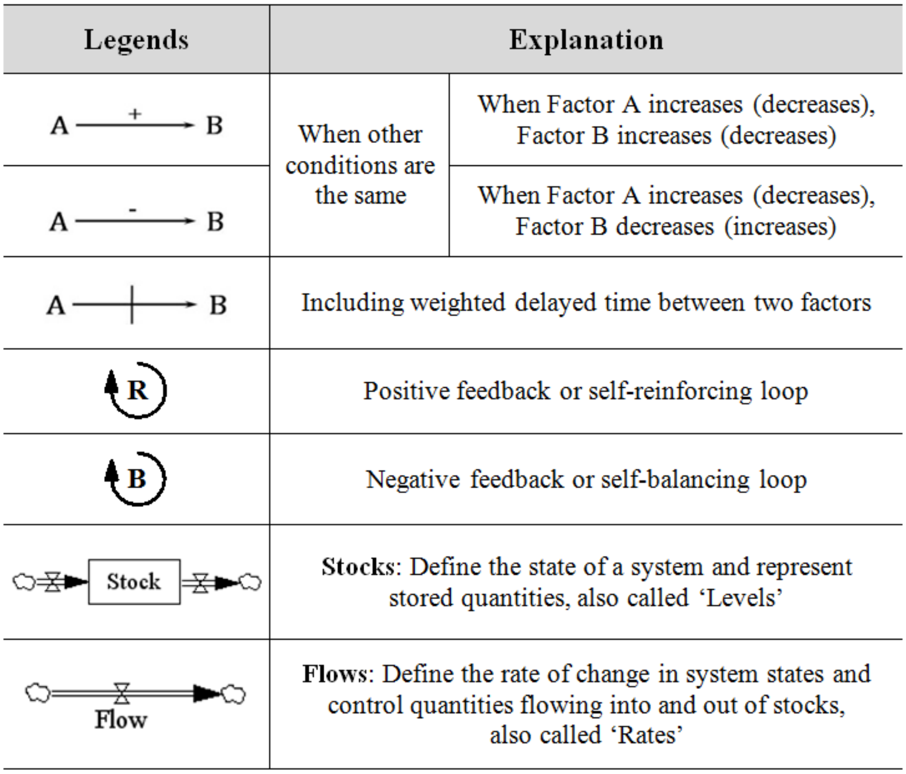

**Figure 2.1** Feedback Process of the Regional Recovery System

However, there exist multiple difficulties in addressing this issue. First, there exists a lack of a comprehensive understanding of the complex and

[7] The built environment in the region include numerous types of facilities and infrastructures such as houses, commercial buildings, public buildings, industrial facilities, roads, and other civil infrastructures.

multiple interdependent restoration processes at the macro level. Second, there may be an insufficient understanding of dynamic changes in the overall recovery system. Third and finally, there also exists a lack of a strategic point of view that can be helpful in capturing the effect of diverse recovery plans/policies (Djanatliev et al. 2012). In this context, SD can provide an analytic solution for complex and dynamic systems by capturing interactions among variables and by understanding their structures (Sterman 2000; Williams 2002). SD modeling with a holistic view can thus be effectively applied to analyze dynamic features of overall recovery efforts with a consideration of governmental recovery plans.

Fig. 2.2 shows notations used in the SD models. SD is based on the theory-based cause-and-effect relationship among variables and the stock-and-flow diagram, which can be applied to model the behavior of a system as a whole. The SD model known as a causal loop diagram consists of variables and their feedback loops. The most important model component, the feedback loop, has two categories: (a) a balancing loop, which is a goal-seeking structure that produces balance and stability in a system; and (b) a reinforcing loop, which generates the growth process through which action generates greater action (Ahmad and Simonovic 2000). In addition, the SD model represents both the current states and changes of variables in the system, by using stock and flow diagrams. While a stock defines the state of a system and stored quantities, a flow defines the rate of change in the system states (Sterman 2000).



| Legends | Explanation | |
|---|---|---|
| A —+→ B | When other conditions are the same | When Factor A increases (decreases), Factor B increases (decreases) |
| A —-→ B | | When Factor A increases (decreases), Factor B decreases (increases) |
| A —┼→ B | Including weighted delayed time between two factors | |
| R | Positive feedback or self-reinforcing loop | |
| B | Negative feedback or self-balancing loop | |
| Stock | **Stocks**: Define the state of a system and represent stored quantities, also called 'Levels' | |
| Flow | **Flows**: Define the rate of change in system states and control quantities flowing into and out of stocks, also called 'Rates' | |

**Figure 2.2** Notations in the Causal Loop Diagrams of SD

(Sterman 2000; Hwang et al. 2013)

SD includes not only the physical aspects but the policies with a strategic point of view, which dominate decision making in overall systems (Abdel-Hamid 1993). For this reason, SD is helpful for dealing with the dynamic and complex problems typical in the field of project management, where many exogenous factors have different effects—both dynamic and compositive—on the endogenous factors at different times and with various participants (Hwang et al. 2013). SD has been previously and successfully applied to solve complex project management problems in the construction industry through the comprehensive analysis of project-related behaviors that affect construction processes such as organizational issues, resource supplies, and managerial policies (Ford and Sterman 1998; Peña-Mora and Li 2001; Lee et al. 2005; Park 2005; Taylor and Ford 2008). SD has also been incorporated into regional-level infrastructure management such as water supply systems (Ahmad and Simonovic 2000), because SD provides a comprehensive understanding and a systematic approach for a whole complex system. Focusing on this analytical capability, SD modeling at the macro level (at a regional level in this study) can thus have strengths at analyzing multiple feedback processes and dynamic changes in the regional recovery systems.

### 2.3.2 Use of Discrete Event Simulation (DES)

Since many uncertain and unpredictable variables dynamically affect facility construction projects (including repair and reconstruction projects), the importance of efficient construction planning has been highlighted in the civil and construction research areas. In general, previous research efforts on construction processes and operations consider resource logistics and schedule performance as a main interest (Peña-Mora et al. 2008). The key issue was how to improve construction processes to reduce construction costs and durations within assigned resources, by effectively handling the complexity and uncertainty of construction operations. This body of research thus includes resource allocation optimizations and advanced scheduling with a detailed level of view (Hegazy 1999; El-Rayes and Moselhi 2001; Ibbs and Nguyen 2007). In particular, DES is regarded as an effective tool for construction process analysis due to its advantages in describing operational details, with a consideration of stochastic durations and resource inputs to reproduce system events (Law and Kelton 2006; AbouRizk et al. 2011). This event-based modeling technique can thus be helpful when dealing with the detailed restoration processes of an individual facility for project planning (e.g., scheduling), where available resources are strictly limited and where the sequence of various activities is highly intricate, and when a restoration work scope and process is different according to diverse facility damage patterns.

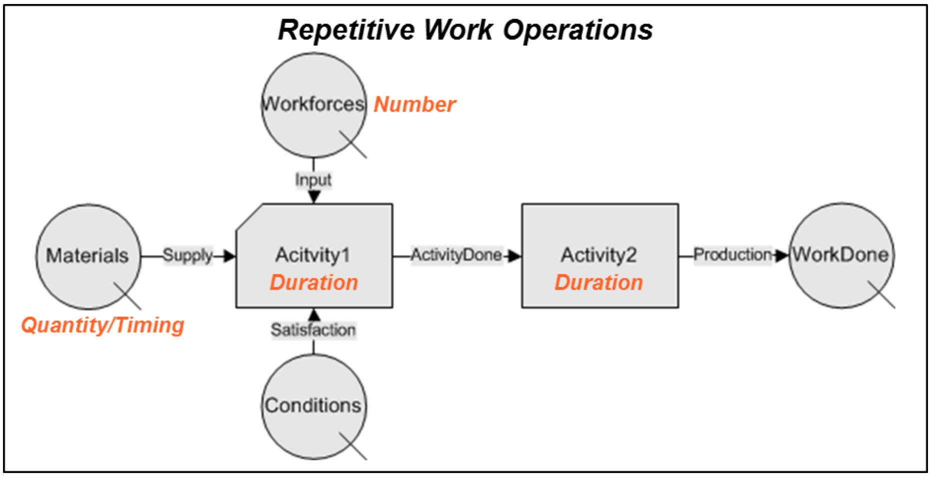



**Figure 2.3** Core Structure of the DES Model

Fig. 2.3 shows the core structure of the DES model. The main element of the DES model is the “activity” that represents tasks. This activity can start whenever available resources (e.g., materials and workforces) for the activities are enough and whenever specific conditions for the activity initiation are satisfied (Martinez 2001). In the sequential work processes, the succeeding activity can start whenever an instance of any preceding activity ends (Martinez 2001). Since the activity have specific durations, the finish times of the activity—at which activity's outcomes (e.g., products) are produced—are determined by both its start time and its length of durations according to the discrete time advancement (Law and Kelton 2006).

### 2.3.3 Need for SD-DES Interactions

Since the regional- and facility-level recovery efforts affect each other, the comprehensive analysis of interactions between regional-level recovery processes and facility-level restoration operations is required for implementing more reliable recovery plans. The reconstruction of the damaged built environment requires great amounts of public expenditures. The government is thus mainly responsible for managing the overall disaster management cycle by implementing all possible measures to support restoration efforts with the cooperation of local governments and communities (RA 2011; RA 2013). Therefore, damage and recovery situations in regions and regional-level recovery plans thus have a significant effect on each facility restoration project (Karlaftis et al. 2007).

In particular, the most severely restricted aspects of the restoration activities in facility restoration projects are their inefficient resource supplies and unavailable services for works, which can vary with overall damage and recovery situations at the damaged region (Holguin-Veras and Jaller 2012). Despite the importance of improving repair/reconstruction processes within assigned resources at a facility level, previous research efforts have thus identified the various elements of post-disaster limited external conditions for a facility restoration project, caused by surrounding damage and recovery situations, as follows: (a) resource availability for each restoration project (e.g., materials, equipment, and workforces) becomes limited due to surrounding damaged production systems and the simultaneous high demands for repair/reconstruction as well as governmental plans for resource prioritizations (Orabi et al. 2010); and (b) poor working environments, such as a lack of available spaces for construction works caused by excessive debris and the damaged transportation systems that interrupt the delivery of resources, may cause delays in construction activities more than expected (Swan 2000; Olshansky et al. 2012; Holguin-Veras and Jaller 2012).

What makes it worse is that these external conditions of facility restoration projects can be different according to various damage and recovery situations in the facility's surrounding areas and continuously change. For instance, when road damage is more severe than the damage to other structures, more resources need to be preferentially input to road restoration projects to rapidly re-activate interrupted transportation systems. According to the functionality recovery of transportation systems over time, restoration working environments such as resource delivery capabilities are gradually improved, and consequently restoration operations get faster. Therefore, the comprehensive analysis of multiple complex recovery efforts at the regional level using SD is required for facility-level restoration operation analysis to understand dynamic features of critical and various external restoration conditions, which can help better planning of facility restoration projects.

**Table 2.2** Comparisons between SD and DES

| **Categories** | **SD** | **DES** |
|---|---|---|
| Time Advancement | Continuous time advancement | Discrete time advancement |
| Level of Analysis | Macro-level analysis (Comprehensive view) | Micro-level analysis (Detailed view) |
| Crucial Mechanisms | Cause-and-effect relationships and feedback processes | Repetitive operations on a single process structure |
| Focuses and Objectives | Capturing dynamic state changes of multiple interdependent process | Representing processes and operational details, including resources |
| Abilities | Analyzing both the physical aspects and policies with a strategic and holistic point of view | Handling the complexity and uncertainty with an event-oriented view |

Table 2.2 offers comparative investigations of both the SD and DES. Despite the usefulness of the SD and DES in recovery management, the use of a single method may not be enough for effective recovery planning in a chaotic situation. For example, the SD has a limited capability in representing the detailed construction operations of a facility restoration project due to the difficulties in modeling sequential processes among discrete sub-activities

(Williams 2002; Peña-Mora et al. 2008). This difficulty may become worse when construction process is different according to diverse facility damage patterns. On the other hand, the DES lacks the capability to incorporate feedback processes in continuously changing surrounding project conditions caused by regional-level recovery situations due to its discrete time advancement (Peña-Mora et al. 2008; Lee et al. 2009).

To address such shortcomings, there has been a growing interest in a hybrid simulation because it can support the comprehensive analysis of complex systems with both strategic and operational points of view (Mosterman 1999). A hybrid simulation approach was successfully used in various fields to comprehensively analyze the impact of higher-level decisions (e.g., enterprise-level production planning) on the lower level (e.g., manufacturing processes) and vice versa (Venkateswaran et al. 2004). Similarly, Lee et al. (2009) applied a hybrid SD-DES simulation to construction in order to better understand the complex interactions between construction operations and context in a large-scale project.[8] Lee et al. (2009) also provided the method to show how continuous variables in SD and discrete variables in DES can interact seamlessly. The integration of both the continuous and discrete time advancement of SD and DES respectively thus

[8] "Context" represents the overall project-related behaviors, which interact with operations and changes dynamically as a result of the feedback between two (Lee et al. 2009). Generally, context, mainly used in the linguistics field, refers to surrounding and implicit, cultural and situational information to be helpful for understanding true meaning of the language, text and the expression (Ghadessy 1999). Context in the construction projects can be also defined as the surrounding and implicit situational information to be helpful for understanding "operation".

enables hybrid simulation to fully utilize the benefit of different simulation methods (i.e., SD and DES), as well as to conduct a comprehensive analysis on restoration management. In this regard, the SD-DES interacted hybrid simulation can provide analytic capabilities for both facility- and regional-level disaster restoration situations, including the operation itself and the critical surroundings affecting the restoration operation.

## 2.4 Relevant Simulation Techniques

### 2.4.1 Useful Existing Techniques

To achieve the goal of this study that aims to develop an interactive simulation by integrating seismic intensity analysis and damage simulation tools into recovery simulations, the most appropriate existing techniques are selected by examining the usability, interoperability, accessibility, and development easiness of each technique. Based on the investigations of existing disaster-related simulations in previous table, Table 2.1, it is found that an open-source simulation is more effective for seamless interactions among simulations because the control over source codes is required for the simulation interactions. Commercial software, which is generally closed-source simulation, still has problems in fully interacting with other simulation software.

In this regard, the *OpenSees* is an open-source and object-oriented software developed at the University of California–Berkeley, and supported by the PEER (Pacific Earthquake Engineering Research Center) and the Nees (Network for Earthquake Engineering Simulation) (McKenna 1997). The *OpenSees* has so far been focused on providing an advanced finite-element computational tool for analyzing the nonlinear response of structural and geotechnical systems subjected to seismic excitations (Jiang et al. 2014). The *OpenSees* also provides a structure library and a visualization module. Since the *OpenSees* facilitates both the collaborative works with a substantial community of developers and users and the integrated implementations with other simulations, the *OpenSees* is identified as an effective tool for structural response analysis required for a facility damage assessment in the distributed restoration simulation.

On the other hand, it can be more meaningful for immediately detecting actual seismic information after an event rather than analyzing a seismic intensity by using disaster prediction and estimation tools before an event, because recovery planning are generally implemented immediately after an earthquake due to the rapidly and unpredictably changing internal and external circumstances in the post-disaster situation. In other words, the use of accurate real-time data retrieval technique can be more effective in appropriate and rapid recovery planning compared to the use of a predicted data. Therefore, this study utilizes web-based real-time earthquake data feeds provided by the U.S. Geological Survey (USGS) as a seismic data retrieval module. The *USGS* provides several ways to obtain real-time earthquake lists, in addition to web-based maps and event pages. Earthquakes are broadcast (i.e., through email, CSV, ATOM, CAP alerts, etc.) within a few minutes for California events, and within 30-minutes for worldwide events (USGS 2014). This earthquake information can be specified using the query methods by requesting interest magnitudes, locations, and time of events. Therefore, the *USGS* seismic data retrieval module can replace the functions of earthquake simulation and prediction tools. By using real-time earthquake information from the *USGS* seismic data retrieval technique, approximate damage patterns throughout the region can be estimated. Structural responses of the facility can also be analyzed by using earthquake information and the *OpenSees* simulation. After both regional and facility damage are assessed based on these data, different recovery processes of both regions and facilities according to damage will be captured using the recovery simulations.

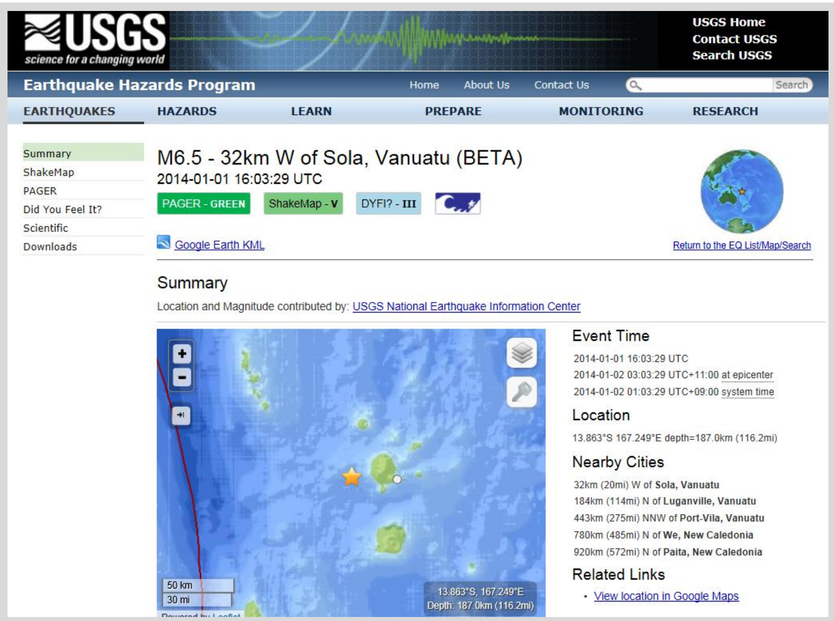



**Figure 2.4** USGS Real-Time Earthquake Data Feeds (USGS 2014)

On the other hand, none of recovery simulation techniques are examined to exactly match our modeling purpose. Since the recovery simulation in this study comprehensively deals with both regional-level interdependent multiple recovery processes and facility-level restoration operations, it is more appropriate to develop our own integrated SD-DES recovery simulation as explained in previous chapter, Ch. 2.3, rather than to utilize existing recovery simulation tools. Since the SD-DES recovery simulation enables a deeper understanding and a dynamic analysis of continuously changing complex recovery systems with both detailed and comprehensive perspectives, it can help to establish more reliable recovery plans appropriate to diverse recovery efforts.

### 2.4.2 Need for Distributed Simulation

Although there exist numerous advanced simulation techniques and models that has different abilities respectively (e.g., a disaster prediction and disaster intensity analysis, a damage analysis, and a disaster recovery analysis), the simultaneous use of multiple simulations is required to incorporate both a disaster intensity analysis and a damage assessment into multiple recovery management.

However, a gap exists between each simulation technique with regard to the target, level, scope, and purpose of an analysis. For instance, disaster prediction and damage simulation simulators have limited capabilities in including numerous types of disasters and damage patterns as well as in supporting recovery planning. On the other hand, disaster recovery simulation tools lack the ability to integrating both regional- and facility-levels of damage and recovery situations. Due to a specific purpose of each technique, a single simulation is hard to satisfy functional needs for a comprehensive analysis of multi-level recovery simulations.

According to Dahmann et al. (1998), the main ideas of the needs of distributed simulation include: first, no one simulation can solve all the functional needs for modeling and simulation of the complex systems; second, the needs of simulation are too diverse as well as change over time; third, it is difficult to anticipate how simulation will be used in the future or in what combinations.

For example, for the purpose of multi-level recovery planning after a catastrophic earthquake in this study, a seismic intensity analysis, a structure damage simulation, and both a regional-level recovery simulation (i.e., SD model) and a facility-level restoration simulation (i.e., DES model) need to be integrated according to each analytical purpose. In this situation, a distributed simulation can concurrently utilize diverse simulation systems and incoming data streams for their own purposes (Yotsukura and Takahashi 2009). According to Dimakis et al. (2010), each simulated object is a dedicated active entity that does not necessarily need to be in the same physical computer location as any other of the entities. Furthermore, all aspects of the simulation are all privately held by each simulated object and this enhances flexibility and security. Finally, a distributed simulation platform adopts a paradigm in which certain parts of the simulation are simulated independently.

Therefore, the distributed simulation has been widely applied to disaster management areas including complex disaster analysis, damage mitigation, evacuation planning, rescue planning, and emergency relief efforts (Koto and Takeuchi 2003; Currion et al. 2007; Hu et al. 2009; Yotsukura and Takahashi 2009; Dimakis et al. 2010). In this regard, a distributed simulation can be effectively utilized to the multi-level recovery management with the interactions among seismic intensity analysis techniques, regional and facility damage assessment techniques, and multi-level recovery analysis modules.

## 2.5 High Level Architecture (HLA)

For the purpose of promoting reusability and extendibility of the distributed simulation in the future, the interactive recovery simulation in this study is developed as a prototype based on the principles defined in the High Level Architecture (HLA). The HLA was first developed by the U.S. Department of Defense (DoD) to provide interoperability and reusability among each simulations by its general rules for distributed simulation environments (Zhang et al. 2011). The HLA, which was standardized by the IEEE (Institute of Electrical and Electronics Engineers) (IEEE 1516), is a collection of general rules that have been created and compiled to guide and manage the development of complex and interoperable simulations capable of running across multiple processes or computers in a distributed network environment (Menassa et al. 2014). HLA thus enables computer simulation to exchange information, coordinate operation, and synchronize simulation action regardless of their technical implementation and platforms because it provides standards for building the individual federates of such environments by different developers while maintaining interoperability between them (AbouRizk 2010).[9] HLA has been applied to the areas in which distributed, scalable, and extensible simulations are needed (AbouRizk 2010).

The HLA consist of three components including HLA rules (IEEE 1516), HLA interface specification (IEEE 1516.1), and Object Model Template (OMT) (IEEE 1516.2). First, HLA rules describe proper interaction of federate in a federation and the responsibilities of federates and federations.[10] Second, HLA interface specification defines Run-Time Infrastructure (RTI) services and interfaces as well as identifies "call back" functions each federate must provide. The RTI is a server-based utility that can run on any of the machines that house one or more of the federate simulations or on its own server. As shown in Figs. 2.5 and 2.6, in the HLA-compliant distributed simulation the RTI coordinates the synchronization and transfer of data between federates, which can run simultaneously in a distributed fashion

---

9 A federate is an individual simulation component such as single simulation model, live participant, and incoming data stream (Kuhl et al. 2000).

10 A federation is a set of federates of a collection of multiple running and interacting federates such as models, participants, or data streams (Kuhl et al. 2000).

(Menassa et al. 2014). Finally, OMT provides standards for documenting HLA object modeling information and consists of three parts: Federation Object Model (FOM), Simulation Object Model (SOM), and Management Object Model (MOM) (AbouRizk 2010; Zhang et al. 2011).

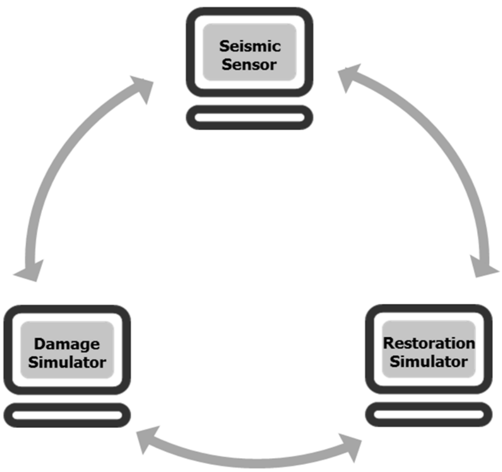



**Figure 2.5** Concept of Distributed Interactive Simulation

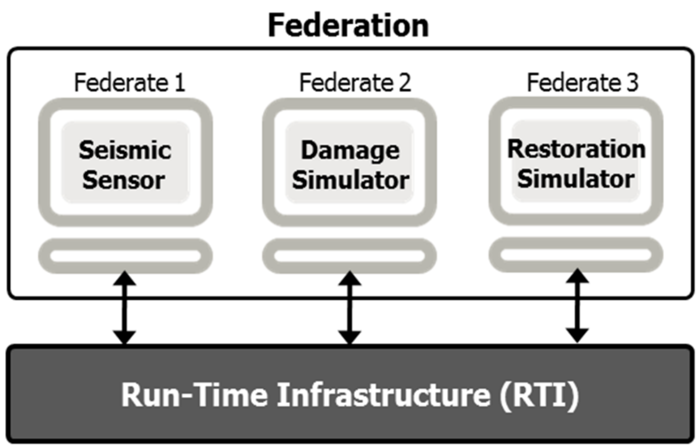



**Figure 2.6** Concept of HLA-Compliant Distributed Simulation

In the HLA-compliant distributed simulation framework, the HLA rules must be enforced if a federate or federation is to be regarded as HLA compliant. Therefore, the HLA is capable of supporting the following interoperability; among federates using different time-advance mechanisms, among heterogeneous hardware platforms, among different implementation languages, and among different network environments (Schulze et al. 1999). It can thus provide the open architecture for existing simulation applications by its reusability and interoperability, and then ultimately reduce the time and cost required to create a new environment while maintaining the advantages of each simulation (Zhang et al. 2011).

The HLA has been applied to diverse area to solve the modeling and simulation issues on the complex problems. It was originally developed for military simulations and war games by addressing the continuing need for interoperability between new and existing simulations within the Department of Defense (Fujimoto 1998). It was further extended to the industrial domain with the aim at improving complex operations including manufacturing systems and their logistics processes in production facilities (McLean et al. 2005). It was also applied to other industrial sectors, such as airport operation systems (Adelantado 2004), and the medical simulation system for training (Petty and Windyga 1999). In addition, research efforts on disaster management using distributed simulation have tried to apply HLA in their simulation platforms, as explained in the previous chapter, Ch.2.4.2 (Hu et al. 2009; Yotsukura and Takahashi 2009; Dimakis et al. 2010).

In the civil and construction domain, the HLA has also been thought of as a synthetic approach to solve the complex problems generated by the uses of diverse planning and management methods as well as the diverse participants in the projects. The example of successful applications of HLA is the transportation simulation which provides the interoperability between a driving simulator federate, a traffic simulator federate, and an observer federate used for visualization within a synthetic environment (Schulze et al. 1999). To provide an integrated alternative approach for modeling and complex simulation, the Construction Synthetic Environment (COSYE) was also developed in the University of Alberta for modeling and simulating construction operations. It was based on the High Level Architecture (HLA) standards (IEEE 1516) for parallel and distributed simulation (AbouRizk et al. 2011). It has successfully been addressed the issues on the construction operation analysis and the visualization of the tunnel excavating, pipe fabrication, and module construction (AbouRizk 2010), as well as student training by introducing a bidding game (AbouRizk et al. 2009).

Although the COSYE facilitates collaborative development, interoperation, and reuse of simulation components, it does not provide fully distributed simulation environment because of the following limitations: first, it does not provide 100% implementation of HLA standards (AbouRizk et al. 2011); second, although the distributed components need to run on the different platforms, it provides the single platform (e.g., Microsoft.NET) to provide an integrated construction simulation. Since our research aims to provide a synthetic environment for countless number of simulation components for multi-level recovery management, the completely distributed environment is further required. In other words, it is not possible to construct a single integrated simulation environment for the numerous components which are (or will be) developed by different agent, by focusing on different facilities and different recovery efforts, by using different simulation methods, and under the different platforms. The fully distributed environment should thus enable numerous components to communicate each other by providing the reusability and interoperability within a fully open architecture.

Based on investigations of several implementations of the HLA RTI, it is found that the CERTI HLA implementation can be effectively applied to design fully distributed simulation environment because CERTI HLA RTI is an open-source and platform/programming language independent RTI by providing the flexibility in the simulation interactions (Noulard et al. 2009; Menassa et al. 2014). Therefore, the use of CERTI HLA RTI is expected to facilitate the integration of other new and existing simulation techniques into the interactive recovery simulation according to further requirements for the analysis of different damage and recovery situations.

## 2.6 Summary

In this chapter, this study introduced simulation methodologies and techniques to be utilized in the development of an interactive simulation for post-disaster recovery planning with regard to both regional and facility levels. By conducting in-depth investigations of previous research efforts and disaster cases, it was found that the computer simulation is effective to analyze complex and uncertain facility restoration projects in a chaotic situation. While existing disaster-related simulations have effectively been applied in disaster management such as disaster predictions and damage estimations with a technical approach, the modeling approaches using diverse simulation methodologies can also provide a comprehensive understanding of the disaster recovery analysis and an insight into the recovery planning. In particular, SD allows for a comprehensive understanding of overall recovery processes in damaged regions, while DES enables the examination of facility restoration operations. The SD-DES interacted simulation is more effective in accurately modeling and comprehensively analyzing the complex interactions among regional- and facility- level recovery efforts in a complex and uncertain disaster situation.

In addition, existing disaster simulation techniques, particularly tools for supporting immediate damage assessments, need to be incorporated in the recovery simulation for more prompt and reliable analysis. After investigating existing techniques, the web-based USGS seismic data retrieval tool can be also utilized to promptly detect earthquake information for the purpose of rapidly estimating regional damage patterns and analyzing disaster intensities. The OpenSees simulation software is also identified as an effective tool for structural response analysis required for assessing facility damage.

Since each single simulation for analyzing disaster (e.g., an USGS seismic data retrieval), structural responses (e.g., an OpenSees Simulator), and post-disaster damage and recovery situations (e.g., an SD-DES simulation) needs to be able to interact with each other for a comprehensive analysis, a distributed disaster simulation approach is applied. For the purpose of promoting reusability and extendibility of distributed simulation in the future, an interactive recovery simulation is developed as a prototype based on the principles defined in the HLA, which is a collection of general rules that have been created to guide and manage the development of complex and interoperable simulations capable of running across multiple processes or computers in a distributed network environment. Fig. 2.7 describes a summary of the interactive recovery simulation development in this study.

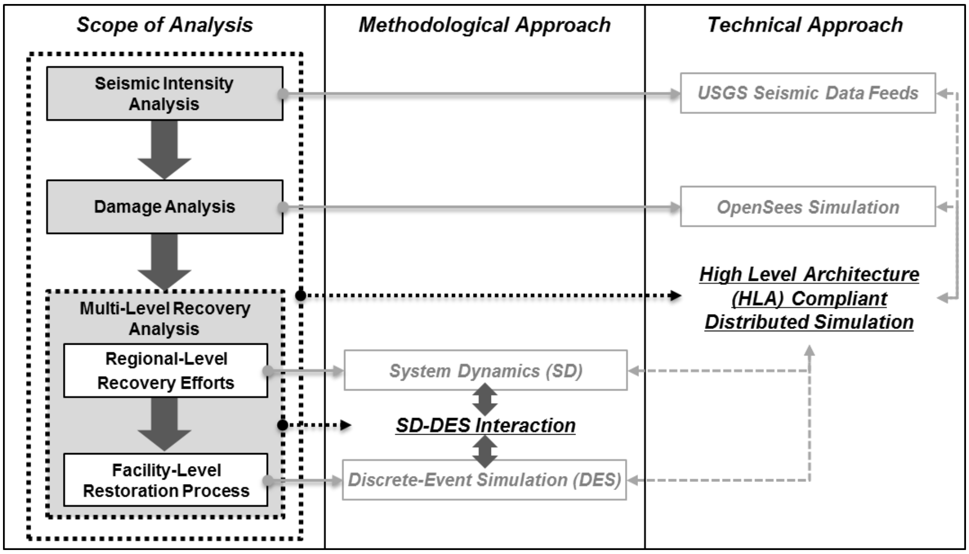



**Figure 2.7** Summary of Interactive Simulation Development

# Chapter 3. Interactive Simulation Framework

In this chapter, this study develops an interactive simulation framework for multiple recovery management, with a focus on different functional needs of simulations for both regional- and facility-level recovery planning. This framework is developed based on the investigations of previous research efforts, useful modeling methodologies, and relevant techniques. By examining input and output information as well as purposes of each simulation model, this study determines the concepts and required functions of each simulation model as well as their interactions for a comprehensive analysis of the post-disaster multi-level recovery management.

## 3.1 Overview of Framework Development

One of the major issues in implementing interactive simulation is the lack of modeling framework (Alvanchi et al. 2011).[11] The interactive simulation framework is helpful for preventing the loss of the modeling scope and purpose when each model in the interactive simulation environment is separately developed, and when interactions among models are determined. Although many research efforts have been trying to develop disaster simulation platforms in the distributed simulation environment with a technical perspective, the lack of framework for the interactive disaster

[11] The modeling framework refers to "the set of provided basic modeling elements and concepts (Alvanchi et al. 2011)."

simulation has prevented the reusability and future extendibility of developed simulations. Therefore, this study develops the interactive recovery simulation framework in advance, with a consideration of diverse analytical requirements for multiple recovery management as well as communications among diverse disaster-related simulations.

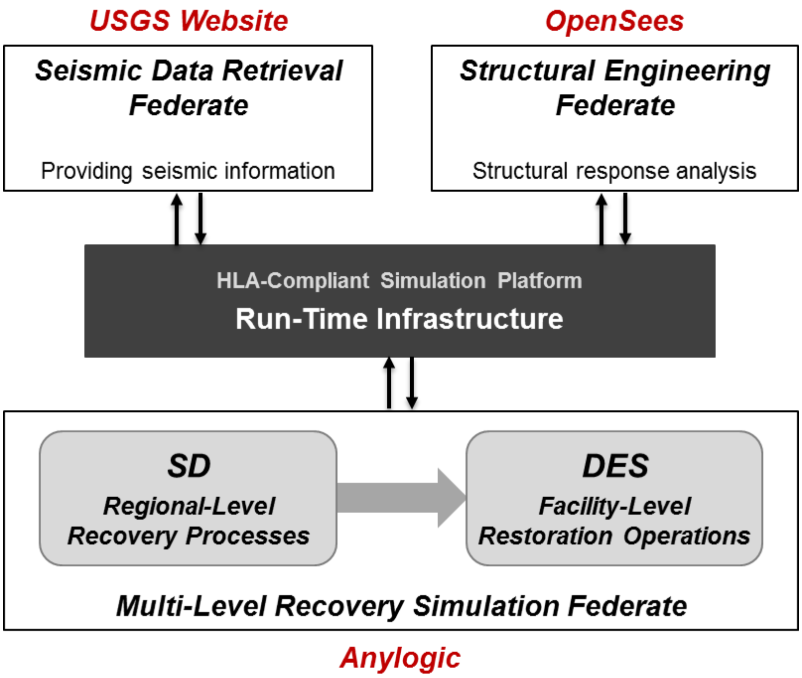



**Figure 3.1** Basic Structure of Interactive Recovery Simulation

Based on the in-depth investigations of useful simulation methodologies and relevant techniques as described in previous chapter, Ch. 2, Fig. 3.1 offers the basic structure of the interactive recovery simulation. In the distributed simulation environment, HLA RTI provides functions for exchanging information, coordinating operations, and synchronizing simulation actions among required sub-simulations which can be referred to "federates". In particular, HLA enables not only the interactions among simulations but also the interactions between data incoming techniques (e.g., an USGS real-time seismic data feeds). Therefore, seismic data retrieval techniques can interact with other relevant simulations for multi-level recovery management.

In the HLA-compliant simulation platform, three federates—including an seismic data retrieval federate using USGS real-time data feeds, a structural response simulation (structural engineering) federate using OpenSees, and a SD-DES multi-level recovery simulation federate—are able to interact with each other. In particular, this study utilizes Anylogic 7 software (Anylogic Company) for the SD-DES recovery simulation because the Anylogic software provides multi-method simulations useful in developing an SD-DES interacted simulation and providing a comprehensive analysis of complex problems in a large system.

## 3.2 Application of Disaster Simulation Techniques

The application of existing techniques for disaster-related data retrievals and simulations can support recovery simulations in more rapidly and accurately figuring out damage status of each facilities and regions. Therefore, this study conducts an investigation of input and output data of both the USGS seismic data retrieval federate and the OpenSees structural response simulation federate (McKenna 1997; USGS 2014). Since these existing techniques are incorporated into the multi-level recovery simulation, more detailed investigations of these techniques can be effective in further development of a seamless interactive recovery simulation.

### 3.2.1 USGS Seismic Data Retrieval Technique

Table 3.1 describes an input-output analysis of the USGS seismic data retrieval technique. The USGS federate requires a set of query information to detect a seismic event of interest. This query information includes the event time, the minimum and maximum values of both the latitude and longitude in the target region, and the minimum value of a magnitude of interest seismic events. The USGS federate then generates earthquake information relevant to requested queries, including an event time, and epicentral location (e.g., latitude and longitude), and a focal depth of a current seismic event.

**Table 3.1** Input and Output Analysis for the USGS Seismic Data Retrieval Federate

| Techniques | Input Data | Output Data |
| --- | --- | --- |
| Seismic Data Retrieval (USGS) | <u>Requested query</u><br>· The minimum and maximum values of interest event time<br>· The minimum and maximum values of both longitude and latitude of interest locations<br>· The minimum value of interest magnitude | <u>Earthquake information</u><br>· Event time<br>· Event location (Longitude, latitude)<br>· Magnitude of events<br>· Focal depth of events |

The USGS real-time earthquake data feeds provides numerous methods (e.g., HTTP GET method and programmatic access) for requesting queries to the USGS server and provides diverse output data formats. Table 3.2 shows the example of query requests for a seismic event using the HTTP GET method. To acquire earthquake information of the M 9.0 earthquake off the Pacific coast of Tohoku in Japan of March 2011 (the M 9.0 2011 Earthquake of Tohoku), the locational information of the region, an interest time, and the magnitude are submitted to the USGS server. A request URL is then constructed based on these query parameters, as follows:

· Request URL (Example):

*http://comcat.cr.usgs.gov/fdsnws/event/1/query?minmag=8.0&starttime=2011-03-07T00:00:00&endtime=2011-03-14T00:00:00&minlatitude=37.0&maxlatitude=39.0&minlongitude=141.0&maxlongitude=143.0*

**Table 3.2** Example of Query Requests to the USGS Server for Seismic Events

| **Query Method Parameters** | **Descriptions** |
|---|---|
| Start time | 2011-03-07T00:00:00 |
| End time | 2011-03-14T00:00:00 |
| Minimum magnitude | 8.0 |
| Minimum latitude | 37.0 |
| Minimum longitude | 141.0 |
| Maximum latitude | 39.0 |
| Minimum longitude | 143.0 |

When the seismic event that corresponds to requested queries is found, the earthquake information of this event is provided. Fig. 3.2 and Table 3.3 show the detected earthquake information that the USGS server provides. According to submitted parameters in this example, the M 9.0 Earthquake of Tohoku of March 2011 is found with the information of the event time, epicentral location, magnitude, and focal depth, as shown in Fig. 3.2 and Table 3.3. This real-time earthquake information can further be utilized to promptly analyze damage situations for regional-level damage estimations and recovery planning. This information can also assist to assess detailed structural damage of facilities by providing intensities of seismic events.


**Figure 3.2** Example of Earthquake Detection: The M 9.0 Earthquake of Tohoku of March 2011

**Table 3.3** Example of Earthquake Information Outputs from the USGS

| Parameters | Data Type | Descriptions |
|---|---|---|
| Event ID | String | pde20110311054624120_29 |
| Time | Long Integer | 2011-03-11T05:46:24.120Z |
| Latitude | Decimal | 38.297 |
| Longitude | Decimal | 142.373 |
| Focal depth value [m] | Decimal | 29,000 |
| Magnitude value | Decimal | 9.0 |

### 3.2.2 OpenSees Structural Response Simulation

Table 3.4 describes an input-output analysis of the OpenSees structural response simulation. The OpenSees federate requires input values for the structure information such as elements and nodes, materials, constraints, and load patterns. Since the object-oriented OpenSees provides library of materials, elements, and analyses, it allows developers to easily construct a new model and modify existing specific model to perform response simulations of specific structures. To analyze structural responses according to a seismic event, the OpenSees simulation also requires the earthquake information such as ground motions at the point of a structure. In particular, ground motions and shaking data which can be represented by accelerograms,[12] as shown in Fig. 3.3, can be directly used in the OpenSees model. Based on these input data, the OpenSees produces structural responses such as structure displacements, damping forces, and so forth.

[12] An Accelerogram is a strong motion data that represents the seismic excitation of the ground. It is recorded by an Accelerometer, which is a device for real-time measurement of seismic accelerations (Bommer and Acevedo 2004).

**Table 3.4** Input and Output Analysis for the OpenSees Structural Response Simulation Federate

| Techniques | Input Data | Output Data |
|---|---|---|
| Structural Response Simulation (OpenSees) | <u>Structure information</u><br>· Nodes and elements<br>· Materials<br>· Nodal coordinates<br>· Transformation<br>· Boundary conditions<br>· Element connectivity<br>· Nodal masses<br><u>Seismic information</u><br>· Ground motions: Acceleration (accelerograms) | <u>Structural responses</u><br>· Displacement<br>· Node velocity and acceleration<br>· Incremental displacement<br>· Eigenvector<br>· Damping forces |

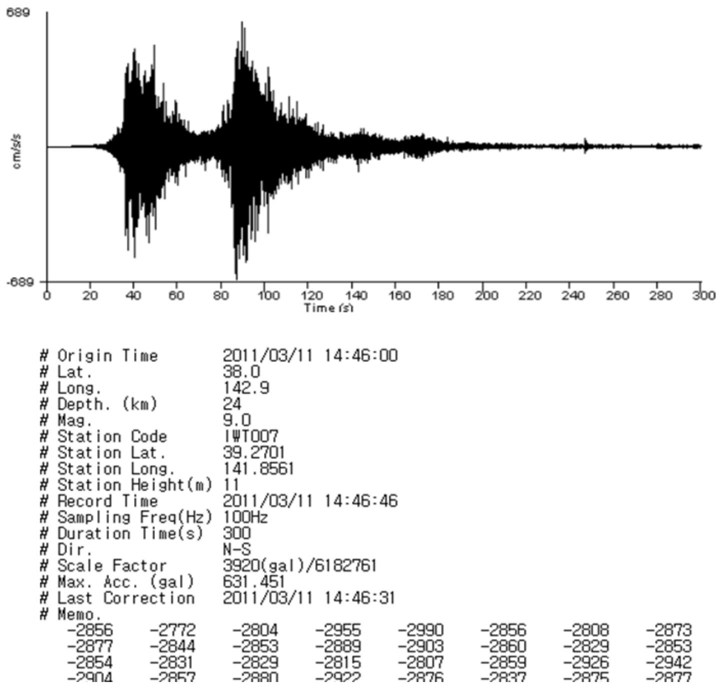



**Figure 3.3** Example of Ground Motion Data (Accelerograms)

Fig. 3.4 shows the example of a typical OpenSees model that represents the generic frame structure with reinforced-concrete sections and steel W-sections. Based on this generic model provided by the OpenSees Website (2013), the simulation models for specific facilities can be constructed by simply revising the values of parameters. Moreover, a structural response analysis for a specific seismic event can be conducted by only changing input files of ground motions.

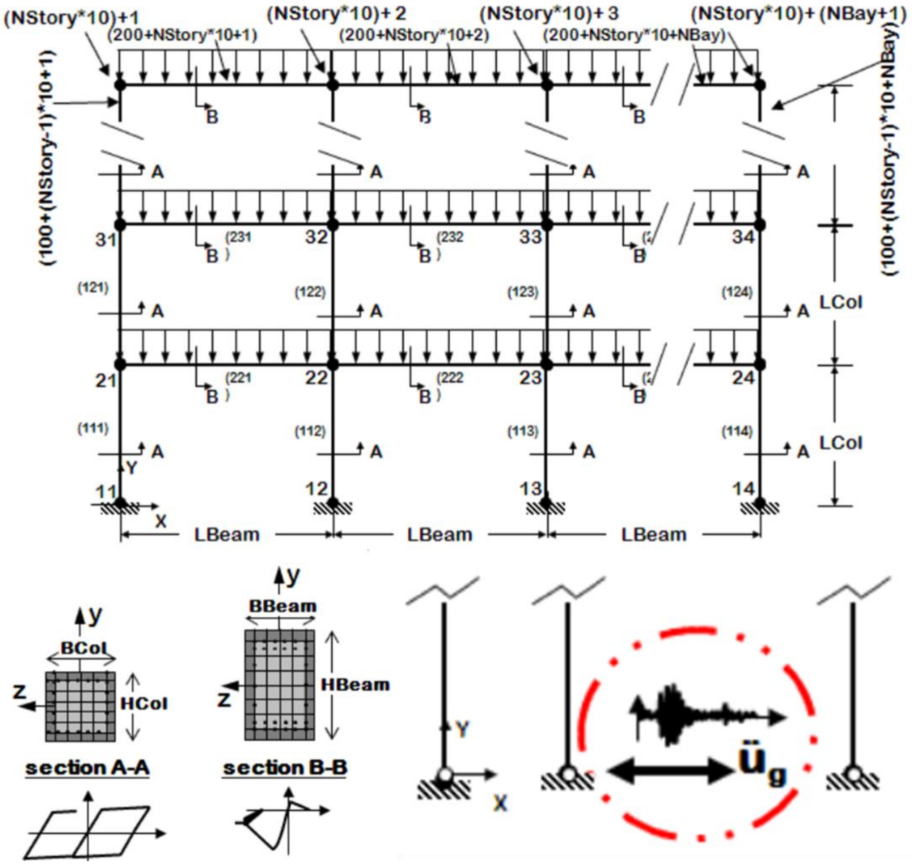



**Figure 3.4** Example of Generic OpenSees Model (OpenSees 2013)

### 3.2.3 Open Source CERTI HLA RTI

With a consideration of fully distributed environment in the future, this study utilizes CERTI HLA RTI, which is an HLA RTI developed since 1996 by ONERA, the French Aerospace Lab. It is an open source RTI available on various platforms including various flavors of Linux, Microsoft Windows, Solaris, FreeBSD, and IRIX, and has multiple language bindings such as C++, JAVA, and Matlab (Noulard et al. 2009). Since the interactive recovery simulation in this study requires multiple language bindings according to the uses of various simulation techniques such as C++ (e.g., OpenSees) and JAVA (e.g., Anylogic), CERTI can be effectively applied to develop an interactive recovery simulation prototype. CERTI HLA RTI follows the IEEE 1516 standard and provides total controls over source code because CERTI is fully opened with the general public license. It also allows users to construct federations from a set of communicating components according to the needs of simulations (Noulard et al. 2009). As shown in Fig. 3.5, the CERTI RTI is a distributed system involving two processes, a local one (RTIA) and a global one (RTIG), as well as a library (libRTI) linked with each federate.

Each federate process interacts locally with an RTI ambassador process (RTIA). This point evolved when we ported CERTI to Windows systems and on multiprocessor architectures. The RTIA processes exchange messages over the network, in particular with the RTI gateway process (RTIG), via TCP (and UDP) sockets, in order to run the various distributed algorithms associated with the RTI services. A specific role of the RTIA is to immediately satisfy some federate requests, while other requests require network message sending or receiving. The RTIA manages memory allocation for the message FIFOs (i.e., first-in, first out) and always listens to both the federate and the network (the RTIG). It is never blocked because the required computation time is reduced. It also plays a great role in the implementation of the tick function.

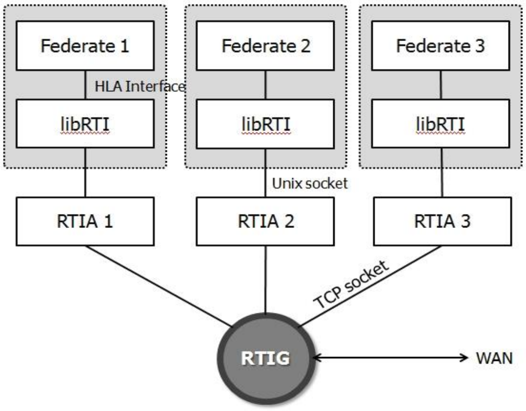



**Figure 3.5** Core Structures of CERTI HLA RTI

The RTIG is a centralization point in the architecture. Its function has been to simplify the implementation of some services. It manages the creation and destruction of federation executions and the publication/subscription of data. It plays a key role in message broadcasting which has been implemented by an emulated multicast approach. When a message is received from a given RTIA, the RTIG delivers it to the interested RTIAs, avoiding a true broadcasting. Since the CERTI HLA RTI follows the IEEE 1516 standard including HLA interface specifications, CERTI HLA RTI also enables dynamic data exchanges among distributed simulations by providing six management areas as defined in the HLA interface specifications (IEEE 1516.1), as shown in Table 3.5.

**Table 3.5** Functions of HLA RTI (DMSO 2001)

| HLA Functions | Descriptions |
|---|---|
| Federation Management | Creating and destroying federations, joining federates to federations, and resigning federates from federations |
| Declaration Management | Data exchange coordination by specifying data types which federate sends and receives and controlling required data |
| Object Management | Enabling each federate to create, modify, and delete object instance and interaction |
| Ownership Management | Enabling federates to distribute the responsibility for updating and deleting object instances with a few restrictions |
| Data Distribution Management | Supporting efficient routing of data, specifying data distribution, and acknowledging routing conditions |
| Time Management | Coordinating time advancement of each federate |

First, RTI performs creating federations, joining federates to federations, observing federation-wide synchronization points, effecting federation-wide saves and restores, resigning federates from federations, and destroying federations during simulations (i.e., federation management). It enables that each components developed in different environment can interact with other components. It also creates federation among related components according to particular simulation purposes (DMSO 2001). Second, RTI provides data exchange coordination by specifying data types which federate sends and receives and controlling what data is required (i.e., declaration management).

Based on this supporting control functions, each federates declare exactly what they are able to generate (i.e., publication) and what they require (i.e., subscription) (DMSO 2001). Third, RTI enables each federate to create, modify, and delete object instance and interaction. It also includes methods for sending and receiving interactions, controlling instance updates by connecting with other federate which it is currently interested in. (i.e., object management) (DMSO 2001). Forth, RTI allows federates to distribute the responsibility for updating and deleting object instances with a few restrictions. In other words, it supports transfer or share of ownership for single object instances among several federates (i.e., ownership management) (DMSO 2001). Fifth, RTI provides a flexible and extensive mechanism for supporting efficient routing of data, specifying data distribution, and acknowledging routing conditions (i.e., data distribution management) (DMSO 2001). Sixth and finally, RTI coordinates time advancement of each federate by establishing or associating events with federate time, regulating interactions, and supporting interactions among federates using different timing schemes (i.e., time management) (DMSO 2001).

By using the distributed simulation technique and HLA concept based on an open source CERTI HLA RTI implementation, this study develops an interactive simulation for multiple recovery management. The open source RTI can promote further development and integration of simulation components for diverse disaster and recovery situations.

## 3.3 Requirements for Recovery Simulation

To develop the interactive simulation framework, this study investigates purposes, abilities, and required input and output information of each simulation model components in the recovery simulation federate. This investigation can be helpful for determining the concept, scope, and required functions of each model components to achieve modeling purposes. It can also be helpful for determining how simulation model components are seamlessly interacted.

To analyze both regional- and facility-level recovery processes, the recovery simulation requires the information of regional and facility damage. Since the USGS federate only provides information of seismic intensities rather than information of damage, regional damage estimation by a seismic event needs to be further conducted. Structure damage needs to also be further calculated based on the structural response information from the OpenSees because the OpenSees federate only provides structural response information. While the USGS federate generates earthquake information such as the magnitude and epicenter, the OpenSees federate requires ground motion data of the point in which the facility resides. Therefore, the method for converting earthquake information to ground motion data needs to be included. As a result, model components in the Anylogic recovery simulation federate need to include not only both the SD model for regional-level recovery analysis and the DES model for facility-level restoration analysis, but also the regional-level damage estimation model and the facility-level structural damage assessment model. Table 3.6 offers required functions and required input and output data for model components in the recovery simulation.

**Table 3.6** Required Functions for Model Components in Recovery Simulation

| Model Components | Input Data | Output Data |
|---|---|---|
| Regional Damage Estimation | Earthquake information<br>▪ Event location and focal depth<br>▪ Magnitude of an event<br>Regional information<br>▪ Location of regions | Regional damage<br>▪ Regional damage ratio<br>▪ Functionality losses |
| Facility Damage Assessment | Earthquake information<br>▪ Event location<br>▪ Magnitude of an event<br>Facility information<br>▪ Location of a facility | Earthquake information<br>▪ Ground motions: Acceleration (accelerograms) |
| | Structural responses<br>▪ Displacement<br>▪ Node and element | Facility damage<br>▪ Structural damage ratio<br>▪ Required amount of work |
| Regional-level Recovery Analysis (SD) | Regional information<br>▪ Built environment information<br>▪ Recovery effort information<br>Regional damage<br>▪ Regional damage ratio<br>▪ Functionality losses | Recovery status<br>▪ Recovery works done<br>▪ Functionalities of facilities<br>Restoration conditions<br>▪ Resource supply ratio<br>▪ Work delay ratio |
| Facility-level Restoration Analysis (DES) | Activity information<br>▪ Required resource and duration<br>Facility damage<br>▪ Structural damage ratio<br>▪ Required amount of work<br>Restoration conditions<br>▪ Resource supply ratio<br>▪ Work delay ratio | Project performance<br>▪ Amount of work done<br>▪ Duration (delay) |

## 3.4 Interactive Recovery Simulation Framework

Based on the input and output analysis of model components in the recovery simulation as described before, this study develops the interactive recovery simulation framework as shown in Fig.3.6. This framework is elaborated based on the conceptual framework provided in Fig. 3.1.

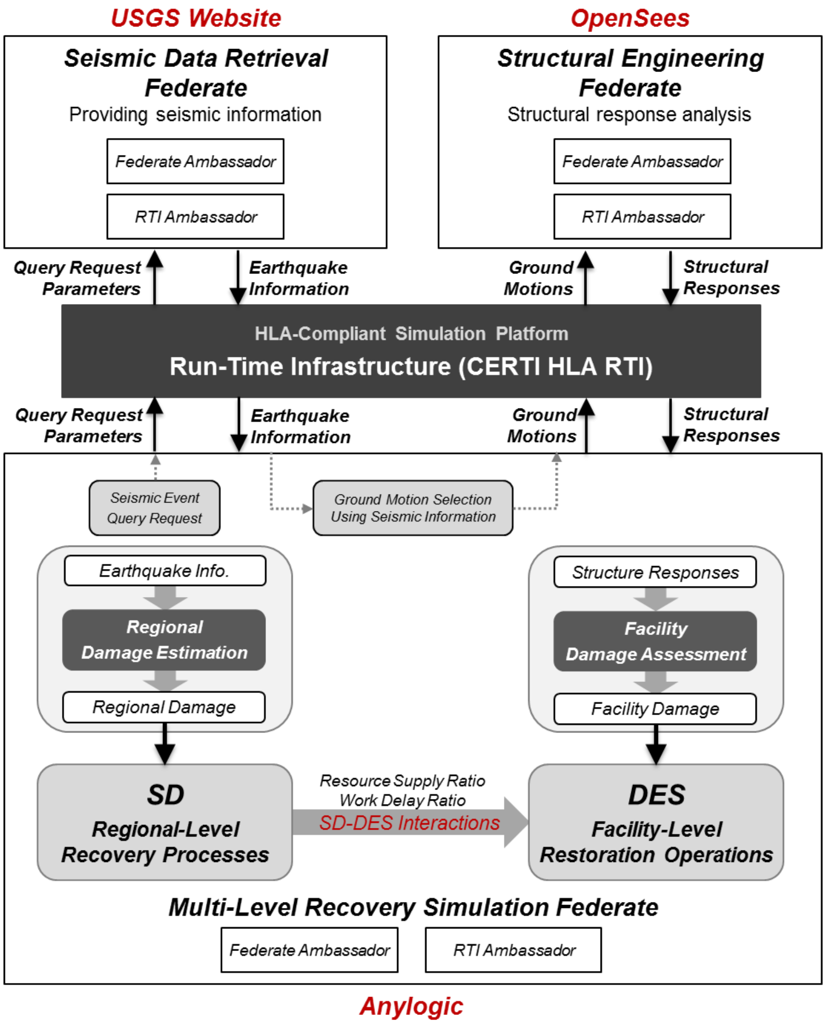



**Figure 3.6** Interactive Recovery Simulation Framework

The main platform of the interactive recovery simulation is the Anylogic federate that performs not only the SD-DES multi-level recovery simulations but also damage assessments of both regions and facilities. Therefore, the Anylogic federate requests earthquake query information to the USGS server for the purpose of rapid recovery simulations through detecting a seismic event. After the USGS federate sends earthquake information according to queries such as the magnitude and epicenter, the Anylogic federate estimates regional damage and converts this information into ground motion data, to send it to the OpenSees federate. Then, the OpenSees federate simulates structural responses of the facility. Based on this structural response data from the OpenSees, the Anylogic federate finally estimates facility damage status.

The HLA-compliant recovery simulation framework is developed by the concept suggested by Menassa et al. (2014). In this framework, the CERTI HLA RTI coordinates the synchronization and transfer of data between federates. In this interactive simulation platform, RTIA (i.e., RTI ambassador in Fig. 3.6) manages communications between the federate (i.e., federate ambassador in Fig. 3.6) and the RTIG (i.e., CERTI HLA RTI in Fig. 3.6), which controls publications and subscriptions of data. When a message for data transfer is received from a given RTIA to the RTIG, the RTIG delivers it to the other interested RTIAs. Then, interested federate can conduct simulation based on the subscribed data. For instance, if the USGS federate creates earthquake information and then sends it to the RTIG and if the Anylogic federate informs the RTIG of its interest to earthquake information, the RTIG delivers this information to the RTIA of the Anylogic federate to activate Anylogic recovery simulation processes.

Figure 3.7 provides detailed descriptions of the recovery simulation framework for both regional and facility levels in the Anylogic federate. This framework is based on the SD-DES interacted simulation for post-disaster multi-level recovery management by determining the interactions among simulation model components in the Anylogic recovery simulation federate, including a regional-level damage estimation, a facility-level damage assessment, a regional-level recovery analysis, and a facility-level restoration operation analysis. Each simulation model produces useful data output based on each model's ability.

First, since the regional-level damage and recovery situations may vary with the damage differences throughout the whole region, an overall regional-level damage estimation caused by a disaster (i.e., an earthquake in this study) needs to be conducted to accurately estimate damage patterns among sub-regions and model recovery processes according to damage (A in Fig. 3.7).

Second, facility restoration operations also may be different according to damage status of facility structures. Therefore, a detailed facility-level damage assessment is required to determine a required amount of repair/reconstruction works in the facility restoration project, with a judgment of expected damage from structural response data (B in Fig. 3.7).

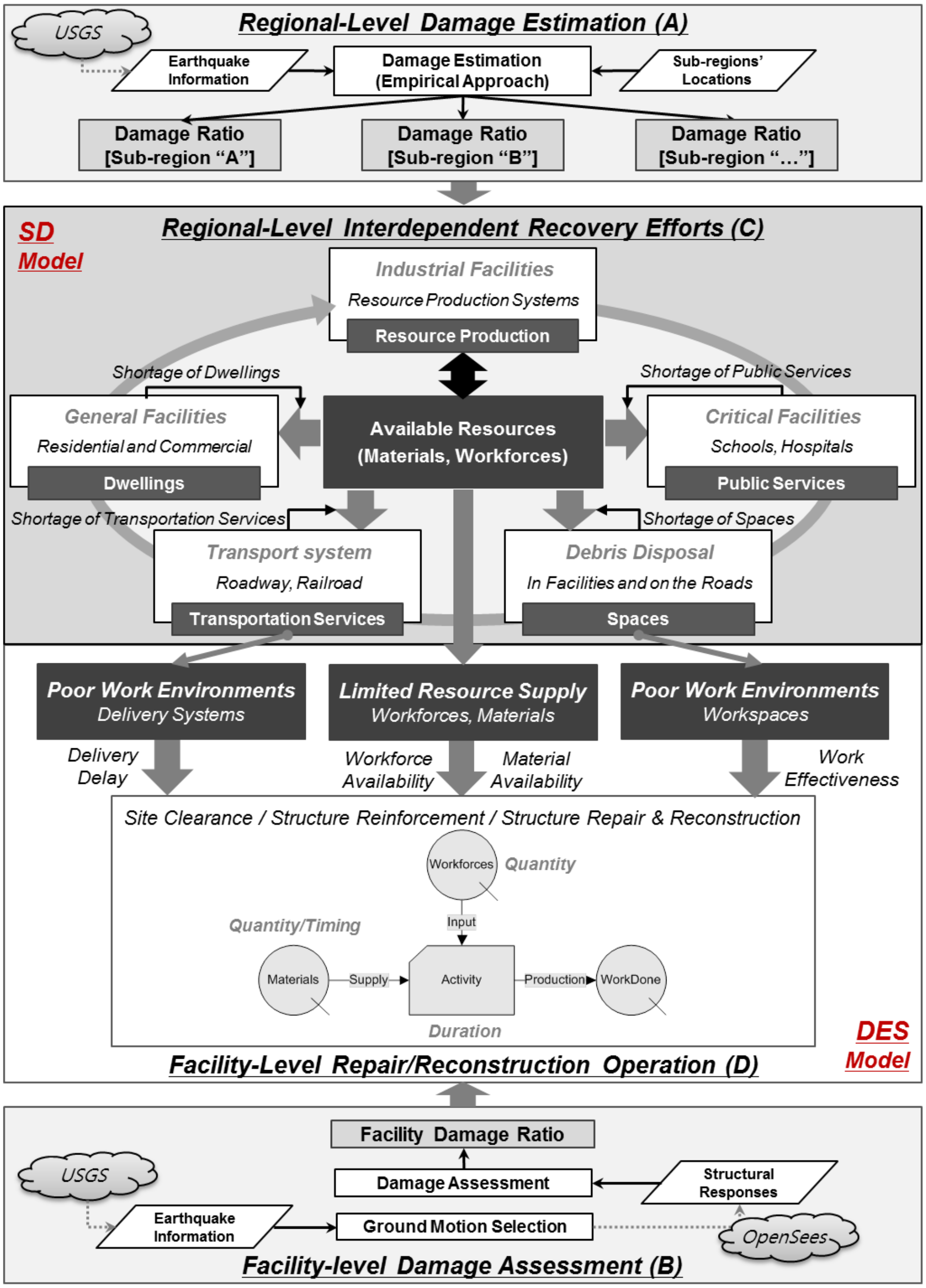



**Figure 3.7** Detailed Descriptions of the Multi-Level Recovery Simulation Framework

Third, the damage in the region causes excessive needs for built environment recovery efforts at a regional level. Since numerous types of recovery works required excessive resources at the same time,[13] resource distribution can mostly be determined by the recovery plans implemented with a consideration of the shortage, relative importance, and interdependency among facilities/infrastructures' functions and services at a certain time. Therefore, regional-level recovery processes—analyzed at a regional level using SD—can vary according to recovery plans (C in Fig. 3.7).

Fourth and finally, due to the structural damage of the facility in the aftermath of a disaster, a considerable amount of repair and reconstruction works are required in the facility restoration projects. In addition, due to different damage and recovery situations among both sub-regions and multiple types of facilities, an in-depth analysis of various restoration conditions for facility-level restoration projects according to sub-regions and facility types is required. In other word, regional-level recovery interdependencies among facilities/infrastructures and regional recovery plans can affect the planning of an individual facility restorations in accordance with the external restoration conditions, such as variable resource supplies and negative work environments that determine the resource availability (e.g., workforces and materials) and work effectiveness (e.g., workspace

[13] According to the diverse functions of facilities and infrastructures, numerous types of recovery works at the damage region can be categorized into restoring general residential and commercial buildings, critical public facilities, industrial facilities, and transportation systems as well as disposing debris (Olshansky et al. 2012).

availability and resource delivery capabilities) for facility-level restoration projects. Therefore, these conditions influence the amounts of resource inputs and activity durations of facility reconstruction operations—analyzed at a facility level using DES—over time (D in Fig. 3.7), and then eventually have a significant effect on overall facility restoration processes.

## 3.5 Summary

In this chapter, this study developed the interactive simulation framework based on the investigation of existing techniques for disaster-related data retrievals and simulations including the USGS seismic data retrieval technique and the OpenSees structural response simulation technique. By examining the applicability of both techniques, it was found that they can be useful for near real-time damage assessment at both regional and facility levels. Despite the ability of both techniques, further development such as a damage assessment model is required to overcome the limitations of these techniques. For example, since the OpenSees federate only provides structural response information rather than direct structure damage information, structure damage needs to be further calculated based on these structural response information.

Based on these investigations, the framework showed how the USGS seismic data retrieval federate, the OpenSees structural response simulation federate, and the Anylogic recovery simulation federate interact with each other in the HLA-compliant distributed simulation environment. In particular, the open source CERTI HLA RTI will be used to improve the flexibility in the development of the prototype. The framework also offered detailed descriptions of how recovery simulation model components—the regional-level damage estimation model, the SD regional-level recovery process analysis model, the facility-level damage assessment model, and the DES facility-level restoration operation analysis model—interact with each other in the single simulation platform of the Anylogic federate, by investigating required input and output information for each simulation model component. The developed interactive simulation framework can be helpful for preventing the loss of the modeling scope and purpose when each model in the interactive simulation is separately developed and when interactions among models are determined in detail. This framework can eventually be useful for promoting the reusability and extendibility of interactive simulations in the future.

# Chapter 4. Simulation Model Components

In this chapter, simulation model components in the interactive recovery simulation are separately developed based on the framework presented in the previous chapter, Ch. 3. Model components for multi-level recovery planning include both the regional-level damage and recovery simulation and the facility-level damage and restoration simulation. In detail, the former consists of a regional damage estimation model and a regional-level recovery analysis model using SD. The latter consists of a facility damage assessment model and a facility-level restoration analysis model using DES. This chapter provides detailed model descriptions and test results using case examples. Interactions and data exchanges among model components are also described.

## 4.1 Regional-Level Damage and Recovery Simulation

### 4.1.1 Regional Damage Estimation Model

***Model Descriptions***

In this section, this study develops a regional damage estimation model by the integrated uses with the USGS seismic data retrieval federate. The regional-level damage estimation model can produce reliable input data for the SD regional recovery simulation model—including direct physical damage and functional losses of the built environment throughout a whole region—to determine the required amounts of restoration works at the overall region. By using detected earthquake data, regional damage can be instantly estimated. Overall damage ratios at the region are assessed based on previous research efforts on damage and loss estimations after a seismic event (Wald et al. 1999; Kanno et al. 2006; Jaiswal and Wald 2013) (See Fig. 4.1).

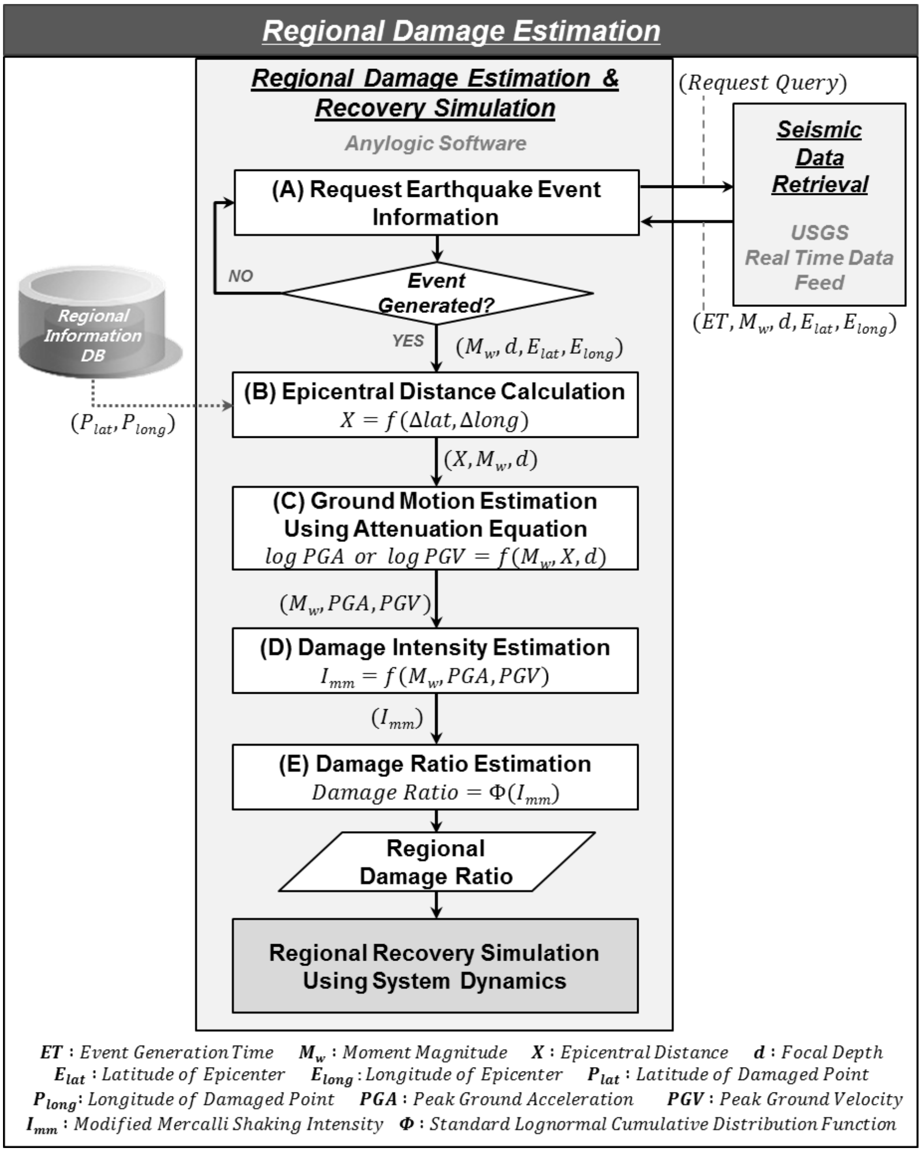



**Figure 4.1** Regional-level Damage Estimation Process

To perform recovery simulation immediately after an earthquake, the damage estimation model in the Anylogic recovery simulation federate generates a set of query information for a target seismic event at the area of interest, including the range of the longitude and latitude of the area, and the magnitude of an earthquake. This information are newly updated at every time step and sent to the USGS federate that provides real-time data feeds within 30-minutes for worldwide seismic events (USGS 2014), in order to detect significant earthquake events. After detecting a target event, the USGS federate sends target earthquake information (e.g., event time, magnitude, epicenter, and focal depth) to activate the damage estimation and recovery simulation of a damaged region (A in Fig. 4.1).

Then, by using each sub-region's locational information and target earthquake information (e.g., epicenter), the epicentral distance (i.e., distance from the damaged point to the epicenter) is calculated in accordance with the following basic equations for distance calculation between two points of the surface of the earth (B in Fig. 4.1):

$$\Delta lat = E_{lat} - P_{lat}$$

$$\Delta long = E_{long} - P_{long}$$

$$a = \sin^2\left(\frac{\Delta lat}{2}\right) + \cos(P_{lat}) \times \cos(E_{lat}) \times \sin^2\left(\frac{\Delta long}{2}\right)$$

$$X = 2R \times \tan^{-1}\left(\sqrt{a} \div \sqrt{1-a}\right) \quad \text{(Eq. 4.1)}$$

where $E_{lat}$ = the latitude of epicenter [Degree]; $E_{long}$ = the longitude of epicenter [Degree]; $P_{lat}$ = the latitude of damaged point [Degree]; $P_{lat}$ = longitude of damaged point [Degree]; $R$ = radius of the earth, approximately 6,370 [km]; and $X$ = distance between the epicenter and damaged area [km].

Based on the calculated epicentral distance ($X$) and target earthquake information (e.g., magnitude ($M_w$), and focal depth ($D$)), ground motions such as the peak ground acceleration ($PGA$) and the peak ground velocity ($PGV$) are determined based on the attenuation relationship suggested by Kanno et al. (2006) (C in Fig. 4.1). These equations have been widely used based on its simplicity because these equations use only two parameters including the moment magnitude and the epicentral distance, in accordance with the following equations:

$$\log(PGA) = a_1 M_w + b_1 X - \log\left(X + d_1 \cdot 10^{e_1 M_w}\right) + c_1 + \varepsilon_1$$

(D ≤ 30 km) (Eq. 4.2a)

$$\log(PGA) = a_2 M_w + b_2 X - \log(X) + c_2 + \varepsilon_2$$

(D ≥ 30 km) (Eq. 4.2b)

where $PGA$ = predicted peak ground acceleration [cm/sec$^2$]; $M_w$ = moment magnitude; $D$ = focal depth [km]; $a_1$ = regression coefficients (0.56); $b_1$ = regression coefficients (-0.0031); $c_1$ = regression coefficients (0.26); $d_1$ =

regression coefficients (0.0055); $\varepsilon_1$ = errors between observed and predicted values (0.37); $e_1$ = regression coefficients (0.5); $a_2$ = regression coefficients (0.41); $b_2$ = regression coefficients (-0.0039); $c_2$ = regression coefficients (1.56); and $\varepsilon_2$ = errors between observed and predicted values (0.40).

$$\log\left(PGV\right) = a_1 M_w + b_1 X - \log\left(X + d_1 \cdot 10^{e_1 M_w}\right) + c_1 + \varepsilon_1$$

(D ≤ 30 km) (Eq. 4.3a)

$$\log\left(PGV\right) = a_2 M_w + b_2 X - \log\left(X\right) + c_2 + \varepsilon_2$$

(D ≥ 30 km) (Eq. 4.3b)

where $PGV$ = predicted peak ground velocity [cm/sec]; $a_1$ = regression coefficients (0.70); $b_1$ = regression coefficients (-0.0009); $c_1$ = regression coefficients (-1.93); $d_1$ = regression coefficients (0.0022); $\varepsilon_1$ = errors between observed and predicted values (0.32); $e_1$ = regression coefficients (0.5); $a_2$ = regression coefficients (0.55); $b_2$ = regression coefficients (-0.0032); $c_2$ = regression coefficients (-0.57); and $\varepsilon_2$ = errors between observed and predicted values (0.36).

Since the Modified Mercalli Intensity scale ($I_{mm}$) has broadly been utilized to measure seismic severity, $PGA$ and $PGV$ are used to estimate $I_{mm}$ using regression relationships developed by Wald et al. (1999) (D in Fig. 4.1). As described in the previous research suggested by Wald et al. (1999), it was found that a combined regression based on *PGV* for the intensity above 7.0 and *PGA* for the intensity below 7.0 is most suitable for reproducing observed *Imm* patterns, in accordance with the following equations:

$$I_{mm} = 3.66\log\left(PGA\right) - 1.66 \quad (5 \leq I_{mm} \leq 8)$$

$$I_{mm} = 3.47\log\left(PGV\right) + 2.35 \quad (5 \leq I_{mm} \leq 10) \qquad \text{(Eq. 4.4)}$$

where $I_{mm}$ = Modified Mercalli Shaking Intensity; *PGA* = peak ground acceleration [cm/sec$^2$]; and *PGV* = peak ground velocity [cm/sec].

Finally, based on an empirical approach, damage ratios for each sub-region—which will be used as input values for the SD model—are determined by using the $I_{mm}$ values (E in Fig. 4.1). This approach suggested by Jaiswal and Wald (2013) was developed by conducting earthquake damage and loss analysis through the uses of historical data for more than 250 past worldwide earthquake events. Since this approach provided parameters for diverse damaged regions and countries through the considerations of differences among regional characteristics, this approach for estimating the loss ratio can be generally applied in worldwide post-earthquake situation, in accordance with the following equation:

$$DI\left(I_{mm}\right) = \Phi\left(\frac{\ln\left(I_{mm}\right) - \ln\left(\theta\right)}{\beta}\right) \qquad \text{(Eq. 4.5)}$$

where $\Phi$ = the standard lognormal cumulative distribution function; $DI(I_{mm})$ = the damage ratio at a given Modified Mercalli Shaking Intensity ($I_{mm}$) scale; $\theta$ = the mean of the natural logarithm of shaking intensity; and $\beta$ = the standard deviation.

Although there exist numerous research efforts and equations for assessing ground motion, damage intensity, and the loss ratio, the reason that the above methods are applied to this study is due to their wide range of uses. For instance, the U.S. Geological Survey website provides damage estimation results by using the above-mentioned procedures. By using the diverse geographical information of numerous sub-regions in the "*DamagedRegion*" class, this model can be utilized to estimate different damage patterns of multiple sub-regions. The whole model can be found in [Appendix B-I].

***Model Test***

To evaluate this model's damage-estimating capability, actual disaster information (e.g., the M 9.0 Earthquake of Tohoku of March 11, 2011) is applied to simulate both ground motions and damage intensities at the damaged Tohoku region, including three sub-regions (i.e., the Iwate, Miyagi, and Fukushima prefectures). Table 4.1 shows the input values for regional-level damage estimations including actual earthquake information (e.g., the epicentral locations, magnitude, and focal depth) and locational information of each sub-region. When the locational information of the middle of each sub-region as well as earthquake information are applied to the model, epicentral distances of the Iwate, Miyagi, and Fukushima prefectures are calculated as 188.6 km, 131.0 km, and 177.3 km, respectively.

**Table 4.1** Input Values for Testing Regional Damage Estimation Model

| Categories | Earthquake Parameters | | Locations | | |
|---|---|---|---|---|---|
| | $M_w$ | Focal Depth | Latitude | Longitude | Epicentral Distances |
| **Seismic Event** | | | | | |
| 2011 Earthquake of Tohoku | 9.0 | 29 km | 38.297 | 142.373 | 0.0 km |
| **Sub-Regions** | | | | | |
| Iwate Prefecture | - | - | 39.704 | 141.153 | 188.6 km |
| Miyagi Prefecture | - | - | 38.269 | 140.872 | 131.0 km |
| Fukushima Prefecture | - | - | 37.761 | 140.467 | 177.3 km |

By using the input values in Table 4.1, the simulation results of the average *PGAs* and $I_{mms}$ of the middle of these three sub-regions in Table 4.2 show observable similarities compared to their actual range of *PGAs* and $I_{mms}$ (Eidinger et al. 2012). For instance, 200–1500 gal (cm/sec2) of *PGAs* and 8.5–11.0 of $I_{mms}$ were found throughout the most severely damaged Miyagi Prefecture. The model of this study simulates the results of 602 gal of *PGA* and 10.3 of $I_{mm}$, which largely correspond to average values of actual *PGAs* and $I_{mms}$. Since disaster intensities vary throughout each region due to wide range of regional area, different locations within each sub-region are also applied in the simulations to capture the ranges of disaster intensities among sub-regions. The simulation results of the range *PGAs* and $I_{mms}$ of these three sub-regions are also similar to the actual range values of *PGAs* and $I_{mms}$.

**Table 4.2** Test Results of Regional Damage Estimation Model

| Parameters | Sub-regions | Actual Data* (Eidinger et al. 2012) | Simulation Results | |
|---|---|---|---|---|
| | | | Values in the Middle Point** | Range Values of 10 Different Points*** |
| **Peak Ground Acceleration (*PGA*)** | Iwate Prefecture | 100–800 gal [cm/sec$^2$] | 336 gal [cm/sec$^2$] | 141–809 gal [cm/sec$^2$] |
| | Miyagi Prefecture | 200–1500 gal [cm/sec$^2$] | 602 gal [cm/sec$^2$] | 250–1416gal [cm/sec$^2$] |
| | Fukushima Prefecture | 100–1000 gal [cm/sec$^2$] | 375 gal [cm/sec$^2$] | 157-912 gal [cm/sec$^2$] |
| **The Modified Mercalli Intensity Scale ($I_{mm}$)** | Iwate Prefecture | 7.5–11.0 | 9.7 | 7.1–10.6 |
| | Miyagi Prefecture | 8.5–11.0 | 10.3 | 8.7–11.1 |
| | Fukushima Prefecture | 8.0–11.0 | 9.8 | 6.9–10.9 |

* Due to the wide ranges of sub-regions, disaster intensities vary throughout sub-regions.

** Simulations are conducted with the locational information of the middle point of each sub-region.

***10 simulations are conducted with different 10 locational information within each sub-region in order to capture the ranges of disaster intensities.

### 4.1.2 Regional-Level Recovery Analysis Model Using SD

***Model Overview***

As shown in Fig. 4.2, post-disaster recovery systems involve repairing and rebuilding houses, commercial buildings, pathways, and critical facilities/infrastructures, such that populations are provided with residential, commercial, transportation, and public services, as well as products (Olshansky et al. 2012).

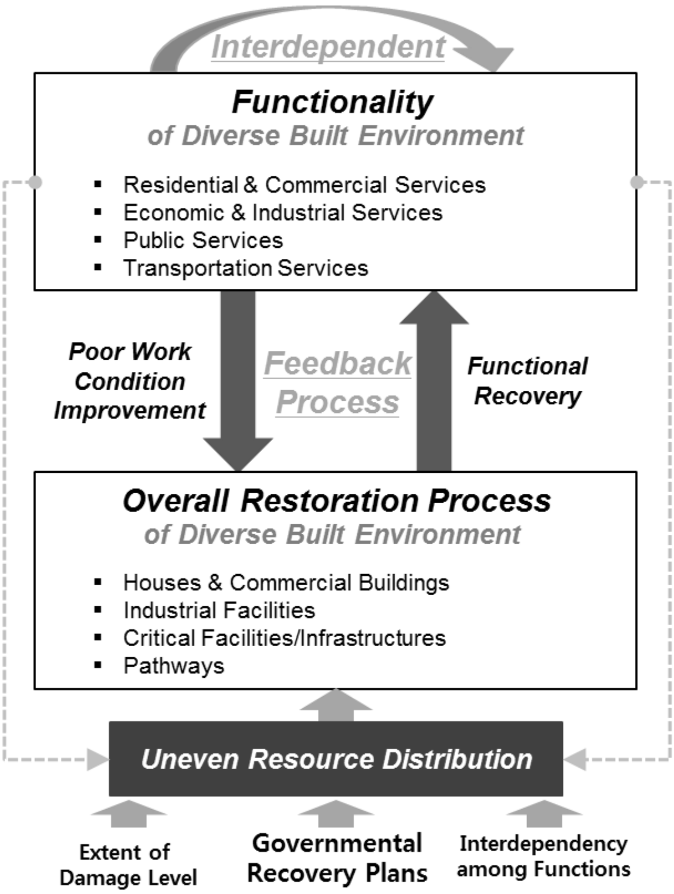



**Figure 4.2** Overview of Regional-Level Recovery Analysis

The analysis of dynamic features of regional-level interdependent multiple recovery processes using SD thus comprehensively deals with various facilities/infrastructures and their functions in the overall damaged region. Since recovery situations change over time under the influence of damage differences among sub-regions as well as multiple restoration efforts and diverse functional recoveries of the damaged built environment, the main focus of the SD model is to understand: (a) complex and interdependent restoration efforts among diverse facilities/infrastructure types and among various sub-regions, and (b) multiple feedback processes between regional restoration works and functionality recoveries, as shown in Fig. 4.2.

To capture the differences between restoration processes among diverse types of facilities throughout a whole region in detail, the SD model in this study uses two-dimensional (2-D) array variables including sub-elements for several sub-regions and for diverse facility/infrastructure types into the "*[Region]*" and "*[Type]*" dimensions, respectively, because the recovery processes can vary according to facility types and the sub-regions. For instance, resource availability for each type of facilities in the region (e.g., housing, public buildings, transportation infrastructures) can be different under the program-based governmental restoration budget distribution plans (e.g., housing development programs and road development programs). In addition, resource delivery capabilities for facility restorations among regions depend on mostly adjacent road damage, rather than overall region's damage.

Therefore, according to the facility/infrastructure's functions, facility types in the "*[Type]*" dimension are categorized into general facilities such as residential and commercial buildings *[Type R]*, critical facilities and infrastructures (e.g., administration offices, police stations, schools, hospitals, and so forth) that provide core public services *[Type C]*, industrial facilities that provide industrial services and produce construction equipment and materials *[Type I]*, transportation infrastructures such as roadways, railroads, and bridges *[Type T]*, as described in Table 4.3 (Jianxue and Jie 1996; Shoji and Toyota 2009). In addition, divisions of the broad range of the devastated area into several sub-regions in the "*[Region]*" dimension are made according to differences in damage. The subdivision of region and facility type can be useful for an in-depth analysis of recovery processes by reflecting more relevant damage and recovery situations.

**Table 4.3** Categorization of Facilities/Infrastructures

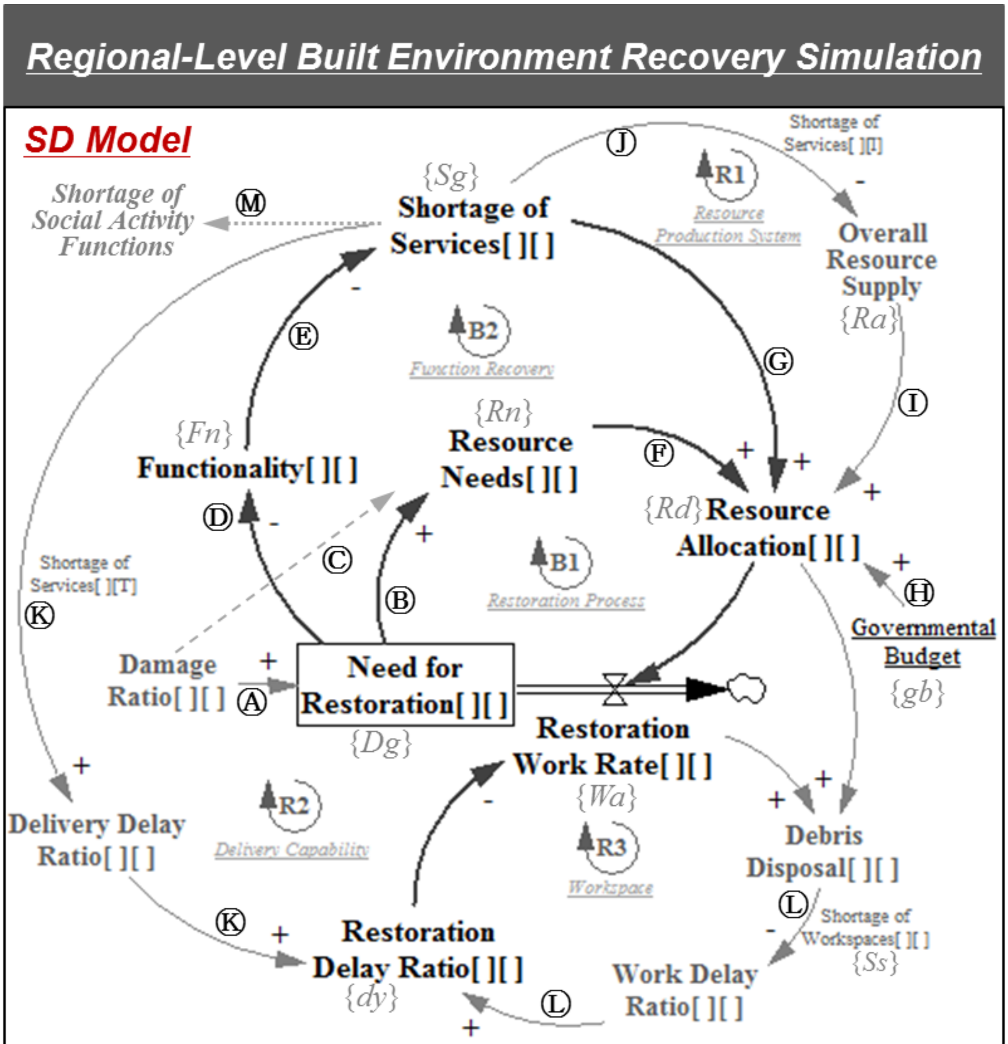
| **Array [Type]** | **Facility Type** | **Facilities Functions** |
|---|---|---|
| General<br>[ R ] | General facilities including<br>residential and commercial buildings | Residential and<br>commercial services |
| Critical<br>[ C ] | Critical facilities<br>(e.g., schools, hospitals, …) | Public services |
| Industrial<br>[ I ] | Industrial facilities and resource<br>production systems | Industrial and<br>economic functions |
| Transportation<br>[ T ] | Transportation system<br>(e.g., roadway, railroad, …) | Transportation<br>services |

***Model Descriptions***

According to Holguin-Veras and Jaller (2012), the problems in resource supply after a disaster include the excessive needs for resources, their temporal evolution, complex interactions among the dozens of supply chains, timing and types of commodities requested, and their relative importance in utilization (Orabi et al. 2009). Therefore, the model in this study focuses on how resource supply problems are generated and how limited resources are utilized over time.

As shown in Fig. 4.3, the disaster event generates considerable needs for facility/infrastructure restorations at the whole region (A in Fig. 4.3). In this situation, the resources available to perform repair/reconstruction works (2-D array variable "*Restoration Work Rate*" in Fig. 4.3) are limited because all kinds of overall damaged built environments (i.e., damaged facilities/infrastructures at the whole region) require a huge amount of materials, personnel, and equipment (B in Fig. 4.3). Although overall structural damage (2-D array variable "*Need for Restoration*" in Fig. 4.3)—which can be determined as the number of damaged facilities multiplied by their damage severity—is the same, resource problems can differ according to both the number of damaged facilities and the degree of damage (2-D array variable "*Damage Ratio*" in Fig. 4.3) in each facility, respectively. For instance, when only small numbers of facilities are severely damaged, resources such as construction equipment are less limited because only small numbers of projects require them. In opposite, when a large number of facilities are damaged (even though they are all slightly damaged) and consequently many projects simultaneously require restoration resources, resource problems can be more severe (C in Fig 4.3). These excessive needs for resources continue until the restorations of the built environment are completed, though they can gradually be alleviated as the restoration operation progresses (B1 Loop in Fig. 4.3).

Since damaged facilities/infrastructures may lose their functionality (2-D array variable "*Functionality*" in Fig. 4.3) after a disaster due to interrupted operations of facilities (D in Fig. 4.3), the delayed restorations within limited resources can cause lengthy shortages in the services from facilities' functions (2-D array variable "*Shortage of Services*" in Fig. 4.3) (E in Fig. 4.3). In order to effectively and rapidly recover the interrupted functionality of facilities where available resources are limited, resource allocations and distributions to multiple restorations can mainly be determined by the shortage level and relative importance among functions of diverse facilities/infrastructures (B2 Loop in Fig. 4.3). An associated interdependency among facility's functions can thus be regarded as the most important issue for understanding overall recovery situations and for implementing recovery plans.




**Array [Facility] [Region]**

| Region | Facility Type | | | |
|---|---|---|---|---|
| | General [ R ] | Critical [ C ] | Industrial [ I ] | Transportation [ T ] |
| | Functions | | | |
| | Residential & Commercial Services | Public Services | Production Capabilities | Transportation Services |
| [ A ] | … | … | … | … |
| [ B ] | … | … | … | … |
| … | … | … | … | … |

**Figure 4.3** SD Model Description

Although both structural and non-structural damage (e.g., operating system shut-down) can affect the functionality losses of facilities (Pachakis and Kiremidjian 2004), facilities' functionality losses in the long term (e.g., 3–10 years of intensive restoration periods) are mainly influenced by structural damage because non-structural damage such as surrounding lifeline systems are already treated at an early recovery phase (Shoji and Toyota 2009).[14] After non-structural damage is recovered, the long-term estimation of facilities' functionality losses (*Fn*) in a wide range of area thus tends to be determined as the amount of a facility's physical damage (*Dg)* divided by the total damage ($Dg_{max}$), as shown in Eq. 4.6 (Davis 2013; Jaiswal and Wald 2013). These losses can be directly reflected in the shortage of related services (*Sg*) in the damaged region without enough reserve rates for such services, as determined in Eq. 4.7. Based on these pieces of evidence from existing research efforts, facility functionality and the shortage of a facility's services are defined in the model according to the following equations:

$$Fn[Region][Type] = 1 - \frac{Dg[Region][Type]}{Dg_{\max}[Region][Type]} \qquad \text{(Eq. 4.6)}$$

$$Sg[Region][Type] = 1 - Fn[Region][Type] \qquad \text{(Eq. 4.7)}$$

[14] In past disaster recovery situations such as the Hurricanes Katrina and Rita and the 2011 Earthquake of Tohoku, it was found that non-structural damage (e.g., an operation system breakdown and power blackout) was generally repaired within several months with fully intensive and short-term emergency recovery efforts made by the central government (Pike 2007; RA 2011; RA 2013).

where $Dg[Region][Type]$ = structural damage of each facility type in each region; $Fn[Region][Type]$ = functionality level (%) of each facility type in each region; and $Sg[Region][Type]$ = shortage level (%) of each facility type's function or functionality loss in each region compared to pre-disaster facilities' functionality level (100%).

As shown in the SD model structure, the differences in the restoration work rate of a damaged facility/infrastructure and its functional recoveries depend on the amount of resource distribution (2-D array variable "*Resource Allocation*" in Fig. 4.3), both to various types of facilities and to diverse sub-regions within limited resources. This is an outcome of resource competitions caused by: (a) different resource requirements (2-D array variable "*Resource Needs*," F in Fig. 4.3) according to the structural damage levels of different sub-regions; (b) the various functional loss levels among facility types and consequent a need for services at the entire region (or a shortage of services) that each facility/infrastructure can provide (2-D array variable "*Shortage of Services*," G in Fig. 4.3); and (c) governmental recovery plans such as budget distribution plans (H in Fig. 4.3) that are implemented with a consideration of the associated interdependency and relative importance of each type of facility functions as well as political issues and public opinions (Kovel 2000; Shoji and Toyota 2009). In the context of the lack of resources (I in Fig. 4.3), the recovery prioritization to each type of facility/infrastructure is inevitable. The resource distribution process thus varies, and then resources that had already been allocated to different restoration projects are continuously adjusted with dynamic changes in a shortage of services, in order to avoid extreme shortage problems (Orabi 2010). In reality, recovery budget spending was unevenly distributed to diverse recovery programs (e.g., housing programs, infrastructure, and economic development) and to numerous sub-regions by considering relative functional shortages (*Sg*) of each facility type's functions and in each sub-region with additional public assistances from central recovery agencies (*gb*) (e.g., FEMA and Recovery Agencies of Japan), as shown in Eq. 4.8 (Pike 2007; RA 2011; RA 2013). Therefore, uneven recovery progress was found because restoration programs for some types of facilities had problems in acquiring enough resources (*Rd*) according to unevenly distributed budgets, as shown in Eq. 4.9 (Rowley 2007; RA 2013). With empirical evidences from the actual recovery plans after both the 2011 Earthquake of Tohoku and Hurricanes Katrina and Rita, resource allocation is applied to the model in accordance with the following equations:

$$\begin{aligned}\delta[Region][Type] &= \frac{Sg[Region][Type]}{\sum Sg[Region][Type]}\cdot(1-BC)\\ &+gb[Region][Type]\cdot BC\end{aligned} \quad \text{(Eq. 4.8)}$$

$$\begin{aligned}&Rd[Region][Type]=\\ &MIN\left(Rn[Region][Type], Ra\cdot\delta[Region][Type]\right)\end{aligned} \quad \text{(Eq. 4.9)}$$

where $\delta[Region][Type]=$ resource distribution ratio (%) for the restoration of each facility type; $Sg[Region][Type]=$ shortage level (%) of each facility type's function; $BC=$ a ratio (%) of central governmental recovery budget to total recovery budget including central- and local-governments and private sectors; $gb[Region][Type] =$ governmental budget distribution (%) for the restoration of each facility type among central-governmental recovery programs; $Rd[Region][Type] =$ resource allocation for the restoration of each facility type; $Rn[Region][Type] =$ needs for resources for restoration of each facility type; and $Ra =$ available resources.

In this situation, functional interdependencies among different facilities determine feedback processes between the diverse functionality recoveries (2-D array variable "*Shortage of Service*" in Fig. 4.3) and multiple restoration operations (2-D array variable "*Need for Restoration*" in Fig. 4.3). The consideration of interdependencies among diverse types of facilities and infrastructures can thus be helpful in understanding how and to what extent the overall restoration efforts can be effective. Within limited resources, in particular, damage to industrial facilities and ensuing functionality losses can interrupt construction resource productions (J in Fig. 4.3). Due to problems in new resource supplies (I in Fig. 4.3), most restoration works can be delayed (R1 Loop in Fig. 4.3). In this situation, the early resource allocation to industrial facility restoration projects can alleviate delays in the overall restoration operations by rapidly re-activating the resource supply system.

The damage to roadways, bridges, and railroads may also result in functionality losses for transportation systems. As a result, the resource supply and debris disposal system may be severely interrupted due to delays in deliveries (K in Fig. 4.3), which in turn can lead to overall restoration project delays (R2 Loop in Fig. 4.3) (Holguin-Veras and Jaller 2012).

On the other hand, when debris disposal capability is overwhelmed by excessive debris, the deployment/delivery of resources and the supply of public services are delayed (Swan 2000). Excessive debris may also obstruct reconstruction activities via a lack of spaces (R3 Loop in Fig. 4.3) because building construction requires space to move, store, and fabricate materials, and to perform work (L in Fig. 4.3) (Riley and Sanvido 1995; Chua et al. 2010). Debris clearance, removal, and disposal efforts thus need to be planned and coordinated prior to other reconstruction activities to alleviate these problems. Based on these theoretical evidences and the basic equations on production rates in the field of construction, the restoration work rate (*Wa*) and work delay ratio (*dy*) are defined according to the following equations:

$$\frac{Wa[Region][Type]}{We[Region][Type]} = (1 - dy[Region][Type]) \cdot \frac{Rd[Region][Type]}{Rn[Region][Type]} \quad \text{(Eq. 4.10)}$$

$$dy[Region][Type] = Sg[Region][T] \cdot Ss[Region][Type] \quad \text{(Eq. 4.11)}$$

where $Wa[Region][Type] =$ actual restoration work rate of each facility type in each region; $We[Region][Type] =$ expected restoration work rate of each facility type in each region (optimistic); $dy[Region][Type] =$ work delay ratio for the restoration of each facility type in each region; $Rd[Region][Type] =$ resource allocation for the restoration of each facility type in each region; $Rn[Region][Type] =$ needs for the resources for the restoration of each facility type in each region; $Sg[Region][T] =$ shortage level (%) of transportation services in each region; and $Ss[Region][Type] =$ shortage level (%) of workspaces for the restoration of each facility type in each region.

An understanding of interdependencies among diverse facilities' functions is required not only to improve restoration work processes but also to improve the efficiency in recovering daily lives of populations (Shoji and Toyota 2009). In particular, the reconstruction of residential buildings needs to be accompanied by a restoration of commercial buildings and critical facilities to recover communities' social activity functions by providing not only dwellings but employments, educations, medical services, and so forth. A shortage of social activity functions, which can be defined as the average value of all types of shortage of services, is thus applied to the model in accordance with the following formula (M in Fig. 4.3):

$$Sg_{SA} = Average(Sg[Region][R], Sg[Region][C], Sg[Region][I], Sg[Region][T]) \quad \text{(Eq. 4.12)}$$

where $Sg_{SA}$ = shortage level (%) of social activity functions; and

$Sg[Region][] =$ shortage level (%) of residential/commercial services [R], public services [C], industrial services [I], and transportation services [T].

Since both the central- and local-governmental recovery planning has a significant effect on overall recovery efforts due to the need for great amounts of expenditures (Karlaftis et al. 2007), an understanding of post-disaster recovery plans and their effects on recovery systems is essential. In reality, central governments played leading and supporting roles for the purpose of advancing the functional recovery of the built environment and revitalizing local communities with consistent and comprehensive plans within limited resources, time, and budget. For example, in the aftermath of catastrophic disasters such as the 2011 Earthquake of Tohoku in Japan, the national government was mainly responsible for managing the overall disaster management cycle by implementing all possible measures to support restoration efforts, with the support of local governments and communities (RA 2011; RA 2013). After the Hurricanes Katrina and Rita in the U.S., the state government led the overall recovery process and supported the local governments that have the primary role of the community's recovery, under the federal government's support (Pike 2007; FEMA 2011).[15]

In detail, to recover and improve functions of the built environment in the recovery phase, the central government implemented diverse initiatives

[15] Despite the gap of the detailed role between Japan and the U.S., both central governments played the most important role in overall recovery efforts with the deployment of a huge amount of public funds and recovery resources.

that include: (a) special measures (e.g., relaxed regulation and procedures) and special programs (e.g., temporary debris clearance, temporary housing, land use policies); and (b) a grant for reconstruction projects proposed by the municipalities, local communities, and collaborative private sectors (RA 2013; Berke et al. 2014). The local governments and communities thus not only autonomously implemented recovery plans using their own budget but also provided support to government-centered core projects. In addition, they also competed for financial grants and resources that are distributed and periodically adjusted under the governmental recovery budget plans, which can vary according to the region and type of facilities.

As a result, the overall restoration efforts were not only determined by internal resource competitions but were also significantly affected by a higher-level recovery plans with both positive and negative effects. Table 4.4 describes how governmental plans (external variables in the model) affect both recovery patterns and model structures with a focus on special recovery programs and restoration budget distribution plans. For example, a temporary debris movement plan can be effective in reducing disturbances to resource deliveries and reconstruction works by temporarily moving excessive debris to temporary storage area. Although this plan requires additional time and efforts at an initial stage, it eventually can alleviate the emergent requirements for debris disposal when the debris disposal capability is overwhelmed. Other policy effects on the restoration system can be found in Table 4.4.

**Table 4.4** Effects of Governmental Plans on the Recovery System

| Governmental Recovery Plans (Independent Variables in the Model) | | System Changes (Dependent Variables in the Model) | +/- | Descriptions of the Governmental Policy Effects |
|---|---|---|---|---|
| Emergency Response Plans | Temporary debris movement | Shortage of workspaces | - | · A plan for debris movement to temporary storage area is effective to reduce shortage of workspace |
| | | Restoration work rate [Region][Type] | + | · Debris movement requires time for temporarily moving debris at an initial stage |
| | Temporary housing development | Shortage of services [Region][R] | - | · A plan for temporary housing is effective to temporarily solve housing problems |
| | | Restoration work rate [Region][Type] | + | · Temporary housing requires time for installing temporary housing at an initial stage |
| Restoration Plans | Land readjustment / Household relocation | Restoration work rate [Region][Type] | + | · A plan for household relocation requires time for spatial planning |
| | | Shortage of workspaces | - | · Household relocation is effective to reduce shortage of workspace by avoiding congestion areas overwhelmed by excessive debris generation |
| | Budget distribution to restoration projects | Resource allocation [Region][Type] | +/- | · Budget distribution plans can reduce/increase restoration resources for each type of facility/infrastructure |

Since the regional-level recovery processes and recovery plans significantly affect each damaged facility restoration projects, an understanding of damage and recovery situations of facility's surrounding regions can assist more reliable facility-level restoration planning as well. The need for this understanding is because inefficient resource supplies and poor work environments (i.e., restoration conditions) may detrimentally impact facility restoration operations in chaotic post-disaster recovery situations where resource and time limitations are strongly imposed and where surrounding areas are severely devastated (Orabi et al. 2010; Olshansky et al. 2012). Since surrounding restoration conditions of damaged facilities change over time according to diverse damage patterns and consequent restoration works as well as functional recoveries of the damaged built environment, this model can also be utilized to understand the extremely increased uncertainties and disturbances of facility restoration projects caused by unfavorable external conditions.

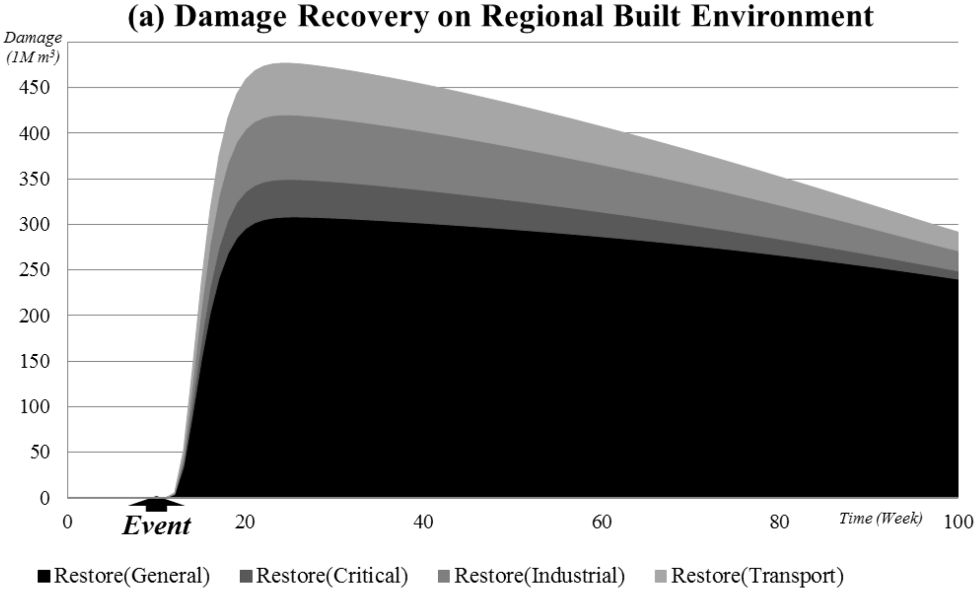
From this model, the recovery progress of each type of facility in each region can be determined by the ratio between the resource needs and the amount and timing of the resource allocations (i.e., resource supply ratio including materials and workforces). Work effectiveness influencing the restoration work rate can be estimated by the shortage level of both transportation capabilities and workspaces (i.e., delivery and work delay ratio).

### *Model Test*

By using the causal loop diagrams and equations presented above, this study constructs a simulation model by defining quantitative relationships among variables. This is based on the existing research efforts and basic equations on the construction processes as well as the supply and demand of resources/services that are widely used in SD modeling. The whole model, including the descriptions of model structures and equations, can be found in [Appendix B-II].

To understand the overall recovery situations and to test the model behaviors, the model is simulated by applying a specific actual damage and recovery scenario from the 2011 earthquake of Tohoku for a case study. In detail, this study utilizes actual disaster, facility, region, and enforced policy information as input values to the model, such as damage intensities of sub-regions (i.e., the Iwate, Miyagi, and Fukushima prefectures), the amount and the scale of each type of facility/infrastructure (general, critical, industrial facilities, and transportation infrastructure) in each sub-region, different seismic importance factors among numerous types of facilities/infrastructures, the amount of restoration resource (budget) input, and the enforced budget distribution plans (Taranath 2005; E-Stat 2013; NPA 2013; RA 2013). Since recovery processes of the whole region are affected by the degrees of damage and the restoration works of each facility type and sub-region, the differences and changes of the damage and restoration progresses among facility types and sub-regions are captured for a more reliable analysis of regional-level recovery processes. By conducting a comparative analysis between simulation results and actual restoration data, simulation results of different damage and recovery processes by facility types (general, critical, industrial facilities, and transportation infrastructure) are tested. Then, simulation results of different damage and recovery processes among sub-regions are further tested.

Fig. 4.4 represents the simulation results of the damage generation (Fig. 4.4a), and the restoration process (Fig. 4.4b) by facility types in the whole Tohoku region of Japan. In addition, Table 4.5 describes the detailed results of the model behavior test with a focus on different damage and recovery processes by facility types. The damage in Fig. 4.4(a) is calculated as a total volume of damaged structures [$m^3$] by considering total floor area, average height, and the number of facilities/infrastructures as well as the regional damage ratios and seismic importance factors. Due to the overwhelming number of residential and commercial buildings (general facilities), there is a great deal of physical damage to general facilities (Fig. 4.4a). Despite difficulties in accurate estimation due to the uncertain nature of the post-disaster situation, these damage simulation results largely correspond to actual damage patterns from the 2011 earthquake in the Tohoku region, as shown in Table 4.5(a). For instance, it was found in both the actual data and the simulation results that damage to general facilities was approximately seven times greater than that of critical facilities (NPA 2013).



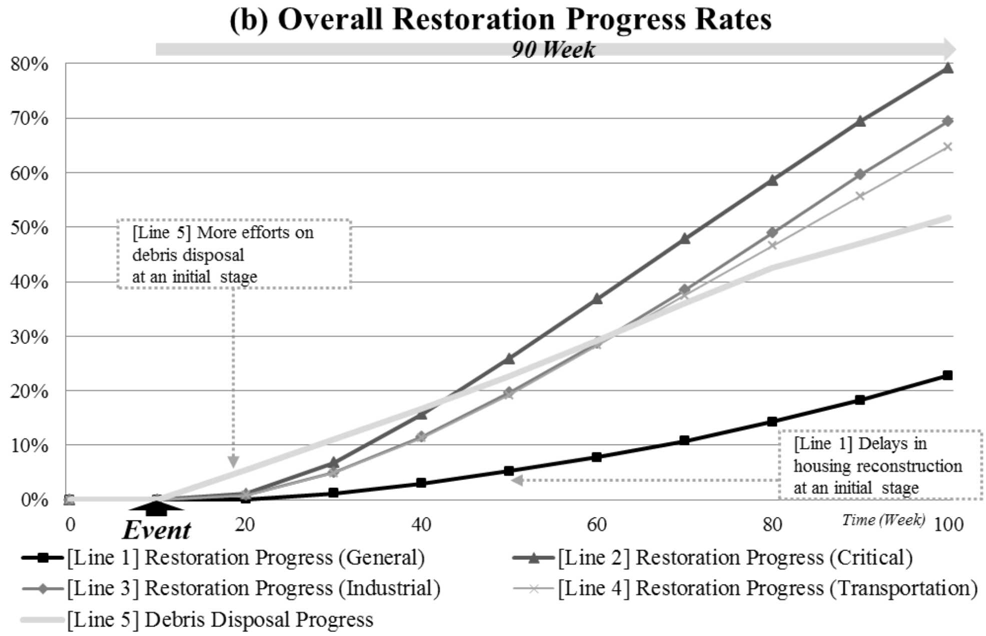




**Figure 4.4** Simulation Results: Regional Damage and Recovery according to Facility Types

**Table 4.5** SD Model Behavior Test Results of Different Damage and Recovery Processes by Facility Types

| Restoration Type | (a) Relative Damage (Damage / Total Damage) | | (b) Restoration Progress (90 Weeks After Disaster) | |
|---|---|---|---|---|
| | Actual Data (NPA 2013; RA 2013) | Simulation Results | Actual Data (RA 2013) | Simulation Results |
| Residential/Commercial (General) Buildings | 60%–65% | 64% | 30%–40% | 24%* |
| Critical Facilities | 5%–10% | 9% | 80%–90% | 79% |
| Industrial Facilities | 15%–20% | 15% | 70%–80% | 70% |
| Transportation Infrastructures | 10%–15% | 12% | 50%–80% | 65% |
| Debris Clearance and Disposal | - | - | 50%–55% | 52% |

* The effects of public and temporary housing development as well as relief efforts in actual data are ignored in the simulation results.

The simulation results of the restoration processes of Fig. 4.4(b) are also similar to actual restoration progress data investigated by the Reconstruction Agency of Japan (RA) (2013). As described in Table 4.5(b), the results of a model behavior test show that there are observable similar behaviors in the differences and degrees of the recovery progresses among diverse restoration works. For instance, housing reconstruction after the 2011 earthquake of Tohoku was delayed due to the lack of resources for restoring massive numbers of residential buildings. The actual data show that housing reconstruction was only 30%–40% completed as of December 2012 (90 weeks after the event), while social and public structure restorations such as schools and hospitals was 80%–90% completed. The model of this study simulates the results of a 24% completion of housing reconstruction (i.e., general buildings, Line 1 in Fig. 4.5b) and a 79% completion of social and public structure restorations (i.e., critical facilities, Line 2 in Fig. 4.5b) when utilizing actual data. The simulation results of the recovery processes of industrial facilities (Line 3 in Fig. 4.5b), transportation systems (Line 4 in Fig. 4.5b), and debris disposal (Line 5 in Fig. 4.5b) are also similar to the actual data. Although there is a gap between the simulation results and the actual data to some degree, particularly in the simulation result of housing reconstruction progress, it is found that this gap exists because the effects of the massive amounts of relief efforts and the migration of refugees to safe regions in reality are beyond the scope of simulation.

In addition, dynamic changes in recovery processes can be identified in the simulation results. For instance, more efforts on debris disposal at an initial restoration stage can be found (Line 5 in Fig. 4.5b) due to significant and negative effects of excessive debris on restoration works. On the other hand, the early stage of housing reconstruction is inevitably more delayed compared to its late stage (Line 1 in Fig. 4.5b) because the recovery of other core services is more urgent. These different dynamic changes in recovery progress among facility types were also found in an actual case, which shows similar behaviors over time with simulation results (RA 2013).

Table 4.6 describes the detailed results of the model test with a focus on different damage and restoration processes among three sub-regions including the Iwate, Miyagi, and Fukushima prefectures. It is found that simulation results of damage and recovery processes of each sub-region show observable similarities compared to their actual data (NIRA 2013; NPA 2013).

In addition, Fig. 4.5 shows behavior test results of recoveries of facilities'/infrastructures' functionalities at the sub-regions by using actual recovery data investigated by Japan's National Institute for Research Advancement (NIRA) (2013). By conducting a comparative analysis between actual data (See Fig. 4.5a) and simulation results (See Fig. 4.5b), the results of a model behavior test show that there are observable similar behaviors in the functional losses of facilities and their recoveries over time among each sub-region. In particular, it is found that the most severe damage and the ensuing tremendous losses of facilities' functions on the Miyagi Prefecture are found both in simulation results and actual data. In addition, both graphs of the Miyagi Prefecture show faster recovery processes than the other two sub-regions because more efforts on restorations are made due to more urgent needs for recoveries. Although there exists a gap between the simulation results and the actual data (e.g., more severe damage at an early recovery phase and a later recovery of Fukushima Prefecture in the actual data), it is found that this gap exists because non-structural damage (e.g., power blackout and radiation leakage in Fukushima) is beyond the scope of simulation.

**Table 4.6** SD Model Behavior Test Results of Different Damage and Recovery Processes among Sub-Regions

| Sub-Regions | (a) Relative Damage (Damage / Total Damage) | | (b) Restoration Progress (90 Weeks After Disaster) | |
|---|---|---|---|---|
| | Actual Data (NPA 2013) | Simulation Results | Actual Data (NIRA 2013) | Simulation Results |
| Iwate Prefecture | 14 % | 16 % | 3.5 % Recovered | 3.8 % Recovered |
| Miyagi Prefecture | 62 % | 58 % | 5.0 % Recovered | 4.4 % Recovered |
| Fukushima Prefecture | 24 % | 26 % | 4.6 % Recovered | 4.0 % Recovered |

From this analysis, the SD model is effective in capturing the dynamic features of diverse recovery processes, and is also effective in explaining why such behaviors have been generated. In particular, since the model reflects functional interdependencies among the built environment and governmental plans, the model can be utilized to reasonably analyze the overall restoration efforts by considering diverse recovery situations. The analytical capability of SD can also be useful in reasonably analyzing surrounding restoration conditions of facility-level restoration projects, which itself is helpful in implementing valid restoration project plans.

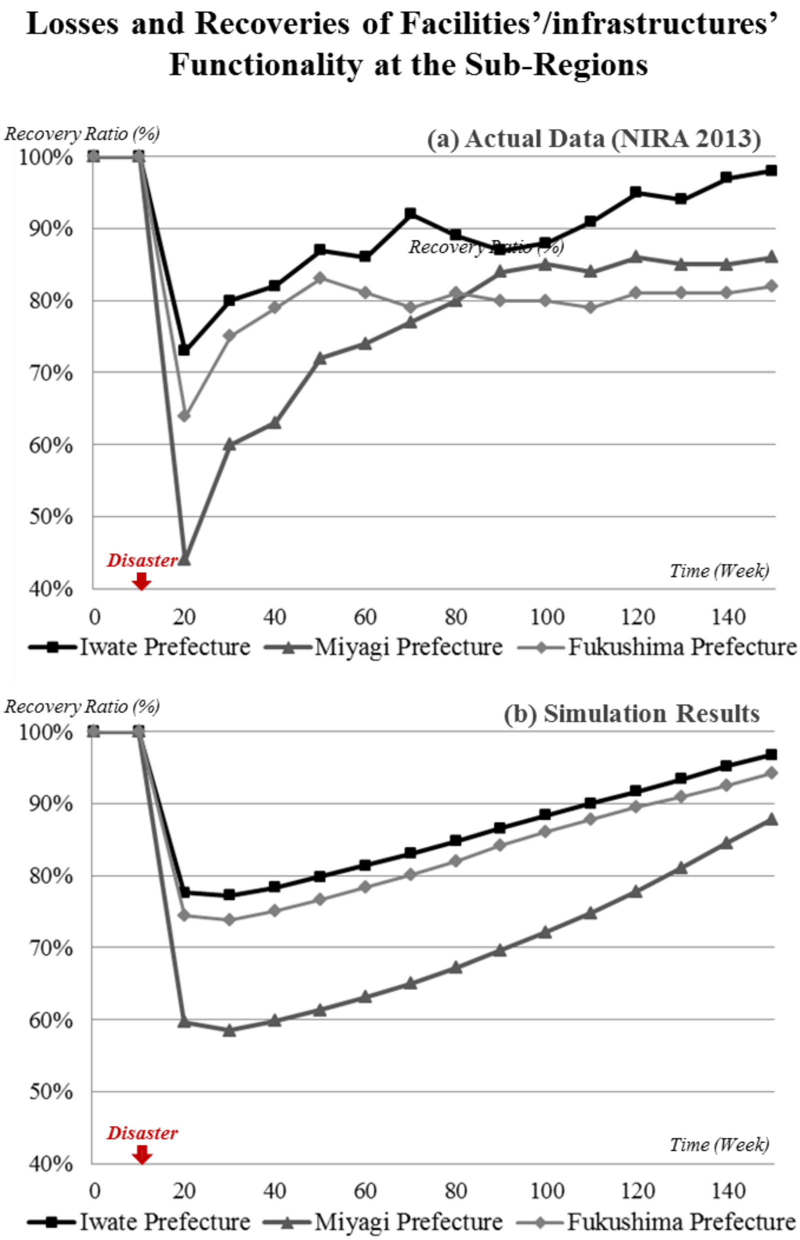



**Figure 4.5** SD Model Behavior Test Results of Facilities' Functionality Recovery among Sub-Regions

## 4.2 Facility-Level Damage and Restoration Simulation

### 4.2.1 Facility Damage Assessment Model

***Model Descriptions***

In this section, this study develops a facility damage assessment model by the integrated uses of the USGS seismic data retrieval federate and the OpenSees structural response simulation federate, in order to assist more rapid and reliable analysis of facility restoration operations. The Anylogic federate plays a main role to assess facility damage as well as analyze facility repair/reconstruction operations using DES in the aftermath of a seismic event. Therefore, this study offers the description of facility damage assessment model in the Anylogic federate, as shown in Fig. 4.6. In this damage assessment model, the USGS federate and the OpenSees federate are integrated with the Anylogic federate to conduct prompt analysis by detecting an earthquake event and analyzing structural responses according to ground shaking, respectively.

As described in previous chapter, Ch. 4.1.1., the USGS federate provides target earthquake information (e.g., event time, magnitude, epicenter, and focal depth) to activate the damage assessment and restoration simulation of a target facility (A in Fig. 4.6). In general, structural response analysis using the OpenSees requires ground motion and shaking data (i.e., accelerograms) at the damaged point. In the case that the ground shaking of a facility can be directly detected by using an accelerometer—which is a device for real-time measurement of seismic accelerations—this ground shaking data can solely be utilized for the structural response analysis without detecting other earthquake information, as shown in Fig. 4.7.

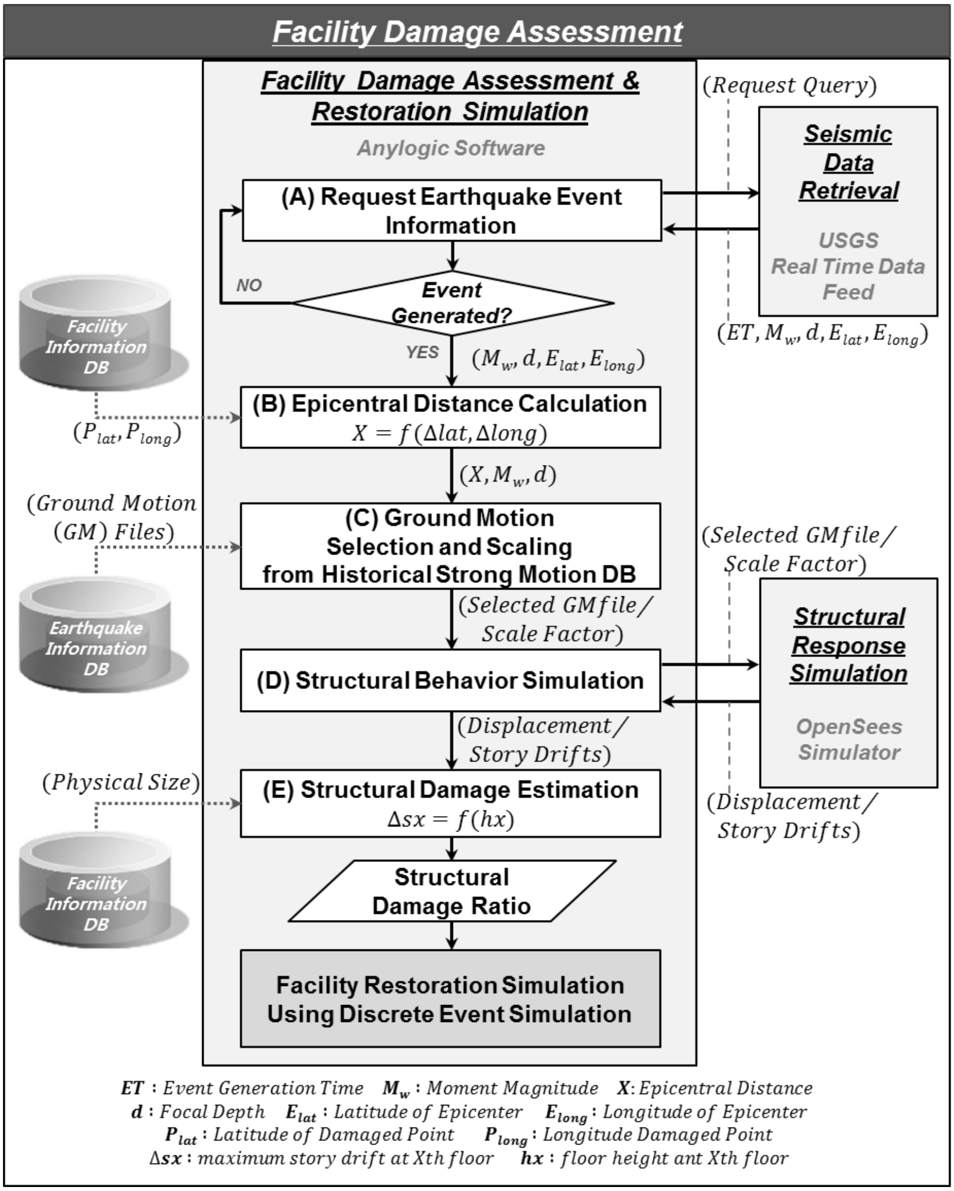



**Figure 4.6** Facility-level Damage Assessment Process

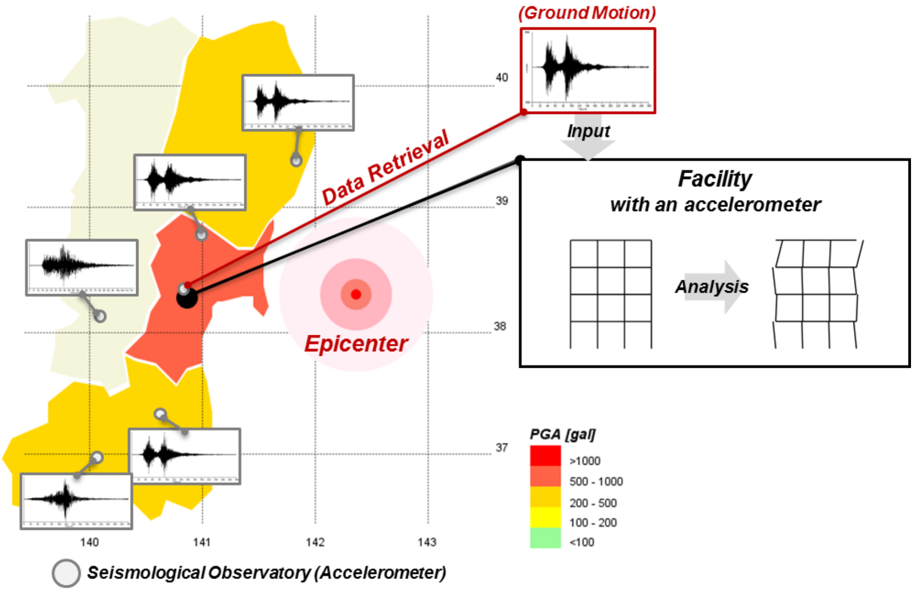



**Figure 4.7** Concept of Damage Assessment for the Facility with an Accelerometer

However, it is hard for most common facilities to obtain accelerograms in real time because only critical facilities or strong motion sensing stations are equipped with accelerometers. To solve this problem, there exist numerous historical ground motion databases for the purpose of analyzing structural responses using the historical data and empirical approaches.[16] The most suitable accelerograms for the target facility can be selected from this database and then scaled with a consideration on differences between the

16 The examples of ground motion databases include "the Pacific Earthquake Engineering Research (PEER) Center Strong Ground Motion Database" and "the Center for Engineering Strong Motion Data (CESMD)" which can be found in following websites, respectively: (a) PEER: http://ngawest2.berkeley.edu/; and (b) CESMD: http://strongmotioncenter.org/. These databases have been widely used in the research area of earthquake engineering and seismic damage assessment.

magnitudes and epicentral distances both of a current event and a past event (See Fig. 4.8) (Bommer and Acevedo 2004; Watson-Lamprey and Abrahamson 2006).

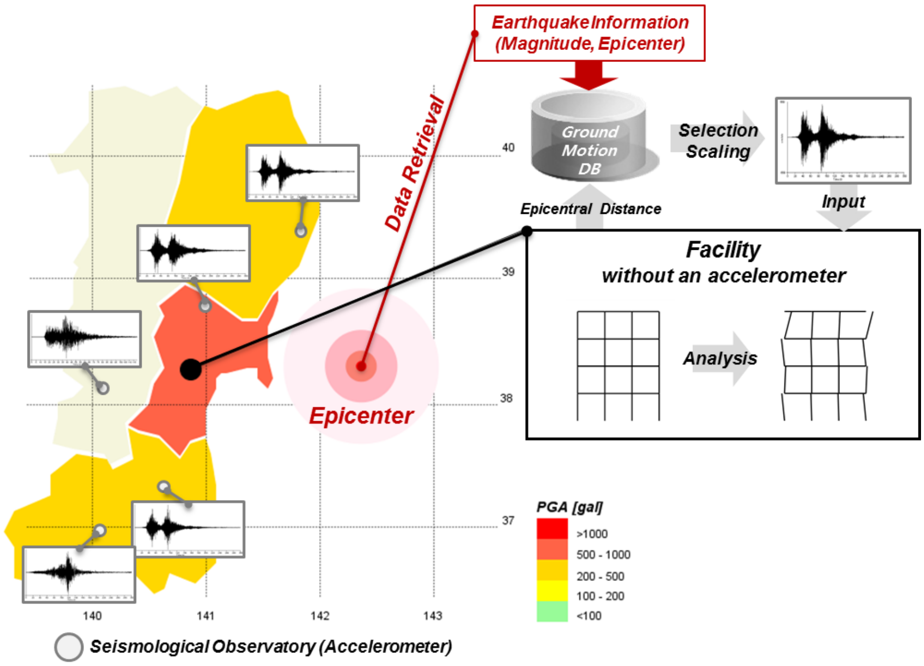



**Figure 4.8** Concept of Damage Assessment for the Facility without an Accelerometer

Therefore, after detecting earthquake information in real time, an epicentral distance is calculated by using facility's locational information (e.g., latitude and longitude) and target earthquake information (e.g., the latitude and longitude of the epicenter) (B in Fig. 4.6). Based on the simple procedures for selection and scaling of ground motion suggested by Katsanos et al. (2010), then, the most relevant ground motion data is selected and the scale factor is calculated for further analysis of structural responses. Since moment magnitude ($M_w$) and epicentral distance ($X$) of an interest seismic event are the most common parameters for the simplest ground motion selection in this procedure, the iterative ground motion selection processes continue by increasing the ranges of selections, until the most suitable data is found within the range of selections as follows:

$$Bin(1):\{M_{w.c} - \Delta m \cdot 1 \leq M_{w.p} \leq M_{w.c} + \Delta m \cdot 1, \left| X_p - X_c \right| \leq \Delta X\,(km) \cdot 1\} \ldots$$

$$Bin(n):\{M_{w.c} - \Delta m \cdot n \leq M_{w.p} \leq M_{w.c} + \Delta m \cdot n, \left| X_p - X_c \right| \leq \Delta X\,(km) \cdot n\}$$

(Eq. 4.13)

where *Bin(1)* = the first range of selecting the most appropriate ground motions in the first selection process; *Bin(n)* = the nth range of selecting the most appropriate ground motions in the nth selection process; $X_c$ = distance between the epicenter and damaged point a current seismic event [km]; $X_p$ = distance between the epicenter and damaged point of past ground motion data [km]; $M_{w.c}$ = moment magnitude of a current seismic event; $M_{w.p}$ = moment magnitude of past ground motion data; $\Delta m$ = increment of the selection range of moment magnitude after each selection process [km]; and $\Delta X$ = increment of the selection range of epicentral distances after each selection process [km].

To reduce the gap between current ground motions and selected past ground motions, the scale factor ($SF$) can be calculated by the ratio between the *PGAs* of both the current event and the past event in accordance with the following equation. Therefore, the accelerograms that shows the minimum *SF* is selected for facility damage assessment (C in Fig. 4.6):

$$SF = \frac{PGA_c}{PGA_p} \tag{Eq. 4.14}$$

where $SF$ = the scale factor of ground motion data; $PGA_c$ = the predicted *PGA* of a current event at the damaged point [cm/sec$^2$]; and $PGA_p$ = the *PGA* of past ground motion data [cm/sec$^2$].

To simulate structure behaviors, the OpenSees federate is interacted in the distributed simulation environment. After sending ground motion data and the scale factor, the OpenSees federate publishes structural response information such as displacements and shear forces (D in Fig. 4.6). As a prototype, this simulation model mainly uses structural displacement information as a main element for estimating structure damage.

This estimated structural displacements and consequent story drifts are utilized to determine the facility damage intensity (i.e., structural damage ratio in the model) based on the structural design standards (Taranath 2005) (E in Fig. 4.6). The maximum story drift of structures with a period ($T$) less or greater than 0.7 seconds are limited as follows:

$$\Delta_{sx} = \alpha \cdot h_{sx} \tag{4.15a}$$

$$\Delta_{sx} \leq 0.025 \cdot h \ \ (T \geq 0.7\,\text{sec}) \tag{4.15b}$$

$$\Delta_{sx} \leq 0.020 \cdot h \ \ (T \leq 0.7\,\text{sec}) \tag{4.15c}$$

where $\Delta_{sx}$ = maximum story drift at X-th floor; $h_{sx}$ = floor height at X-th floor; and $\alpha$ = 0.010 (for S-seismic classification), 0.015 (for I-seismic classification), and 0.02 (for II-seismic classification).

The information of structural damage ratio will be used as input values for the DES model to determine required types and amounts of restoration works of a damaged facility restoration. The detailed description of the whole facility damage assessment model can be found in [Appendix B-III].

***Model Test***

In this test, the actual earthquake information of the M 9.0 2011 Earthquake of Tohoku is also applied to evaluate this model's capability of facility damage assessment. With three hypothetical cases of facilities located in the inland, middle, and coastal areas of Tohoku, respectively, this test simulates both ground motions and damage patterns at these three case facilities. The main functions of facility damage assessment model include: (a) the selection of appropriate ground motions of the facility from the historical database; and (b) structure damage calculations based on analysis data of structural responses from the OpenSees federate. To improve confidence in the model, this study utilizes actual facility information from the 8-story residential/commercial complex building for three hypothetical cases of facilities located in different areas.

**Table 4.7** Test Results of Facility Damage Assessment Model

| Facilities | Peak Ground Acceleration (PGA) | | Damage Status (0.0–1.0) | |
|---|---|---|---|---|
| | Actual Data (Eidinger et al. 2012) | Simulation Results | Actual Data (Eidinger et al. 2012) | Simulation Results |
| A Facilities in the Coastal Area of Tohoku<br>***- Lat.: 38.9***<br>***- Long.:141.5*** | 500–1000 gal [cm/sec$^2$] | ***PGA of selected GM (A):***<br>998 gal [cm/sec$^2$]<br>***Scale Factor (SF) (B):***<br>0.839<br>***Estimated PGA (C=A*B):***<br>837 gal [cm/sec$^2$] | Very heavy (0.8–1.0) | 1.0 |
| B Facilities in the Middle Area of Tohoku<br>***- Lat.: 39.7***<br>***- Long.:141.2*** | 200–500 gal [cm/sec$^2$] | ***PGA of selected GM (A):***<br>591 gal (cm/sec$^2$)<br>***Scale Factor (SF) (B):***<br>0.569<br>***Estimated PGA (C=A*B):***<br>336 gal [cm/sec$^2$] | Moderate (0.4–0.6) | 0.5 |
| C Facilities in the Inland Area of Tohoku<br>***- Lat.: 37.1***<br>***- Long.:139.4*** | 50–200 gal [cm/sec$^2$] | ***PGA of selected GM (A):***<br>227 gal [cm/sec$^2$]<br>***Scale Factor (SF) (B):***<br>0.542<br>***Estimated PGA (C=A*B):***<br>123 gal [cm/sec$^2$] | Very light (0.0–0.2) | 0 |

As shown in Table 4.7, the simulation results of the estimated *PGAs* of three facilities show observable similarities compared to their actual range of *PGAs* (Eidinger et al. 2012). For three facilities, the historical ground motion data, which were measured by the stations fewer than 50 km away from these facilities in the past, were selected respectively. Since there exist gaps between the actual *PGAs* and the past *PGAs* of selected data due to the differences of epicentral distances and earthquake magnitude among past and current events, the appropriate scale factors for three facilities are respectively calculated to adjust estimation errors for a better analysis of structural responses.

By applying selected ground motion data and estimated scale factors, the simulation results of structural damage of three facilities largely correspond to actual data that were roughly investigated as the ratio of facilities' damage at each region from a past seismic event. Although damage can vary according to structural, geographical, and geological features, simulation results in this study can be effective in the approximate project planning stage. These hypothetical cases of damaged facilities will be further utilized in the model experiment sections.

### 4.2.2 Facility-Level Restoration Analysis Model Using DES

***Model Descriptions***

In this section, reconstruction processes of common buildings are modeled by investigating actual building reconstruction processes and using the DES model that can be useful for facility-level restoration planning. The focus of the DES model includes a whole reconstruction process (e.g., removal/demolition of interior/external materials of the building, structural reinforcement, deck and frame installation, and curtain wall work process), excluding interior finishing works.

Fig. 4.9 describes a detailed description of the DES model developed by using Anylogic 7 (Anylogic Company) software. To help an understanding of the DES model in the Anylogic software described in Fig. 4.9, this study explains objects used for representing resources and activities in the construction processes (Anylogic Company 2014). For instance, the "*Source*" object (e.g., "*CwallSupply*" in Fig. 4.9) generates entities such as construction materials for simulation. The "*Delay*" object (e.g., "*CwallDelivery*" in Fig. 4.9) represents construction activities that have specific work durations. According to activity's duration, the entities that entered to the "*Delay*" object wait for a given amount of time to move to the succeeding object. In the "*Queue*" object (e.g., "*CwallStorage*" in Fig. 4.9), which represents an idle entities, the entities wait to be accepted by the next objects in the construction process flows. The "*Service*" or "*Assembler*" objects (e.g., "*CwallInstall*" in Fig. 4.9) represent construction activities that require other resources such as construction workforces to initiate these activities. The "*ResourcePool*" objects (e.g., "*CwallCrew*" in Fig. 4.9) defines a set of resource units such as construction workforces that performs construction activities. The detailed descriptions of these objects are found in Fig. 4.10. In addition, the detailed descriptions of the DES model, including input data for resources and durations of activities, can be found in [Appendix B-IV].

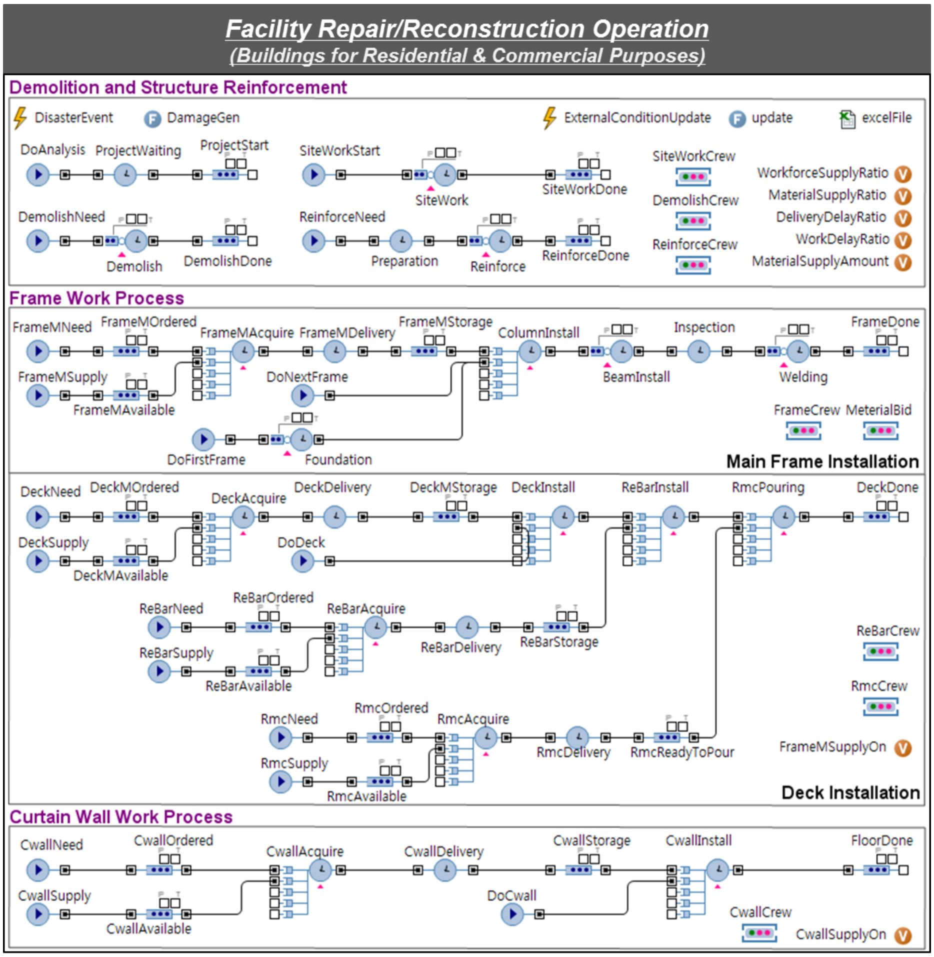



**Figure 4.9** DES Model Descriptions

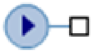
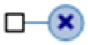
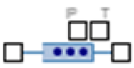
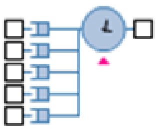
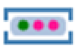
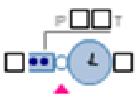
| Legends | Explanation |
|---|---|
| source  | "*Source*" generates entities such as construction materials or specific conditions required for construction activities. "*Source*" is usually a starting point of a process model. |
| sink  | "*Sink*" disposes entities. "*Sink*" is usually an end point in a process model. |
| delay  | "*Delay*" delays entities for a given amount of time. "*Delay*" can represent construction activities that have specific durations, which can be commenced without any specific conditions. The delay time is evaluated dynamically, may be stochastic and may depend on the entity as well as any other conditions. |
| queue  | "*Queue*" is a buffer of entities waiting to be accepted by the next object(s) in the process flow. "*Queue*" can represent construction materials waiting to be used in construction activities. |
| assembler  | "*Assembler*" allows certain number of entities from several sources (5 or less) to be joined into a single entity. All arrived entities wait inside the object until all required entities arrive. Once the new entity can be build, the assembly operation starts. This operation takes the delay time specified. |
| resourcePool  | "*ResourcePool*" defines a set of resource units such as construction workforces performing construction activities that can be seized and released by agents. |
| service  | "*Service*" seizes a given number of resource units, delays the entity, and releases the seized units. "*Service*" can represent construction activities that have specific durations, which can be commenced with workforces or by satisfying conditions. |

**Figure 4.10** Descriptions of the Objects Used in the DES Model

(Anylogic Company 2014)

***Model Test***

To improve confidence of the model, this study utilizes actual reconstruction data from the 8-story residential/commercial complex building as an example case, as described in Table 4.8.

**Table 4.8** Descriptions of the Case Restoration Project

| Sub Process | Work Scope | Planned Production Rates | Planned Work Duration |
|---|---|---|---|
| Site Preparation/ Interior/External Material Demolition | 8 floors ($7,154 M^2$ of total floor area) | - | 5 weeks |
| Structural Reinforcement | 8 floors | 2 weeks/floor | 16 weeks |
| Frame Installation | 8 floors | 2 weeks/floor | 16 weeks |
| Deck Installation | | 1.8 weeks/floor | 14 weeks |
| Curtain Wall Installation | | 2 weeks/floor | 16 weeks |

* The main frame installation works including the overlapped frame, deck, and curtain wall works took almost 18 weeks.

As shown in Fig. 4.11, the reconstruction process simulation is based on the actual data of the activity schedule and resource utilization in the case example. In reality, reconstruction preparation (e.g., demolition of interior and external materials) and structural reinforcement took about 21 weeks, and the main frame reconstruction process of the building took almost 18 weeks, including the overlapped frame, deck, and curtain wall works (See Table 4.8 and Fig. 4.11a). Simulation results in Fig. 4.11(b) largely correspond to the actual data of the construction processes.

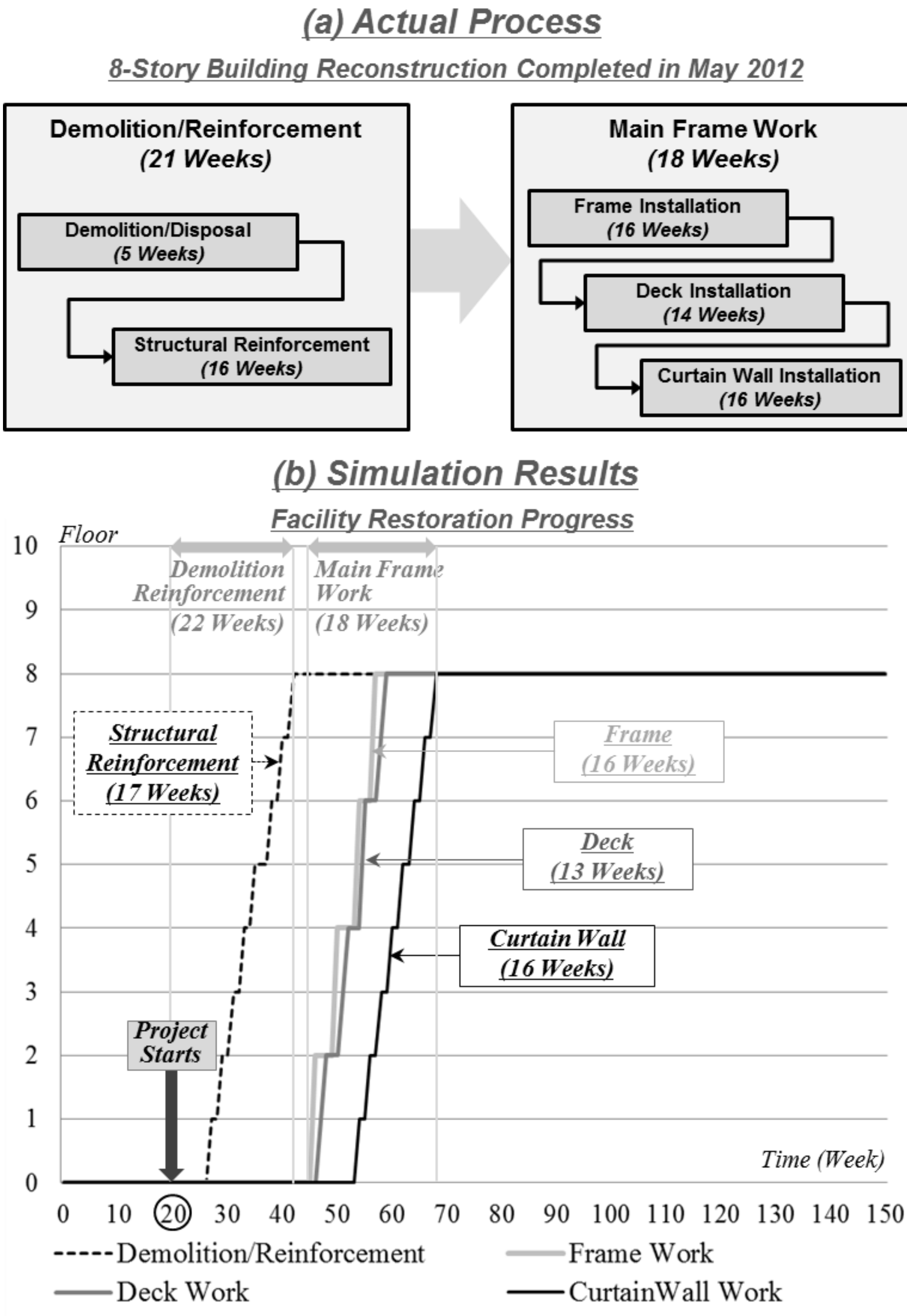



**Figure 4.11** Building Restoration Process Simulation and Test

## 4.3 Descriptions of Data Exchange among Simulations

Within the recovery simulation, four simulation model components—including a regional damage assessment model, a SD regional-level recovery analysis model, a facility damage assessment model, and a DES facility-level restoration analysis model—are interacted with dynamic data exchanges. This study thus offers the descriptions of the multi-level SD-DES simulations in the recovery simulation federate with a consideration of the roles of both damage assessment models.

In detail, SD-DES interactions show interdependent recovery processes among both regional-level and facility-level recovery efforts because not only regional recovery situations affect facility-level restoration projects but also individual facility restorations can also have an impact on overall recovery processes in region. The regional-level damage and recovery situations can particularly be critical to restoration conditions of facility restoration projects by determining the resource availability and the work effectiveness of each restoration process. For example, when the regional recovery planning has a focus on restoring critical public buildings and core infrastructures, resources for housing reconstructions can be limited. On the other hand, a functional recovery of the facility can also be critical for improvement of overall region's recovery situations. For instance, the early restoration of an industrial facility that produces construction materials can be helpful in recovering overall resource supply capabilities. However, while the restoration process of the individual facility is significantly affected by regional-level recovery tendencies, a restored individual facility may have little effects on supporting regional-level recoveries, unless functional recoveries of important facilities (e.g., power plants) can be considerably helpful for improvement of surrounding region's recovery works. Since facility-level restoration planning in this study focuses on most common buildings, therefore, this study only pays attention on the effects of regional-level recoveries on project conditions of individual facility restorations.

On the other hand, by subscribing earthquake information (e.g., magnitude and epicentral location, and focal depth) and the structural response data (e.g., displacement), the recovery simulation estimates both regional and facility damage to analyze regional-level overall recovery processes and facility restoration operations. Table 4.9 shows the data exchange among four simulation model components in the Anylogic recovery simulation federate. Based on the earthquake information from the USGS seismic data retrieval module, a regional damage estimation model produces data of regional damage ratio. By using this information, a SD model provides useful information for regional-level recovery planning by simulating regional recovery situations, as well as information for restoration conditions of facility restoration projects including the resource availability and working environment.

**Table 4.9** Descriptions of Data Exchange among Recovery Simulation Model Components

| Model Components / Data | Regional Damage Estimation | Regional Recovery Analysis (SD) | Facility Damage Assessment | Facility Restoration Analysis (DES) |
|---|---|---|---|---|
| Earthquake | | | | |
| ▪ EventID | Subscribe | | Subscribe | |
| ▪ Time | Subscribe | | Subscribe | |
| ▪ Longitude | Subscribe | | Subscribe | |
| ▪ Latitude | Subscribe | | Subscribe | |
| ▪ Depth | Subscribe | | | |
| ▪ Magnitude | Subscribe | | Subscribe | |
| Structural Displacement | | | | |
| ▪ Displacements | | | Subscribe | |
| RegionalDamage | | | | |
| ▪ Region Name | Publish | Subscribe | | |
| ▪ Facility Type | Publish | Subscribe | | |
| ▪ Damage Ratio | Publish | Subscribe | | |
| FacilityDamage | | | | |
| ▪ Intensity | | | Publish | Subscribe |
| Restoration Conditions | | | | |
| ▪ Region Name | | Publish | | Subscribe |
| ▪ Facility Type | | Publish | | Subscribe |
| ▪ Delivery Delay Ratio | | Publish | | Subscribe |
| ▪ Material Supply Ratio | | Publish | | Subscribe |
| ▪ Workforce Supply Ratio | | Publish | | Subscribe |
| ▪ Work Delay Ratio | | Publish | | Subscribe |

In addition, based on the structural displacement data from the OpenSees structural response simulation, a facility damage estimation model produces data of facility damage intensities. As a result, not only information of the facility damage intensity are applied in the DES model at the event time but also information of restoration conditions—affected by facility's surrounding damage and recovery situations analyzed by SD—are updated to the DES model at every time step. Through these data subscriptions, the DES model simulates construction operations and analyzes the building reconstruction project (in particular, duration estimation) in the post-disaster situation according to facility damage levels and external restoration condition changes.

Since the restoration conditions caused by regional recovery situations dynamically change affecting facility restoration projects over, a more detailed description of interactions between SD and DES models is required to examine the effects of facility restoration operations according to changes in regional-level recovery situations. Fig. 4.12 offers a dynamic data exchange between SD and DES models.

Based on actual data of reconstruction projects taken during a normal situation (i.e., not a post-disaster situation), as shown in Fig. 4.12, required input values in the DES model are determined as both the amount of resources and the duration for initiating each activity. However, in a post-disaster situation, limited material supplies (e.g., quantity and timing) may interrupt each activity's initiation. Poor working environments may cause longer-than-normal activity durations. These unfavorable conditions eventually lead to changes (delays) in overall reconstruction operations compared to a normal situation. The seamless interaction between SD and DES enables the facility restoration simulation to approximately estimate the effects of poor restoration conditions (i.e., SD model) and their dynamic changes (e.g., restoration condition improvement over time as devastated surrounding areas are recovered) on reconstruction operations (i.e., DES model).

In the integrated model, output values from an array element in the SD model—the values are relevant to each facility type in each sub-region (e.g., a general type of building in an A region in this example case) at each time step—are continuously updated to a DES model. The interface variables, which refers to the main contact points between SD and DES (Alvanchi et al. 2011), represent the dynamically changing external conditions of restoration projects. These interface variables in the SD-DES interactions are defined in Table 4.10, with the detailed descriptions of their definitions, calculations from the SD model, their dynamic features, and their effects to the DES model. These interface variables include the delivery delay ratio (DDR), material supply ratio (MSR), workforce supply ratio (WSR), and work delay ratio (WDR).

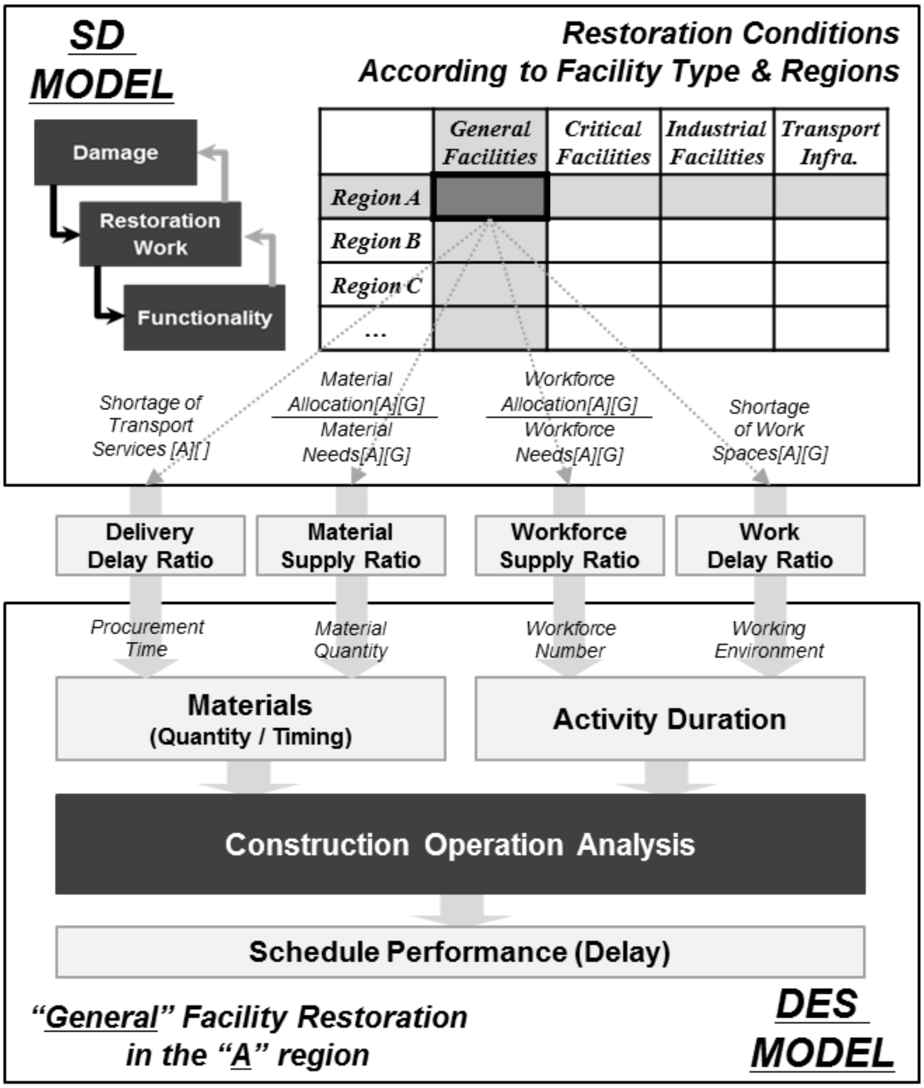



**Figure 4.12** Descriptions of SD-DES Interactions

**Table 4.10** Descriptions of Interface Variables in the SD-DES Interactions

| SD-DES Interface Variables | Calculation from SD model | Value changes | Effects to DES model |
|---|---|---|---|
| Delivery Delay Ratio (DDR)<br>*(most negative value: 1)*<br>*(most positive value: 0)* | ▪ The shortage level of delivery capability caused by limited transportation services at the facility's surrounding sub-region. | ▪ As regional damage is severe, DDR is negative due to damage in the transportation system.<br>▪ DDR is improved as transportation system is restored over time. | ▪ Determining the timing of material procurement for construction activities. |
| Material Supply Ratio (MSR)<br>*(most negative value: 0)*<br>*(most positive value: 1)* | ▪ Total material allocations divided by the total material needs of the relevant facility type at the facility's surrounding sub-region. | ▪ As regional damage is severe, MSR is negative due to massive needs for restoration materials.<br>▪ MSR is improved as restoration works progresses. | ▪ Determining the amounts of material supplies for construction activities. |
| Workforce Supply Ratio (WSR)<br>*(most negative value: 0)*<br>*(most positive value: 1)* | ▪ Total workforce allocations divided by the total workforce needs of the relevant facility type at the facility's surrounding sub-region. | ▪ As regional damage is severe, WSR is negative due to massive needs for restoration workforces.<br>▪ WSR is improved as restoration works progresses. | ▪ Determining the degree of overall extension in construction activity durations due to the lower-than-expected number of workforces. |
| Work Delay Ratio (WDR)<br>*(most negative value: 1)*<br>*(most positive value: 0)* | ▪ The shortage level of workspaces caused by debris at the facility's surrounding sub-region. | ▪ As regional damage is severe, WDR is negative due to massive debris.<br>▪ WDR is improved as debris are removed over time. | ▪ Determining construction activity durations due to poor working environments. |

By applying sub-dimensions (i.e., sub-regions and facility types) in the comprehensive SD model, more detailed and reliable values for subdivided surrounding conditions (i.e., different restoration conditions according to the sub-region and facility type) can be produced. Due to the uncertain disaster situations, the above variables are utilized as median values for the stochastic DES model rather than utilized as deterministic values. In the model, triangular distributions, which have widely been used in project management, are applied to represent variable external conditions and activities' durations.[17] The whole integrated model, including the detailed descriptions of model structures and equations of a two damage assessment models, a SD model, and a DES model, as well as their interactions, can be found in [Appendix B].

## 4.4 Summary

In this chapter, this study developed four simulation model components in the recovery simulation including a regional damage estimation model (in Ch. 4.1.1), a regional-level recovery process analysis model using SD (in Ch. 4.1.2), a facility damage assessment model (in Ch. 4.2.1), and a facility-level restoration operation analysis model using DES (in Ch. 4.2.2).

Firstly, in the regional-level damage estimation model, when earthquake information can be detected by the USGS seismic data retrieval federate,

[17] Triangular distributions, as well as beta distributions, effectively express subjective knowledge especially when a number of observations are limited. Moreover, triangular distributions require further reduced processes for specifying model parameters than beta distributions require (Back et al. 2000; Martinez 2010).

overall damage ratios at the regions are estimated based on the empirical approaches on damage and loss estimations after a seismic event. By conducting the model test, these estimated values can be utilized in the SD model to more accurately analyze overall damage and recovery situations in regions.

Secondly, the SD regional-level recovery process analysis model was developed to understand the multiple and complex recovery processes in an overall region. To capture the differences between recovery processes among diverse types of facilities throughout a whole region in detail, the SD model in this study uses two-dimensional (2-D) array variables including sub-elements for several sub-regions and diverse facility/infrastructure types. By conducting the model test, it was found that the subdivision of regions and facility types can be useful for reflecting more reliable damage patterns and recovery efforts. Since SD is effective in explaining why specific system behaviors have been generated, this analytical capability of SD can be useful in comprehensively understanding complex and interdependent multiple restoration operations and diverse functional recovery processes at the overall region as well as the governmental recovery policy effects.

Thirdly, in the facility-level damage assessment model, the facility damage intensities can be determined to more rapidly and accurately analyze facility restoration operations, by the integrated uses of an USGS seismic data retrieval federate, an OpenSees structural response simulation federate, and the damage assessment model in the Anylogic restoration simulation federate. The test results of the facility damage assessment model showed the effectiveness of rapid damage assessment in the post-disaster restoration project management of facilities.

Finally, the DES restoration operation analysis model was developed to analyze restoration projects, and then tested by applying actual reconstruction data. In addition, the descriptions of the SD-DES interactions in the multi-level recovery simulation with a consideration of the roles of both damage assessment models were provided. Based on the data exchanges among simulation model components, it is identified that SD can assist higher-level (regional-level) recovery planning by estimating regional damage patterns and understanding dynamic features of complex and interdependent multiple recovery processes. The SD can also be useful in reasonably analyzing surrounding restoration conditions of facility restoration projects. The DES also enables the examination of facility restoration operations according to facility damage patterns and external condition changes, in order to support project planners in rapid project planning

# Chapter 5. Interactive Simulation Development

In this chapter, this study provides detailed descriptions of the interactive simulation development for multiple recovery management. To help readers in understanding this issue, an overview of simulation interactions will be offered. Then, this study describes the detailed data exchanges and synchronizations among federates for both damage assessments and recovery process analyses, in the HLA-compliant simulation architecture. The development process of an interactive recovery simulation prototype and its executions will be also provided. Simulation interactions within the HLA-compliant distributed simulation environment enable to organize different combinations of simulations according to different analytical purposes such as a comprehensive analysis of regional-level recovery processes and a detailed analysis of facility-level restoration operations.

## 5.1 Federate Descriptions

To support immediate and reliable damage and recovery analyses after a seismic event, the interactive recovery simulation consists of three federates that interact with each other in the HLA-compliant distributed simulation environment, including an USGS seismic data retrieval federate (i.e., USGS federate), an OpenSees structural response simulation federate (i.e., OpenSees federate), and an Anylogic multi-level recovery simulation federate (i.e., Anylogic federate). These three federates are briefly described as follows:

(a) The Anylogic federate is an AnyLogic-based modeling and damage/recovery simulation solution. Its purpose is to simulate the damage incurred during a natural disaster at both the facility and regional levels. These data are shared with both the regional-level recovery process simulation and the facility-level restoration operation simulation which, along with the damage assessment modules, represent the core functionality of the HLA-compliant distributed simulation. Therefore, the Anylogic federate—described in the previous chapter, Ch. 4—includes four simulation model components of a regional damage estimation model, a SD regional-level recovery process analysis model, a facility damage assessment model, and a DES facility-level restoration operation analysis model. Among them, both regional and facility damage assessment models only interact with an USGS federate and an OpenSees federate for the purpose of instant and reliable damage assessments with the information of seismic intensities and structural responses.

(b) The USGS federate communicates with a USGS server to retrieve earthquake and seismic event data for the damaged region in which the facility resides in near-real time. These earthquake data (i.e. location, depth, and magnitude of the event) are used for estimating overall damage patterns at the whole region as well as computing ground motions from which structural displacements and facility damage are assessed.

(c) The OpenSees federate is used for generating a 2-D model of the facility and calculating the structural responses, especially displacements at each of the key nodes during a seismic event. These data are utilized by the facility damage assessment model in the Anylogic federate for estimating the structural damage of a facility incurred during an earthquake.

## 5.2 Overview of Federate Interactions

In the HLA-compliant distributed simulation, *Objects* in the RTI refers to simulated entities that are of interest to more than one federate and handled by the RTI. (Kuhl et al. 2000). Each *Object Class* has a set of named data called *attributes*. In addition, *Interactions* in the RTI refers a collection of non-persisting data fields (i.e. an event) in the simulation that can be published and/or subscribed to by any number of federates. Each *Interaction Class* has a single data field called *parameter* (Kuhl et al. 2000).[18]

Due to the non-persistent features of exchanged data (e.g., earthquake information) in the interactive recovery simulation, this study utilizes four *interactions* for the communication between federates occurred via HLA, including the *USGSRequest interaction, Earthquake interaction, GroundMotion interaction,* and *StructuralDisplacement interaction*.

[18] In the HLA development, *Objects* and *Interactions* are interchangeable. Any federation model can be written in terms of both *Objects* and *Interactions*. A good example of the HLA-compliant distributed simulation using both *Objects* and *Interactions* is a simulation of a "conveyor-belt sushi" restaurant that includes simulations of sushi production by a chef, sushi transportation in a dish, and sushi consumption for dinner. Since a dish is a persisting entity, a dish can be represented as an *Object* and delivered from a chef to a consumer. On the other hand, it is more effective to represent a sushi as an *Interaction* because a non-persisting sushi will be consumed for a dinner (Kuhl et al. 2000).

**Table 5.1** Descriptions of Interactions in the Interactive Simulation

| **Interactions** (Parameters) | **Explanations** (Data Type) |
|---|---|
| USGSRequest<br>▪ Start Time<br>▪ End Time<br>▪ MinMagnitude<br>▪ MinLatitude<br>▪ MaxLatitude<br>▪ MinLongitude<br>▪ MaxLongitude | <br>▪ Minimum time of events to download (`long`)<br>▪ Maximum time of events to download (`long`)<br>▪ Minimum magnitude of events to download (`double`)<br>▪ Minimum origin latitude of events to download (`double`)<br>▪ Maximum origin latitude of events to download (`double`)<br>▪ Minimum origin longitude of events to download (`double`)<br>▪ Maximum origin longitude of events to download (`double`) |
| Earthquake<br>▪ EventID<br>▪ Time<br>▪ Longitude<br>▪ Latitude<br>▪ Depth<br>▪ Magnitude | <br>▪ A unique ID for the event (`string`)<br>▪ The time at which the event occurred (`long`)<br>▪ The longitudinal location of the event's origin (`double`)<br>▪ The latitudinal location of the event's origin (`double`)<br>▪ The depth from ground level of the event's origin (`int`)<br>▪ The magnitude of the event (`double`) |
| GroundMotion<br>▪ Scale Factor<br><br>▪ Acceleration FilePath | <br>▪ A scaling factor for normalizing ground motion data between a similar historical event and a current event (`double`)<br>▪ The file path for an appropriate acceleration file determined by Damage (`string`) |
| Structural Displacement<br>▪ Displacements | <br>▪ The nodal displacements (C++: `vector<double>`, Java: `ArrayList<double>`) |

The *USGSRequest interaction* represents a request made by the Anylogic federate to download one week's worth of seismic event data from the USGS server. The *Earthquake interaction* represents a single seismic event downloaded from the USGS server by the USGS federate. The *GroundMotion interaction* communicates the ground motion data (computed from the data contained in *Earthquake interaction* by the Anylogic federate) to the OpenSees federate. Finally, the *StructuralDisplacement interaction* represents the nodal displacements in the facility resulting from a seismic event and ground shaking. Table 5.1 offers a detailed description of interactions and their parameters used in the interactive recovery simulations. Fig. 5.1 also shows the HLA-based communication architecture of an interactive recovery simulation, including three federates, four interactions, and publishing/subscribing schemes, based on the simulation framework in the previous chapter, Ch. 3.

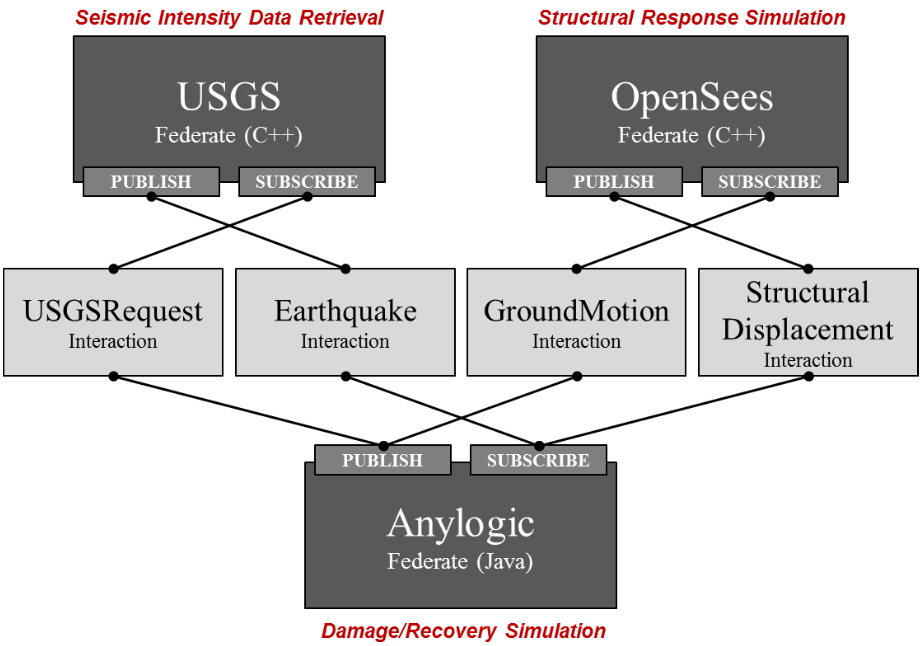



**Figure 5.1** HLA-Based Communication Architecture in the Interactive Recovery Simulation

**Table 5.2** Data Publications and Subscriptions among Federates

| Interactions \ Federate | USGS | OpenSees | Anylogic |
|---|---|---|---|
| USGSRequest | | | |
| · Start Time | Subscribe | | Publish |
| · End Time | Subscribe | | Publish |
| · MinMagnitude | Subscribe | | Publish |
| · MinLatitude | Subscribe | | Publish |
| · MaxLatitude | Subscribe | | Publish |
| · MinLongitude | Subscribe | | Publish |
| · MaxLongitude | Subscribe | | Publish |
| Earthquake | | | |
| · EventID | Publish | | Subscribe |
| · Time | Publish | | Subscribe |
| · Longitude | Publish | | Subscribe |
| · Latitude | Publish | | Subscribe |
| · Depth | Publish | | Subscribe |
| · Magnitude | Publish | | Subscribe |
| GroundMotion | | | |
| · Scale Factor | | Subscribe | Publish |
| · AccelerationFilePath | | Subscribe | Publish |
| StructuralDisplacement | | | |
| · Displacements | | Publish | Subscribe |

In addition, Table 5.2 represents a detailed description of parameter publications/subscriptions of four interactions among federates. The Anylogic federate publishes the request for earthquake information (i.e., *USGSRequest interaction*) which is used as an input data for the USGS federate. After the USGS federate subscribes the *USGSRequest interaction*, it publishes earthquake information (i.e., *Earthquake interaction*). Then, the Anylogic federate subscribes the *Earthquake interaction* and then publishes estimated ground motion data at the facility's location (i.e., *GroundMotion interaction*) required for the OpenSees federate as input data. Finally, after the OpenSees federate subscribes the *GroundMotion interaction*, it publishes structural response data (i.e., *StructuralDisplacement interaction*), which are subscribed by the Anylogic federate.

## 5.3 Design of Recovery Simulation Federation

Based on the federate interactions determined previously, the recovery federation (i.e., the recovery federation is a collection of federates in the interactive recovery simulation that are integrated via HLA) is designed and developed as a prototype. In this prototype, the API handles all of the CERTI—the specific RTI used in this prototype—function calls necessary for creating or joining a federation, publishing or subscribing *objects* and *interactions*, sending and receiving data update. This prototype can be utilized in any HLA-based distributed simulation system because it provides a collection of libraries of a recovery simulation federation.

### 5.3.1 Class Structure and Federation Object Model (FOM)

Since the interactive recovery simulation prototype makes liberal use of object-oriented programming techniques, an understanding of the structure of classes is important. Fig. 5.2 shows the class diagram outlining the basic inheritance and composition scheme used in the interactive recovery simulation prototype.

In the interactive recovery simulation prototype, there are two primary namespaces including *RTI* and *HLA*. The *RTI namespace* is used in the CERTI API—all CERTI classes and methods are members of this namespace. The *HLA namespace* is used by the recovery simulation prototype (but not the models or simulations within the interactive recovery simulation).

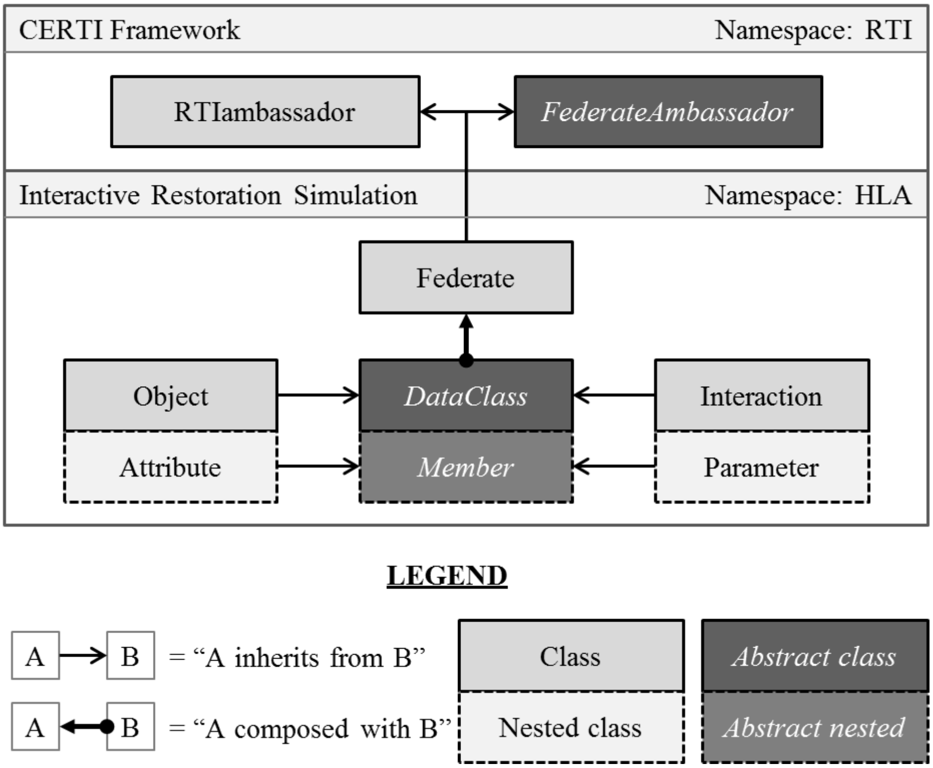



**Figure 5.2** Class Diagram for Interactive Recovery Simulation

In Fig. 5.2, ***RTI::RTIambassador*** handles outgoing data transfer and federation management requests from the *Federate* to the *RTI* via calls to *RTIambassador* methods. These calls should be handled by *Federate* method calls whenever possible. However, all *RTIambassador* functionality is exposed to the user through public inheritance. ***RTI::FederateAmbassador*** handles incoming data transfers from the *RTI* to the *Federate* via calls to abstract methods for which implementation must be provided in any class that inherits from *FederateAmbassador*. These methods are handled by *Federate* and should never be called directly. The inheritance is public, thus modifications to these methods can be made by deriving from *Federate* and overriding any or all of the *FederateAmbassador* methods. However, the user should always call the superclass method when doing so to ensure undesirable side effects are avoided. ***HLA::Federate*** handles all calls to the *RTI namespace* and the creation of *Object* and *Interaction* instances. It is designed to be used via composition, where a simulation will instantiate one instance of *Federate* as either a data member or a global variable through which HLA functions will be handled. All *DataClass* child instances are stored within the *Federate* instance.

***HLA::DataClass*** is the abstract parent class from which *Object* and *Interaction* inherit from, each of which represents a single *HLA object* or *interaction* respectively. Data updates to (sending) and from (reflecting) the *RTI* are handled by this class. Any utilization of *Object* or *Interaction* instances which can be applied to both *class* types should be made through a *DataClass* reference or pointer. *DataClass::Member* instances are stored within the *DataClass* instance. ***HLA::DataClass::Member*** is the abstract parent class from which *Attribute* and *Parameter* inherit from. It stores the data associated with a single data field (attribute or parameter) and handles the encoding/decoding of the data for *RTI* compatibility. Any utilization of *Attribute* or *Parameter* instances which can be applied to both *class* types should be made through a *DataClass::Member* reference or pointer.

***HLA::Object*** is the *DataClass* child responsible for handling HLA objects. This class should not be instantiated directly—the class *Federate* supplies a method for creating *Object* instances and returning a pointer. ***HLA::Object::Attribute*** is the *DataClass::Member* child responsible for handling HLA object attributes. This class should not be instantiated directly—the class *Object* supplies a method for creating *Object::Attribute* instances and returning a pointer. ***HLA::Interaction*** is the *DataClass* child responsible for handling HLA interactions. This class should not be instantiated directly—the class *Federate* supplies a method for creating Interaction instances and returning a pointer. ***HLA::Interaction::Parameter*** is the *DataClass::Member* child responsible for handling HLA interaction parameters. This class should not be instantiated directly—the class *Interaction* supplies a method for creating *Interaction::Parameter* instances and returning a pointer.

For every federation, a Federation Object Model (FOM) (i.e., FOM is a common object model for the data exchanged between federates in a federation, which is represented as a form of a “.fed file”) needs to be created in order to standardize the names of the objects and attributes to allow cross-federate access. The FOM file for the interactive recovery simulation prototype is shown in Fig. 5.3. The FOM file describes what data is going to be sent through the CERTI RTI. When each federate tries to initialize itself, this FOM file has to be provided so that the federate can either create a new federation or join an existing federation (Menassa et al. 2014).

```
;; Interactive Recovery Simulation Prototype
(Fed
    (Federation IRSP)
(FedVersion v1.3)
    (Federate "fed" "Public")
    (Spaces)
    (Objects)
    (Interactions
        (Class InteractionRoot BEST_EFFORT RECEIVE
            (Class RTIprivate BEST_EFFORT RECEIVE)
            (Class USGSRequest RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter StartTime)
                (Parameter EndTime)
                (Parameter MinMagnitude)
                (Parameter MinLatitude)
                (Parameter MaxLatitude)
                (Parameter MinLongitude)
                (Parameter MaxLongitude)
            )
            (Class Earthquake RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter EventID)
                (Parameter Time)
                (Parameter Longitude)
                (Parameter Latitude)
                (Parameter Depth)
                (Parameter Magnitude)
            )
            (Class GroundMotion RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter ScaleFactor)
                (Parameter AccelerationFilePath)
            )
            (Class StructuralDisplacement RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter Displacements)
            )
        )
    )
)
```

**Figure 5.3** FOM File for Interactive Recovery Simulation Federation (.fed file)

The FOM file contains four *interactions* including *USGSRequest interaction* with seven parameters (*StartTime, EndTime, MinMagnitude, MinLatitude, MaxLatitude, MinLongitude*, and *MaxLongitude*), *Earthquake interaction* with six parameters (*EventID, Time, Longitude, Latitude, Depth*, and *Magnitude*), *GroundMotion interaction* with two parameters (*ScaleFactor* and *AccelerationFilePath*), and *StructuralDisplacement interaction* with one parameter (*Displacements*).

### 5.3.2 Federation Execution

With the federates and interactions defined before, this study provides a general outline of the HLA data exchanges occurring during a federation execution, as shown in Fig. 5.4.

1) The Anylogic federate, which plays main functions for multi-level damage and recovery simulations, increments the simulation time by one week.
2) The Anylogic federate creates and sends an *USGSRequest interaction*.
   a) *StartTime* = current simulation time minus one week.
   b) *EndTime* = current simulation time.
   c) All other parameters are static.
3) The USGS federate receives *USGSRequest* and send queries to the USGS server.

4) If seismic data conforming to the parameters in *USGSRequest* is found, the USGS federate creates and sends an *Earthquake interaction.*

5) The Anylogic federate receives *Earthquake* and computes ground motions of the point where a facility resides. The recovery simulation in the Anylogic federate also utilizes the earthquake information in order to estimate overall damage patterns of the damaged region and simulate regional-level recovery processes.

6) The Anylogic federate creates and sends a *GroundMotion interaction*.

7) The OpenSees federate receives *GroundMotion* and computes structural displacement.

8) The OpenSees federate creates and sends a *StructuralDisplacement interaction*.

9) The Anylogic federate receives *StructuralDisplacement* and computes facility damage patterns based on the information of nodal displacement.

10) The recovery simulation in the Anylogic federate makes use of the facility damage data and simulates the facility restoration progress.

Steps 1-3 are continuously executed during simulation execution. Steps 4-10 are immediately executed once for every *Earthquake interaction* sent.

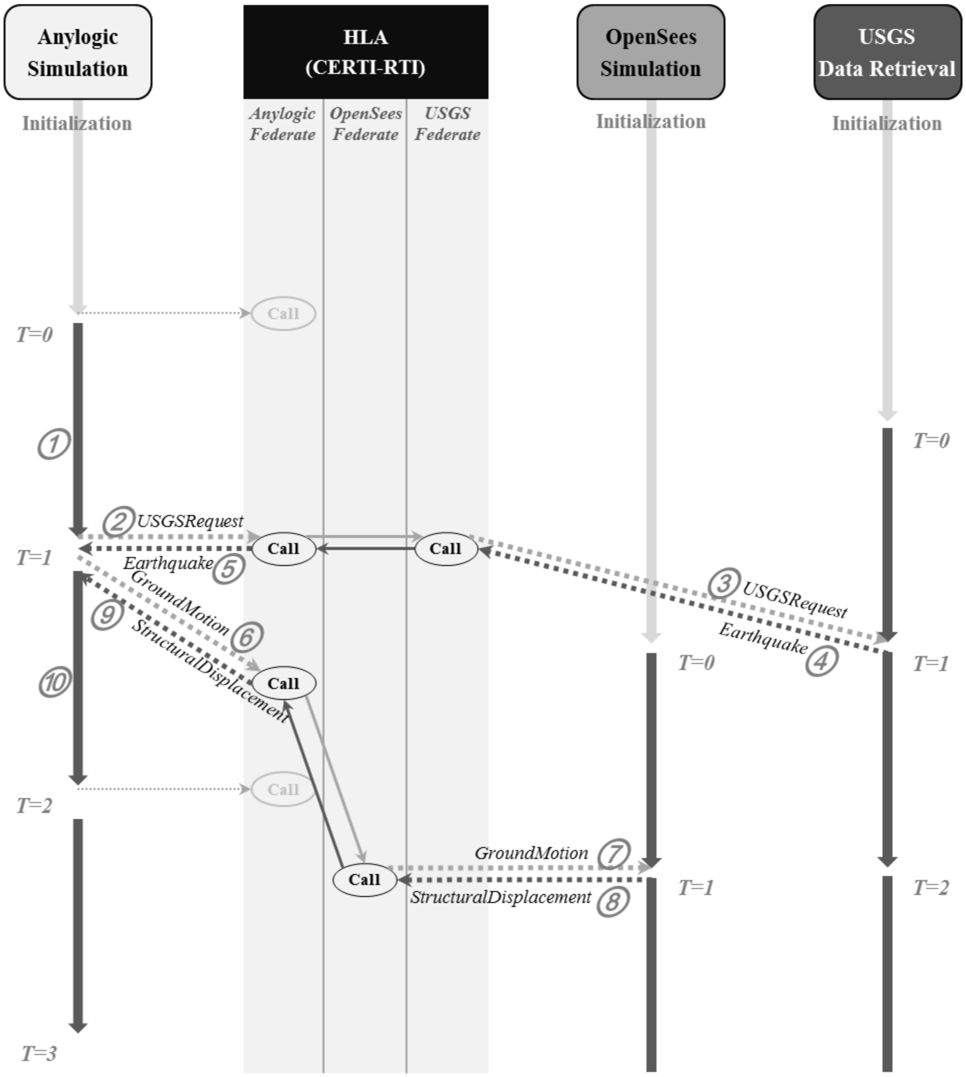



**Figure 5.4** Data Exchange among Federates during Federation Execution

In detail, Figs. 5.5–5.10 describes the procedures to execute interactive recovery simulations in the HLA-compliant distributed simulation environment. After opening the recovery simulation model in the Anylogic simulation, the desired facility location and earthquake magnitude parameters can be set in the Damage module of the Anylogic recovery simulation, as shown in Fig. 5.5. In the Damage module, two simulation model components, including a regional-level damage estimation model and a facility-level damage assessment model, are integrated.

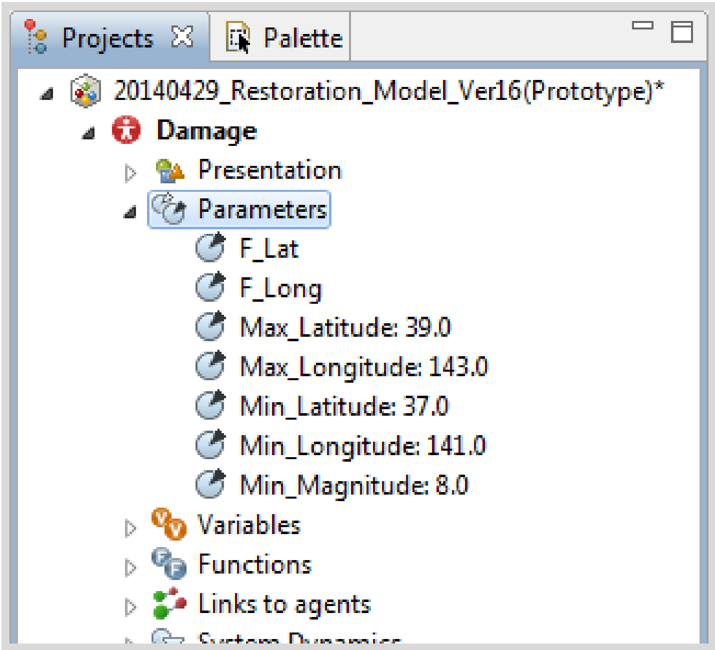



**Figure 5.5** Damage Module Parameters for a Seismic Request

CERTI-RTI is then launched in a command window, as shown in Fig. 5.6. The executables for both the USGS federate and the OpenSees federate are also launched for interactive simulation, as shown in Fig. 5.7 and Fig. 5.8, respectively.

After launching both federates, *USGSRequest interaction* and *Earthquake interaction* in the USGS federate are created to subscribe earthquake request data and publish earthquake information while *GroundMotion interaction* and *StructuralDisplacement interaction* in the OpenSees federate are created to subscribe ground motion data and publish structural displacement data.

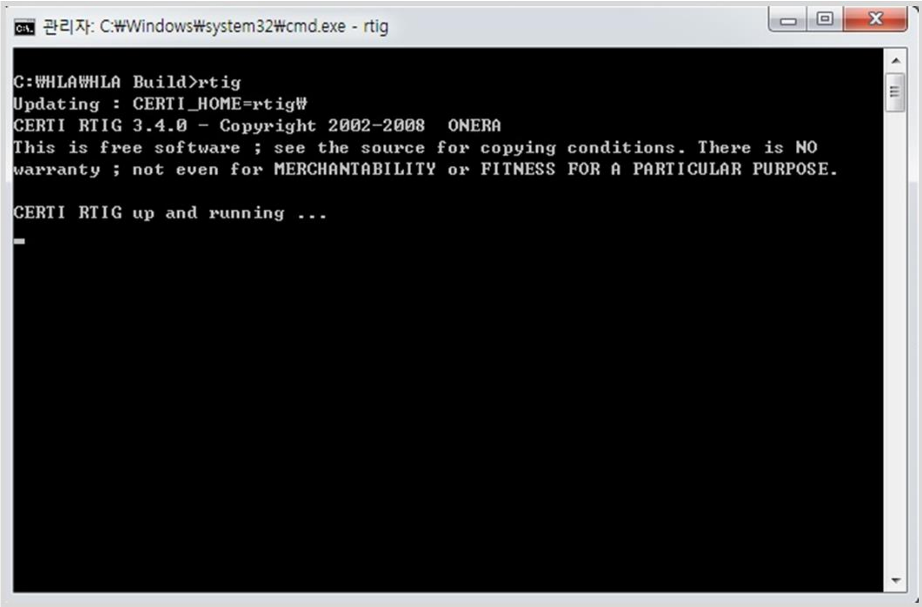



**Figure 5.6** Launching the CERTI RTI

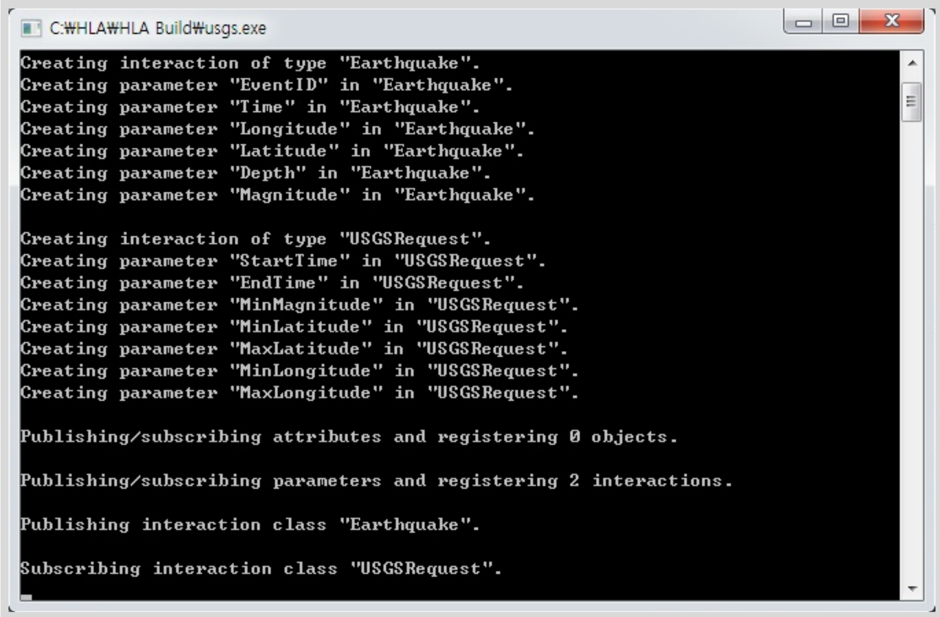



**Figure 5.7** Launching the USGS Federate

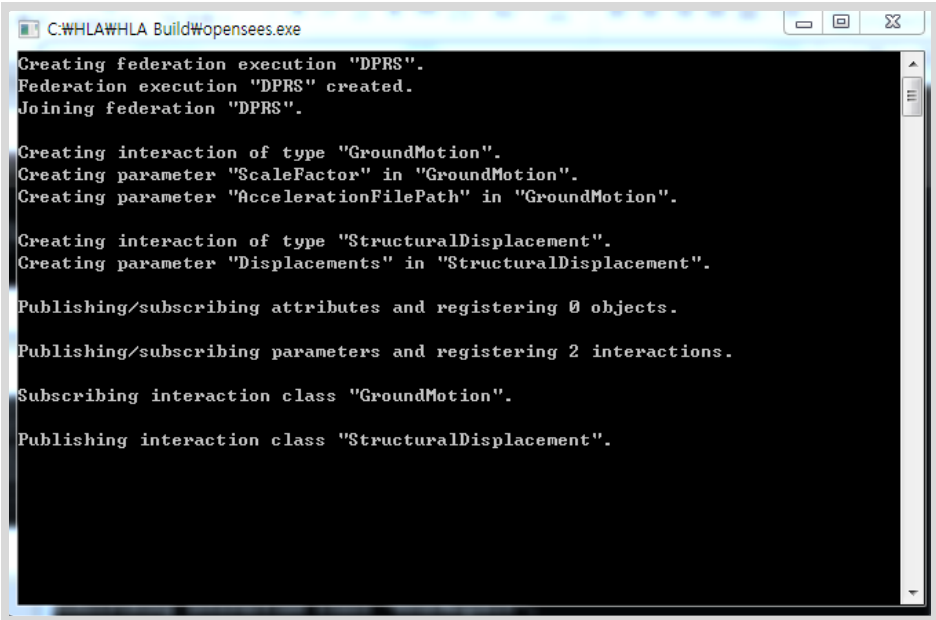



**Figure 5.8** Launching the OpenSees Federate

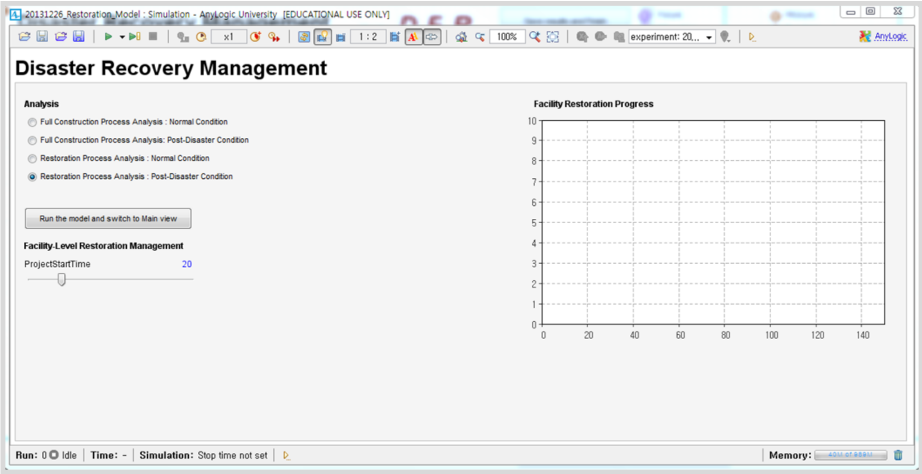



**Figure 5.9** Launching the Anylogic Recovery Simulation

In the Anylogic federate, a multi-level recovery simulation model is simulated as shown in Fig. 5.9. Then, the interactive recovery simulation prototype is executed with the interaction among three federates. Figs. 5.10–5.12 shows runtime information for federates during execution including the USGS federate, the OpenSees federate, and the Anylogic federate.

In the USGS federate in Fig. 5.10, detected earthquake is shown with the information of an event time, epicentral location, magnitude, and focal depth. In the OpenSees federate in Fig. 5.11, structural displacements for every nodes of the structure are calculated. Based on this information, The Anylogic federate in Fig. 5.12 simulates both regional and facility damage and both regional- and facility-level recovery processes and operations. In particular, the Damage module in the Anylogic recovery simulation publishes query request for a seismic event, subscribes earthquake information from the USGS federate to estimate regional damage, publishes ground motion data, and finally subscribes structural response data from the OpenSees federate to assess facility damage in order of precedence, as shown in Fig. 5.13.

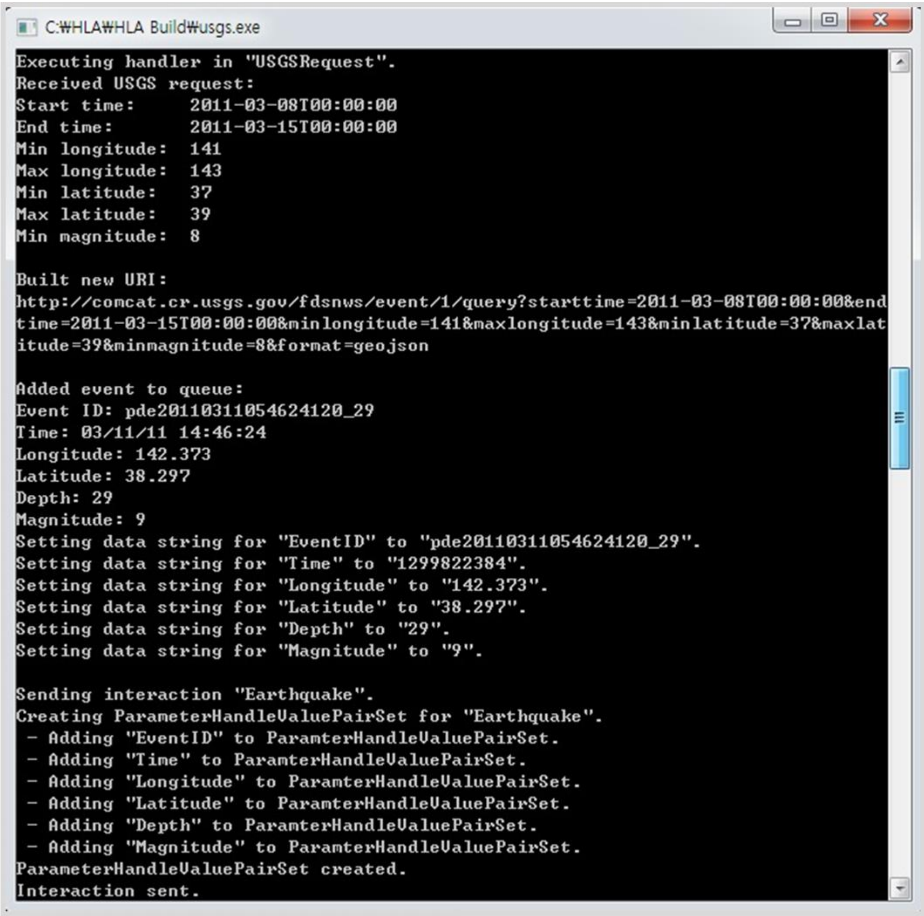



**Figure 5.10** USGS Federate Output during Federation Execution

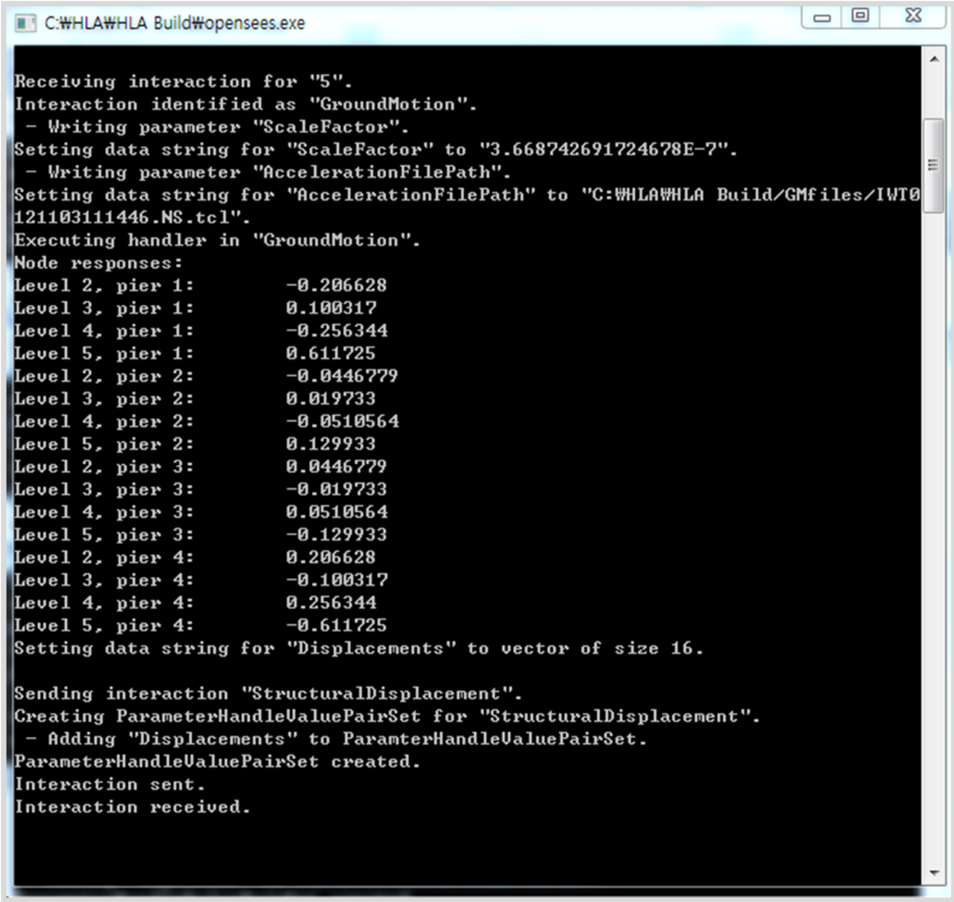



**Figure 5.11** OpenSees Federate Output during Federation Execution

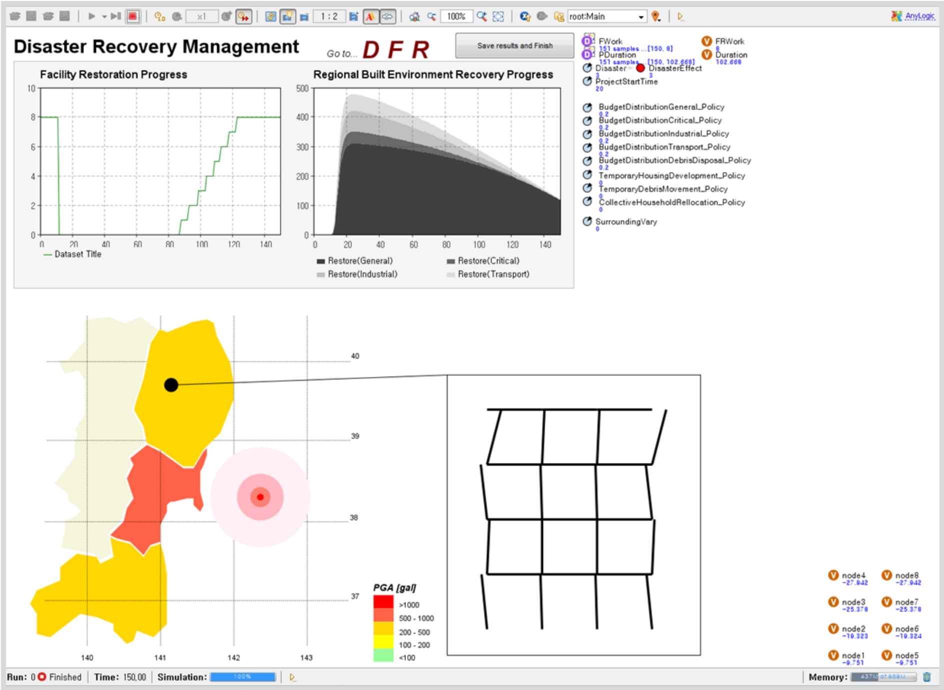



**Figure 5.12** Anylogic Recovery Simulation Window

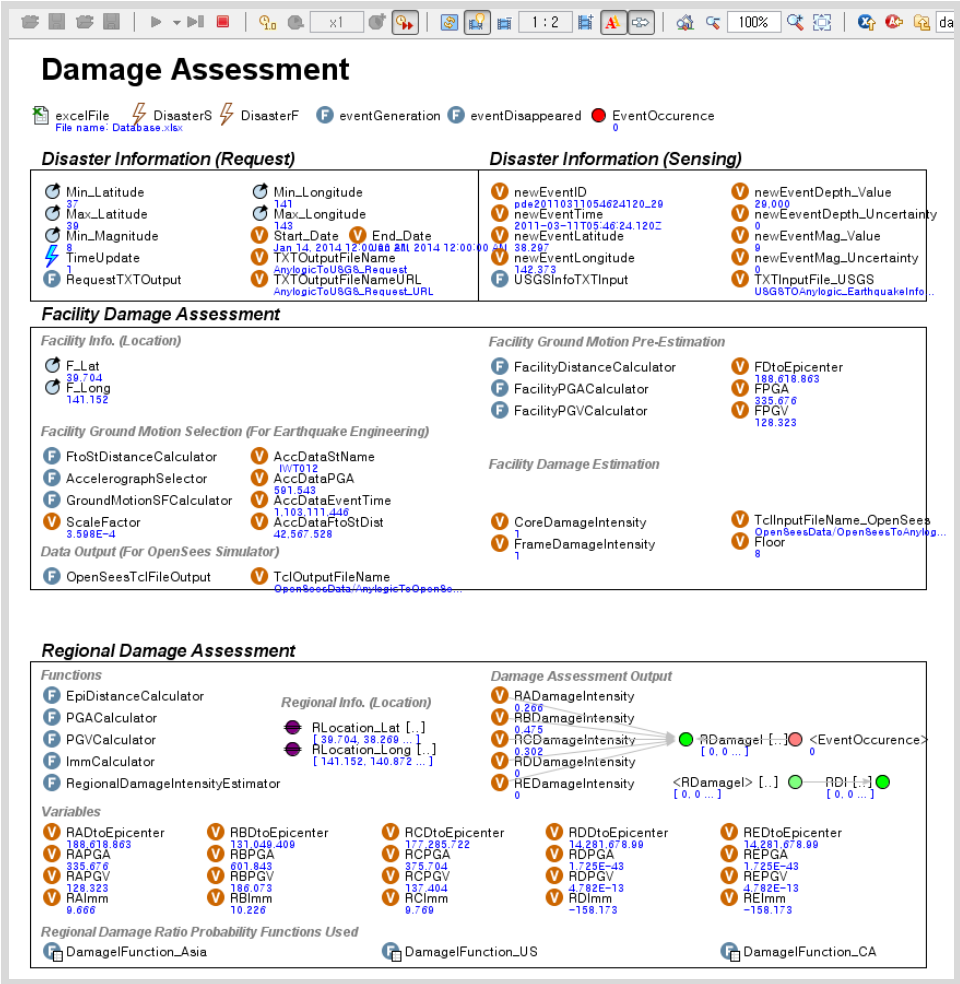



**Figure 5.13** Damage Module in the Anylogic Recovery Simulation during Federation Execution

By using estimated damage of regions and facilities, both SD regional-level recovery process simulations and DES facility-level restoration operation simulations are activated. As shown in Fig. 5.14, the simulation model analyzes overall recovery processes among diverse types of facilities/infrastructures and among sub-regions.

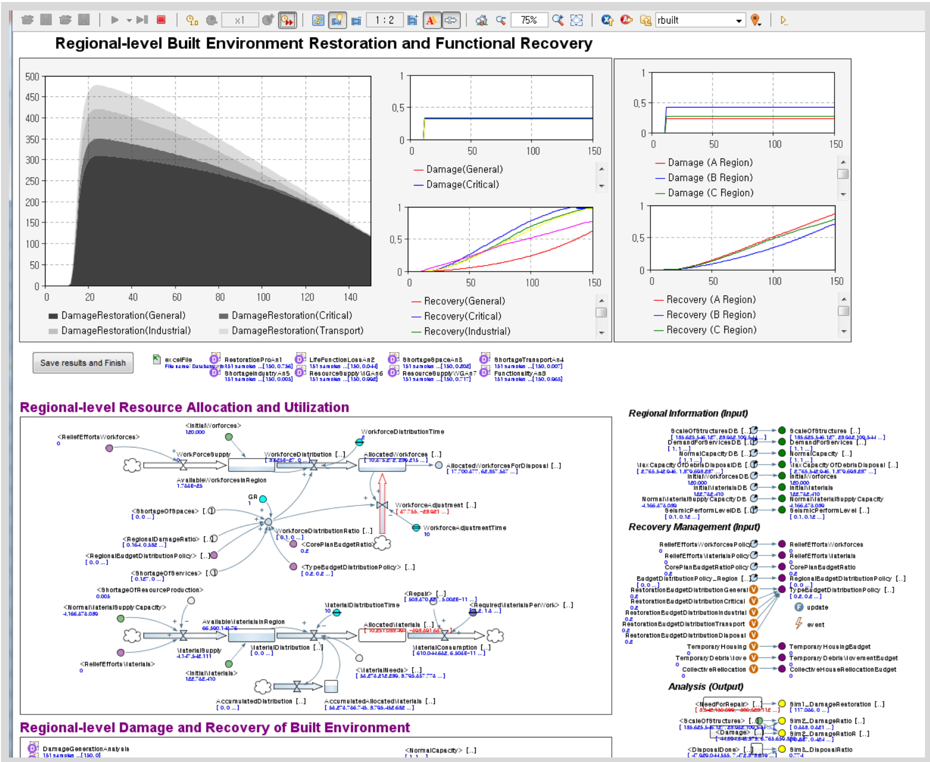



**Figure 5.14** Anylogic Regional-Level Recovery Simulation Using SD during Federation Execution

Figs. 5.15 and 5.16 show facility-level structure repair/reconstruction operations. When only DES is simulated to represent a restoration project in the normal situation, as shown in Fig. 5.15, restoration condition variables (e.g., workforce supply ratio, material supply ratio, delivery delay ratio, and work delay ratio) are static and optimistic, and consequently restoration operations progress as well as expected. On the other hand, When SD and DES are interacted to represent a post-disaster restoration project, as shown in Fig. 5.16, restoration condition variables (e.g., workforce supply ratio, material supply ratio, delivery delay ratio, and work delay ratio) from SD are negative and change over time, by causing delays in restoration operations.

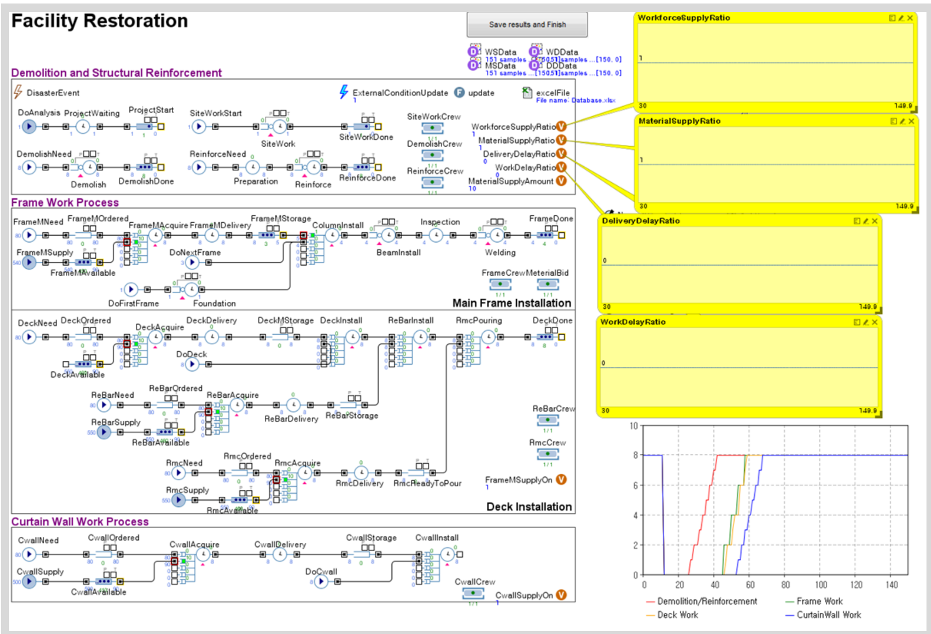



**Figure 5.15** Anylogic Facility-Level Restoration Simulation Using DES during Federation Execution (Normal Situation: DES Only)

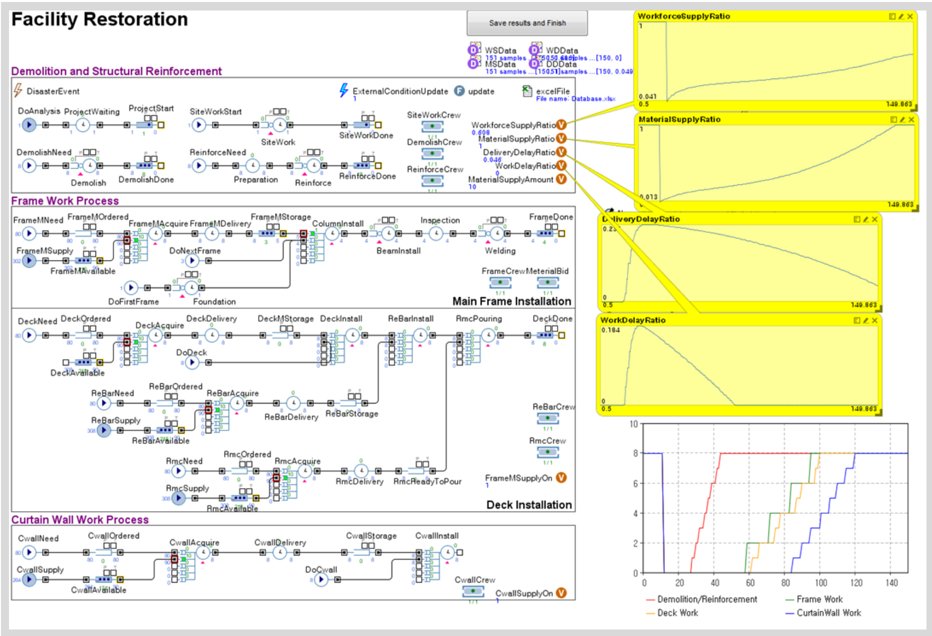



**Figure 5.16** Anylogic Facility-level Restoration Simulation Using DES during Federation Execution (Post-Disaster Situation: SD-DES Interactions)

## 5.4 Summary

In this chapter, this study provided technical descriptions for the development of the interactive recovery simulation in the HLA-compliant distributed simulation environment. By determining the interactions among federates including an USGS seismic data retrieval federate, an OpenSees structural response simulation federate, and an Anylogic damage and recovery simulation federate, the interactive recovery simulation federations was designed with the class structure and the Federation Object Model (FOM). Finally, this study offered federation execution processes for the purpose of incorporating near real-time seismic data retrieval and structural response analysis into rapid recovery simulation of both regional and facility levels. The detailed description of the prototype development can be found in [Appendix C].

# Chapter 6. Case Simulation and Experiment

In this chapter, this study conducts case simulations for both a comprehensive understanding of regional recovery processes and a detailed analysis of facility restoration works, in order to inspect the usefulness of interactive simulations with regard to interoperability, reusability, and extendibility according to each purpose of analysis for multi-level (the regional level and the facility level) recovery planning. The developed prototype of interactive recovery simulations is tested in advance by using the actual data of a past disaster case (e.g., the M9.0 2011 earthquake of Tohoku). Based on the tested prototype, this study conducts both government- and project-level recovery policy experiments to assist with the recovery manager's decision-making in the early recovery planning phase with both regional and facility levels. These analyses and experiments with discussions aim at providing insights and policy implications into both regional recovery management and facility restoration management.

## 6.1 Test of Interactive Recovery Simulation Prototype

To obtain reliable simulation results, the validity of the simulation model needs to be evaluated against the modeling purpose (Banks et al. 2005). As described in the previous chapter, Ch. 4, four recovery simulation model components—including a regional damage assessment model that interacts with the USGS federate, a SD regional-level recovery analysis model, a facility damage assessment model that interacts with the USGS federate and the OpenSees federate, and a DES facility-level restoration analysis model—have been separately tested using case examples. Since distributed recovery simulation in this study aims to analyze how and to what extent the different regional and facility damage patterns as well as recovery plans respectively affect both regional recovery situations and facility restoration projects, this study will focus on the verification of the synchronization and dynamic data exchange among recovery simulation model components and among different federates.

Firstly, this study tests SD-DES interacted recovery simulation in the Anylogic federate to capture if interactions among regional- and facility-level recovery simulations can be well-articulated in the model. In particular, since regional-level damage and recovery situations can be critical to restoration conditions of facility restoration projects, the test focuses on if dynamic features of restoration conditions—analyzed by SD model—are well-applied to facility restoration operations—analyzed by DES model—over time. Secondly, this study conducts a test of interactions among an USGS federate for seismic data retrieval, an OpenSees federate for structural response simulation, and an Anylogic federate for recovery simulation in the HLA-compliant distributed simulation environment, to check the accuracy of near real-time disaster intensity analysis and regional and facility damage assessment for multi-level recovery management.

### 6.1.1 Test for SD-DES Interacted Recovery Simulation

Table 6.1 provides test results of SD-DES interactions in the Anylogic multi-level recovery simulation with a focus on fully damaged facilities located in a severely damaged region. In the SD model, interface variables—which represent external restoration conditions for general facility restoration including delivery delay ratio, material supply ratio, workforce supply ratio, and work delay ratio—change over time from the most pessimistic values (PVs) at the time of the project start (approximately 0.237, 0.081, 0.113, and 0.184, respectively) to the most optimistic values (OVs) at the time of the project finish (0.120, 0.719, 0.453, and 0.000, respectively) within a single simulation run for 150 weeks. The changes of these values are due to the improvement of limited external conditions for facility restoration projects as regional-level multiple recovery works progress. In the SD-DES interacted simulation, these dynamic values need to thus be continuously updated in the DES model at every time step.

To check if dynamic data exchanges of interface variables are implemented as well as expected in the SD-DES interacted simulation, this study first develops an intermediate DES model to which the changing values for restoration conditions from the SD model are applied as linear functions representing restoration condition changes (i.e., linear changes from PVs to OVs) manually, based on the method for verifying the hybrid model suggested by Peña-Mora et al. (2008). On the other hand, all values are automatically applied to the DES model by dynamic data exchange between SD and DES in the SD-DES interacted simulation. Then, this study conducts a comparative analysis between the intermediate DES model and the integrated SD-DES model.

**Table 6.1** Verification Results: Statistics for Simulated Project Durations of Intermediate DES Model and SD-DES Interacted Model

| **Measurement** | **Intermediate DES Model (A)*** | **SD-DES Model (B)**** | **Comparison (A/ B)** |
|---|---|---|---|
| Sample Size | 100 | 100 | 1.000 |
| Min | 98.7 weeks | 99.9 weeks | 0.988 |
| 25% Quartile | 100.7 weeks | 101.6 weeks | 0.992 |
| Mean | 101.6 weeks | 102.3 weeks | 0.993 |
| Median | 101.8 weeks | 102.3 weeks | 0.995 |
| 75% Quartile | 102.5 weeks | 103.0 weeks | 0.995 |
| Maximum | 104.3 weeks | 104.4 weeks | 0.998 |
| Standard Deviation | 1.237 weeks | 1.010 weeks | 1.225 |

* Linear functions representing restoration condition changes calculated by the SD model are applied to the DES model manually.
** Dynamic restoration conditions (interface variables) are automatically updated to the DES model at every time step with SD-DES interactions.

As described in Table 6.1, both simulation results of project durations respectively from the intermediate model and the SD-DES interacted model show high similarities even though there exist gaps in values due to nonlinear changes of interface variables in the SD-DES interacted model. Therefore, it is identified that changes of interface variables over time from the SD model are well reflected to the DES model, and that the SD-DES interacted simulation is valid for post-disaster multi-level recovery management by considering interactions among regional- and facility-level recovery efforts.

### 6.1.2 Test for Distributed Simulation

Table 6.2 provides test results of interactions among an interactive recovery simulation federation in the HLA-compliant distributed simulation environment. Since a facility-level restoration simulation model interacts with both the USGS federate and the OpenSees federate while a regional-level recovery simulation model only interacts with the USGS federate, the test of simulation interactions is conducted based on the facility-level restoration simulation and its synchronization with the USGS and OpenSees federates. In this test, this study also develops an intermediate simulation model that only the Anylogic recovery simulation runs based on the hypothetical case of partially damaged common 8-story building restoration after the M 9.0 2011 earthquake of Tohoku. In this hypothetical case, the disaster event takes place at week 10 after simulation start time and the 50% of structural damage of the building located in the middle area of Tohoku is estimated by using actual earthquake information and facility information. In the intermediate recovery simulation model, these earthquake information and damage information are input to the Anylogic manually while these information are automatically applied to the Anylogic by dynamic data exchanges among three federates in the interactive recovery simulation.

By simulating interactive recovery prototype that three federates including an USGS, an OpenSees, and an Anylogic federate are interacted in the HLA-compliant distributed simulation environment, the M 9.0 2011 earthquake of Tohoku is also detected 10 weeks after a simulation by the USGS federate and the 50% of structural damage is also estimated by the structural response data from the OpenSees federate.

**Table 6.2** Verification Results: Statistics for Simulated Project Durations of Intermediate Simulation Model and Interactive Simulation Prototype

| **Measurement** | **Intermediate Simulation Model (A)*** | **Interactive Simulation Prototype (B)**** | **Comparison (A/ B)** |
|---|---|---|---|
| Sample Size | 10 | 10 | 1.000 |
| Min | 36.1 weeks | 37.1 weeks | 0.945 |
| 25% Quartile | 37.1 weeks | 37.5 weeks | 0.966 |
| Mean | 38.2 weeks | 38.6 weeks | 0.965 |
| Median | 38.2 weeks | 38.3 weeks | 0.976 |
| 75% Quartile | 39.1 weeks | 39.7 weeks | 0.962 |
| Maximum | 40.3 weeks | 41.1 weeks | 0.955 |
| Standard Deviation | 1.360 weeks | 1.407 weeks | 0.951 |

* Only Anylogic is simulated with earthquake and damage information input manually.

** The earthquake and damage information are automatically applied to the Anylogic federate by dynamic data exchange among federates.

For the purpose of identifying if the requirements for data exchange in the interactive recovery simulation are satisfied, this study conducts a comparative analysis of simulation results of both the intermediate simulation model and the interactive recovery simulation prototype, with regard to facility restoration operations (in particular, project durations) according to earthquake and damage information. As described in comparative analysis results of Table 6.2, both simulation results of project durations show high similarities. Therefore, it is verified that three federates in the interactive recovery simulation prototype is synchronized as well as expected. Since the accuracy of near real-time disaster data retrieval and damage assessment for multi-level recovery management are guaranteed in the interactive simulation prototype, it is valid that this prototype can be effective for post-disaster recovery management at both regional and facility levels.

As an in-depth investigation of test results, this study also checks in detail whether all data are timely and accurately interacted among federates. In the interactive recovery simulation prototype, three federates including an USGS federate for seismic data retrieval, an OpenSees federate for structural response simulation, and an Anylogic federate for multi-level recovery simulations are interacted. The Anylogic recovery simulation, which has the core functionality of the interactive recovery simulation, is simulated by updating query request information for a seismic event. After detecting target earthquake, earthquake information for the USGS federate are sent to Anylogic recovery simulation, then, Anylogic recovery simulation estimates regional damage ratio for regional-level recovery simulations and creates ground motion data in order to send them to the OpenSees federate. Using this data, the OpenSees federate produces structural displacement data and sends them to the Anylogic recovery simulation. The Anylogic recovery simulation finally calculates facility damage by subscribing this data, for the purpose of performing a facility-level restoration operation simulation. In this test, recovery simulations are activated by detecting the 2011 earthquake of Tohoku. By changing simulation start time, it is found that interacted data including 16 parameters in 4 interactions are timely and accurately published and subscribed by federated as well as expected, as shown in Table 6.3, which demonstrates the accuracy of dynamic data exchange among federates.

**Table 6.3** Verification Results: Dynamic Data Exchange among Federates

| **Simulation Starting Time*** | **Earthquake Detection Time**** | **Accuracy of Data Exchange***** |
|---|---|---|
| 02-05-2011 (5 weeks before an event) | 7 weeks | 100% |
| 02-12-2011 (4 weeks before an event) | 6 weeks | 100% |
| 02-19-2011 (3 weeks before an event) | 5 weeks | 100% |
| 02-26-2011 (2 weeks before an event) | 4 weeks | 100% |
| 03-05-2011 (1 weeks before an event) | 3 weeks | 100% |

* Different simulation starts to conduct recovery simulations by subscribing earthquake information of the 2011 earthquake of Tohoku in March 11, 2011.

** The time of activating recovery simulations in the Anylogic federate by detecting earthquake information and subscribing structural response.

*** The comparison results of data interactions (16 parameters in 4 interactions) between the intermediate simulation model and interactive simulation prototype.

## 6.2 Simulation for Regional-Level Recovery Planning

In this section, this study presents how the interactive recovery simulation can be effectively utilized in the regional-recovery planning. By the interaction with the USGS seismic data retrieval federate, the SD regional-level recovery simulation model can reasonably analyze overall recovery processes immediately after a disaster by using estimated regional damage information according to a seismic intensity and understanding complex and interdependent recovery systems. Since an understanding of overall damage and recovery situations is critical to regional-level recovery planning mostly implemented by the government, the experiments are performed to analyze damage and restoration work progresses as well as functional losses and recoveries of overall facilities/infrastructures according to damage intensities and recovery plans. Based on investigated analytical requirements of regional-level recovery management in a chaotic disaster situation, case simulations aim at providing insights with regard to following key questions: (a) how do overall recovery processes at the damaged region differently change over time among sub-regions and facility types according to overall damage patterns and multiple restoration works?; and (b) how do recovery plans affect overall restoration processes and recovery situations?

### 6.2.1 Experimental Design

To properly understand regional-level diverse damage and recovery situations as well as the effectiveness of governmental plans on the overall complex and multiple recovery processes, this study analyzes the changes in the overall recovery system over time according to diverse recovery scenarios with a consideration of interdependencies among numerous facilities and their diverse functions. This simulation is based on the developed prototype and based on the actual data of a past disaster case of the M 9.0 2011 earthquake of Tohoku. The developed model simulates the differences between variables and their changes, and then this study compares these shifts with a base case that represents the situation in which the recovery budget is evenly distributed (i.e., there is no recovery priority).

As described in the SD regional-level recovery simulation test results in the previous chapter, Ch. 4.1.2, the simulations and experiments utilize a set of actual regional information and enforced governmental policy information after the M 9.0 2011 earthquake of Tohoku, with a consideration of different damage intensities of sub-regions (i.e., the Iwate, Miyagi, and Fukushima prefectures), the diverse types of facilities/infrastructures (e.g., general, critical, industrial facilities, and transportation infrastructure), and the enforced budget distribution plans (E-Stat 2013; NPA 2013; RA 2013). Since the detailed case simulation results are previously presented with model behavior tests (see Ch. 4.1.2), this chapter only focuses on the analysis of overall recovery situations according to high-level recovery planning.

Table 6.4 describes four virtual scenarios for diverse recovery patterns according to governmental resource distribution plans. The values in Table 6.4

indicate how much of the total recovery budget are invested [%] for each restoration project. The experimental values in Table 6.4 are adjusted from the base case within an allowable range as found in the actual policies of previously investigated past disaster situations, in order to show which recovery program is regarded as the most important with more budget investment in each recovery scenario (i.e., underlined values in Table 6.4). The scenarios include diverse budget distribution plans that recovery budgets are unevenly distributed to numerous types of restoration projects with different restoration priorities (Lines 1–4 in Figs. 6.1–6.4). In particular, the first Line (Line 1) shows a base case that has no recovery priority.

**Table 6.4** Case Simulation Scenarios: Diverse Recovery Patterns at the Regional Level

| **Budget Distribution Ratio Per Total Recovery Budget [%]** Budget distribution to diverse restoration projects | | | | | | |
|---|---|---|---|---|---|---|
| **Line No. in Graphs** | **Description of Recovery Patterns** | **General Facilities (R)** | **Critical Facilities (C)** | **Industrial Facilities (I)** | **Transport Infra. (T)** | **Debris Disposal (D)** |
| Line 1 | No recovery priority (Base case) | 20 | 20 | 20 | 20 | 20 |
| Line 2 | Recovery priority of restoring residential/ commercial buildings | **60** | 10 | 10 | 10 | 10 |
| Line 3 | Recovery priority of restoring critical/ industrial facilities | 10 | **30** | **40** | 10 | 10 |
| Line 4 | Recovery priority of transportation system/ debris disposal | 10 | 10 | 10 | **30** | **40** |

* Underlined numbers indicate restoration projects (programs) that are regarded as the most important with more budget investment within each recovery planning scenario.

In addition, Table 6.5 shows the scenarios for governmental policy experiments with a focus on implementing special programs such as temporary housing development in an early phase (Line 5 in Figs. 6.1–6.4), temporary debris movement in an early phase (Line 6 in Figs. 6.1–6.4), and household relocation in a restoration phase (Line 7 in Figs. 6.1–6.4). The scenarios are based on the evidences from actual enforced governmental policies after the past disaster case. This experiment can demonstrates that an understanding of multiple interdependent recovery processes in an overall region can assist with the regional recovery manager's decision-making in implementing valid and diverse recovery plans.

**Table 6.5** Policy Experiment Scenarios: Governmental Policies (Special Recovery Programs)

| **Additional Budget investment (Special Recovery Programs) [%]** | | | | |
|---|---|---|---|---|
| **Line No. in Graphs** | **Description of Governmental Special Recovery Programs** | **Temporary Housing Development (THD)** | **Temporary Debris Move (TDM)** | **Household Relocation (CHR)** |
| Line 1 | No recovery priority (Base case) | 0 | 0 | 0 |
| Line 5 | Temporary housing development in an early stage | **20** | 0 | 0 |
| Line 6 | Temporary debris movement in an early stage | 0 | **20** | 0 |
| Line 7 | Household relocation in restoration stage | 0 | 0 | **20** |

* Underlined numbers indicate restoration projects (programs) that are regarded as the most important with more budget investment within each recovery planning scenario.

### 6.2.2 Simulation Results for Regional-level Recovery Patterns

Based on damage situations and diverse recovery patterns as well as recovery plans (special programs), both the overall restoration process (Fig. 6.1) and recoveries from functional losses in communities' activities (Fig. 6.2) are analyzed. Since regional restoration processes can be severely delayed by a lack of workspaces and by interrupted delivery systems, the debris disposal process (Fig. 6.3) and the functional recovery of a transportation system (Fig. 6.4) are also analyzed.

All values of Fig. 6.1 indicate the overall restoration progress [%] calculated by the ratio between works done and initial works to do at each time. All values of Fig. 6.2 mean the average ratio of shortages in all kinds of required services from damaged facilities/infrastructures [%] (including residential, commercial, public, industrial and transportation) that are essential for refugees. Fig. 6.3 and Fig.6.4 show, respectively, the degree of shortages in construction workspaces by debris [%], and shortages in transportation functions [%] by calculating the ratio between the supplies and demands of these services. In particular, the amounts of required restoration works, functional losses of diverse facilities/infrastructures, and the generation of debris can be analyzed by the disaster intensity from the USGS seismic data retrieval federate and by the estimated damage ratio in the damaged regions.

In the case that there exists a high-priority (i.e., prior budget investment) restoration of residential and commercial buildings at the region (Line 2 in Figs. 6.1–6.4), overall physical structure damage can be more rapidly repaired (Line 2 in Fig. 6.1) compared to a base case (Line 1 in Fig. 6.1). This is of course due to a great number of residential and commercial buildings, accounting for about 50% of the overall built environment in the damaged region (E-Stat 2013). Although these plans can accelerate the overall restoration process to some degree and reduce the shortage of housing, a lack of core services and economic functions caused by the delayed restoration of other facilities (e.g., public and industrial facilities) can more interrupt the recovery of the function of communities' social and economic activities (Line 2 in Fig. 6.2), compared to a base case. This is because social and economic activities in communities depend not only on dwellings but also on many commodities, businesses, utilities, and public services. In reality, the primary purpose of the governmental recovery plans is to recover daily lives of refugees to pre-disaster conditions as soon as possible, rather than to optimize the overall restoration operations. From the results, it is found that the actual budget plan that focused on core infrastructures and critical facilities at an early stage after the 2011 earthquake of Tohoku achieved the primary goal of meeting all kinds of basic requirements of living and recovering the daily life of communities (Lines 1 or 3 in Fig. 6.2) even though it caused delays in housing reconstruction.

The recovery plans that focus on critical facilities such as schools and hospitals, as well as on industrial facilities (Line 3 in Figs. 6.1–6.4), can slightly impede the overall restoration process due to their relatively smaller amount of physical structures that need restoration (Line 3 in Fig. 6.1). This result may be also due to delays in restorations of housing and transportation infrastructures as well as delayed debris clearance that may be more helpful for rapid restoration works. Despite these plans' inefficiency in improving restoration processes, these negative impacts on overall restorations can be ignored due to low sensitivity of these impacts. Rather, this plan may be common and inevitable after disaster due to its more positive influence on satisfying emergency needs for rescue and first aid goods and services, and social and economic supports for refugees (Line 3 in Fig 6.2), compared to other plans (e.g., Lines 2 and 4 in Fig. 6.2).

On the other hand, and compared to other plans, the recovery plans that focus on transportation infrastructures and debris disposal work (Line 4 in Figs. 6.1–6.4) can alleviate and improve the poor work conditions from excessive debris (Line 4 in Fig. 6.3) and the limited delivery capability (Line 4 in Fig.6.4). This plan can thus be effective in reducing delays in receiving, storing, shipping, delivering, managing, or utilizing construction resources. However, this plan may be hard to apply in reality because the effective plan to recover basic functions (e.g., housing, public, and economic services) for refugees will be considered as the first priority (e.g., the case of Lines 1 or 3 in Figs 6.2). From this result, it is found that the best plan for alleviating inconveniences in living may not always be helpful for rapid restoration work operations (Lines 1 or 3 in Fig. 6.3 and 6.4).

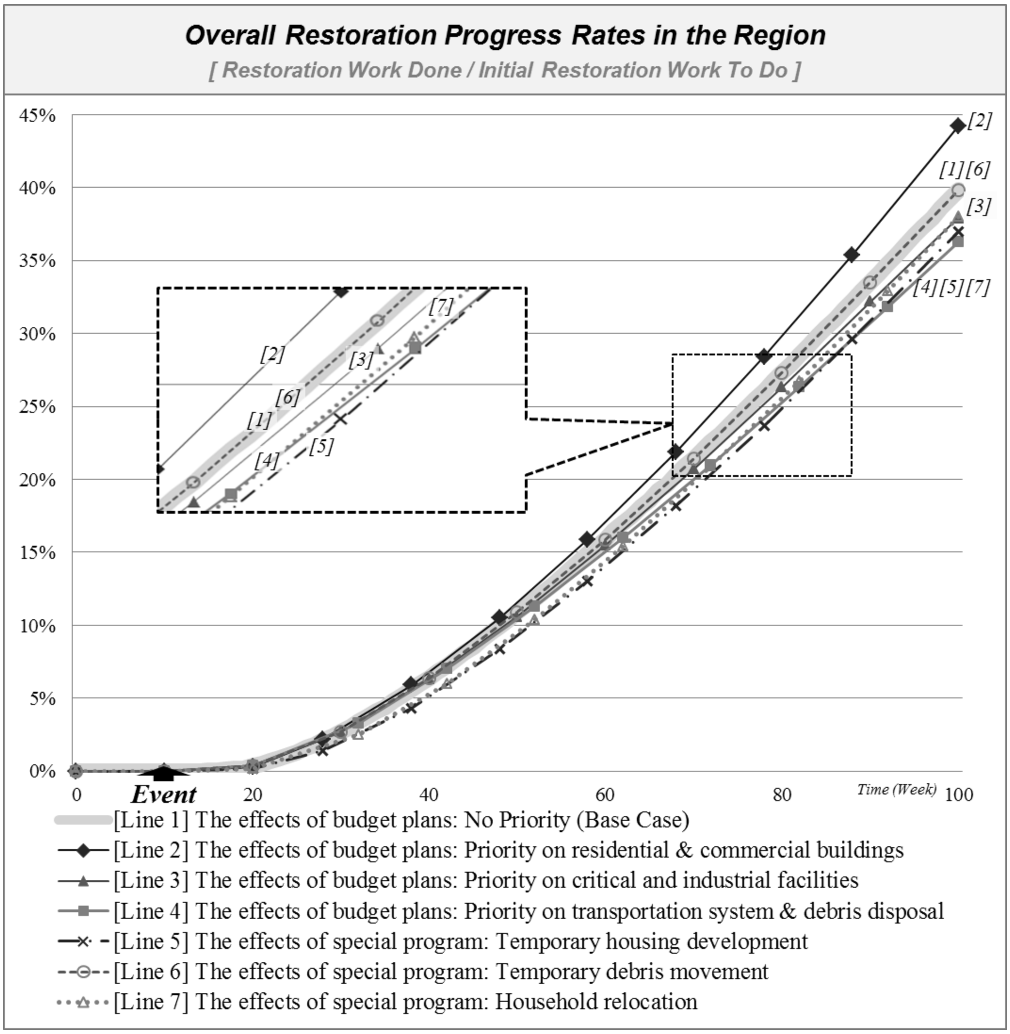



**Figure 6.1** Overall Restoration Efforts of Built Environment at the Region according to Diverse Recovery Patterns and Plans

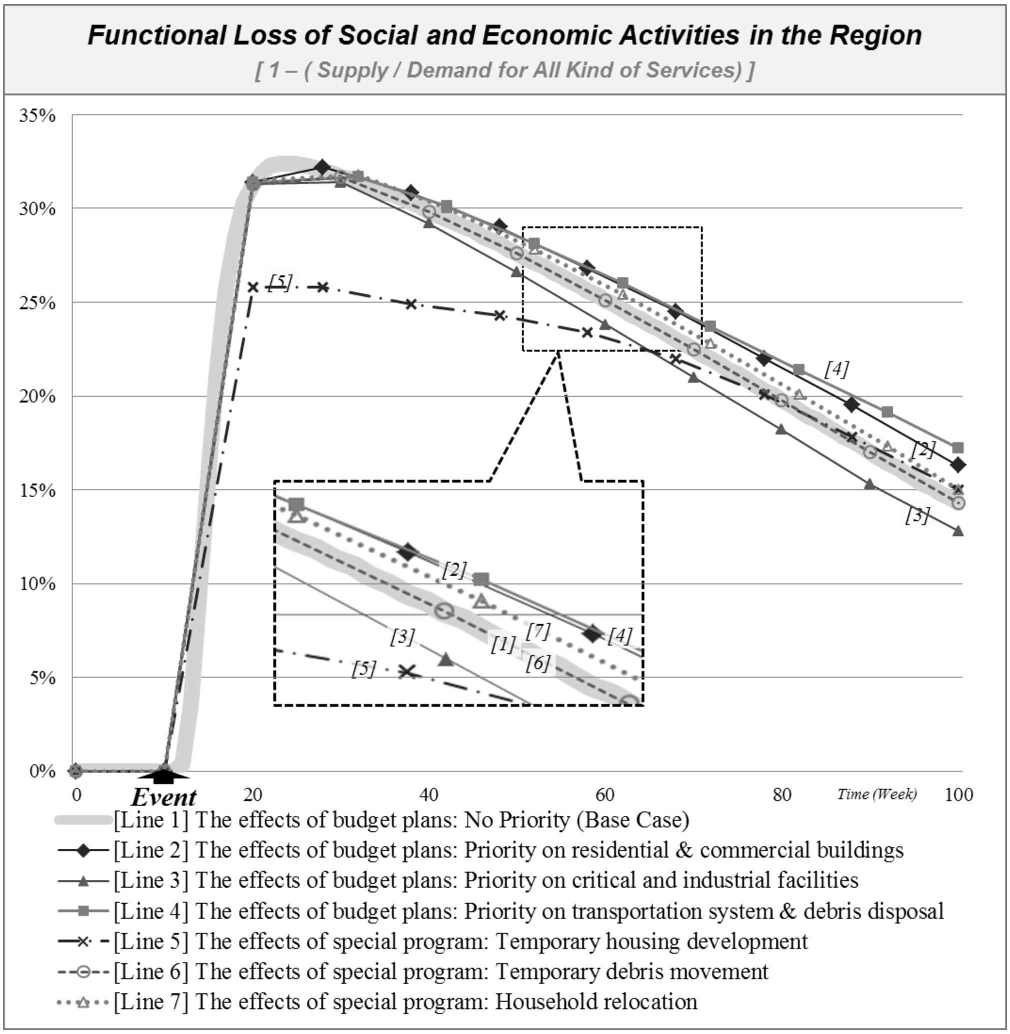



**Figure 6.2** Recovery of Functional Loss of Social Activities at the Region according to Diverse Recovery Patterns and Plans

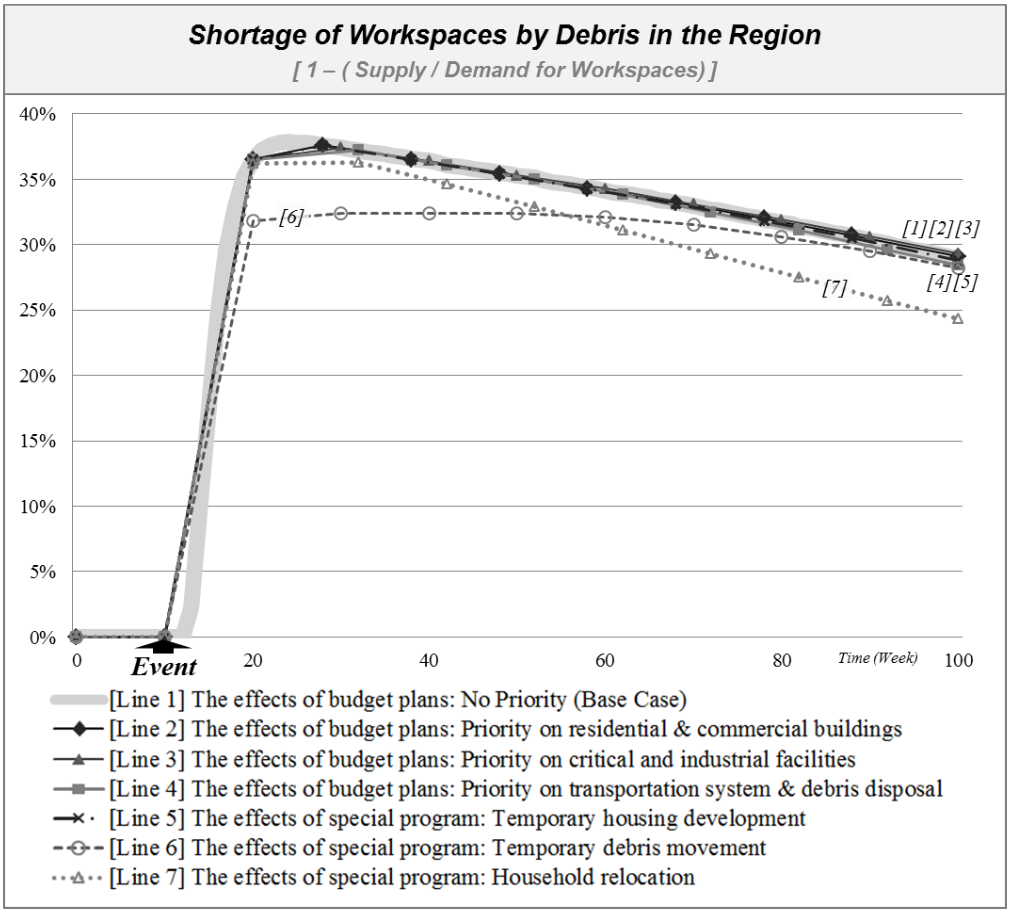



**Figure 6.3** Restoration Work Environment according to Diverse Recovery Patterns and Plans: Shortage of Workspaces by Debris

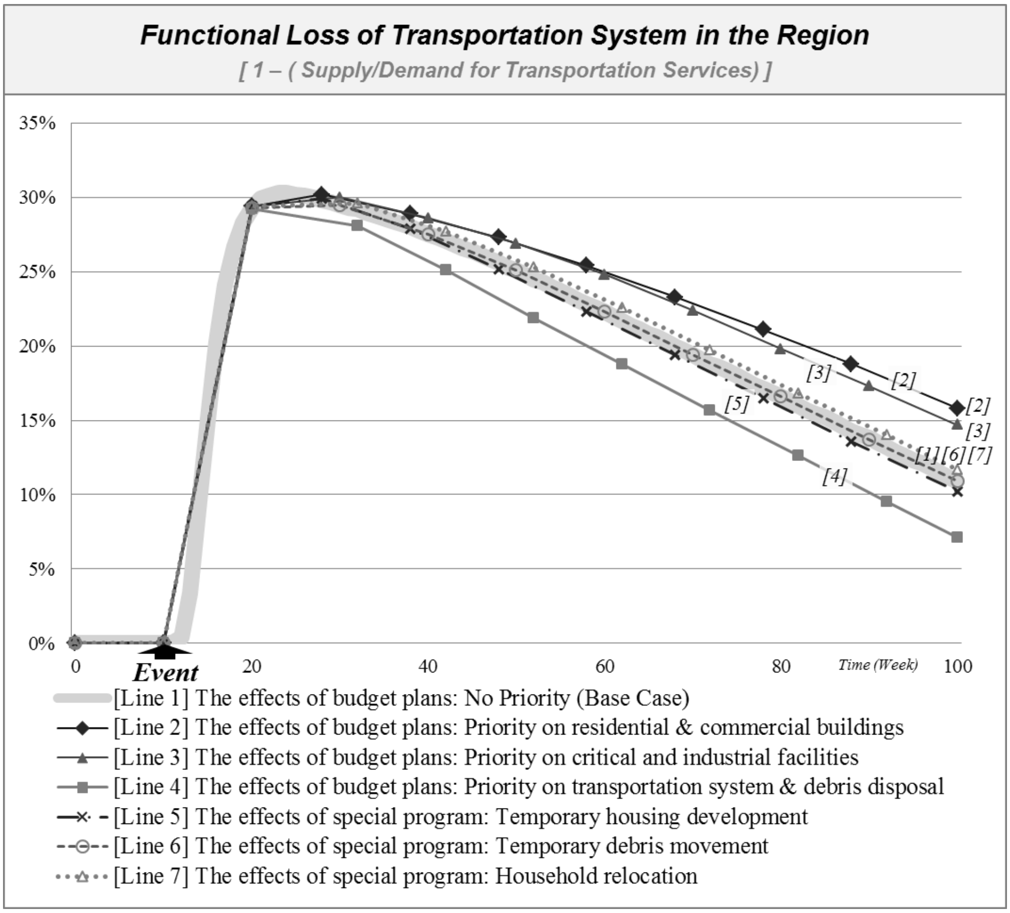



**Figure 6.4** Restoration Work Environment according to Diverse Recovery Patterns and Plans: Functional Loss of Transportation System

### 6.2.3 Experiment of Governmental Recovery Policy Effects

Since the severe shortage of shelters for refugees can be a significant problem in the disaster situation, alternative plans for housing are generally regarded as one of the most important issues in the overall recovery efforts. The implementation of a special program such as temporary housing development in an early phase (Line 5 in Figs. 6.1–6.4), can thus be helpful to temporarily overcome some of these problems (Line 5 in Fig. 6.2). However, a temporary housing development plan has little negative impact on improving an overall restoration process because it requires time, budget, and resources for planning and policy implementation in an early stage. This negative impact can be manageable, though, due to its low sensitivity on restoration processes (Line 5 in Fig. 6.1). Rather, it may be helpful to accelerate the restoration process in the long term with pre-planned temporary housing. This is due to the greater number of chances to allocate resources for restorations of critical and industrial facilities when emergency needs for housing restorations are diminished (i.e., when a shortage of residential services in an early stage is temporarily reduced). More attentions on core facilities and civil infrastructures within this plan can also lead to better supporting emergency responses (e.g., rescue, aid to the injured, and relief efforts) in an early phase.

On the other hand, special programs such as temporary debris movement (Line 6 in Figs. 6.1–6.4) and household relocation (Line 7 in Figs. 6.1–6.4) are respectively intended to avoid a poor work environment during a recovery phase and/or to improve a living environment after a restoration's completion. Although temporary debris movement and household relocation have very little impact on improving either the overall restoration process or the functional recovery (Lines 6 and 7 in Figs. 6.1 and 6.2), these special programs for treating debris can help restoration work to more effectively progress. Since excessive debris requires lengthy debris disposal durations before restoration works commence, initiatives for temporary debris movement to pre-designated areas can significantly reduce the emergency of site clearance and can ensure spaces for restoration works in an early phase (Line 6 in Fig. 6.3). Pre-planned household relocation to less-damaged (or less overwhelmed by debris) areas can also improve poor work conditions in the long-term as there is less need for deploying budget, time, and resources for debris disposal (Line 7 in Fig. 6.3). These programs can thus be effective in accelerating restoration operations in spite of their inefficiency in rapidly recovering social and economic activities, as shown in Lines 6 and 7 in Fig. 6.2.

By describing the effects of governmental policies—including resource distribution plans and special recovery programs—on overall recovery processes caused by the excessive damage of facilities/infrastructures at the region, a better understanding of the complex and multiple recovery efforts at the regional level can be effective in implementing immediate and appropriate regional government-centered recovery plans (e.g., restoration budget plans, temporary housing development plans, and emergency response plans). The analytical capability of the overall damage situations and the recovery system can also support recovery planners in developing effective special recovery programs as well as providing improved institutional strategies (e.g., tax breaks, financial assistance, and the laws for special zones).

## 6.3 Simulation for Facility-Level Restoration Planning

In this section, this study shows the effectiveness of the interactive recovery simulation in facility-level restoration planning. By the interaction with the USGS seismic data retrieval federate and the OpenSees structural response simulation federate, the DES facility-level restoration simulation model can be utilized for damaged facility restoration planning after a disaster by immediately assessing facility damage. In addition, the interaction with a SD regional-level recovery simulation model enables a facility restoration operation analysis to reasonably understand the effects of external project conditions—affected by regional-level damage and recovery situations and governmental plans—on the restoration project. Therefore, the experiments aim to provide a set of relevant information of facility restoration operations according to different structural damage, changing critical external conditions, and project's managerial policies, and then provide insights with regard to following key questions into facility restoration management: (a) does a restoration project's performance and uncertainty in a post-disaster situation considerably differ from that in a pre-disaster situation?; (b) does a facility's type or surrounding situation significantly affect project conditions and a project's performance and uncertainty?; (c) does the effect of post-disaster external conditions on a project's performance and uncertainty significantly vary according to the different facility's damage patterns and ensuing different work scope of the restoration project?; and (d) how do managerial actions affect project's performance and uncertainty in the post-disaster situation?

### 6.3.1 Experimental Design

Based on the developed prototype and the actual data of a past disaster case of the M 9.0 2011 earthquake of Tohoku, this study conducts a case study of facility restoration projects to examine the effects of different facility damage and an unfavorable post-disaster resource supplies and working environments on individual facility restoration operations. The general assumptions are set for case simulations, as the disaster event takes place 10 weeks after the simulation start time, and when the hypothetical case of the damaged building reconstruction project—including a site preparation work, an interior/external material demolition work, a structural reinforcement work, a frame installation work, a deck installation work, and a curtain wall installation work for a 8-story building—starts 20 weeks after the simulation start time. The simulations are also based on the assumption that facility's structural damage is hard to be determined with an observation while the damage of the facility's exterior (e.g., curtain walls) is easily identified by an observation. Therefore, this study focuses on several facilities in different locations with different disaster intensities.

Furthermore, project conditions (e.g., resource availability) for the restoration of a critical facility for public uses (e.g., hospitals and rescue centers) can be better than project conditions for the restoration of a general facility (e.g., a common residential/commercial building), with more intensive governmental supports on core facilities at an early disaster recovery phase.

Therefore this study compares a couple of different types of facilities that have different functionalities respectively (i.e., a critical facility that provides public services such as education and medical services, and a general facility that provides only residential and commercial services), in order to examine the effects of different external conditions—caused by governmental recovery resource distribution plans—on the facility restoration project.

As shown in Table 6.6, This case simulation is implemented by six scenarios for three hypothetical facilities located in different area with different damage levels and for two types of hypothetical facilities with different surrounding situations of restoration projects, as follows: (a) a fully damaged general facility restoration in the severely damaged area under the more unfavorable restoration conditions (Case 1 in Table 6.6); (b) a fully damaged critical facility restoration in the severely damaged area under the less unfavorable restoration conditions with more governmental supports (Case 2 in Table 6.6); (c) a partially damaged general facility restoration in the less severely damaged area under the more unfavorable restoration conditions (Case 3 in Table 6.6); (d) a partially damaged critical facility restoration in the less severely damaged area under the less unfavorable restoration conditions with more governmental supports (Case 4 in Table 6.6); (e) an only exterior-damaged general facility restoration in slightly damaged area under the more unfavorable restoration conditions (Case 5 in Table 6.6); and (f) an only exterior-damaged critical facility restoration under the less unfavorable restoration conditions with more governmental supports (Case 6 in Table 6.6).

The six cases for the experiments of post-disaster facility restoration projects in Table 6.6 consider the different structural damage patterns of facilities into restoration operations by the interactions with damage simulations using both the USGS seismic data retrieval federate and the OpenSees structural response simulation federate. These scenarios also consider diverse post-disaster restoration conditions into facility restoration operations by the SD-DES interactions.

**Table 6.6** Case Simulation Scenarios: Diverse Restoration Situations at the Facility Level

| **Facility Types and Project Conditions** / **Facility Locations and Damage** | **General Facility Restoration** More unfavorable post-disaster restoration conditions | **Critical Facility Restoration** Less unfavorable conditions under the governmental supports |
|---|---|---|
| Fully damaged building restoration located in severely damaged area (100% structure damage) ***Lat.: 38.9; Long.:141.5*** | **Case 1** | **Case 2** |
| Partially damaged building restoration located in less severely damaged area (50% structure damage) ***Lat.: 39.7; Long.:141.2*** | **Case 3** | **Case 4** |
| Only-exterior damaged building restoration located in slightly damaged area (0% structure damage) ***Lat.: 37.1; Long.:139.4*** | **Case 5** | **Case 6** |

### 6.3.2 Case Simulations and Results for Facility Restorations

To conduct a comparative analysis between pre- and post-disaster restorations, this study firstly simulates a fully-damaged building restoration project in the normal conditions (i.e., pre-disaster situations), as a base case. Fig. 6.5, which was tested using a case example in the previous figure, Fig. 4.11, shows the results of only simulating a DES model without any interaction with other simulations. By comparing the restoration operations in the post-disaster situation using the interactive simulation with a base case, the simulation results can show the need for the interactive simulation in post-disaster facility-restoration planning. A case simulation in Fig. 6.5 displays reconstruction works of a fully damaged facility completed in the 40–50 weeks after the project's commencement in the pre-disaster situation where the surroundings are assumed to be normal.

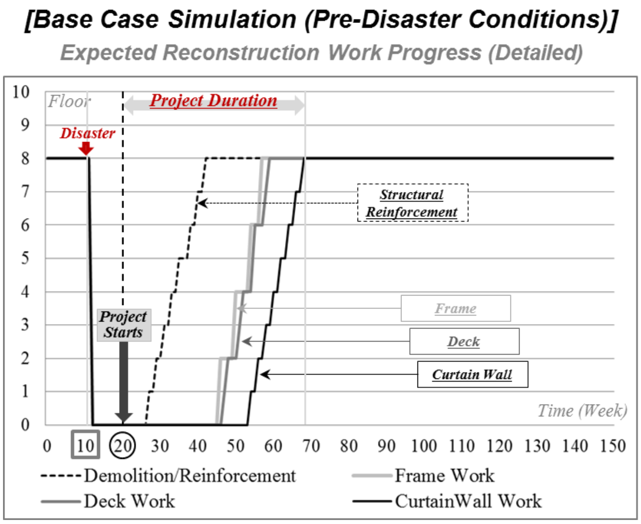



**Figure 6.5** Fully Damaged Facility Restoration Operations in the Normal (Pre-disaster) Conditions (Base Case)

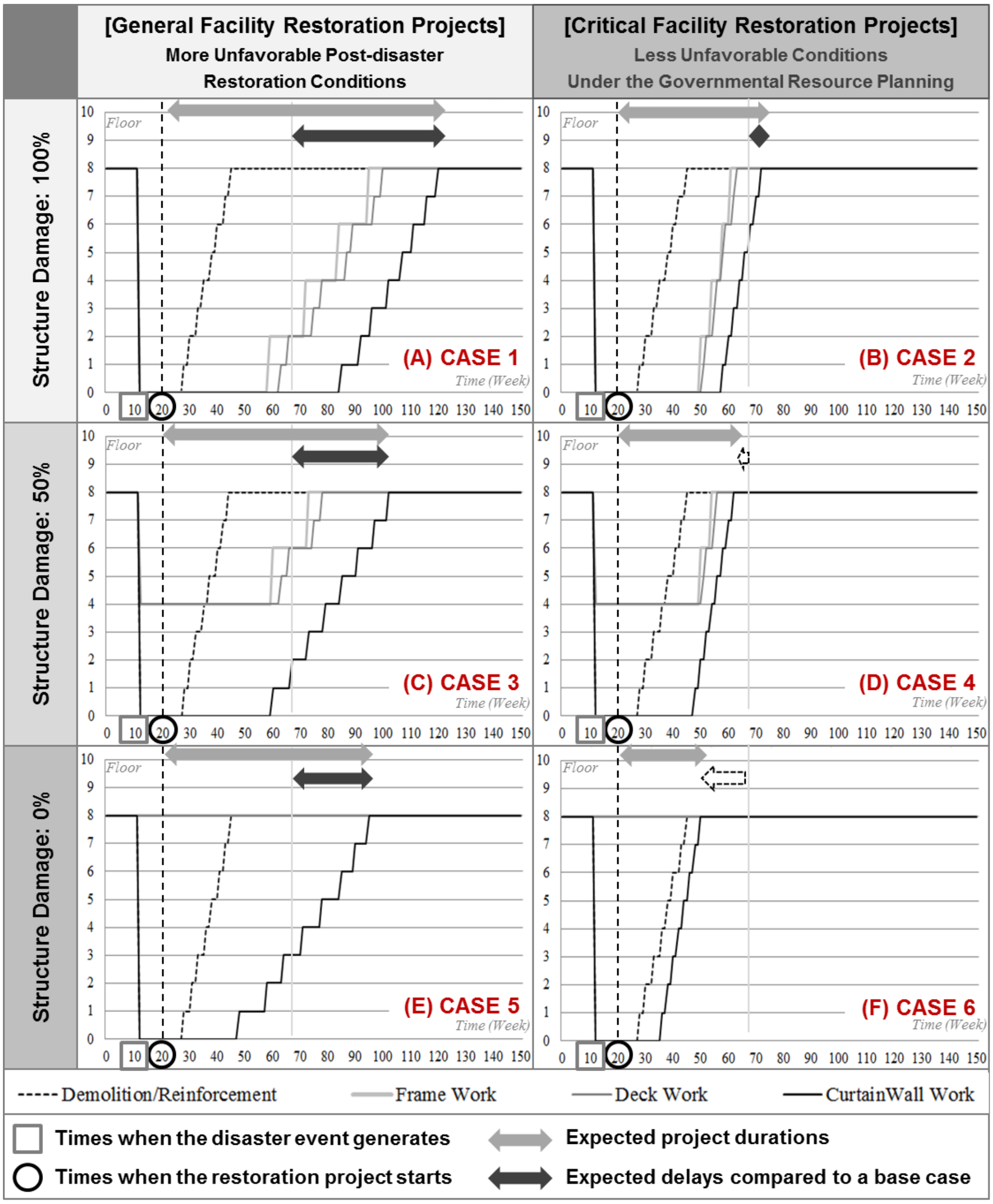



**Figure 6.6** Simulation Results for Detailed Restoration Operations of Facilities by Different Damage and External Conditions

Fig. 6.6 shows the effects of different damage patterns and diverse unfavorable project conditions on the damaged facility restoration operations. In particular, facility's damage differences are due to their locations because disaster intensities vary throughout the region. Restoration project conditions can also be different according to the facility's importance because critical facilities can have more opportunities for governmental supports under the regional-level resource distribution planning.

As shown in Graph (A) of Fig. 6.6 (Case 1), the fully damaged general facility restoration shows significant project delays—delays of more than 50–60 weeks—in the unfavorable post-disaster conditions compared to a base case. As mentioned before, this is due to problems in resource availability and delivery as well as excessive debris that interrupt construction activities. Massive debris from structure damage particularly leads to the lengthy duration of restoration preparation (e.g., demolition, debris disposal, and structural reinforcement) at an early stage compared to the normal situation. After finishing restoration preparation, though, a lack of resources has continuously greater influences on project delays in the long term.

In the post-disaster recovery situation, these resource problems may gradually be improved by alleviating project delays as overall recovery processes in the surrounding regions progress. However, the chances for resource acquisition of general building restoration projects can be more limited for a long period, compared to those of other core facility types (e.g., Graph B of Fig. 6.6) within limited resources. This is due to the massive numbers of restoration projects for common residential and commercial buildings and their relatively lesser need for urgent restoration. The two years after the 2011 earthquake of Tohoku, in reality, saw an overall restoration of general buildings of about only 30%; this was due to the excessive need for reconstruction, while other core facility restorations were almost done within the same time (RA 2013). From this actual recovery tendency, it is found that delays in residential/commercial building restoration operations can be more excessive than expected, and more excessive than other core facilities' delays. However, under the governmental planning that aims at rapidly recovering core facilities with more supports, the project delays of critical facility restoration—that requires urgent functional recovery due to societal needs—can be alleviated much faster as shown in Graph (B) of Fig. 6.6 (Case 2). As a result, the simulation results imply the significantly different impact of external recovery conditions on the progress of facility restoration operations according to the type and importance of the facility. These results also highlight the necessity of comprehensively considering surrounding areas and external conditions for post-disaster construction project management.

Although both SD and DES models are developed by respectively applying actual data, performing an accurate analysis of post-disaster restoration may be hindered due to the lack of detailed data availability as well as the rapidly and unpredictably changing internal and external circumstances (e.g., aftershock and diverse policy implementation).

Nonetheless, an evidence of the negative effects of post-disaster restoration conditions in Graph (A) of Fig. 6.6 can be supported by previous research efforts that analyzed the mounting needs for and shrinking supply of recovery resources (Orabi et al. 2010; Olshansky et al. 2012). In addition, numerical estimation for the availability of resources and commodities in Holguin-Veras and Jaller (2012)'s research showed similarity with the presented model's behavior. This empirical evidence supports that this study's findings can warrant the usefulness and analytical capability of SD-DES interacted simulation. Although the primary purpose of this study is to model both internal restoration processes and their dynamically changing critical external conditions, more empirical studies must be conducted to fully validate the interacted model before it can be applied to the real world situations.

On the other hand, Graph (C) of Fig. 6.6 (Case 3) displays the delays in reconstruction works of a partially damaged general facility can be shortened compared to fully damaged facility restoration as shown in Graph (A) of Fig. 6.6. Due to a less damage and a reduced work scope and process, reconstruction works of a partially damaged general facility may be completed approximately 10–20 weeks before the completion of a fully damaged facility. However, significant delays—delays of more than 30 weeks—are also shown for a partially damaged general facility in the post-disaster conditions compared to a base case. Although unfavorable restoration conditions can of course be significant to partially damaged facility restoration projects of both general and critical facilities, a less amount of restoration works by partial damage can reduce overall project durations compared to a base case, especially as shown in the critical facility restoration with more governmental supports (Case 4, Graph D of Fig. 6.6). This information of expected project duration depending on various facility damage and an ensuing different work scope and process can be effective for implementing a valid restoration plans in an early stage especially when the facility damage is hard to be estimated with an observation, and when damage assessment requires lengthy durations because of the massive amount of damaged facilities in the whole region that wait deliberate structure examination by lacking skilled engineers.

The information of different project durations according to different project conditions and damage patterns can be found in Graphs (E) and (F) of Fig. 6.6 (Cases 5 and 6), which show the restoration operations of the buildings (both general and critical facilities respectively) that only exteriors are damaged. The detailed analysis of post-disaster facility restoration operations using DES can provide the information of what kinds of restoration activities are required and how each activity progresses. The examples of Graphs (E) and (F) of Fig. 6.6 particularly shows only a curtain wall installation work is required because no structural damage is estimated.

As a result, due to the uncertain and complex post-disaster situation, a deeper understanding of external restoration conditions and an immediate assessment of facility damage are essential for rapid and appropriate project planning during the early recovery phase. As shown in Fig. 6.6, in detail, an understanding of the limited restoration conditions and ensuing possible delays of project durations using the SD-DES simulation can be helpful for restoration work scheduling. In addition, both the approximate and rapid assessment of facility damage patterns and determination of the amount and the type of required repair/reconstruction works (e.g., demolition works, structural reinforcement, frame installation, or finishing works) using an interactive recovery simulation can support project planners in project work scope decisions.

Since facility restoration plans need to be rapidly implemented in a chaotic situation where the repair duration and cost are highly variable (Pachakis and Kiremidjian 2004), the main interest in rapid restoration project planning may recognize inherent uncertainties in project duration and cost, by offering both a deeper understanding of limited external conditions and an information of approximate facility damage that needs to be repaired. This study thus conducts sensitivity analyses of the duration variation probability of the restoration projects for not only a base case but also six experimental cases in previous table, Table 6.6. Sensitivity analysis in a sample of 200 simulations is conducted based on the Monte-Carlo simulation using random values of stochastic variables (e.g., activity durations, and probability that external conditions get better or worse by chance). Table 6.7 offers the summary of sensitivity analysis for possible variations of project durations with a base case and six simulation cases.

**Table 6.7** Summary of Sensitivity Analysis Results

| Categories / Restoration Simulation Cases | Possible Project Durations (Weeks) | | | |
|---|---|---|---|---|
| | **Average Value** | **Minimum Value (A)** | **Maximum Value (B)** | **Variation (B-A)** |
| *Fully damaged facility in normal conditions (Base case in Fig. 6.7)* | *49* | *47* | *52* | *5* |
| Fully damaged general facility in more unfavorable restoration conditions (Case 1 in Fig. 6.8) | 100 | 55 | 112 | 57 |
| Fully damaged critical facility in less unfavorable restoration conditions (Case 2 in Fig. 6.8) | 52 | 48 | 66 | 18 |
| Partially damaged general facility in more unfavorable restoration conditions (Case 3 in Fig. 6.8) | 82 | 44 | 95 | 51 |
| Partially damaged critical facility in less unfavorable restoration conditions (Case 4 in Fig. 6.8) | 42 | 40 | 51 | 11 |
| Only-exterior damaged general facility in more unfavorable restoration conditions (Case 5 in Fig. 6.8) | 75 | 44 | 89 | 45 |
| Only-exterior damaged critical facility in less unfavorable restoration conditions (Case 6 in Fig. 6.8) | 30 | 26 | 37 | 11 |

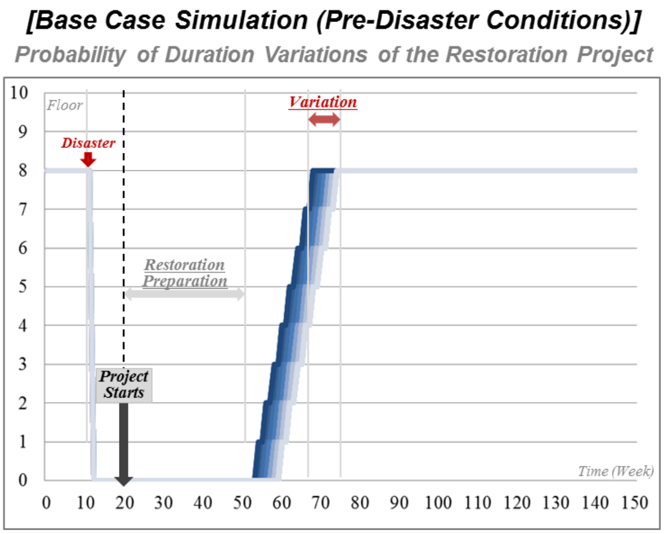



**Figure 6.7** Project Duration Variations of a Fully Damaged Facility in the Normal (Pre-disaster) Conditions (Base Case)

Fig. 6.7 shows the duration variation of the fully-damaged facility restoration in a pre-disaster situation by only simulating a DES model. In the pre-disaster condition, possible project duration variations—mainly determined by solely intrinsic on-site conditions—are expected to be insignificant as shown in a base case. Since project managers generally have a lot of experience in these variations at the normal situation, these variations can easily be managed in the project planning phase.

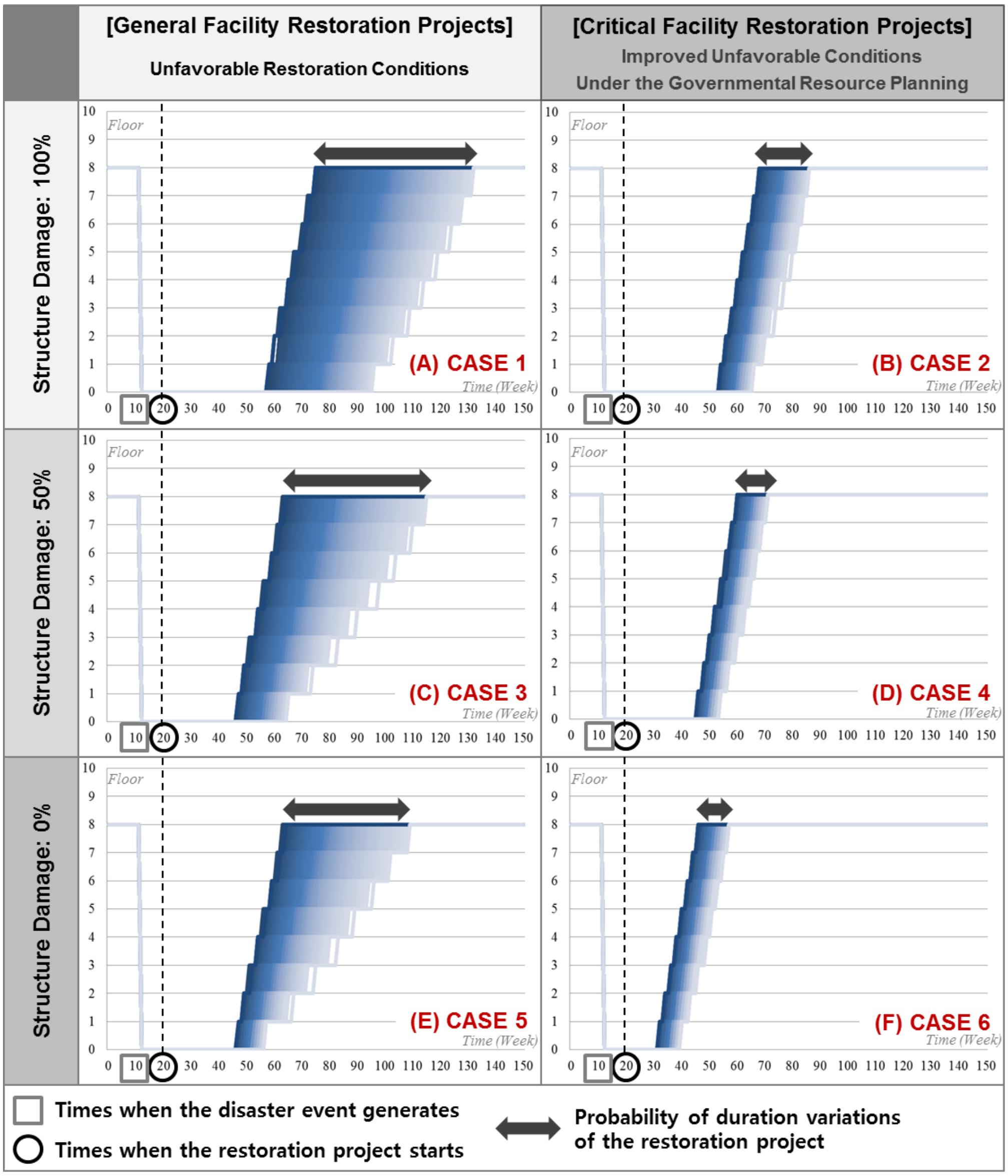



**Figure 6.8** Sensitivity Analysis Results for Project Duration Variations of Facilities by Different Damage and External Conditions

However, for the post-disaster situation, as shown in Fig. 6.8, the high possibility of remarkable duration variations, which cause confusions in implementing project plans, can be found in the simulation results of restorations for a fully damaged general facility (Case 1, Graph A of Fig. 6.8) and a partially damaged general facility (Case 3, Graph C of Fig. 6.8), as well as an only exterior-damaged general facility (Case 5, Graph of in Fig. 6.8). Although a partially damaged facility restoration requires shorter project duration than a fully damaged facility restoration, the possible project duration of a partially damaged facility restoration is also highly variable due to the significance of uncertain project conditions. These results imply that some facilities can easily acquire restoration resources at an early time (i.e., an optimistic situation) while some can be extremely delayed (i.e., a pessimistic situation) depending on various restoration conditions.

On the other hand, possible project duration variations in the critical facility restoration projects can be less variable according to governmental resource distribution plans although restoration project conditions are uncertain and damage patterns of structures are different, as shown in the Graph (B) (Case 2), Graph (D) (Case 4), and Graph (F) (Case 6) of Fig. 6.8. In reality, it was found that the resource availability and the working environments for each facility restoration tended to considerably vary, according to various damage patterns in the neighboring area, the functional importance of the facility, political issues, and numerous unforeseen reasons. To more effectively implement restoration plans in a chaotic situation when there was insufficient surrounding information, as shown in Fig. 6.8, an understanding of the project's increasing uncertainty caused by variable external conditions is a key in avoiding unexpected disturbances to the project. An additional key insight arises regarding the different effects of external conditions on restoration projects according to diverse project's work scope and processes by different facility damage levels, which can be helpful for implementing reliable contingency plans in terms of project cost and time.

### 6.3.3 Experiment of Managerial Policy Effects

In this section, this study demonstrates what kinds of information can assist with the project manager's decision-making in the approximate planning phase (in particular, determining project initiation time) through a policy experiment. When limited resource supplies and working environments significantly affect the general restoration projects of a fully damaged facility, Graph (A) and Graph (B) of Fig. 6.9 imply that reconstruction durations greatly depend on the project's start time after a disaster event. In the case of a fully damaged critical facility restoration project, as shown in Fig. 6.10, the project's start time also affect the uncertainty of a project's duration due to the significance of negative restoration conditions at an early project phase. However, the impact of project start time on reducing project's uncertainty of a critical facility is less significant than the impact on general facility.

If restoration projects of both fully damaged general and critical facilities are initiated shortly after the occurrence of the disaster (an early project starts after week 20 as shown in Graph A of Fig. 6.9 and Graph A of Fig. 6.10), the entire project durations will become highly unpredictable due to the projects' increasing uncertainty and the inevitably that they will be prolonged until restoration resources are available. In the case that the major goal of restoration is to minimize the facility's operation profit losses or faster completions, the managerial action of an early project start may be essential for recovering profit losses and meeting the societal needs with earlier project completion even though overall project durations and costs can increase. In the damaged critical facility restoration project, this managerial action of an early project start (Graph A of Fig. 6.10) can particularly be aggressively applied due to facility's important functions on regions and due to a less uncertainty of project durations with the expectancy for governmental supports, compared to a high uncertainty of project durations in the general facility restoration (Graph A of Fig. 6.9).

However, both the Graph (B) of Fig. 6.9 and Graph (B) of Fig.6.10 show more effective reconstruction work processes and reduced project's uncertainties as the project initiations are postponed to avoid unfavorable restoration conditions (a postponed project starts after week 50 as shown in Graph B of Fig. 6.9 and Graph B of Fig. 6.10). In terms of both time and cost management, the latter managerial action of postponed project starts displays more efficient behavior by reducing both wasted time and the indirect cost of running the unproductive site.

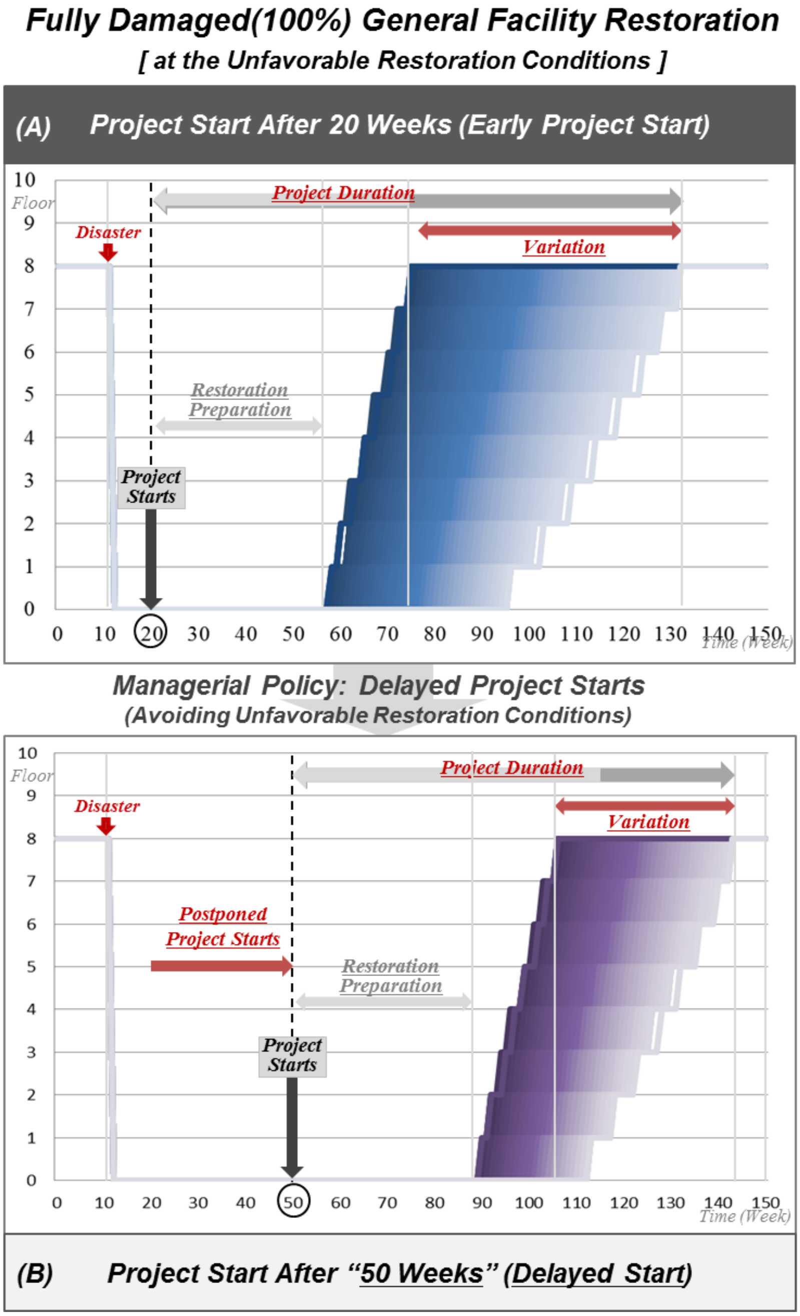



**Figure 6.9** Experimental Results of Managerial Actions on General Facility Restoration: Decision on Project Start Time

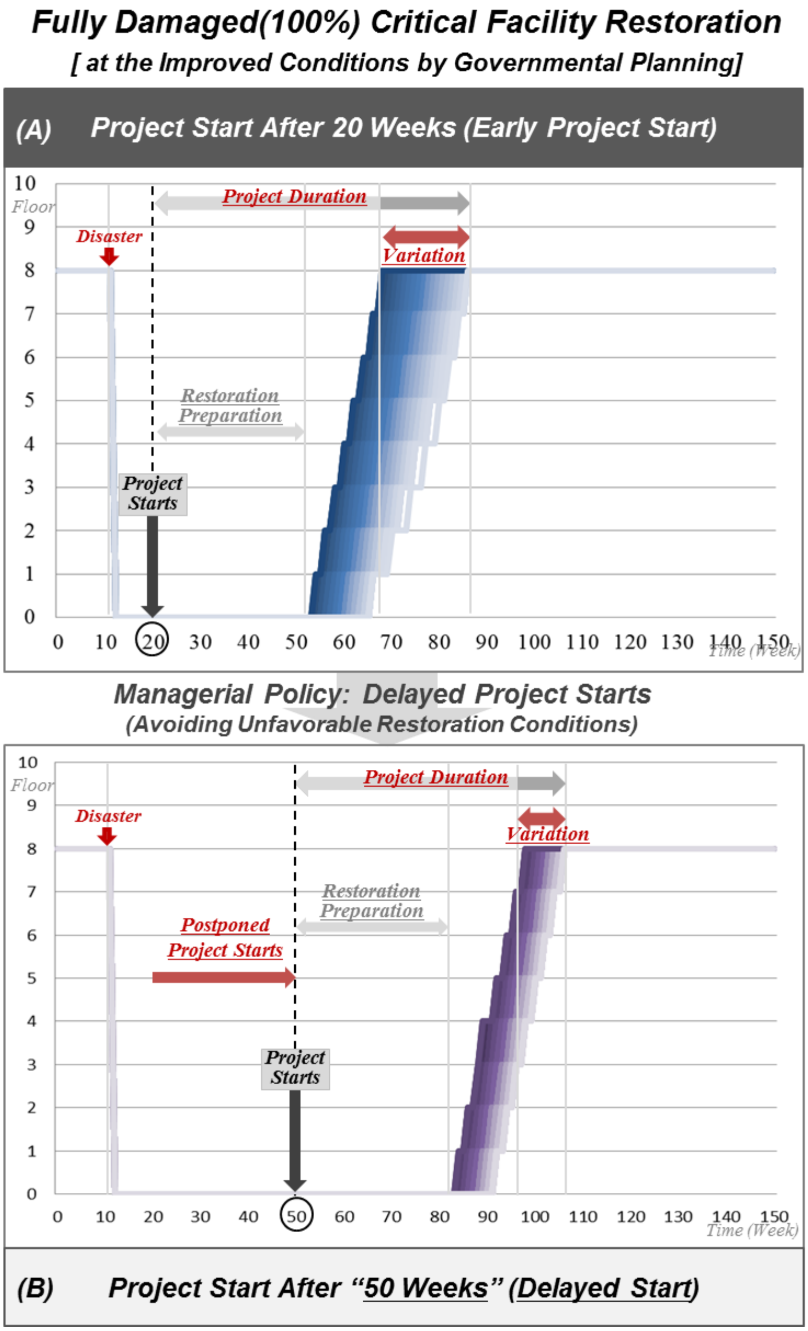



**Figure 6.10** Experimental Results of Managerial Actions on Critical Facility Restoration: Decision on Project Start Time

If a project's uncertainty is foreseeable to some degree and can be gradually resolved during the project's execution such as critical facilities with governmental supports, the project can be commenced earlier with active crisis management strategies for coping with a considerable uncertainty, such as developing flexible scheduling or preparing contingency policies (Pich et al. 2002; Herroelen and Leus 2005) (e.g., Graph A of Fig. 6.10). However, in a chaotic post-disaster recovery situation where sudden changes may occur and unfavorable external conditions are hard to be controlled such as general facility restoration projects, the late project start strategy for avoiding project uncertainty in a passive way (e.g., Graph B of Fig. 6.9)—which is one of the strategies for handling uncertainty suggested by Pich et al. (2002)—can be more effective at preventing the possibility of a project failure regarding project time and cost, compared to pushing the project to be started earlier with inadequate information (e.g., Graph A of Fig. 6.9). The late project start strategy also provide time for more accurately assessing facility damage and implementing detailed work scheduling before the project commencement.

By suggesting the effects of managerial policies—particularly different project start times—on reconstruction work progress of both general and critical facilities respectively, these results can be utilized to assist project managers during the approximate project planning phase by providing information on possible variations on and the extrinsic uncertainty of the project, especially when encountering difficulties in resource supplies and effective work processes as well as difficulties in work scope decisions.

Further, by modeling diverse critical external project conditions (e.g., weather changes, fluctuations in material demand and prices, and/or changes in workforce supply market) in more detail, the interactive simulation framework in this study has the potential to be widely applied to cope with various construction project uncertainties.

## 6.4 Summary and Discussions

In this chapter, for the purpose of providing the effectiveness of the interactive simulation in satisfying diverse analytical requirements for a multiple levels and types of disaster recovery management in different situations, the case simulations using different combinations of simulations were conducted. Before conducting simulations, this study tested the interactive recovery simulation with a focus on the verification of the synchronization and dynamic data exchange among previously tested recovery simulation model components and among different federates, in order to obtain reliable simulation results. Based on the tested prototype, simulation results showed the need for diverse combinations of simulations for different levels of recovery planning because their required information on damage and recovery situations are different. For example, since regional-level recovery planning is mainly implemented by the government with a focus on resource distribution plans and special recovery programs (e.g., debris treatment, temporary housing development) with a holistic view, overall damage patterns and losses at the whole region needs to be comprehensively estimated and multiple recovery efforts for diverse types of facilities/infrastructures among different sub-regions also need to be comprehensively understood. On the other hand, since facility-level restoration planning is mainly implemented by the local communities and private sectors with a focus on project development (e.g., work scope decisions, scheduling, and resource procurement plans), detailed structural damage needs to be assessed and detailed reconstruction processes also need to be captured. In addition, an understanding of changes in external project conditions over time and their dynamic effects on a facility restoration project are essential because facility's surrounding damage and recovery situations as well as high-level recovery plans significantly affect facility restoration projects in a post-disaster situation.

Based on different functional requirements of simulations, both regional- and facility-level recovery simulations were respectively conducted with different combinations of simulations. First, simulation results for regional-level recoveries using an SD implied that regional recovery processes can significantly vary with overall damage patterns, multiple recovery efforts, and their associated interdependencies as well as recovery plans. In particular, an associated interdependency and relative importance among the built environment functions needs to be considered as the most important issue when implementing recovery plans. With this consideration, governmental recovery plans can play a significant role in improving the overall recovery processes. In addition, it was found that the main purpose of governmental plans is generally recovering the inconvenient and impoverished daily lives of populations to pre-disaster conditions as soon as possible by swiftly recovering diverse functions of the built environment, rather than optimizing the overall restoration operations.

Although well-implemented recovery plans can be effective to some degree in terms of alleviating the negative impacts on the repair/reconstruction operations, there are obstacles in effectively improving the restoration process as expected under the common objective to meet all kinds of basic requirements of living. In other words, the best plan for recovering daily life in a region may not always be helpful for rapid restoration work operations. In this situation, the timely use of government-centered recovery programs that each have advantages—including temporary housing, temporary debris movement, and household relocation—has the potential to support not only the rapid functional recovery of the built environment but also the improvement of the poor restoration work environment. Government-centered recovery programs may be more effective when they are pre-planned and well-combined.

By conducting a comprehensive analysis, it is expected that a better understanding of both the overall damage patterns and ensuing complex and multiple recovery efforts at the regional level can be effective in implementing immediate and appropriate regional government-centered recovery plans (e.g., restoration budget plans, temporary housing development plans, and emergency response plans). The analytical capability of the overall recovery system can also support recovery planners in developing effective special recovery programs as well as in providing improved institutional strategies (e.g., tax breaks, financial assistance, and the laws for special zones) with a holistic view. Furthermore, policy implications and lessons learned from a past recovery case can be useful in providing specific guidelines for future disaster events, such as how limited resources need to be distributed and periodically adjusted, how much relief efforts are required over time, and/or which facility functions need to be more rapidly recovered. These research contributions can eventually be helpful in satisfying the requirements of both rapid structure restoration and the swift recovery of daily lives in populations

Second, simulation results for facility-level restoration using a DES implied that the restoration projects' uncertainty tends to be highly variable in a post-disaster situation, according facility's damage patterns, the importance of facility's functions, and facility's surrounding damage and recovery situations. In particular, this study's outcomes provided key insights into post-disaster restoration planning, including to what extent the facility damage is different according to disaster severity and facility's locations and structures, to what extent the restoration conditions are unfavorable according to a facility's type and the facility's surroundings, how critical restoration conditions change over time, and to what extent the different work scope and process by different facility damage and dynamic external conditions impact a project's performance and uncertainty (e.g., variation of a project's duration) over time in a chaotic post-disaster situation. A detailed view of restoration process in the SD-DES simulation also enabled an in-depth analysis of different effects of external conditions on each activity in the restoration projects according to diverse work scope and process by different facility damage levels.

Furthermore, this study also conducted managerial policy experiments to assist with the project manager's decision-making with a focus on the decision on time for project commencement. By conducting policy experiments, it was found that a postponed project initiation plan to avoid unfavorable restoration conditions for damaged facility restorations can be pursued to reduce a project's uncertainty, by alleviating the possible increase in wasted time and the indirect cost of running the unproductive site, and by more accurately assessing facility damage. The plan for postponed project initiation can be especially helpful when external conditions are unforeseeable and uncontrollable and when facility damage is hard to be estimated. On the other hand, if a project's uncertainty is foreseeable to some degree and can be rapidly resolved during the project's execution with intensive governmental supports, an early project initiation can be implemented to minimize the facility's operation profit losses and to meet the societal needs with earlier project completion despite the need for enduring unproductive running of restoration works.

Therefore, by understanding of external restoration conditions and by the accurate and immediate assessment of facility damage, an analysis of project's uncertainty in an early time can assist the project manager in immediate and appropriate project planning in the approximate restoration planning phase, such as project work scope decisions and project scheduling. In particular, required types and amounts of repair activities (e.g., demolition, structural reinforcement, frame installation, or finishing works) can be approximately understood, which can be useful for work scheduling or resource procurement plans. By understanding project uncertainties, reliable contingency plans in terms of project cost and time can also be made. To more effectively implement restoration plans in a chaotic situation when there was insufficient surrounding information, an understanding of the project's increasing uncertainties—caused by variable external conditions and different facility damage levels—is a key in avoiding unexpected disturbances to the project. A test tool for restoration planning scenarios in this study can eventually be helpful for reducing project time/cost and swiftly regaining the functionality of the facility.

To sum up, Table 6.8 shows the different requirements of analyses for both regional- and facility-level recovery planning and detailed analytical capabilities of the interactive recovery simulations with diverse combinations of simulations. Table 6.8 also displays the expected uses of interactive simulations in the recovery phase.

**Table 6.8** Usefulness of Interactive Recovery Simulation

| Categories | Regional-level Recovery Planning | Facility-level Restoration Planning |
|---|---|---|
| Requirement for Analysis | ▪ Overall damage patterns.<br>▪ Overall recovery processes.<br>▪ Interdependencies among facilities and infrastructures.<br>▪ The effects of recovery plans. | ▪ Detailed structural damage.<br>▪ Detailed construction process.<br>▪ External project conditions such as resource availabilities and work delay factors. |
| Combination of Simulation | ▪ Seismic data retrieval for a regional damage estimation.<br>▪ SD simulation for a comprehensive analysis of overall recovery efforts. | ▪ Seismic data retrieval for a disaster intensity analysis.<br>▪ Structural response simulation for a facility damage assessment.<br>▪ DES simulation for a detailed analysis of restoration operations.<br>▪ SD simulation for capturing surrounding project conditions. |
| Abilities of Interactive Simulation | ▪ Comprehensive understanding of overall damage patterns and multiple recovery processes. | ▪ Detailed analysis of facility's damage, facility's surroundings, and facility restoration operations. |
| Detailed Analytical Capability | ▪ Regional damage and functionality losses according to the disaster severity and region's locations.<br>▪ Changes and patterns of multiple recovery processes according to damage and recovery plans.<br>▪ Functional recoveries of diverse facilities and infrastructures according to restoration works and recovery plans. | ▪ Facility damage according to the disaster severity and facility's locations, and structures.<br>▪ Required types and amounts of repair activities.<br>▪ Differences of restoration conditions according to a facility's type and surroundings over time.<br>▪ Changes of a project's performance and uncertainty by different facility damage, external conditions, and managerial policies. |
| Expected Utilization | ▪ Regional-level government-centered recovery plans (e.g., restoration budget plans, emergency responses).<br>▪ Improvement of institutional strategies (e.g., tax breaks). | ▪ Contingency plans (cost and time).<br>▪ Work scheduling or resource procurement plans.<br>▪ Project work scope decisions in an early phase. |

# Chapter 7. Extendable Applications

In this chapter, this study describes the expected future uses of the developed interactive simulation with several examples of detailed application scenarios, based on the reusability and extendibility of the framework and the prototype. Although the interactive disaster simulation in this study focuses on the disaster recovery stages, an HLA-compliant distributed disaster simulation framework can be easily reused and extended to all-time disaster management through the incorporation of other simulation techniques and modules into the developed interactive simulation. By examining analytical requirements for diverse areas of disaster management and determining communication architectures among diverse simulations, the extended framework with regard to diverse damage situations and an overall disaster management cycle is provided. Applications using this extended framework not only deal with recovery stages but also cover emergency disaster response stages including evacuations and emergency recovery efforts. These applications also consider the non-structural damage such as noxious gases by the outbreak of a fire and electric power shortages resulted from an electric power supply system shut-down.

## 7.1 Reusability and Extendibility of the Framework

### 7.1.1 Extended Framework for Interactive Disaster Simulation

Although an interactive disaster simulation developed in this study involves diverse combinations of simulations for multiple recovery planning at both the regional level and the facility level, applications may be limited to recovery stages only with a consideration of structural damage. To assist an overall cycle of disaster management at multiple levels, it is required to consider diverse disaster and damage situations as well as numerous disaster responses. The examples of these considerations include: (a) serial disaster and damage generations such as the outbreak of fires caused by structural damage; (b) the non-structural damage such as an electric power shortage and an operating system shut-down; and (c) emergency disaster responses such as evacuations, rescue efforts, emergency operations and restorations of core infrastructures (e.g., electric power supply system).

To overcome these hurdles in the future, this study provides several scenarios of how the developed framework can be reused and extended to assist an overall cycle of disaster management in complex disaster situations. Fig. 7.1 shows a detailed example of the extended framework for interactive disaster simulations based on the assumption of specific complex damage situations, where a facility's structural damage by a seismic event generates a building fire, and where both structural damage and non-structural damage such as electric power shortages are occurred throughout the region. Since prompt evacuation and rescue planning as well as electric power supply recovery planning need to be made according to these complex damage patterns, the extended framework include not only existing recovery simulation modules but also an evacuation and recue simulation module and an emergency power supply system operation and restoration simulation module.

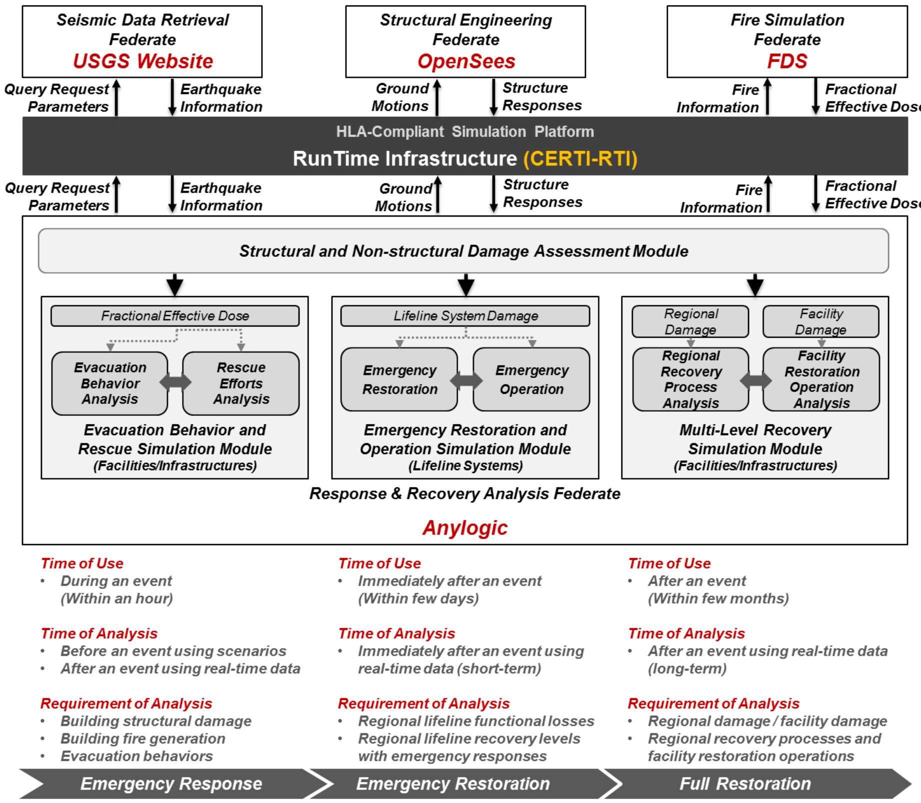



**Figure 7.1** Detailed Examples of Extended Framework for Interactive Disaster Simulation

Since the Anylogic simulation software provides multi-method modeling including not only the SD and DES but also the ABM, this interactive simulation platform facilitates further developments of diverse disaster response simulation modules. For instance, many evacuation and rescue simulations in current research efforts have been developed by using an ABM with a focus on the interactions among agents including evacuees and rescue teams (Chu et al. 2012). Therefore, an evacuation and rescue simulation module can be seamlessly integrated in the interactive simulation to analyze reliable evacuation and rescue processes in a disaster situation, by using detected disaster data from an USGS federate and structural response data from an OpenSees federate. Moreover, the interrupted functionality of the lifeline systems should be taken into account by considering their networks and functional interdependencies. ABM has also been used to represent interdependent emergency restoration processes of lifeline systems by determining strong interdependencies and connections among these systems (Sanford Bernhardt and McNeil 2008), such as electric power supply systems that include power plants and substations. Therefore, the developed interactive simulation can be extended to multiple disaster simulations by incorporating not only an evacuation behavior and rescue simulation module but also an emergency restoration and operation simulation module.

On the other hand, the interactive simulation can interact with other exiting disaster simulation techniques to analyze diverse damage situations focusing on its extendibility. To capture the building fire situations caused by structural damage, for instance, the integration of a fire simulation federate (e.g., an FDS simulation technique) can offer the information on fires and toxic gas diffusions inside the building, which can be helpful for reliable building evacuation and rescue planning. The existing techniques previously incorporated into the interactive simulation can also provide reusability of the developed prototype. For example, the detected seismic data such as the magnitude and epicenter is effectively used to estimate lifeline system damage patterns according to the existing empirical approach for lifeline system damage and loss estimations (FEMA 2003). As shown in these examples, the interactive disaster simulation can be further utilized in diverse disaster and response situations in the future, by reusing communication architectures among simulations in the developed framework and extending the framework through the interactions with other simulation techniques.

### 7.1.2 Design of Extended Disaster Simulation Federation

By determining the interactions between newly incorporated simulation techniques (e.g., FDS) and existing federations that include the USGS, OpenSees, and Anylogic federates, the interactive disaster simulation platform enables both existing and new federates to communicate with each other through interactions in the HLA RTI. As an example case, this section briefly introduces technical requirements for incorporating an FDS simulator into the existing interactive recovery simulation federation, in order to extend the range of use of the developed prototype.

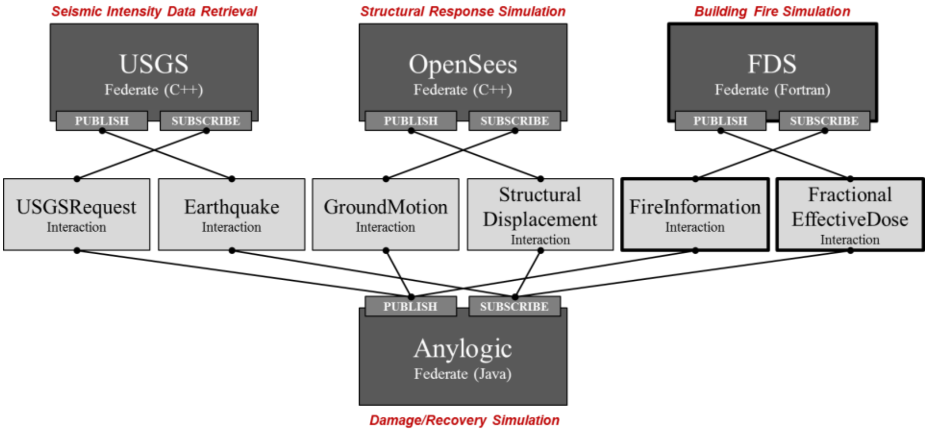



**Figure 7.2** Example of Extended Communication Architecture in the HLA-Compliant Interactive Disaster Simulation

Fig. 7.2 shows the extended HLA-compliant communication architecture that includes interactions with the FDS simulation. While the existing communication architecture in the previous figure (Fig. 5.1) only include four *interactions* such as the *USGSRequest*, *Earthquake*, *GroundMotion*, and *StructuralDisplacement*, the extended federation includes *interactions* with FDS such as *FireInformation*, and *FractionalEffectiveDose (FED).*[19] By passing input values (i.e., ignition point and hear release rate of a fire) from the Anylogic Federate through the *FireInformation interaction*, the FDS federate runs a fire simulation that analyzes the amount of toxic gases including carbon dioxide ($CO_2$), carbon monoxide (CO), and oxygen ($O_2$) at the ignition point at every time. Along with the fire gas data, the FDS federate

[19] To analyze the effects of toxic gases on humans, Fractional Effective Dose (FED), which is an estimation of human incapacitation due to a limited set of combustion gases such as CO, $CO_2$, and $O_2$, has widely been used (Purser 1992).

publishes simulation time and gas velocity values and the *FED interaction* simultaneously sends out five types of data (i.e., levels of carbon dioxide/carbon monoxide/oxygen, simulation time, and gas movement velocity) changing in each time step. These data can be effectively utilized to conduct more reliable building evacuation behavior and rescue simulation. Furthermore, these interactions are included in the existing FOM (see Fig. 5.3) which standardizes names of federate elements for cross-federate access. Fig. 7.3 summarizes the contents of the FOM file with different parameters for each interaction. In detail, *FireInformation interaction* with two parameters (*IgnitionPoint* and *HeatReleaseRate*), and *FractionalEffectiveDose (FED) interaction* with five parameters (*FDSSimulationTime*, *CarbonMonoxide*, *CarbonDioxide*, *Oxigen*, and *GasVelocity*) are newly determined in the extended FOM.

```
(Fed
    (Federation IRSP)
(FedVersion v1.3)
    (Federate "fed" "Public")
    (Spaces)
    (Objects)
    (Interactions
        (Class InteractionRoot BEST_EFFORT RECEIVE
            (Class RTIprivate BEST_EFFORT RECEIVE)
            (Class USGSRequest RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter StartTime)
                (Parameter EndTime)
                (Parameter MinMagnitude)
                (Parameter MinLatitude)
                (Parameter MaxLatitude)
                (Parameter MinLongitude)
                (Parameter MaxLongitude)
            )
            (Class Earthquake RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter EventID)
                (Parameter Time)
                (Parameter Longitude)
                (Parameter Latitude)
                (Parameter Depth)
                (Parameter Magnitude)
            )
            (Class GroundMotion RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter ScaleFactor)
                (Parameter AccelerationFilePath)
            )
            (Class StructuralDisplacement RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter Displacements)
            ) // [Newly Determined Interactions]
            (Class FireInformation RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter IgnitionPoint)
                (Parameter HeatReleaseRate)
            )
            (Class FractionalEffectiveDose RELIABLE TIMESTAMP
                (Sec_Level "Public")
                (Parameter FDSSimulationTime)
                (Parameter CarbonMonoxide)
                (Parameter CarbonDioxide)
                (Parameter Oxigen)
                (Parameter GasVelocity)
            )
        )
    )
)
```

**Figure 7.3** Extended FOM for Interactive Disaster Simulation Federation

## 7.2 Extendible Future Use Scenarios

Based on the extended interactive disaster simulation framework, this study describes several example scenarios for future applications to the all-time disaster management. Fig. 7.4 shows the detailed utilization scenarios to building evacuation and rescue planning. For the preparation of efficient evacuation and rescue efforts in emergency, it is important to consider the effects of a disaster on occupants' and rescue agents' behaviors beforehand (Purser 1996). In this context, the FDS simulator enables interactive disaster simulations to analyze the concentrations of fire toxic gases generated in the building and provide such information to the evacuation and rescue simulation module. In the distributed simulation environment, the fire ignition point can be predicted using the information of structural response and damage provided by the interaction with the OpenSees simulation, and using the facility information such as the hazardous area. FDS then simulates the information of toxic gas diffusions (e.g., CO, $CO_2$, and $O_2$ concentrations) inside the building. By calculating the effects of toxic gases on individuals (e.g., FED level), the evacuation behaviors and rescue efforts are captured in the ABM model with a consideration of human interactions and toxic gas effects on human behaviors. In detail, this simulation investigates possible changes in the movement and decision-making process of the occupants with the effects of fire. The analysis outcome can be utilized at the break of a disaster for fast responses to figure out the crowded areas, which can be helpful for decision making in rescue efforts. In the pre-disaster situation, simulations using disaster scenarios can also be effective to assist evacuation and rescue training.

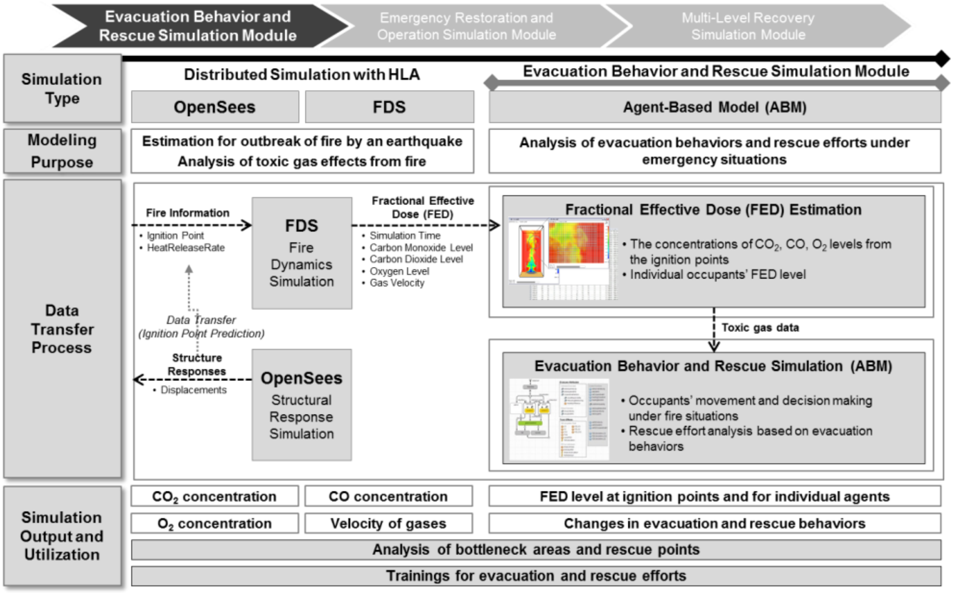



**Figure 7.4** Extended Application Scenario: Evacuation Behavior and Rescue Simulation

Fig. 7.5 represents another application scenario to electric power supply system operation and restoration planning in the emergency response stages. The earthquake information from the USGS seismic data retrieval technique is used not only to estimate facility's damage but also to calculate non-structural damage such as the ratio of expected probability in power supply system's malfunctioning, according to damage and loss estimating functions suggested by FEMA (2003).

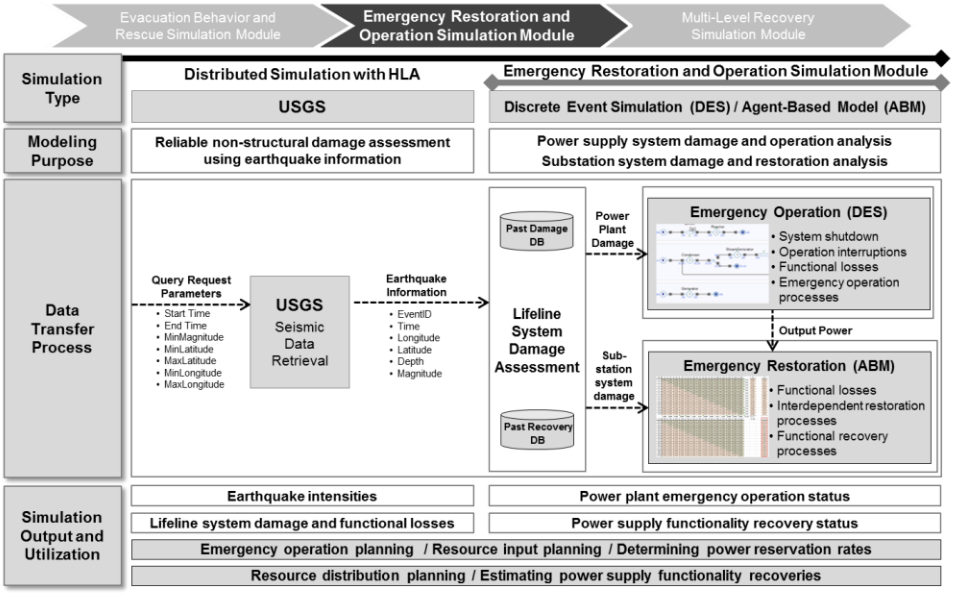



**Figure 7.5** Extended Application Scenario: Emergency Restoration and Operation Simulation

The emergency restoration and operation simulation module can be further involved in the interactive disaster simulation by re-using existing modules in the interactive simulation. In this simulation module, the DES enables to represent emergency operation processes of the electric power supply systems caused by power plant damage while ABM is effective to simulate emergency restorations of interdependent power supply system networks (e.g., substations). This simulation is expected to analyze the recoveries of electric power supply functionalities according to recovery plans such as resource distributions, which assist emergency response planning to rapidly regain lifeline system functions.

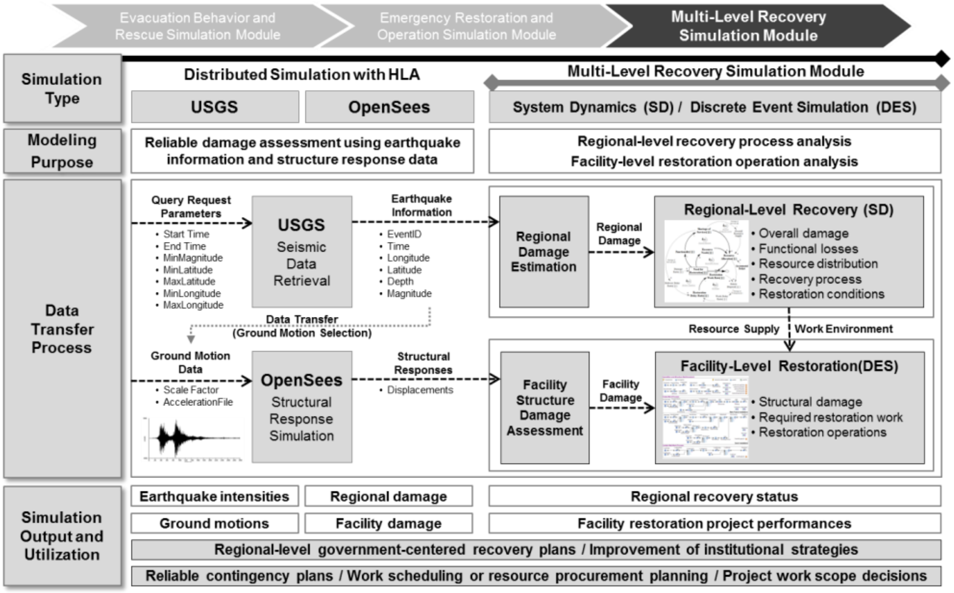



**Figure 7.6** Extended Application Scenario: Multi-Level Recovery Simulation

By summarizing the use of an interactive simulation in the recovery stages, as described throughout this study, Fig. 7.6 shows the application scenarios for both regional- and facility-level recovery planning. As shown in presented example scenarios (Figs. 7.4–7.6), the interactive simulation can be reused and extended to assist all-time disaster management in diverse disaster and damage situations. Since the distributed simulation is capable of running across multiple processes or computers in a distributed network environment, distributed disaster simulation can enable diverse disaster management sectors (e.g., disaster research centers, regional disaster and safety managers, facility manages, and so forth) to communicate and cooperate with each other in a complex disaster situation, as shown in Fig. 7.7. With an integration of numerous databases and information input modules provided by diverse disaster management sectors into a distributed disaster simulation platform, the HLA-compliant interactive disaster simulation is expected to serve as a technical platform in establishing the network-based disaster control tower in the future.

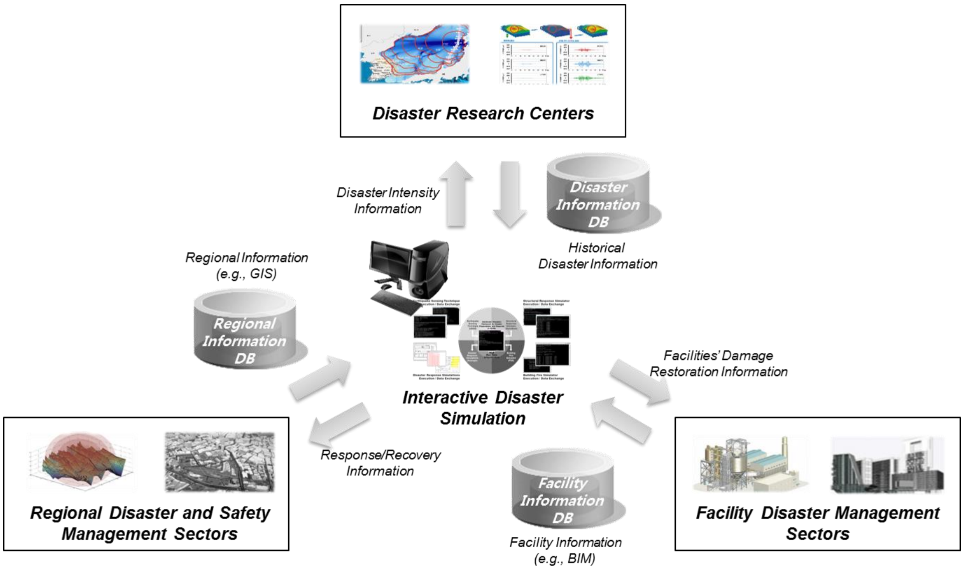



**Figure 7.7** Expected Roles of Interactive Disaster Simulation

## 7.3 Summary

This study provided several example scenarios for extendible future uses of the developed interactive simulation. With the multi-level recovery simulations described throughout this study, future use scenarios included an evacuation behavior and rescue simulation, and an emergency restoration and operation simulation of lifeline systems focusing on electric power supply systems. The detailed descriptions of scenarios showed the reusability and extendibility of the framework into the all-time disaster management in diverse disaster and damage situations. Further, it is expected that the interactive simulation in a distributed network environment support diverse disaster management sectors in communicating and cooperating with each other to implement more effective disaster response and recovery plans.

# Chapter 8. Conclusions

In this chapter, this study describes research results of the developed interactive simulation for post-disaster recovery management. In particular, the abilities of the interactive recovery simulation for satisfying diverse analysis requirements for multiple levels of recovery efforts are presented. Then, this study explains methodological and technical contributions to the body of knowledge in the field of construction and disaster management, with regard to reusability and extendibility of the developed framework in the future. Finally, the limitations and required future works are provided for the further practical uses of this study's outcomes in the real world situations.

## 8.1 Research Results

This study developed an interactive simulation framework for prompt and appropriate post-disaster recovery management in a complex and uncertain disaster situation, by enabling comprehensive and rapid interactive analysis among disaster intensity, damage, and multiple recovery efforts. As a technical approach of the interactive recovery simulation, the distributed simulation through the use of the High Level Architecture (HLA) was applied to promote reusability and extendibility of the interactive recovery simulation in the future, with regard to numerous types of disasters, various damage situations, and diverse and complex disaster response and recovery efforts in diverse facilities and regions performed by different agents at both macro and micro levels of management (e.g., the regional level and the individual facility level). To present the interoperability, reusability, and extendibility of the interactive simulation, this study further developed a prototype of the interactive recovery simulation for both regional- and facility-level damage situations in the aftermath of a seismic event, in order to assist multi-level (i.e., both the regional level and the facility level) recovery planning.

In the interactive recovery simulation prototype, an USGS seismic data retrieval federate (i.e., USGS federate) and an OpenSees structural response simulation federate (i.e., OpenSees federate) are interacted with an SD-DES integrated multi-level damage and recovery simulation federate (i.e., Anylogic federate) by dynamic data exchanges among three federates. According to different analysis requirements among regional- and facility-level recovery planning respectively, different combinations of simulations can be made in the distributed simulation environment.

First, regional-level recovery planning—mostly implemented by the central and local governments—requires a comprehensive understanding of overall damage patterns and multiple interdependent recovery efforts at the whole region. Therefore, an USGS seismic data retrieval federate is interacted with regional-level damage and recovery simulation models in the Anylogic federate. The near real-time detection of disaster information such as the magnitude and the epicentral location from the USGS federate enables a regional-level damage simulation to rapidly and approximately estimate overall damage patterns and functional losses. As a methodological approach, a SD model for regional-level recovery simulations allows for a comprehensive understanding of multiple feedback processes and dynamic changes in the complex and multiple recovery efforts, with a consideration of policy factors. By analyzing recovery situations in an overall region according to damage patterns, multiple interdependent recovery efforts, and governmental plans, interactive recovery simulation at a regional-level can be effective in implementing immediate and valid government-centered recovery plans, which can be helpful in satisfying the requirements of both rapid restorations and the swift recovery of daily lives in populations in an widespread devastated region.

Second, facility-level restoration planning—mostly implemented by the project managers in the local communities and private sectors—requires a rapid and detailed analysis of both structural damage and construction operations for a facility restoration project. Therefore, facility restoration operation simulations need to rapidly consider the facility damage by detecting disaster intensities in near real-time and then analyzing structural responses of facilities. Facility-level damage and restoration simulation models in the Anylogic federate facilitate immediate damage assessment by using the detected earthquake information from an USGS federate and structural response analysis results from an OpenSees federate. As a methodological approach, a DES model for facility-level restoration simulations enables the examination of facility restoration operations according to different facility damage patterns. On the other hand, since regional-level damage and recovery situations affect facility-level restoration projects by determining external project conditions (e.g., resource availability and work effectiveness), an SD-DES interacted restoration simulation offers reliable facility restoration operation analyses, with the ability of understanding the effects of unfavorable external conditions on facility restoration projects over time. By using an interactive recovery simulation at a facility-level, a comprehensive understanding of both facility's damage and facility's surroundings as well as ensuing project's performances and uncertainties in an early time is expected to assist the project manager in immediate and appropriate facility-level project planning.

As described before, diverse combinations of simulations in the interactive recovery simulation can provide appropriate information to respectively assist multi-level (i.e., both regional and facility level) recovery planning performed by different agents—such as governments, local communities, and private sectors—in different situations. By describing diverse future applications of the developed interactive simulations into overall disaster management areas with several examples of detailed scenarios, this study further examined reusability and extendibility of the developed framework through diverse combinations of simulations as well as interactions with other simulations (e.g., a fire simulator, an evacuation behavior and rescue simulation model, or an emergency restoration and operation simulation model).

## 8.2 Research Contributions

The interactive recovery simulation in this study provided generic model structures and a base technology for disaster-related simulations and recovery management by developing the interactive simulation framework and the prototype. By integrating seismic intensity, structural damage, and recovery process and operation analyses of both regional and facility levels, the interactive simulation can be further utilized to analyze various damage and recovery situations when serial and complex disaster are generated (e.g., aftershocks). Furthermore, by figuring out diverse analytical requirements in real-world recovery situations and accumulating knowledge of post-disaster recovery management into the interactive simulation framework based on the existing research, theories, and the actual cases, the research outcome can be practically applied in supporting complex and multiple recovery efforts with relevant insights and policy implications. The areas of multiple recovery management include many types of activities (e.g., loss estimation, debris disposal, and restoration) for numerous facilities and infrastructures (e.g., houses, plants, and transportation systems), performed by diverse agents (e.g., governments, local communities, and private sectors) at different levels (e.g., the regional level and the individual facility level). The interactive analysis approach (i.e., SD-DES interaction) between regional-level recovery processes (i.e., SD analysis) and facility-level restoration operations (i.e., DES analysis) can be further applied in supporting cooperative efforts of disaster recovery management among diverse levels and agents.

Although this study developed the interactive simulation for disaster management only with a focus on recovery efforts after a catastrophic earthquake, the HLA-compliant simulation framework—where innumerable simulations (including disaster detections/simulations, damage simulations, and disaster response and recovery simulations) interact with each other in a distributed simulation platform—can provide future extendibility and reusability. In other words, the interoperability within the integrated sub-simulations potentially provide the simulation with extendibility to numerous types of disasters (e.g., hurricanes, landslides, and fires), to various damage situations including non-structural damage (e.g., power blackout, facility's operating system shut-down, and a leakage of toxic gases), and to the all-time disaster management (e.g., evacuation planning, rescue planning, and emergency restoration and operation planning such as lifeline systems) when diverse disaster-related simulations are interacted with this prototype.

Further, since the distributed simulation is capable of running across multiple processes or computers in a distributed network environment, a distributed disaster simulation can support multiple disaster management sectors in communicating and cooperating with each other to implement more effective disaster response and recovery plans, by providing a platform to establish the network-based disaster control tower in the future.

## 8.3 Limitations and Future Research

Despite the contributions of this study, further development is required to enhance the applicability of the proposed research outcome. First, as a prototype, the facility damage was estimated by only considering structure displacement and using historical ground motion data. In the future, more accurate and elaborate damage assessment needs to be made by analyzing diverse structural responses (e.g., shear forces, deformations, eigenvectors, and node velocities and accelerations) and directly detecting real-time ground motion data.

Second, more detailed geographical and geological information of regions and spatial and structural information of a facility/infrastructure need to be supplemented to more accurately assess both regional and facility damage and analyze multiple recovery processes.

Third, although both structural and non-structural damage (e.g., power blackout and operating system shut-down) should be considered for recovery management, non-structural damage was beyond the scope of simulations in the prototype of this study because non-structural damage can generally be treated at an early phase. In the future, the consideration of non-structure damage will be necessary to facilitate not only long-term recovery planning but also short-term emergency response and restoration planning.

Fourth, to strengthen the usefulness of the interactive simulation, further development of data input module is required. For instance, the integration of the BIM (i.e., building information model), the GIS (i.e., geographic information system), and facility/region information databases into the interactive simulation can enhance the reusability of the developed prototype by rapidly and accurately reflecting both regional and facility information.

Fifth, advanced time management functionality of the HLA-compliant distributed simulation should be added. In the developed prototype, federation time advancement was handled by a single federate only (i.e., Anylogic federate). For federations where multiple federates can advance the federation time, additional functionality within the framework will be further necessary.

Sixth and finally, the integration of diverse types of disaster-related simulators can extend the range of uses of the developed prototype by enabling disaster simulation to cope with complex disaster situations. For instance, an existing FDS building fire simulation can be interacted with the interactive simulation prototype to analyze complex disasters of both earthquakes and fires. In addition, since the OpenSees simulation is used to a lateral load analysis of structures, the prototype in this study can be applied in the post-hurricane situation by using a hurricane sensing module. With the future works described above, the interactive simulation prototype in this study can have the potential to be applied to real-world disaster management by more accurately reflecting reality.

# References

Abdel-Hamid, T. (1993). “Thinking in Circles.” *American Programmer*, 6(5), 3–9.

AbouRizk, S., Hague, S., and Moghani, E. (2009). “Developing a Bidding Game Using High Level Architecture.” *Proceedings of the 2009 ASCE International Workshop on Computing in Civil Engineering*, ASCE, Reston, VA, U. S., 513–522.

AbouRizk, S. (2010). “Role of Simulation in Construction Engineering and Management.” *Journal of Construction Engineering and Management*, ASCE, 136(10), 1140–1153.

AbouRizk, S., Halpin, D., Mohamed, Y., and Hermann, U. (2011). “Research in Modeling and Simulation for Improving Construction Engineering Operations.” *Journal of Construction Engineering and Management*, ASCE, 137(10), 843–852.

Adelantado, M. (2004). “Rapid Prototyping of Airport Advanced Operational Systems and Procedures through Distributed Simulation.” *Simulation*, 80(1), 5–20.

Ahmad, S., and Simonovic, S. P. (2000). “System Dynamics Modeling of Reservoir Operations for Flood Management.” *Journal of Computing in Civil Engineering*, ASCE, 14(3), 190–198.

Alvanchi, A., Lee, S., and AbouRizk, S. (2011). “Modeling Framework and Architecture of Hybrid System Dynamics and Discrete Event Simulation for Construction.” *Computer-Aided Civil and Infrastructure Engineering*, 26(2), 77–91.

Anil, E. B., Akinci, B., Garrett, J. H., and Kurc, O. (2013). “Representation of Damage Information for Post-Earthquake Damage Assessment of Reinforced Concrete Frames.” *Proceedings of Computing in Civil Engineering*, ASCE, Reston, VA, U. S., 825–832.

Anylogic Company Website. “Anylogic Help.” < http://www.anylogic.com/anylogic/help/> (Jun. 22, 2014)

Back, W. E., Boles, W. W., and Fry, G. T. (2000). “Defining Triangular Probability Distributions from Historical Cost Data.” *Journal of Construction Engineering and Management*, ASCE, 126(1), 29–37.

Banks, J., Carson, J. S., Nelson, B. L., and Nicol, D. M. (2005). *Discrete-Event System Simulation*, 4th Edition, Prentice Hall, Upper Saddle River, NJ, U. S.

Berke, P. R., Lyles, W., and Smith, G. (2014). "Impacts of Federal and State Hazard Mitigation Policies on Local Land Use Policy." *Journal of Planning Education and Research*, 34(1), 60–76.

Bommer, J. J., and Acevedo, A. B. (2004). "The Use of Real Earthquake Accelerograms as Input to Dynamic Analysis." *Journal of Earthquake Engineering*, 8(S1), 43–91.

Chen, Y. W., and Tzeng, G. H. (1999). "A Fuzzy Multi-Objective Model for Reconstructing the Post-Quake Road-Network by Genetic Algorithm." *International Journal of Fuzzy Systems*, 1(2), 85–95.

Chu, M. L., Law, K., and Latombe, J. (2012). "A Computational Framework for Egress Analysis with Realistic Human Behaviors." CIFE (Center for Integrated Facility Engineering) Technical Report #TR209.

Chua, D. K. H., Yeoh, K. W., and Song, Y. (2010). "Quantification of Spatial Temporal Congestion in Four-Dimensional Computer-Aided Design." *Journal of Construction Engineering and Management*, ASCE, 136(6), 641–649.

ClientRunner. <http://www.clientrunner.net/> (July 21, 2014).

Cooke, D. L., and Rohleder, T. R. (2006). "Learning from Incidents: From

Normal Accidents to High Reliability." *System Dynamics Review*, 22(3), 213–239.

Currion, P., Silva, C. D., and Van de Walle, B. (2007). "Open Source Software for Disaster Management." *Communications of the ACM*, 50(3), 61–65.

Dahmann, J. S., Kuhl, F., and Weatherly, R. (1998). "Standard for Simulation: As Simple As Possible But Not Simpler the High Level Architecture for Simulation." *Simulation*, 71(6), 378–387.

Davis, C. A. (2013). "Quantifying Post-Earthquake Water System Functionality." *International Efforts in Lifeline Earthquake Engineering* (CD-ROM), ASCE, Reston, VA, U. S.

Dimakis, N., Filippoupolitis, A., and Gelenbe, E. (2010). "Distributed Building Evacuation Simulator for Smart Emergency Management." *The Computer Journal*, 53(9), 1384–1400.

Djanatliev, A., Kolominsky-Rabas, P., Hofmann, B. M., and German, R. (2012). "Hybrid Simulation with Loosely Coupled System Dynamics and Agent-Based Models for Prospective Health Technology Assessments." *Proceedings of the 2012 Winter Simulation Conference*, Institute of Electrical and Electronic Engineers (IEEE), Inc., Piscataway, NJ, U. S., 770–781.

Eidinger, J., Davis, C., Tang, A., and Kempner, L. (2012). “M 9.0 Tohoku Earthquake March 11 2011 Performance of Water and Power Systems.” G&E Engineering Systems Inc.

El-Anwar, O., El-Rayes, K., and Elnashai, A. (2009). “Optimizing Large-Scale Temporary Housing Arrangements After Natural Disasters.” *Journal of Computing in Civil Engineering*, ASCE, 23(2), 110–118.

El-Rayes, K., and Moselhi, O. (2001). “Optimizing Resource Utilization for Repetitive Construction Projects.” *Journal of Construction Engineering and Management*, ASCE, 127(1), 18–27.

E-Stat: Portal Site of Official Statistics of Japan. “Statistics of Japan.” <http://www.e-stat.go.jp/SG1/estat/eStatTopPortalE.do> (Aug. 12, 2013)

Federal Emergency Management Agency (FEMA) (2003). “HAZUS-MH MR4 Technical Manual.” <http://www.fema.gov/medialibrary-data/20130726-1716-25045-6422/hazus_mr4_earthquake_tech_manual.pdf> (Jan. 19, 2014)

Federal Emergency Management Agency (FEMA) (2011). “National Disaster Recovery Framework: Strengthening Disaster Recovery for the Nation.” <https://www.fema.gov/pdf/recoveryframework/ndrf.pdf> (May 13, 2014).

Ford, D. N., and Sterman, J. D. (1998). “Dynamic Modeling of Product Development Process.” *System Dynamics Review*, 14(1), 31–68.

Fujimoto, R. M. (1998). “Time Management in the High Level Architecture.” *Simulation*, 71(6), 388–400.

Ghadessy, M. (1999). *Text and Context in Functional Linguistics*, John Benjamins Publishing, Amsterdam, Holland.

Ghobarah, A., Abou-Elfath, H., and Biddah, A. (1999). “Response-Based Damage Assessment of Structures.” *Earthquake Engineering and Structural Dynamics*, 28(1), 79–104.

HallQuist, J. O. (2000). “LS-DYNA User’s Manual.” Livermore Software Technology Corporation.

Harrison, J. R., Lin, Z., Carroll, G. R., and Carley, K. M. (2007). “Simulation Modeling in Organizational and Management Research.” *Academy of Management Review*, 32(4), 1229–1245.

Hartmann, D., Breidt, M., Nguyen, V. V., Stangenberg, F., Höhler, S., Schweizerhof, K., ... and Liebscher, M. (2008). “Structural Collapse Simulation under Consideration of Uncertainty–Fundamental Concept and Results.” *Computers and Structures*, 86(21), 2064–2078.

Hegazy, T. (1999). "Optimization of Resource Allocation and Leveling Using Genetic Algorithms." *Journal of Construction Engineering and Management, ASCE*, 125(3), 167–175.

Herroelen, W., and Leus, R. (2005). "Project Scheduling under Uncertainty: Survey and Research Potentials." *European Journal of Operational Research*, 165(2), 289–306.

Holguin-Veras, J., and Jaller, M. (2012). "Immediate Resource Requirements After Hurricane Katrina." *Natural Hazards Review*, ASCE, 13(2), 117–131.

Hu, W., Liu, H., and Yang, Q. (2009). "HLA-Based Emergency Response Plan Simulation and Practice over Internet." *Advances in Neural Networks–ISNN 2009*, Springer-Verlag Berlin Heidelberg, 1203–1211.

Hwang, S., Park, M., Lee, H., Lee, S., and Kim, H. (2013). "Dynamic Feasibility Analysis of the Housing Supply Strategies in a Recession: Korean Housing Market." *Journal of Construction Engineering and Management*, ASCE, 139(2), 148–160.

Ibbs, W., and Nguyen, L. D. (2007). "Schedule Analysis under the Effect of Resource Allocation." *Journal of Construction Engineering and Management*, ASCE, 133(2), 131–138.

Imamura, F. (2006). *TUNAMI-N2 Manual draft*, Tohoku University's Numerical Analysis Model for Investigation of Near Field Tsunamis Version 2.

Jaiswal, K., and Wald, D. J. (2013). "Estimating Economic Losses from Earthquakes Using an Empirical Approach." *Earthquake Spectra*, 29(1), 309–324.

Janssen, M. A., and Ostrom, E. (2006). "Empirically Based, Agent-Based Models." *Ecology and Society*, 11(2), 37–49.

Jiang, J., Jiang, L., Kotsovinos, P., Zhang, J., Usmani, A., McKenna, F., and Li, G. Q. (2014). "Opensees Software Architecture for the Analysis of Structures in Fire." *Journal of Computing in Civil Engineering*, ASCE, 10.1061/(ASCE)CP.1943-5487.0000305, 04014030, *In Press*.

Jianxue, S. O. N. G., and Jie, L. (1996). "Simulation on Accessibility of Post-Earthquake Urban Transportation System." *Journal of Natural Disasters*, 1(5), 73–78.

Kanno, T., Narita, A., Morikawa, N., Fujiwara, H., and Fukushima, Y. (2006). "A New Attenuation Relation for Strong Ground Motion in Japan Based on Recorded Data." *Bulletin of the Seismological Society of America*, 96(3), 879–897.

Karlaftis, M. G., Kepaptsoglou, K. L., and Lambropoulos, S. (2007). "Fund Allocation for Transportation Network Recovery Following Natural Disasters." *Journal of Urban Planning and Development*, ASCE, 133(1), 82–89.

Katsanos, E. I., Sextos, A. G., and Manolis, G. D. (2010). "Selection of Earthquake Ground Motion Records: A State-of-the-Art Review from a Structural Engineering Perspective." *Soil Dynamics and Earthquake Engineering*, 30(4), 157–169.

Koto, T., and Takeuchi, I. (2003). "A Distributed Disaster Simulation System that Integrates Sub-Simulators." *1st International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster* (CD-ROM).

Kovel, J. P. (2000). "Modeling Disaster Response Planning." *Journal of Urban Planning and Development*, ASCE, 126(1), 26–38.

Kuhl, F., Dahmann, J., and Weatherly, R. (2000). *Creating Computer Simulation Systems: An Introduction to the High Level Architecture*. Prentice Hall PTR, Upper Saddle River, NJ, U. S.

Lambert, J. H., and Patterson, C. E. (2002). "Prioritization of Schedule Dependencies in Hurricane Recovery of Transportation Agency." *Journal of Infrastructure Systems*, ASCE, 8(3), 103–101.

Law, A., and Kelton, W. D. (2006). *Simulation Modeling and Analysis*, 3th Edition, McGraw-Hill, New York, NY, U. S.

Lee, S., Pena-Mora, F., and Park, M. (2005). "Quality and Change Management Model for Large Scale Concurrent Design and Construction Projects." *Journal of Construction Engineering of Management*, ASCE, 131(8), 890–902.

Lee, S., Han, S., and Peña-Mora, F. (2009). "Integrating Construction Operation and Context in Large-Scale Construction Using Hybrid Computer Simulation." *Journal of Computing in Civil Engineering*, ASCE, 23(2), 75–83.

Ma, T. Yang, H. T., and Chang, C. C. (2005). "Structural Damage Diagnosis and Assessment under Seismic Excitations." *Journal of Engineering and Mechanics*, 131(10), 1036–1045.

Martinez, J. C. (2001). "EZStrobe: General-Purpose Simulation System Based on Activity Cycle Diagrams." *Proceedings of the 2001 Winter Simulation Conference*, Institute of Electrical and Electronic Engineers (IEEE), Inc., Piscataway, NJ, U. S., 1556–1564

Martinez, J. C. (2010). "Methodology for Conducting Discrete-Event Simulation Studies in Construction Engineering and Management."

*Journal of Construction Engineering and Management*, ASCE, 136(1), 3–16.

Masurier, J. L., Wilkinson, S., and Shestakova, Y. (2006). “An Analysis of the Alliancing Procurement Method for Reconstruction Following an Earthquake.” *Proceedings of the 8th US National Conference on Earthquake Engineering* (CD-ROM), Sage, San Francisco, CA, U. S.

McKenna, F. T. (1997). *Object-Oriented Finite Element Programming: Frameworks for Analysis, Algorithms and Parallel Computing*. Doctoral Dissertation, University of California, Berkeley, CA, U. S.

McKenna, F., Fenves, G. L., Scott, M. H., and Jeremic, B. (2000). “Open System for Earthquake Engineering Simulation (OpenSees). Pacific Earthquake Engineering Research Center.” University of California, Berkeley, CA, U. S.

McLean, C., Riddick, F., and Lee, Y. T. (2005). “An Architecture and Interfaces for Distributed Manufacturing Simulation.” *Simulation*, 81(1), 15–32.

Menassa, C. C., Kamat, V. R., Lee, S., Azar, E., Feng, C., and Anderson, K. (2014). “Conceptual Framework to Optimize Building Energy Consumption by Coupling Distributed Energy Simulation and Occupancy Models.” *Journal of Computing in Civil Engineering*, ASCE, 28(1), 50–62.

Menches, C. L., Markman, A. B., and Jones, R. J. (2008). “Innovative Method for Investigating the Facility Damage Assessment Process.” *Proceedings of Architectural Engineering Conference* (AEI) 2008, ASCE, Reston, VA, U. S., 1–10.

Mosterman, P. J. (1999). “An Overview of Hybrid Simulation Phenomena and Their Support by Simulation Packages.” *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science, 1569, 165–177.

National Institute for Research Advancement of Japan (NIRA) (2013). “Status of Recovery and Current Problems in Three Disaster-hit Prefectures: What the Data Tells Us Indexes for Recovery and Reconstruction Following the Great East Japan Earthquake (IV).”<http://www.nira.or.jp/pdf/1301english_summary.pdf> (Jul. 08, 2014).

National Police Agency of Japan (NPA) (2013). “Damage Situation and Police Countermeasures Associated with 2011Tohoku district-off the Pacific Ocean Earthquake.” <http://www.npa.go.jp/archive/keibi/biki/higaijokyo_e.pdf> (Aug. 12, 2013).

Noulard, E., Rousselot, J., and Siron, P. (2009). “CERTI, an Open Source RTI, Why and How.” *Spring Simulation Interoperability Workshop*, Simulation Interoperability Standards Organization (SISO), Orlando, FL, U. S., 23–27.

Olshansky, R. B., Hopkins, L. D., and Johnson, L. A. (2012). “Disaster and Recovery: Processes Compressed in Time.” *Natural Hazards Review*, ASCE, 13(3), 173–178.

OpenSees Website. “OpenSees Example Manual.” <http://opensees.berkeley.edu/wiki/index.php/Examples_Manual/> (Aug. 28, 2013)

Orabi, W., El-Reyes, K., Senouci, A. B., and Al-Derham, H. (2009). “Optimizing Postdisaster Reconstruction Planning for Damaged Transportation Networks.” *Journal of Construction Engineering and Management*, ASCE, 135(10), 1039–1048.

Orabi, W., Senouci, A. B., El-Rayes, K., and Al-Derham, H. (2010). “Optimizing Resource Utilization during the Recovery of Civil Infrastructure Systems.” *Journal of Management in Engineering*, ASCE, 26(4), 237–246.

Pachakis, D., and Kiremidjian, A. S. (2004). “The Use of Simulation in Disaster Response Planning and Risk Management of Ports and Harbors.” *Proceedings of the 6th U.S. Conference and Workshop on Lifeline Earthquake Engineering*, ASCE, Reston, VA, U. S. 425–434.

Park, M. (2005). “Model-Based Dynamic Resource Management for Construction Projects.” *Automation in Constructi*on, 14(5), 585–598.

Peña-Mora, F., and Li, M. (2001). “Dynamic Planning and Control Methodology for Design/Build Fast-Track Construction Projects.” *Journal of Construction Engineering of Management*, ASCE, 127(1), 1–17.

Peña-Mora, F., Han, S. Lee, S., and Park, M. (2008). “Strategic-Operational Construction Management: Hybrid System Dynamics and Discrete Event Approach.” *Journal of Construction Engineering and Management*, ASCE, 134(9), 701–710.

Peña-Mora, F., Thomas, J. K., Glparvar-Fard, M., and Aziz, Z. (2012). “Supporting Civil Engineers during Disaster Response and Recovery Using a Segway Mobile Workstation Chariot.” *Journal of Computing in Civil Engineering*, ASCE, 26(3), 448–455.

Petty, M. D., and Windyga, P. S. (1999). “A High Level Architecture-Based Medical Simulation System.” *Simulation*, 73(5), 281–287.

Pich, M. T., Loch, C. H., and Meyer, A. D. (2002). “On Uncertainty, Ambiguity, and Complexity in Project Management.” *Management Science*, 48(8), 1008–1023.

Pike, J. (2007). “GulfGov Reports: Spending Federal Disaster Aid: Comparing the Process and Priorities in Louisiana and Mississippi in the Wake of Hurricanes Katrina and Rita.” *Nelson A. Rockefeller Institute of Government and Public Affairs Research Council of Louisiana*, Albany, NY, U. S. <http://www.rockinst.org/disaster_recovery/> (May 15, 2014).

Pinelli, J., Simiu, E., Gurley, K., Subramanian, C., Zhang, L., Cope, A., Filliben, J. J., and Hamid, S. (2004). “Hurricane Damage Prediction Model for Residential Structures.” *Journal of Structural Engineering*, ASCE, 130(11), 1685–1691.

Pradhan, A., Laefer, D. F., and Rasdorf, W. J. (2007). “Infrastructure Management Information System Framework requirements for Disasters.” *Journal of Computing in Civil Engineering*, ASCE, 21(2), 90–101.

Purser, D. A. (1992). “The Evolution of Toxic Effluents in Fires and the Assessment of Toxic Hazard.” *Toxicology Letters*, 64, 247–255.

Purser, D. (1996). “Behavioural Impairment in Smoke Environments.” *Toxicology*, 115(1), 25–40.

Reconstruction Agency of Japanese Government (2011). “Basic Guidelines for Reconstruction in response to the Great East Japan Earthquake.”

<http://www.reconstruction.go.jp/topics/basic_guidelines_reconstruction.pdf> (Aug. 12, 2013)

Reconstruction Agency (RA) of Japanese Government (2013). "Current Status and Path Toward Reconstruction." <http://www.reconstruction.go.jp/english/130328_CurrentStatus_PathToward_FINAL.pdf> (May 20, 2013)

Riley, D. R., and Sanvido, V. E. (1995). "Patterns of Construction-Space Use in Multistory Buildings." *Journal of Construction Engineering and Management*, ASCE, 121(4), 464–473.

Rowley, K. (2007). "GulfGov Reports: A Year and a Half after Katrina and Rita, an Uneven Recovery" *Nelson A. Rockefeller Institute of Government and Public Affairs Research Council of Louisiana*. <http://www.rockinst.org/disaster_recovery/> (May 15, 2014).

Sanford Bernhardt, K. L., and McNeil, S. (2008). "Agent-Based Modeling: Approach for Improving Infrastructure Management." *Journal of Infrastructure Systems*, ASCE, 14(3), 253–261.

Schulze, T., Strasburger, S., and Klein, U. (1999). "Migration of HLA into Civil Domains: Solutions and Prototypes for Transportation Applications." *Simulation*, 73(5), 296–303.

Shen, L. Y., Tam, V. W. Y., Tam, C. M., and Drew, D. (2004). "Mapping Approach for Examining Waste Management on Construction Sites." *Journal of Construction Engineering and Management*, ASCE, 130(4), 472–481.

Shiraki, N., Shinozuka, M., Moore, J. E., Chang, S. E., Kameda, H., and Tanaka, S. (2007). "System Risk Curves: Probabilistic Performance Scenarios for Highway Networks Subject to Earthquake Damage." *Journal of Infrastructure Systems*, ASCE, 13(1), 43–54.

Shoji, G., and Toyota, A. (2009). "Modeling of Restoration Process Associated with Critical Infrastructure and Its Interdependency to a Seismic Disaster." *Proceedings of the 2009 Lifeline Earthquake Engineering in a Multihazard Environment*, ASCE, Reston, VA, U. S., 647–658.

Song, F., Yang, X. K., and Du, L. (2010). "The Development of PARAMICS Based Metropolitan Emergency Evacuation Transportation Simulation System-MEETSIM." *Proceedings of the 10th International Conference of Chinese Transportation Professionals*, ASCE, Reston, VA, U. S., 388–400.

Stephens, J. E., and Yao, J. T. P. (1987). "Damage Assessment Using Response Measurements." *Journal of Structural Engineering*, ASCE, 113(4), 787–801.

Sterman, J. (2000). *Business Dynamics: System Thinking and Modeling for a Complex World*, Boston: Irwin McGraw-Hill, New York, NY, U. S.

Swan, R. C. (2000). “Debris Management Planning for the 21st Century.” *Natural Hazards Review*, ASCE, 1(4), 222–225.

Taranath, B. S. (2005). *Wind and Earthquake Resistant Buildings: Structural Analysis and Design, Drift Limitation*, CRC Press Taylor and Francis Group, Boca Raton, FL, U. S.

Taylor, T. R. B., and Ford, D. N. (2008). “Managing Tipping Point Dynamics in Complex Construction Projects.” *Journal of Construction Engineering and Management*, ASCE, 134(6), 421–431.

U.S. Defense Modeling and Simulation Office (DMSO) (2001). *RTI 1.3-Next Generation Programmer's Guide Version 4*. U.S. Department of Defense, Washington, DC, U. S.

U.S. Geological Survey (USGS) Website. “Real Time Data Feeds.” <http://earthquake.usgs.gov/earthquakes/?source=sitenav> (Mar. 3, 2014).

Venkateswaran, J., Son, Y., and Jones, A. (2004). “Hierarchical Production Planning Using a Hybrid System Dynamic-Discrete Event Simulation Architecture.” *Proceedings of the 2004 Winter Simulation Conference*, Institute of Electrical and Electronic Engineers (IEEE), Inc., Piscataway, NJ, U. S., 1094–1102

Vickery, P. J., Skerlj, P. F., Lin, J., Twisdale Jr., L. A., Young, M. A., and Lavelle, F. M. (2006). “HAZUS-MH Hurricane Model Methodology. II: Damage and Loss Estimation.” *Natural Hazards Review*, ASCE, 7(2), 94–103.

Wald, D. J., Quitoriano, V., Heaton, T. H., and Kanamori, H. (1999). “Relationships between Peak Ground Acceleration, Peak Ground Velocity, and Modified Mercalli Intensity in California.” *Earthquake Spectra*, 15(3), 557–564.

Wald, D. J., Jaiswal, K. S., Marano, K. D., and Bausch, D. (2011). “Earthquake Impact Scale.” *Natural Hazards Review*, ASCE, 12(3), 125–139.

Watson-Lamprey, J., and Abrahamson, N. (2006). “Selection of Ground Motion Time-Series and Limit on Scaling.” *Soil Dynamics and Earthquake Engineering*, 26(5), 477–482.

Wikipedia Website. <http://en.wikipedia.org/> (Aug. 04, 2014).

Williams, T. (2002). *Modelling Complex Projects*, Wiley, New York, NY, U. S.

Yotsukura, S., and Takahashi, T. (2009). "Framework of an Emergency Management System Using Different Rescue Simulators." *Advanced Robotics*, 23(9), 1233–1246.

Zhang, Y., AbouRizk, S M., Xie, H., and Moghani, E. (2011). "Design and Implementation of Loose-coupling Visualization Components in a Distributed Construction Simulation Environment with High Level Architecture (HLA)." *Journal of Computing in Civil Engineering*, ASCE, 26(2), 248–258.

# Appendices

# Appendix A: Terminology

| Term. | Acronym | Explanation |
|---|---|---|
| Accelerogram | | A strong motion data that represents the seismic excitation (Bommer and Acevedo 2004). |
| Accelerometer | | A device for real-time measurement of seismic accelerations (Bommer and Acevedo 2004). |
| Agent-Based Model | ABM | A simulation modeling for capturing the emergent behaviors by the adaptive actors and analyzing how they interact to influence one another with a focus on the behaviors of the actors (agents) (Harrison et al. 2007). |
| Attribute | | A single data field within an object (Kuhl et al. 2000). |
| CERTI | | The specific RTI used for interactive recovery simulations (Noulard et al. 2009). |
| Discrete Event Simulation | DES | A simulation modeling for an event-based process analysis with a consideration of stochastic durations and resource inputs to reproduce system events according to the discrete time advancement (Law and Kelton 2006; AbouRizk et al. 2011). |
| Distributed Simulation | | The simulation platform where different simulations interact with each other (Schulze et al. 1999; Menassa et al. 2014) |

| Term. | Abbreviation | Explanation |
|---|---|---|
| Epicenter | | The point on the Earth's surface that is directly above the hypocenter, the point where an earthquake or underground explosion originates (Taranath 2005; Wikipedia 2014). |
| Epicentral distance | | A distance from the surface of the damaged point to the epicenter (Taranath 2005; Wikipedia 2014). |
| Federate | | A single simulation within the federation such as a single simulation model, a live participant, and an incoming data stream (Kuhl et al. 2000; DMSO 2001). |
| Federation | | A set of federates of a collection of multiple running and interacting federates that are integrated via HLA (Kuhl et al. 2000; DMSO 2001). |
| Federation Object Model | FOM | A common object model for the data exchanged between federates in a federation (Kuhl et al. 2000; DMSO 2001). |
| Focal Depth | | The depth below the Earth's surface of the hypocenter of an earthquake (Taranath 2005; Wikipedia 2014). |
| Framework | | The set of provided basic modeling elements and concepts (Alvanchi et al. 2011). |
| High Level Architecture | HLA | A general purpose specification for distributed simulation systems (Menassa et al. 2014). |

| Term. | Abbreviation | Explanation |
|---|---|---|
| Hybrid Simulation | | The idea of combining different simulation methods (Mosterman 1999). |
| Interaction | | A collection of non-persisting data fields (i.e. an event) in the simulation that can be published to and/or subscribed by any number of federates (Kuhl et al. 2000). |
| Interaction Class | | A data structure of Interactions that has a single data field called a parameter (Kuhl et al. 2000). |
| Interface Variables | | The main contact points between simulation models in the interactive simulation (Alvanchi et al. 2011). |
| Management Object Model | MOM | A model that identifies objects and interactions used to manage a federation (Kuhl et al. 2000; DMSO 2001). |
| Model | | The behavior of a system as it evolves over time is studied by developing a model. The model usually takes the form of a set of assumptions concerning the operation of the system. These assumptions are expressed in mathematical, logical, and symbolic relationships between the entities, or objects of interest, of the system (Banks et al. 2005). |
| Moment Magnitude | Mw | A measure the size of earthquakes in terms of the energy released (Taranath 2005; Wikipedia 2014). |

| Term. | Abbreviation | Explanation |
|---|---|---|
| Object | | A collection of persisting data fields that can be published and/or subscribed to by any number of federates (Kuhl et al. 2000). |
| Object Class | | A data structure of Objects that has a set of named data called attributes (Kuhl et al. 2000). |
| Object Model Template | OMT | Standards for defining HLA object modeling information, which includes the data to be handled by the RTI when a simulation federate executes, the objects and interactions that are responsible for managing the federation execution, and the documentation describing the functionality for each simulation federate (Kuhl et al. 2000; DMSO 2001). |
| Parameter | | A single data field within an interaction (Kuhl et al. 2000). |
| Peak Ground Acceleration | PGA | A measure of earthquake acceleration on the ground and an important input parameter for earthquake engineering (Taranath 2005; Wikipedia 2014). |
| Peak Ground Velocity | PGV | A measure of earthquake velocity on the ground and an important input parameter for earthquake engineering (Wald et al. 1999; Wikipedia 2014). |
| Prototype | | An early sample, model, or release of a product built to test a concept or process or to act as a thing to be replicated or learned from (Wikipedia 2014). |

| Term. | Abbreviation | Explanation |
|---|---|---|
| Run-Time Infrastructure | RTI | An implementation of HLA which provides the binary executable for executing data transfers between simulators, as well as the application programming interface (API) for writing software that integrates with the RTI (DMSO 2001). |
| Run-Time Infrastructure Ambassador | RTIA | A process which is automatically launched by the federate with CERTI (Noulard et al. 2009). |
| Run-Time Infrastructure Gateway | RTIG | A process which coordinates the HLA simulation with CERTI, there should be at least one RTIG process for each federation. However a single RTIG may be used for several federations (Noulard et al. 2009). |
| Simulation | | A simulation is the imitation of the operation of a real-world process or system over time. Whether done by hand or on a computer, simulation involves the generation of an artificial history of a system, and the observation of that artificial history to draw inferences concerning the operating characteristics of the real system (Banks et al. 2005). |
| Simulation Object Model | SOM | A model that describes salient characteristics of a federate with a focus on the federate's internal operation (Kuhl et al. 2000; DMSO 2001). |

| Term. | Abbreviation | Explanation |
|---|---|---|
| System | | A collection of entities (e.g., people or machines) that act and interact together toward the accomplishment of some logical end (Law and Kelton 2006). |
| System Dynamics | SD | A simulation modeling based on the theory-based cause-and-effect relationship among variables and the stock-and-flow diagram, which can be applied to model the behavior of a complex and dynamic system as a whole by capturing interactions among variables and understanding their structures according to the continuous time advancement (Sterman 2000; Williams 2002). |
| The Mercalli Intensity Scale | Imm | A seismic scale used for measuring the intensity of an earthquake (Taranath 2005). |
| Validation | | Validation is the determination that a model is an accurate representation of the real system (Banks et al. 2005). |
| Verification | | Verification pertains to the computer program prepared for the simulation model (Banks et al. 2005). |

# Appendix B: Descriptions of Simulation Components

In this Appendix, the detailed descriptions including model structures and equations as well as model input variables are presented with regard to the simulation model components in the recovery simulation, based on the previous chapter, Ch. 4. Model components include a regional damage estimation model, a regional-level recovery analysis model using SD, a facility damage assessment model, and a facility-level restoration analysis model using DES.

## *B-I Regional Damage Estimation Model*

A regional-level damage estimation model in the Anylogic recovery simulation federate aims to estimate direct physical damage and functional losses of the built environment throughout a whole region by using earthquake information from the USGS seismic data retrieval federate. The estimated damage can be used as input values in the SD recovery simulation model.

To capture overall damage patterns throughout sub-regions in the whole damage region, this model uses data structures in the "*DamagedRegion*" class to reflect diverse geographical information of numerous sub-regions. Based on the simulation process of this model as shown in previous figure (Fig. 4.1) in Ch. 4.1.1, computational codes for this model is provided in detail.

**[“*DamagedRegion*” Class Structure]**

```
/* DamagedRegion class for regional damage estimation */

public class DamagedRegion implements java.io.Serializable {

     double lat = 0;         //latitude of regions
     double lon = 0;         //longitude of regions
     double epidist = 0;     //epicentral distance of regions
     double pga = 0;     //PGA at regions
     double pgv = 0;     //PGV at regions
     double imm = 0;     //Imm at regions
     double damagei = 0;//damage intensity of regions

    /* Default constructor */
    public DamagedRegion(){
    }

    /* Constructor initializing the fields */
    public DamagedRegion(double lat,double lon,double epidist,
            double pga, double pgv, double imm, double damagei){
         this.lat = lat;
         this.lon = lon;
         this.epidist = epidist;
         this.pga = pga;
         this.pgv = pgv;
         this.imm = imm;
         this.damagei = damagei;
    }
     @Override
    public String toString() {
         return
             "lat = " + lat +" " +
             "lon = " + lon +" " +
             "epidist = " + epidist +" " +
             "pga = " + pga +" " +
             "pgv = " + pgv +" " +
             "imm = " + imm +" " +
             "damagei = " + damagei +" ";
    }
    private static final long serialVersionUID = 1L;
}
```

## [Detailed Descriptions of the Model]

```
/* Main function for regional damage estimation */
void
 RegionalDamageIntensityEstimator(  ) {
  String sheet = "Regional Info";
  int regionnum = 5;
  int row = 29; //regional information from Excel file (DB)
  DamagedRegion [] dr;
  dr = new DamagedRegion[regionnum];
  for (int i=0; i<regionnum; i++)
  {
    dr[i] = new DamagedRegion();
    dr[i].lat = excelFile.getCellNumericValue(sheet, row, 15);
          /* data input from excel file (database) */
    dr[i].lon = excelFile.getCellNumericValue(sheet, row, 18);
          /* data input from excel file (database) */
    dr[i].epidist = EpiDistanceCalculator(dr[i].lat, dr[i].lon);
    dr[i].pga = PGACalculator(newEventMag_Value,
                    newEventDepth_Value, dr[i].epidist);
    dr[i].pgv = PGVCalculator(newEventMag_Value,
                    newEventDepth_Value, dr[i].epidist);
    dr[i].imm = ImmCalculator(dr[i].pga, dr[i].pgv);
    if (dr[i].imm >= 5) {
        dr[i].damagei =  DamageIFunction_Asia(dr[i].imm);
        /* damage function using lognormal distribution */
        }
    row++;
  }
}

/* Function for regional epicentral distance calculation */
double
 EpiDistanceCalculator( double lat, double lon ) {

  double dLat, dLong, a, c, distance;
  dLat = 0;
  dLong = 0;
  a = 0;
  c = 0;
  distance = 0;
  if(newEventMag_Value > 0)
    dLat = newEventLatitude - lat;
    dLong = newEventLongitude - lon;
    a = pow((sin(dLat*3.141592/180/2)),2) +
        cos(lat*3.141592/180) * cos(newEventLatitude*3.14/180)
        * pow((sin(dLong*3.141592/180/2)),2);
    c = 2 * atan(sqrt(a)/sqrt(1-a));
    distance = c*6370000;

return distance;
}
```

```
/* Function for regional PGA calculation */
double
 PGACalculator( double intensity, double focaldepth,
 double epidistance ) {

  double a1, b1, c1, d1, e1, a2, b2, c2, e2, pga;
  /* regression coefficients */
  a1=0.56;
  b1=-0.0031;
  c1=0.26;
  d1=0.0055;
  e1=0.37;
  a2=0.41;
  b2=-0.0039;
  c2=1.56;
  e2=0.40;
  pga = 0;
  if(focaldepth <= 30000) {
    pga = pow(10, (a1 * intensity + b1 * (epidistance/1000)
          -log10((epidistance/1000) + d1 * pow(10, 0.5*intensity))
          + c1 + e1));
     } else {
        pga = pow(10, (a2 * intensity + b2 * (epidistance/1000)
               -log10((epidistance/1000)) + c1 + e1));
  }
 return pga;  /* by Kanno et al. (2006)'s Equations  */
}

/* Function for regional PGV calculation */
double
 PGVCalculator( double intensity, double focaldepth,
 double epidistance ) {

  double a1, b1, c1, d1, e1, a2, b2, c2, e2, pgv;
  /* regression coefficients */
  a1=0.70;
  b1=-0.0009;
  c1=-1.93;
  d1=0.0022;
  e1=0.32;
  a2=0.55;
  b2=-0.0032;
  c2=-0.57;
  e2=0.36;
  pgv = 0;
  if(focaldepth <= 30000) {
    pgv = pow(10, (a1 * intensity + b1 * (epidistance/1000)
          -log10((epidistance/1000) + d1 * pow(10, 0.5*intensity))
          + c1 + e1));
     } else {
        pgv = pow(10, (a2 * intensity + b2 * (epidistance/1000)
               -log10((epidistance/1000)) + c1 + e1));
 }
  return pgv; /* by Kanno et al. (2006)'s Equations  */
 }
```

```
/* Function for Imm calculation */
double
 ImmCalculator( double pga, double pgv ) {

  double imm = 0;

  if (pga < 232.4) {
    imm = 3.66 * log10(pga) - 1.66;
  } else {
    imm = 3.47 * log10(pgv) + 2.35;
  }

return imm;
    } /* by Waid et al. (1999)'s Equations */
```

**[Input Values from Excel Files: Regional Locational Information]**

The diverse geographical information of numerous sub-regions is provided in the Excel databases, as shown below. These values are automatically input to the regional-level damage estimation model when initiating the simulations.

| | M | N | O | P | Q | R | S | T |
|---|---|---|---|---|---|---|---|---|
| 25 | | | | | | | | |
| 26 | | Hyper array Data | | | | | | |
| 27 | | [Japan] | | | | | | |
| 28 | | **Latitude** | | | **Longitude** | | | |
| 29 | | Iwate Pref. | 39.7 | | Iwate Pref. | 141.2 | | |
| 30 | | Miyagi Pref. | 38.3 | | Miyagi Pref. | 140.9 | | |
| 31 | | Fukushima Pref. | 37.8 | | Fukushima Pref. | 140.5 | | |
| 32 | | RegionD | 0 | | RegionD | 0.0 | | |
| 33 | | RegionE | 0 | | RegionE | 0.0 | | |

**[Input Function for Earthquake Loss Estimation]**

The standard lognormal cumulative distribution function for estimating the loss ratio of the damage region is also input to the regional-level damage estimation model when initiating the simulations. This function is based on the previous approach suggested by Jaiswal and Wald (2013), as shown in the previous equation (Eq. 4.5) in Ch. 4.1.1.

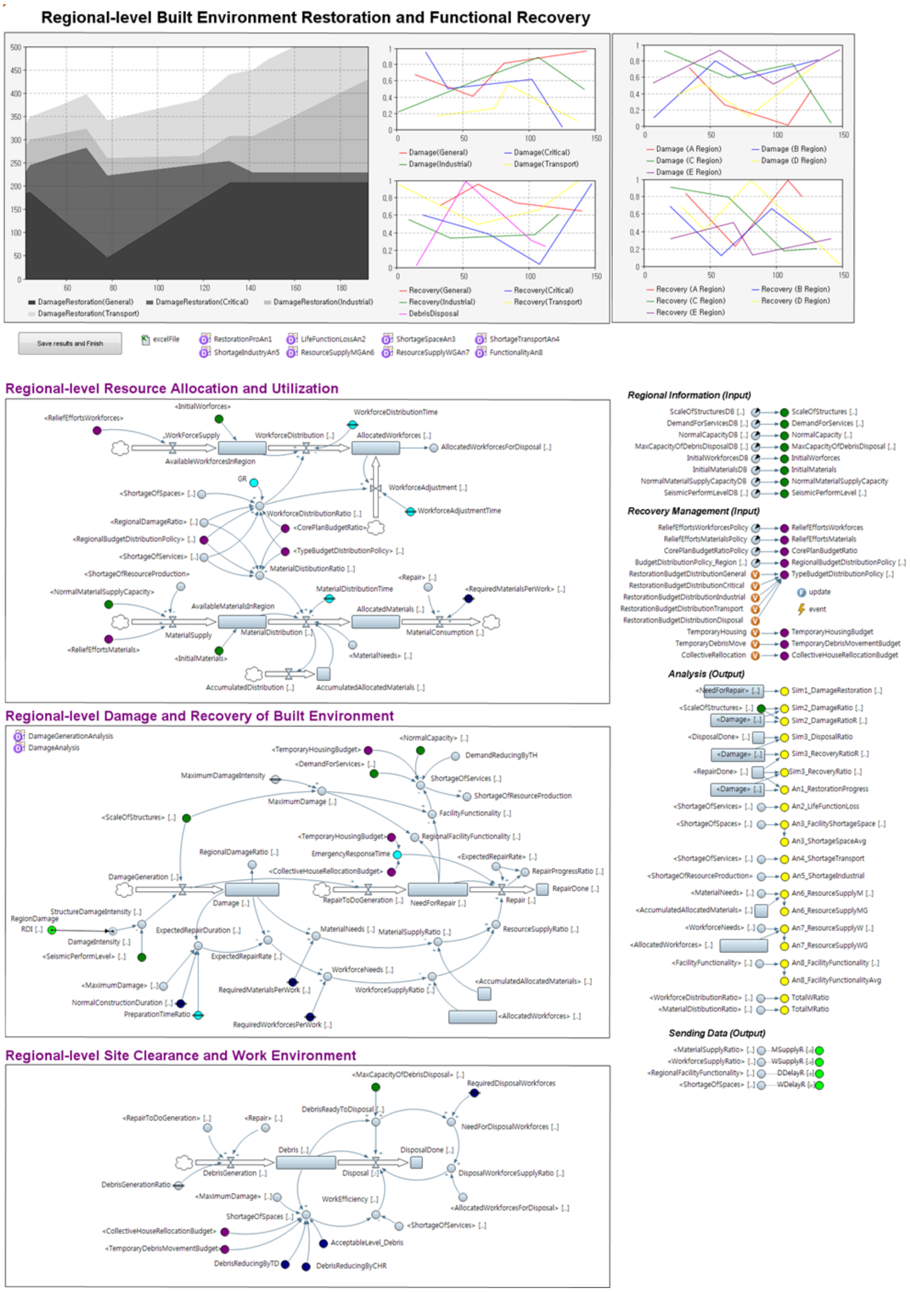
| | A | B | C |
|---|---|---|---|
| 1 | [East Asia] | | |
| 2 | The Mean of Natural Logarithm of Imm | | **10.29** |
| 3 | standard deviation of ln(I) | | **0.1** |
| 4 | **Imm** | **ln(Imm)** | **DI(Imm)** |
| 5 | 5.0 | 1.609437912 | 2.65059E-13 |
| 6 | 5.1 | 1.62924054 | 1.11475E-12 |
| 7 | 5.2 | 1.648658626 | 4.392E-12 |
| 8 | 5.3 | 1.667706821 | 1.62629E-11 |
| 9 | 5.4 | 1.686398954 | 5.67667E-11 |
| 10 | 5.5 | 1.704748092 | 1.87319E-10 |
| 11 | 5.6 | 1.722766598 | 5.85885E-10 |
| 12 | 5.7 | 1.740466175 | 1.7413E-09 |
| 13 | 5.8 | 1.757857918 | 4.92923E-09 |
| 14 | 5.9 | 1.774952351 | 1.33196E-08 |
| 15 | 6.0 | 1.791759469 | 3.4428E-08 |
| … | | | |
| 73 | 11.8 | 2.468099531 | 0.914542525 |
| 74 | 11.9 | 2.4765384 | 0.926979496 |
| 75 | 12.0 | 2.48490665 | 0.937895087 |
| 76 | 12.1 | 2.493205453 | 0.947419192 |
| 77 | 12.2 | 2.501435952 | 0.95568171 |
| 78 | 12.3 | 2.509599262 | 0.962809865 |
| 79 | 12.4 | 2.517696473 | 0.96892606 |
| 80 | 12.5 | 2.525728644 | 0.974146249 |
| 81 | 12.6 | 2.533696814 | 0.978578772 |
| 82 | 12.7 | 2.541601993 | 0.982323614 |
| 83 | 12.8 | 2.549445171 | 0.985472011 |
| 84 | 12.9 | 2.557227311 | 0.988106358 |
| 85 | 13.0 | 2.564949357 | 0.990300359 |

## *B-II Regional-Level Recovery Analysis Model Using SD*

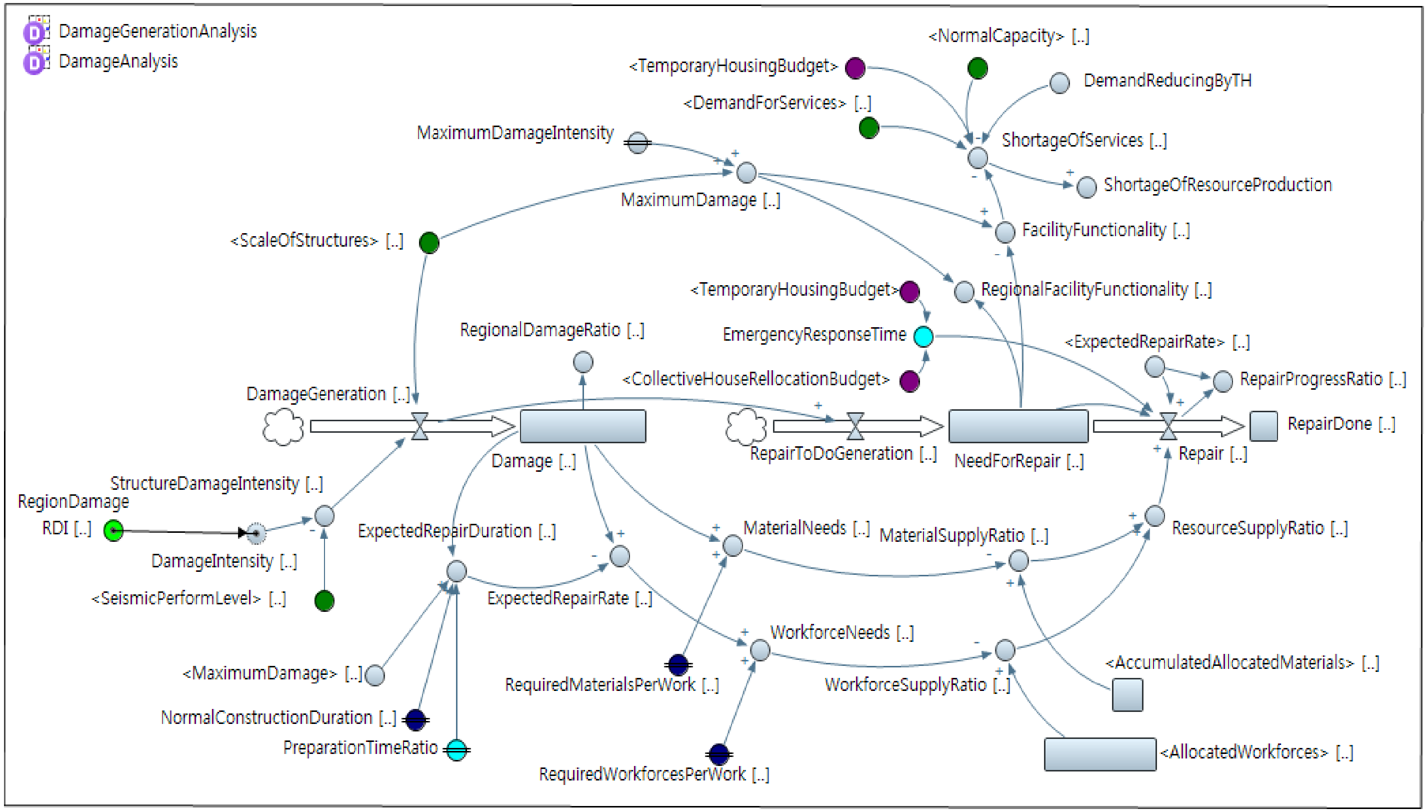
An SD regional-level recovery analysis model in the Anylogic recovery simulation federate aims to understand multiple and complex recovery processes in an overall region, according to estimated regional damage from the regional damage estimation model. To capture the differences between recovery processes among diverse types of facilities throughout a whole region in detail, the SD model in this study uses two-dimensional (2-D) array variables including sub-elements for several sub-regions and for diverse facility/infrastructure types into the "*[Region]*" and "*[Type]*" dimensions, respectively,

By using SD's cause-and-effect relationships and stock and flow diagrams, the main focus of the model is to understand complex and interdependent restoration efforts among diverse facilities/infrastructure types and various sub-regions, and to capture multiple feedback processes between regional restoration works and functionality recoveries of facilities. Based on the simulation model as shown in previous figure (Fig. 4.3) in Ch. 4.1.2, this study provides the descriptions of model structures and equations as well as input values.

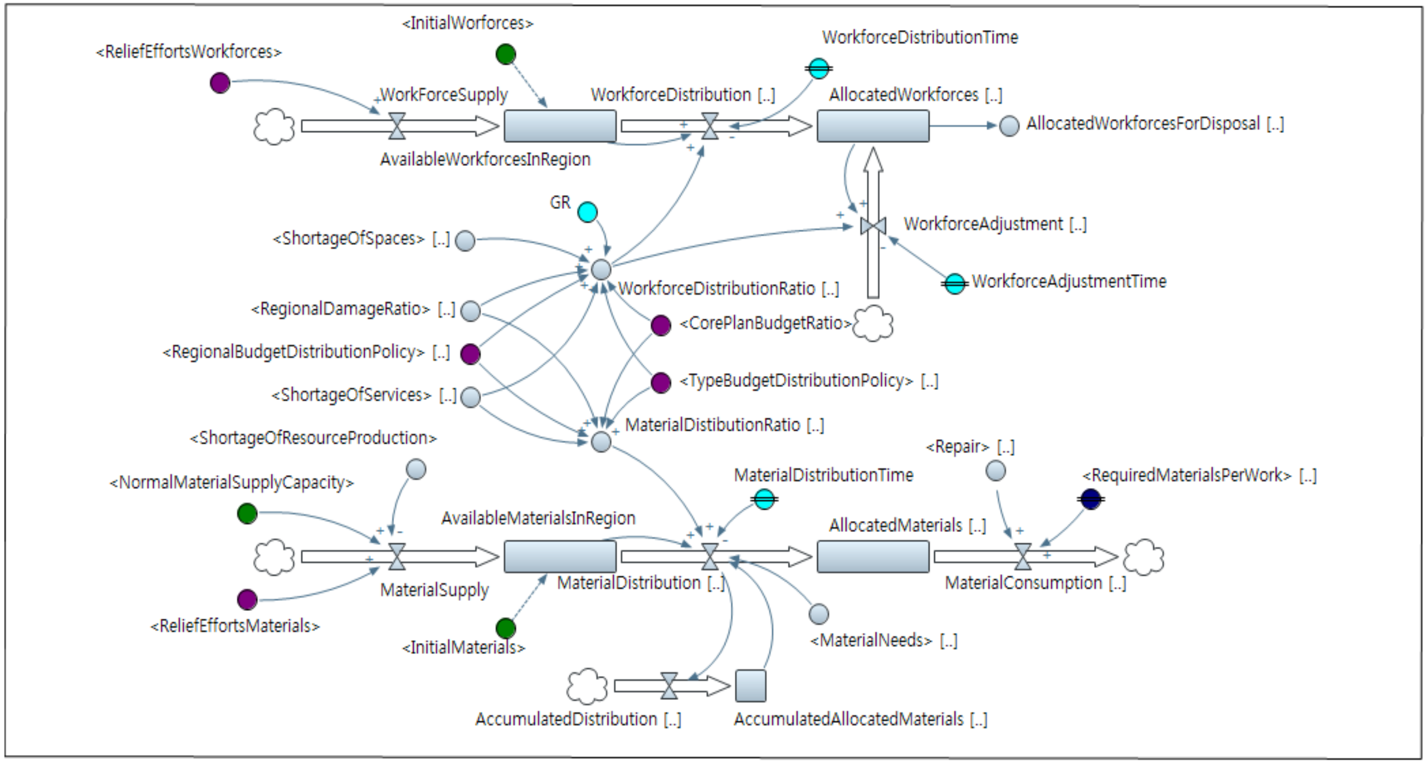
# [Model Structures]



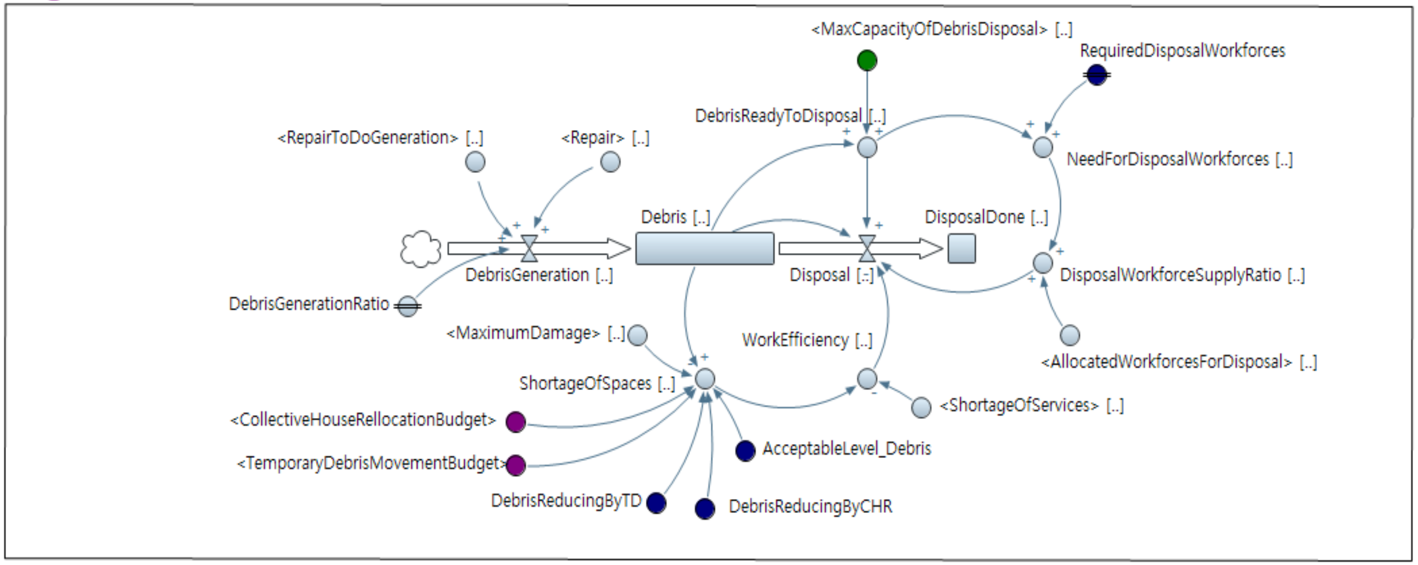
# Regional-level Damage and Recovery of Built Environment



## Regional-level Resource Allocation and Utilization



## Regional-level Site Clearance and Work Environment



**[Variables and Equations]**

| Variables | Equations |
|---|---|
| NeedForRepair [Facilities, Region] | **d(NeedForRepair [Facilities, Region])/dt =**<br>RepairToDoGeneration[Facilities,Region] - Repair[Facilities,Region] |
| FacilityFunctionality [Facilities] | **FacilityFunctionality [Facilities] =**<br>min (1 , 1 -( NeedForRepair.sum (Facilities,INDEX_CAN_VARY) / MaximumDamage.sum (Facilities,INDEX_CAN_VARY))) |
| ShortageOfSpaces [Facilities, Region] | **ShortageOfSpaces [Facilities, Region] =**<br>Debris[Facilities,Region] > 0 ? max ( 0 , Debris[Facilities,Region]* (1-TemporaryDebrisMovementBudget* DebrisReducingByTD)*(1-CollectiveHouseRellocationBudget* DebrisReducingByCHR) / MaximumDamage[Facilities,Region] - AcceptableLevel_Debris ) : 0 |
| Debris [Facilities, Region] | **d(Debris [Facilities, Region])/dt =**<br>DebrisGeneration[Facilities,Region] - Disposal[Facilities,Region] |
| DebrisReadyTo Disposal [Region] | **DebrisReadyToDisposal [Region]=**<br>Debris.sum(INDEX_CAN_VARY, Region) > 0 ? MaxCapacityOfDebrisDisposal[Region] : 0 |
| Disposal [Facilities, Region] | **Disposal [Facilities, Region] =**<br>Debris[Facilities,Region] > 0 ? min ( Debris[Facilities,Region], (Debris[Facilities,Region] / Debris.sum(INDEX_CAN_VARY,Region)) * WorkEfficiency[Facilities,Region] * DebrisReadyToDisposal[Region] * DisposalWorkforceSupplyRatio[Region] ) : 0 |
| NeedForDisposalWorkforces [Region] | **NeedForDisposalWorkforces [Region]=**<br>DebrisReadyToDisposal[Region] * RequiredDisposalWorkforces |
| DisposalWorkforce SupplyRatio [Region] | **DisposalWorkforceSupplyRatio [Region] =**<br>NeedForDisposalWorkforces[Region] > 0 ? min ( 1 , ( AllocatedWorkforcesForDisposal[Region] / NeedForDisposalWorkforces[Region] ) ) : 0 |
| WorkEfficiency [Facilities, Region] | **WorkEfficiency [Facilities, Region]=**<br>( 1 - ShortageOfSpaces[Facilities,Region] ) * (1 - ShortageOfServices.sum(Transport) ) |
| AvailableMaterials InRegion | **AvailableMaterialsInRegion=**<br>MaterialSupply - MaterialDistribution.sum() |
| DamageGeneration [Facilities, Region] | **DamageGeneration [Facilities, Region] =**<br>ScaleOfStructures[Facilities,Region]* * StructureDamageIntensity[Facilities,Region] |
| StructureDamage Intensity [Facilities, Region] | **StructureDamageIntensity [Facilities, Region] =**<br>(1-SeismicPerformLevel[Facilities]) * DamageIntensity[Region] |
| Repair [Facilities, Region] | **Repair [Facilities, Region] =**<br>delay3 (NeedForRepair[Facilities,Region] > 0 ? min ( NeedForRepair[Facilities,Region],ExpectedRepairRate[Facilities,Region] * ResourceSupplyRatio[Facilities,Region]) : 0 , EmergencyResponseTime)//delay3 ( DamageGeneration[Facilities] , EmergencyResponseTime ) |

| Variables | Equations |
| --- | --- |
| ShortageOfServices [Facilities] | **ShortageOfServices [ResidentialCommercial] =**<br>FacilityFunctionality.sum(General) > 0 ?<br>max( ( DemandForServices[ResidentialCommercial] -<br>TemporaryHousingBudget*DemandReducingByTH -<br>FacilityFunctionality.sum(General) *<br>NormalCapacity[ResidentialCommercial] ) /<br>NormalCapacity[ResidentialCommercial], 0 ) : 0<br><br>**ShortageOfServices [Public] =**<br>FacilityFunctionality.sum(Critical) > 0 ?<br>max( ( DemandForServices[Public] -<br>TemporaryHousingBudget*DemandReducingByTH -<br>FacilityFunctionality.sum(Critical) * NormalCapacity[Public] ) /<br>NormalCapacity[Public], 0 ) : 0<br><br>**ShortageOfServices [Economic] =**<br>FacilityFunctionality.sum(Industrial) > 0 ?<br>max( ( DemandForServices[Economic] -<br>FacilityFunctionality.sum(Industrial) * NormalCapacity[Economic] ) /<br>NormalCapacity[Economic], 0 ) : 0<br><br>**ShortageOfServices [Transport] =**<br>FacilityFunctionality.sum(Pathway) > 0 ?<br>max( ( DemandForServices[Transport] -<br>FacilityFunctionality.sum(Pathway) * NormalCapacity[Transport] ) /<br>NormalCapacity[Transport], 0 ) : 0 |
| MaterialDistibution Ratio [Facilities, Region] | **MaterialDistibutionRatio [GeneralRestore, Region] =**<br>ShortageOfServices[ResidentialCommercial] > 0 ?<br>((ShortageOfServices[ResidentialCommercial]/<br>(ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOfServices[ResidentialCommercial]+ShortageOfServices[Transport]))*(1-CorePlanBudgetRatio)+(TypeBudgetDistributionPolicy[GeneralRestore]<br>/(TypeBudgetDistributionPolicy[GeneralRestore]+TypeBudgetDistributionPolicy[CriticalRestore]+TypeBudgetDistributionPolicy[IndustrialRestore]+<br>TypeBudgetDistributionPolicy[TransportRestore]))*CorePlanBudgetRatio)<br>*(RegionalDamageRatio[Region]+RegionalBudgetDistributionPolicy[Region]) : 0<br><br>**MaterialDistibutionRatio [CriticalRestore, Region] =**<br>ShortageOfServices[Public] > 0 ?<br>((ShortageOfServices[Public]/<br>(ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOfServices[ResidentialCommercial]+ShortageOfServices[Transport]))*(1-CorePlanBudgetRatio)+(TypeBudgetDistributionPolicy[CriticalRestore]<br>/(TypeBudgetDistributionPolicy[GeneralRestore]+TypeBudgetDistributionPolicy[CriticalRestore]+TypeBudgetDistributionPolicy[IndustrialRestore]+<br>TypeBudgetDistributionPolicy[TransportRestore]))*CorePlanBudgetRatio)<br>*(RegionalDamageRatio[Region]+RegionalBudgetDistributionPolicy[Region]) : 0 |

| Variables | Equations |
|---|---|
| MaterialDistibution<br>Ratio<br>[Facilities, Region] | **MaterialDistibutionRatio [IndustrialRestore, Region] =**<br>ShortageOfServices[Economic] > 0 ?<br>((ShortageOfServices[Economic]/<br>(ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOf<br>Services[ResidentialCommercial]+ShortageOfServices[Transport]))*(1-<br>CorePlanBudgetRatio)+<br>(TypeBudgetDistributionPolicy[IndustrialRestore]<br>/(TypeBudgetDistributionPolicy[GeneralRestore]+TypeBudgetDistribution<br>Policy[CriticalRestore]+TypeBudgetDistributionPolicy[IndustrialRestore]+<br>TypeBudgetDistributionPolicy[TransportRestore]))*CorePlanBudgetRatio)<br>*(RegionalDamageRatio[Region]<br>+RegionalBudgetDistributionPolicy[Region]) : 0<br><br>**MaterialDistibutionRatio [TransportRestore, Region] =**<br>ShortageOfServices[Transport] > 0 ?<br>((ShortageOfServices[Transport]/<br>(ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOf<br>Services[ResidentialCommercial]+ShortageOfServices[Transport]))*(1-<br>CorePlanBudgetRatio)+<br>(TypeBudgetDistributionPolicy[TransportRestore]<br>/(TypeBudgetDistributionPolicy[GeneralRestore]+TypeBudgetDistribution<br>Policy[CriticalRestore]+TypeBudgetDistributionPolicy[IndustrialRestore]+<br>TypeBudgetDistributionPolicy[TransportRestore]))*CorePlanBudgetRatio)<br>*(RegionalDamageRatio[Region]+RegionalBudgetDistributionPolicy[Regi<br>on]) : 0<br><br>**MaterialDistibutionRatio [DebrisDisposal, Region]** = 0 |
| MaterialDistribution<br>[Facilities, Region] | **MaterialDistribution [GeneralRestore, Region] =**<br>MaterialNeeds[General,Region] ><br>AccumulatedAllocatedMaterials[GeneralRestore,Region] ?<br>AvailableMaterialsInRegion *<br>MaterialDistibutionRatio[GeneralRestore,Region] /<br>MaterialDistributionTime : 0<br><br>**MaterialDistribution [CriticalRestore, Region] =**<br>MaterialNeeds[Critical,Region] ><br>AccumulatedAllocatedMaterials[CriticalRestore,Region] ?<br>AvailableMaterialsInRegion *<br>MaterialDistibutionRatio[CriticalRestore,Region] /<br>MaterialDistributionTime : 0<br><br>**MaterialDistribution [IndustrialRestore, Region] =**<br>MaterialNeeds[Industrial,Region] ><br>AccumulatedAllocatedMaterials[IndustrialRestore,Region] ?<br>AvailableMaterialsInRegion *<br>MaterialDistibutionRatio[IndustrialRestore,Region] /<br>MaterialDistributionTime : 0 |

| Variables | Equations |
|---|---|
| MaterialDistribution [Facilities, Region] | **MaterialDistribution [TransportRestore, Region] =** MaterialNeeds[Pathway,Region] > AccumulatedAllocatedMaterials[TransportRestore,Region] ? AvailableMaterialsInRegion * MaterialDistibutionRatio[TransportRestore,Region] / MaterialDistributionTime : 0<br>**MaterialDistribution [DebrisDisposal, Region] =** 0 |
| ResourceSupply Ratio [Facilities, Region] | **ResourceSupplyRatio [Facilities, Region] =** min (WorkforceSupplyRatio[Facilities,Region] , MaterialSupplyRatio[Facilities,Region] ) |
| MaterialSupplyRatio [Facilities, Region] | **MaterialSupplyRatio [Facilities, Region] =** MaterialNeeds[Facilities,Region] > 0 ? min ( 1 , AccumulatedAllocatedMaterials[Facilities,Region] / MaterialNeeds[Facilities,Region] ) : 0 |
| MaterialNeeds [Facilities, Region] | **MaterialNeeds [Facilities, Region] =** Damage[Facilities,Region] * RequiredMaterialsPerWork[Facilities] |
| Damage [Facilities, Region] | **d(Damage [Facilities, Region])/dt =** DamageGeneration[Facilities,Region] |
| ExpectedRepair Rate [Facilities, Region] | **ExpectedRepairRate [Facilities, Region]=** ExpectedRepairDuration[Facilities,Region] > 0 ? Damage[Facilities,Region] / ExpectedRepairDuration[Facilities,Region] : 0 |
| ExpectedRepair Duration [Facilities, Region] | **ExpectedRepairDuration [Facilities, Region] =** MaximumDamage[Facilities,Region] > 0 ? ( PreparationTimeRatio + Damage[Facilities,Region] / MaximumDamage[Facilities,Region] ) * NormalConstructionDuration[Facilities] : 0 |
| WorkforceNeeds [Facilities, Region] | **WorkforceNeeds [Facilities, Region] =** ExpectedRepairRate[Facilities,Region] * RequiredWorkforcesPerWork[Facilities] |
| RepairProgress Ratio [Facilities, Region] | **RepairProgressRatio [Facilities, Region] =** ExpectedRepairRate[Facilities,Region] > 0 ? Repair[Facilities,Region] / ExpectedRepairRate[Facilities,Region] : 0 |
| WorkforceSupply Ratio [Facilities, Region] | **WorkforceSupplyRatio [Facilities, Region] =** WorkforceNeeds[Facilities,Region] > 0 ? min ( 1 , AllocatedWorkforces[Facilities,Region] / WorkforceNeeds[Facilities,Region] ) : 0 |
| Material Consumption [Facilities, Region] | **MaterialConsumption [GeneralRestore, Region] =** Repair[General,Region] * RequiredMaterialsPerWork[General]<br><br>**MaterialConsumption [CriticalRestore, Region] =** Repair[Critical,Region] * RequiredMaterialsPerWork[Critical]<br><br>**MaterialConsumption [IndustrialRestore, Region] =** Repair[Industrial,Region] * RequiredMaterialsPerWork[Industrial]<br><br>**MaterialConsumption [TransportRestore, Region] =** Repair[Pathway,Region] * RequiredMaterialsPerWork[Pathway]<br><br>**MaterialConsumption [DebrisDisposal, Region] =** 0 |

| Variables | Equations |
|---|---|
| Workforce DistributionRatio [Facilities, Region | **WorkforceDistributionRatio [GeneralRestore, Region] =**<br>GR >= 1 ? ((ShortageOfServices[Public] + ShortageOfServices[Economic] + ShortageOfServices[Transport]) > 0 ? ((ShortageOfServices[ResidentialCommercial]/ (ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOfServices[ResidentialCommercial]+ShortageOfServices[Transport]+ShortageOfSpaces.average()))*(1-CorePlanBudgetRatio)+ TypeBudgetDistributionPolicy[GeneralRestore]*CorePlanBudgetRatio) *(RegionalDamageRatio[Region]+RegionalBudgetDistributionPolicy[Region]) : 1) : 0<br><br>**WorkforceDistributionRatio [CriticalRestore, Region] =**<br>ShortageOfServices[Public] > 0 ? ((ShortageOfServices[Public]/ (ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOfServices[ResidentialCommercial]+ShortageOfServices[Transport]+ShortageOfSpaces.average()))*(1-CorePlanBudgetRatio)+ TypeBudgetDistributionPolicy[CriticalRestore]*CorePlanBudgetRatio) *(RegionalDamageRatio[Region]+RegionalBudgetDistributionPolicy[Region]) : 0<br><br>**WorkforceDistributionRatio [IndustrialRestore, Region] =**<br>ShortageOfServices[Economic] > 0 ? ((ShortageOfServices[Economic]/ (ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOfServices[ResidentialCommercial]+ShortageOfServices[Transport]+ShortageOfSpaces.average()))*(1-CorePlanBudgetRatio)+ TypeBudgetDistributionPolicy[IndustrialRestore]*CorePlanBudgetRatio) *(RegionalDamageRatio[Region]+RegionalBudgetDistributionPolicy[Region]) : 0<br><br>**WorkforceDistributionRatio [TransportRestore, Region] =**<br>ShortageOfServices[Transport] > 0 ? ((ShortageOfServices[Transport]/ (ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOfServices[ResidentialCommercial]+ShortageOfServices[Transport]+ShortageOfSpaces.average()))*(1-CorePlanBudgetRatio)+ TypeBudgetDistributionPolicy[TransportRestore]*CorePlanBudgetRatio) *(RegionalDamageRatio[Region]+RegionalBudgetDistributionPolicy[Region]) : 0<br><br>**WorkforceDistributionRatio [DebrisDisposal, Region] =**<br>ShortageOfSpaces.average() > 0 ? ((ShortageOfSpaces.average()/ (ShortageOfServices[Public]+ShortageOfServices[Economic]+ShortageOfServices[ResidentialCommercial]+ShortageOfServices[Transport]+ShortageOfSpaces.average()))*(1-CorePlanBudgetRatio)+ TypeBudgetDistributionPolicy[DebrisDisposal]*CorePlanBudgetRatio) *(RegionalDamageRatio[Region] |

| Variables | Equations |
|---|---|
| Workforce Distribution [Recovery, Region] | **Workforce Distribution [Recovery, Region]=** AvailableWorkforcesInRegion * WorkforceDistributionRatio[Recovery,Region] / WorkforceDistributionTime |
| Available WorkforcesInRegion | **AvailableWorkforcesInRegion=** WorkForceSupply - WorkforceDistribution.sum() |
| Workforce Adjustment [Facilities, Region] | **WorkforceAdjustment [Recovery, Region] =** AllocatedWorkforces.sum() > 0 ? (WorkforceDistributionRatio[Recovery,Region] - AllocatedWorkforces[Recovery,Region] / AllocatedWorkforces.sum()) * AllocatedWorkforces.sum() /WorkforceAdjustmentTime : 0 |
| Allocated Workforces [Recovery, Region] | **d(AllocatedWorkforces [Recovery, Region])/dt =** WorkforceAdjustment[Recovery,Region]+ + WorkforceDistribution[Recovery,Region] |
| Allocated Workforces ForDisposal [Region] | **AllocatedWorkforcesForDisposal [Region] =** AllocatedWorkforces.sum(INDEX_CAN_VARY, Region) |
| RegionalFacility Functionality [Facilities, Region] | **RegionalFacilityFunctionality [Facilities, Region] =** min(1,MaximumDamage[Facilities,Region] > 0 ? 1-(NeedForRepair[Facilities,Region] /MaximumDamage[Facilities,Region]) : 0) |
| ShortageOfResourceProduction | **ShortageOfResourceProduction=** ShortageOfServices.sum(Economic) |
| MaterialSupply | **MaterialSupply=** ReliefEffortsMaterials + NormalMaterialSupplyCapacity * ( 1 - ShortageOfResourceProduction ) |
| Repair [Facilities, Region] | **Repair [Facilities, Region]=** delay3 (NeedForRepair[Facilities,Region] > 0 ? min ( NeedForRepair[Facilities,Region], ExpectedRepairRate[Facilities,Region] * ResourceSupplyRatio[Facilities,Region]) : 0 , EmergencyResponseTime)//delay3 ( DamageGeneration[Facilities] , EmergencyResponseTime ) |
| DisposalDone [Facilities, Region] | **d(DisposalDone [Facilities, Region])/dt =** Disposal[Facilities,Region] |
| Disposal [Facilities, Region] | **Disposal [Facilities, Region] =** Debris[Facilities,Region] > 0 ? min ( Debris[Facilities,Region], (Debris[Facilities,Region] / Debris.sum(INDEX_CAN_VARY,Region)) * WorkEfficiency[Facilities,Region] * DebrisReadyToDisposal[Region] * DisposalWorkforceSupplyRatio[Region] ) : 0 |
| DebrisGeneration [Facilities, Region] | **DebrisGeneration [Facilities, Region] =** RepairToDoGeneration[Facilities,Region] + Repair[Facilities,Region] * DebrisGenerationRatio |
| Workforce Distribution [Facilities, Region] | **WorkforceDistribution [Facilities, Region] =** AvailableWorkforcesInRegion * WorkforceDistributionRatio[Facilities, Region] / WorkforceDistributionTime |

| Variables | Equations |
|---|---|
| Workforce Adjustment [Facilities, Region] | **WorkforceAdjustment [Facilties, Region]=** AllocatedWorkforces.sum() > 0 ? (WorkforceDistributionRatio[Facilities,Region] - AllocatedWorkforces[Facilities,Region] / AllocatedWorkforces.sum()) * AllocatedWorkforces.sum() /WorkforceAdjustmentTime : 0 |
| MaterialSupply | **MaterialSupply=** ReliefEffortsMaterials + NormalMaterialSupplyCapacity * ( 1 - ShortageOfResourceProduction ) |
| AllocatedMaterials [Recovery, Region] | **d(AllocatedMaterials [Recovery, Region])/dt =** MaterialDistribution[Recovery,Region] - MaterialConsumption[Recovery,Region] |
| MaterialDistribution [Facilities, Region] | **MaterialDistribution [GeneralRestore, Region] =** MaterialNeeds[Facilities,Region] > AccumulatedAllocatedMaterials[Facilities,Region] ? AvailableMaterialsInRegion * MaterialDistibutionRatio[Facilities,Region] / MaterialDistributionTime : 0 |
| Material Consumption [Facilities, Region] | **MaterialConsumption [Facilities, Region] =** Repair[Facilities, Region] * RequiredMaterialsPerWork[Facilities] |
| DamageGeneration [Facilities, Region] | **DamageGeneration [Facilities, Region] =** ScaleOfStructures[Facilities,Region] * StructureDamageIntensity[Facilities,Region] |

**[Interface Variables: Restoration Conditons]**

| Variables | Equations |
|---|---|
| DeliveryDelayRatio [Region] | DeliveryDelayRatio[Region]= 1-RegionalFacilityFunctionality[Transport, Region] |
| MaterialSupply Ratio [Facilities, Region] | MaterialSupplyRatio[Facilities, Region]= MaterialNeeds[Facilities, Region] > 0 ? min ( 1 , AccumulatedAllocatedMaterials[Facilities, Region] / MaterialNeeds[Facilities, Region] ) : 0 |
| WorkforceSupply Ratio [Facilities, Region] | WorkforceSupplyRatio[Facilities, Region]= WorkforceNeeds[Facilities, Region] > 0 ? min ( 1 , AllocatedWorkforces[Facilities, Region] / WorkforceNeeds[Facilities, Region] ) : 0 |
| WorkDelayRatio [Facilities, Region] | DeliveryDelayRatio[Region]= ShortageOfSpaces[Facilities,Region] |

## [Input Values from Excel Files: Regional Locational Information]

| | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 25 | | | | | | | | | | | | |
| 26 | | **Parameters** | | | | | | | | | | |
| 27 | | **[Japan]** | | | | | | | | | | |
| 28 | | Initial Workforce | | 120000 | | | | | | | | |
| 29 | | Initial Materials | | 122732410 | | | | | | | | |
| 30 | | Material Supply | | 4166373.089 | | | | | | | | |
| 31 | | Seismic Performance Level | | {.1, .12, .12, .12} | | Damping Ratio 10%, Seismic Importance Factor I=1.25(Essential), I=1.0 | | | | | | |
| 32 | | Construction Duration | | {150, 210, 210, 240} | | | | | | | | |
| 33 | | Required Materials Work | | 1.2 | | Calculated by ratio between Budget and Damage | | | | | | |
| 34 | | CorePlanBudgetRatio | | 0.171005917 | | | | | | | | |
| 35 | | | | | | | | | | | | |
| 36 | | | | | | | | | | | | |
| 37 | | | | | | | | | | | | |
| 38 | | | | | | | | | | | | |
| 39 | | | | | | | | | | | | |
| 40 | | *Total Economic Damage* | | *177700* | | *MUSD* | (RA 2013) | | | | | |
| 41 | | *Total Budget Scale* | | *199781.0651* | | *MUSD* | (RA 2013) | | | | | |
| 42 | | *Initial Budget* | | *42200* | | *MUSD* | (RA 2013) | | | | | |
| 43 | | *Additional Budget* | | *157581.0651* | | *MUSD* | (RA 2013) | | | | | |
| 44 | | *Physical Damage* | | *484194541.5* | | *M3* | | | | | | |
| 45 | | *Required Resources* | | *581033449.8* | | *M3* | | | | | | |
| 46 | | *Core Project Budget* | | *34163.74427* | | | | | | | | |
| 47 | | | | | | | | | | | | |
| 48 | | | | | | | | | | | | |

| | M | N | O | P | Q | R | S | T |
|---|---|---|---|---|---|---|---|---|
| 25 | | | | | | | | |
| 26 | | **Hyper array Data** | | | | | | |
| 27 | | **[Japan]** | | | | | | |
| 28 | | **Latitude** | | | **Longitude** | | | |
| 29 | | Iwate Pref | 39.7 | | Iwate Pref | 141.2 | | |
| 30 | | Miyagi Pref | 38.3 | | Miyagi Pref | 140.9 | | |
| 31 | | Fukushima Pref | 37.8 | | Fukushima Pref | 140.5 | | |
| 32 | | RegionD | 0 | | RegionD | 0.0 | | |
| 33 | | RegionE | 0 | | RegionE | 0.0 | | |
| 34 | | | | | | | | |
| 35 | | **BE Scale** | Iwate Pref | Miyagi Pref | Fukushima Pref | RegionD | RegionE | |
| 36 | | R | 185,625,546 | 430,165,701 | 303,187,249 | 0 | 0 | |
| 37 | | C | 28,932,110 | 54,506,501 | 49,652,099 | 0 | 0 | |
| 38 | | I | 56,732,448 | 110,962,621 | 49,424,052 | 0 | 0 | |
| 39 | | T | 63,650,616 | 68,324,273 | 55,124,950 | 0 | 0 | |
| 40 | | | | | | | | |
| 41 | | **Disposal Capacity** | | | | | | |
| 42 | | Iwate Pref | 2,765,543 | | | | | |
| 43 | | Miyagi Pref | 1,879,698 | | | | | |
| 44 | | Fukushima Pref | 1,354,759 | | | | | |
| 45 | | RegionD | 0 | | | | | |
| 46 | | RegionE | 0 | | | | | |

## *B-III Facility Damage Assessment Model*

A facility-level damage assessment model in the Anylogic recovery simulation federate aims to immediately assess facility's structural damage by the integrated uses of an USGS seismic data retrieval federate and an OpenSees structural response simulation federate. The estimated damage can be used as input values in the DES restoration simulation model.

In this model, earthquake information from the USGS federate is converted to ground motion data that will be used as an input data for the OpenSees structural response simulation. By using the historical ground motion database and the data structure in the "*AccHistory*" Class, most suitable ground motion data is selected from the ground motion database to estimate structural responses. Then, the nodal displacement values of structures from the OpenSees federate are subscribed to the facility-level damage assessment model in the Anylogic federate by using data structure in the "*Displ*" Class. Based on the simulation process of this model as shown in previous figure (Fig. 4.6) in Ch. 4.2.1, computational codes for this model are provided in detail.

**[“*AccHistory*” Class Structure]**

```
/* AccHistory Class for ground motion selection from database */

public class AccHistory implements java.io.Serializable {
     double mw;      // mw of accelerograph (acc.) history data
     int eventtime;     // event time of acc. history data
     double eventlat;   // event latitude of acc. history data
     double eventlon;   // event longitude of acc. history data
     double epitoepidist;   // distance btw epicenters
     String stname;     // station that measured acc. history data
     int dataid;    // data number of acc. history data
     double stlat;          // latitude of station
     double stlon;          // longitude of station
     double stpga;          // PGA of  measured acc. history data
     double stepidist;      // distance btw epicenter and station
     double ftostdist;      // distance btw facility and station
 /* Default constructor */
    public AccHistory(){
         this.mw = 0;
         this.eventtime = 0;
         this.eventlat = 0;
         this.eventlon = 0;
         this.epitoepidist = 0;
         this.stname = null;
         this.dataid = 0;
         this.stlat = 5;
         this.stlon = 0;
         this.stpga = 0;
         this.stepidist = 0;
         this.ftostdist = 0;
    }
/* Constructor initializing the fields */
    public AccHistory(double mw, int eventtime, double eventlat,
                      double eventlon,  double epitoepidist,
                      String stname, int dataid, double stlat,
                      double stlon, double stpga, double stepidist,
                      double ftostdist){
         this.mw = mw;
         this.eventtime = eventtime;
         this.eventlat = eventlat;
         this.eventlon = eventlon;
         this.epitoepidist = epitoepidist;
         this.stname = stname;
         this.dataid = dataid;
         this.stlat = stlat;
         this.stlon = stlon;
         this.stpga = stpga;
         this.stepidist = stepidist;
         this.ftostdist = ftostdist;
    }
```

**[“*Displ*” Class Structure]**

```
/* Displ Class for assessing damage from displacement data */

public class Displ implements java.io.Serializable {
     int floornum;      //facility's floors
     double height;     //facility's floor height
     double drift;      //facility's story drift
     double okdrift;//facility's acceptable story drift

    /* Default constructor */
    public Displ(){
    }

    /* Constructor initializing the fields */
    public Displ(int floornum, double height, double drift,
                double okdrift){
         this.floornum = floornum;
         this.height = height;
         this.drift = drift;
         this.okdrift = okdrift;
    }

    @Override
    public String toString() {
         return
              "floornum = " + floornum +" " +
              "height = " + height +" " +
              "drift = " + drift +" " +
              "okdrift = " + okdrift +" ";
    }
    private static final long serialVersionUID = 1L;
}
```

**[“*FederationAmbassador*” Class Structure]**

```
import jhla.*;
import jhla.Federate.*;
import certi.rti.impl.*;
import hla.rti.*;
import hla.rti.jlc.*;

/* FedAmbassador class for interactions with USGS and OpenSees */

public class FederationAmbassador extends Federate implements
Serializable {
    public class USGSRequestInteraction implements Serializable {
        public Interaction parent;
        public Interaction.Parameter StartTime;
        public Interaction.Parameter EndTime;
        public Interaction.Parameter MinMagnitude;
        public Interaction.Parameter MinLatitude;
        public Interaction.Parameter MaxLatitude;
        public Interaction.Parameter MinLongitude;
        public Interaction.Parameter MaxLongitude;

        private static final long serialVersionUID = 1L;
    }

    public class EarthquakeInteraction implements Serializable {

        public Interaction parent;
        public Interaction.Parameter EventID;
        public Interaction.Parameter Time;
        public Interaction.Parameter Magnitude;
        public Interaction.Parameter Depth;
        public Interaction.Parameter Latitude;
        public Interaction.Parameter Longitude;

        public EarthquakeHandler Handler =
                    new EarthquakeHandler();

        private static final long serialVersionUID = 1L;

        public class EarthquakeHandler implements UpdateHandler,
        Serializable {

            public void execute(DataClass sender) {
                owner.OnEarthquakeUpdate(sender);
            }

            private static final long serialVersionUID = 1L;
        }
    }
```

**[Detailed Descriptions of the Model]**

```
/* Main function for facility damage assessment */

void OnStructuralDisplacementUpdate(  ) {

  System.out.println("Got StructuralDisplacement");

  Floor = FRestore.Nfloor;
  int AnalFloor = 4;
  double fheight = 6.7;
  double driftlmt = 0.02;

  CustomTypes.DoubleArray values = FedAmb.StructuralDisplacement.
  Displacements.getCustom(CustomTypes.DoubleArray.class);

  for (double val : values)
  {
    System.out.println(val);
  }

  int N = AnalFloor*2;
  double[] vals = new double[N];
  for( int i=0; i<N; i++ ) {
  vals[i] = values.get(i);
  }

  int datanum = AnalFloor;
  int rowout = 0;
  int rowin = AnalFloor;
  Displ [] flout;
  flout = new Displ[AnalFloor];

  for (int i=0; i<AnalFloor; i++)
  {
    flout[i] = new Displ();
    flout[i].floornum = i+1;
    flout[i].height = flout[i].floornum * fheight;
    if (i == 0) {flout[i].drift = abs(vals[i]);}
    if (i != 0)
    {
        flout[i].drift = abs(vals[i]-vals[i-1]);
    }

    flout[i].okdrift = flout[i].height * driftlmt;
  }
```

```
  Displ [] flin;
  flin = new Displ[AnalFloor];

  for (int i=0; i<AnalFloor; i++)
  {
    flin[i] = new Displ();
    flin[i].floornum = i+1;
    flin[i].height = flin[i].floornum * fheight;
    if (i == 0) {flin[i].drift = abs(vals[i+AnalFloor]);}
    if (i != 0)
    {
        flin[i].drift=abs(vals[i+AnalFloor]-vals[i+AnalFloor-1]);
    }
    flin[i].okdrift = flin[i].height * driftlmt;
  }
  int dmgout = 0;
  int dmgin = 0;

  for (int i=0; i<AnalFloor; i++)
  {
    if (flout[i].drift > flout[i].okdrift)
    {dmgout++;}
    else if (i != 0 && flout[i-1].drift > flout[i-1].okdrift)
   {dmgout++;}
    if (flin[i].drift > flin[i].okdrift)
    {dmgin++;}
    else if (i != 0 && flin[i-1].drift > flin[i-1].okdrift)
   {dmgin++;}
  }

  CoreDamageIntensity = (double) dmgin / AnalFloor;
  FrameDamageIntensity = (double) dmgout / AnalFloor;
}
    /* Sets StartTime and EndTime in FedAmb.USGSRequest, then
  updates the interaction with the HLA. All other parameters in
  FedAmb.USGSRequest should already be set on initialization. */

/* Function for facility epicentral distance calculation */
 void FacilityDistanceCalculator( ) {

    double dLat, dLong, a, c;
    dLat = 0;
    dLong = 0;
    a = 0;
    c = 0;
    if(newEventMag_Value > 0)
    dLat = newEventLatitude - F_Lat;
    dLong = newEventLongitude - F_Long;
    a = pow((sin(dLat*3.141592/180/2)),2)
        + cos(F_Lat*3.141592/180)*cos(newEventLatitude*3.14/180)
        * pow((sin(dLong*3.141592/180/2)),2);
    c = 2 * atan(sqrt(a)/sqrt(1-a));
    FDtoEpicenter = c*6370000;
}
```

```
/* Function for facility PGA calculation */

 void FacilityPGACalculator( ) {

    double a1, b1, c1, d1, e1, a2, b2, c2, e2;
    a1=0.56;
    b1=-0.0031;
    c1=0.26;
    d1=0.0055;
    e1=0.37;
    a2=0.41;
    b2=-0.0039;
    c2=1.56;
    e2=0.40;
    if(newEventDepth_Value <= 30000) {
        FPGA = pow(10, (a1 * newEventMag_Value + b1
               *(FDtoEpicenter/1000) - log10((FDtoEpicenter/1000)
               + d1 * pow(10, 0.5*newEventMag_Value)) + c1 + e1));
    }else {
        FPGA = pow(10, (a2 * newEventMag_Value + b2
               * (FDtoEpicenter/1000)
               - log10((FDtoEpicenter/1000)) + c1 + e1));
     } /* by Kanno et al. (2006)'s Equations */
 }

/* Gunction for facility PGV calculation */

 void FacilityPGVCalculator( ) {

    double a1, b1, c1, d1, e1, a2, b2, c2, e2;
    a1=0.70;
    b1=-0.0009;
    c1=-1.93;
    d1=0.0022;
    e1=0.32;
    a2=0.55;
    b2=-0.0032;
    c2=-0.57;
    e2=0.36;
    if(newEventDepth_Value <= 30000) {
        FPGV = pow(10, (a1 * newEventMag_Value + b1
               * (FDtoEpicenter/1000) - log10((FDtoEpicenter/1000)
               + d1 * pow(10, 0.5*newEventMag_Value)) + c1 + e1));
    }else {
        FPGV = pow(10, (a2 * newEventMag_Value + b2
               * (FDtoEpicenter/1000)
               - log10((FDtoEpicenter/1000)) + c1 + e1));
     } /* by Kanno et al. (2006)'s Equations */
 }
```

```
/* Function for initiating HLA */

 void InitHLA( ) {

    System.out.println(this.toString())
    if (!this.toString().equals("root.damage"))
    {
    System.out.println("initHLA");
    try
    {
        FedAmb = new FederationAmbassador(this);

        FedAmb.USGSRequest.MinMagnitude.setString
          (String.valueOf(Min_Magnitude));

          FedAmb.USGSRequest.MinLatitude.setString
          (String.valueOf(Min_Latitude));

          FedAmb.USGSRequest.MaxLatitude.setString
          (String.valueOf(Max_Latitude));

          FedAmb.USGSRequest.MinLongitude.setString
          (String.valueOf(Min_Longitude));

          FedAmb.USGSRequest.MaxLongitude.setString
          (String.valueOf(Max_Longitude));
    }
    catch (RTIexception ex)
    {
        System.out.println(ex.getMessage());
        ex.printStackTrace();
    }
    }
}
  /* This function automatically exectutes when FedAmb.Earthquake
  is updated from the HLA. It should read the updated values from
  FedAmb.Earthquake, perform any necessary manipulations, and
  then update FedAmb.GroundMotion. */
```

```
/* Function for earthquake information request to USGS */

 void UpdateUSGSRequest(  ) {

   double MaxTime = time() + 1;
   double MinTime = time();

   End_Date = timeToDate(MaxTime);
   Start_Date = timeToDate(MinTime);
   java.text.SimpleDateFormat sdf
   = new java.text.SimpleDateFormat( "yyyy-MM-dd" );

   try
   {
    FedAmb.USGSRequest.StartTime.setString
     (sdf.format(Start_Date) + "T00:00:00");
    FedAmb.USGSRequest.EndTime.setString(sdf.format(End_Date)
     + "T00:00:00");
    FedAmb.USGSRequest.parent.update();
   }
   catch (RTIexception ex)
   {
    System.out.println("Error in UpdateUSGSRequest(): "
     + ex.getMessage());
    ex.printStackTrace();
   }
}
   /* Sets all parameters in FedAmb.GroundMotion before updating
   the interaction with the HLA. */

/* Function for Time Update */

 if (FedAmb != null)
 {
    UpdateUSGSRequest();
    try
    {
        FedAmb.tick();
    }
    catch (RTIexception ex)
    {
        System.out.printIn("Error in TimeUpdate(): "
         + ex.getMessage());
        ex.printStackTrace();
    }
 }
```

```
/* Function for earthquake information Retrieval */

 void OnEarthquakeUpdate( Federate.DataClass sender ) {

   eventGeneration();

   try
   {
    /* TODO reset earthquake values from test values */
    newEventMag_Value = 9.0;
         /* Double.parseDouble(sender.getMember("Magnitude").
         getString()); */
    newEventDepth_Value = 29000.0;
         /* Double.parseDouble(sender.getMember("Depth").
         getString());*/
    newEventLatitude = 38.297;
         /* Double.parseDouble(sender.getMember("Latitude").
         getString()); */
    newEventLongitude = 142.373;
         /* Double.parseDouble(sender.getMember("Longitude").
         getString());*/
   }
   catch (Exception ex)
   {
    System.out.println("Error in OnEarthquakeUpdate(): "
     + ex.getMessage());
    ex.printStackTrace();
    return;
   }
   FacilityDistanceCalculator();
   FacilityPGACalculator();
   FacilityPGVCalculator();
   RegionalDamageIntensityEstimator();
   AccelerographSelector();
   GroundMotionSFCalculator();

   UpdateGroundMotion();

 }
```

```
/* Function for selecting ground motion history data */

 void AccelerographSelector( ) {

   String sheet = "Ground Motion"
   int datanum = (int)excelFile.getCellNumericValue(sheet,5,15);
   int row = 10;
   AccHistory [] acc;
   acc = new AccHistory[datanum];
   for (int i=0; i<datanum; i++)
   {
    acc[i] = new AccHistory();
    acc[i].mw = excelFile.getCellNumericValue(sheet, row, 3);
    acc[i].eventtime = (int)excelFile.getCellNumericValue
                    (sheet, row, 4);
    acc[i].eventlat = excelFile.getCellNumericValue
                    (sheet, row, 5);
    acc[i].eventlon = excelFile.getCellNumericValue
                    (sheet, row, 6);
    acc[i].epitoepidist = EpiDistanceCalculator
                    (acc[i].eventlat, acc[i].eventlon);
    acc[i].stname = excelFile.getCellStringValue(sheet, row, 8);
    acc[i].dataid = (int)excelFile.getCellNumericValue
                    (sheet, row, 1);
    acc[i].stlat = excelFile.getCellNumericValue
                    s(heet, row, 10);
    acc[i].stlon = excelFile.getCellNumericValue
                    (sheet, row, 11);
    acc[i].stpga = excelFile.getCellNumericValue
                    (sheet, row, 15);
    acc[i].stepidist = EpiDistanceCalculator
                    (acc[i].stlat, acc[i].stlon);
    acc[i].ftostdist = FtoStDistanceCalculator
                    (acc[i].stlat, acc[i].stlon);
    row++;
 }
   double mwerr = 0.1;
   /* Mw error for selecting history data */
   double epidisterr = 5000;
   /*(m) Distance error for selecting history data */
   int candimwnum = 0;
   int candidistnum = 0;
   int candinum = 0;

   AccHistory [] candidate;
   candidate = new AccHistory[datanum];
```

```
    while (candinum == 0) {
    for (int i=0; i<datanum; i++)
    {
     candidate[i] = new AccHistory();
     if (abs(newEventMag_Value-acc[i].mw) < mwerr)
     {
         candimwnum++;
         if (acc[i].epitoepidist < epidisterr)
         {
             candidate[i] = acc[i];
             candidistnum++;
             candinum = min(candimwnum, candidistnum);
         }
     }
    }
    if (candimwnum == 0) {mwerr += 0.1;}
    if (candidistnum == 0) {epidisterr += 5000;}
    }
    AccHistory target = new AccHistory();
    for (int i=0; i<datanum; i++)
    {
     if (candidate[i].dataid != 0)
         {
             target = candidate[i];
             break;
         }
    }
    for (int i=0; i<datanum; i++)
    {
     if (candidate[i].dataid != 0)
     {
         if (target.ftostdist > candidate[i].ftostdist)
         {
             target.stname = null;
             target = candidate[i];
         }
         if (target.ftostdist == candidate[i].ftostdist)
         {
             if ( abs(target.stpga-FPGA) >
                 abs(candidate[i].stpga-FPGA) )
             {
                 target.stname = null;
                 target = candidate[i];
             }
         }
     }
    }
    AccDataStName = target.stname;
    AccDataPGA = target.stpga;
    AccDataEventTime = target.eventtime;
    AccDataFtoStDist = target.ftostdist;

    return;
}
```

```
/* Function for calculating scale factor */

void GroundMotionSFCalculator( ) {

   double GMFileSF = 0;
   String sheet = "Ground Motion";

   GMFileSF = excelFile.getCellNumericValue(sheet, 4, 15);
   ScaleFactor = GMFileSF * FPGA / AccDataPGA;
}

/* Function for sending ground motion data */

void UpdateGroundMotion( ) {

  if (FedAmb != null)
  {
    String gmFilePath = System.getProperty("user.dir")
     + "/GMfiles/" + AccDataStName.substring(1,7)
     + AccDataEventTime + ".NS.tcl";

     Double sFactor = ScaleFactor / 980.665;

    eventDisappeared();

    try
    {
        FedAmb.GroundMotion.AccelerationFilePath.setString
          (gmFilePath);

    FedAmb.GroundMotion.ScaleFactor.setString(sFactor.
          toString());
        FedAmb.GroundMotion.parent.update();
    }
    catch (Exception ex)
    {
        System.out.println("Error in UpdateGroundMotion(): "
          + ex.getMessage());
        ex.printStackTrace();
    }
  }
}
```

## [Input Values from Excel Files: Facility Locational Information]

The data including facility location, size, and structural information is provided in the Excel databases, as shown below. These values are automatically input to the facility-level damage assessment model when initiating the simulations.

| | A | B | C |
|---|---|---|---|
| 9 | | | |
| 10 | | **Parameters** | |
| 11 | | **[Y building]** | |
| 12 | | Latitude | 38.2692 |
| 13 | | Longitude | 140.8719 |
| 14 | | Basement | |
| 15 | | Floor | 8 |
| 16 | | PH | |
| 17 | | Physical Scale | 30,106 |
| 18 | | Outriggar Floor1 (F) | |
| 19 | | Outriggar Floor2 (F) | |

## [Input Values from Excel Files: Ground Motion Database]

The historical ground motion data is also provided in the Excel databases, as shown below. The most suitable ground motion data for assessing facility's structural responses is automatically input to the facility-level damage assessment model when initiating the simulations.

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | ***DISASTER MANAGEMENT DATABASE*** | | | | | | | | | | | | | | | |
| 2 | ***Ground Motion*** | | | | | | | | | | | | | | | |
| 3 | CESMD Engineering Strong Motion Data Center | | | | | 39.7036 | | | | | | | | | | |
| 4 | http://www.strongmotioncenter.org/ | | | | | 141.153 | | | | *Time Step* | *0.01* | *(s)* | | *Scale Factor* | *0.000634021* | |
| 5 | | | | | | | | | | *Duration* | *300* | *(s)* | | *No. of Data* | *12* | |
| 6 | | | | | | | | | | | | | | | | |
| 7 | | | | | | | | | | | | | | | | |
| 8 | Data # | Event ID | Mw | Event Time | Location of Event | | Station | Code | Network | Location of St. | | Distance (km) | | Horiz Apk (g) | Horiz Apk (gal) | File Name |
| 9 | | | | | Lat. | Long. | | /ID | | Lat. | Long. | Epic. | Fault | Ground | Ground | |
| 10 | 1 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | Tsukidate - MYG004 | MYG004 | KNET | 38.7292 | 141.022 | 125.9 | 75.1 | 2.755 | 2702.655 | MYG0041103111446.NS |
| 11 | 2 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | Shiogama - MYG012 | MYG012 | KNET | 38.3175 | 141.019 | 118.1 | 67.4 | 2.009 | 1970.829 | MYG0121103111446.NS |
| 12 | 3 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | HITACHI | IBR003 | KNET | 36.5915 | 140.645 | 245.2 | 58.7 | 1.631 | 1600.011 | IBR0031103111446.NS |
| 13 | 4 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | Sendai - MYG013 | MYG013 | KNET | 38.2663 | 140.929 | 126.1 | 71.8 | 1.548 | 1518.588 | MYG0131103111446.NS |
| 14 | 5 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | HOKOTA | IBR013 | KNET | 36.1587 | 140.489 | 292.3 | 77.8 | 1.383 | 1356.723 | IBR0131103111446.NS |
| 15 | 6 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | SHIRAKAWA | FKS016 | KNET | 37.1228 | 140.191 | 233.6 | 98 | 1.322 | 1296.882 | FKS0161103111446.NS |
| 16 | 7 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | OHMIYA | IBR004 | KNET | 36.5516 | 140.41 | 262 | 71.4 | 1.309 | 1284.129 | IBR0041103111446.NS |
| 17 | 8 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | KOHRIYAMA | FKS018 | KNET | 37.3961 | 140.362 | 204.4 | 91.6 | 1.091 | 1070.271 | FKS0181103111446.NS |
| 18 | 9 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | FUNEHIKI | FKS008 | KNET | 37.4363 | 140.567 | 186.6 | 78.3 | 1.032 | 1012.392 | FKS0081103111446.NS |
| 19 | 10 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | ICHINOSEKI | IWT010 | KNET | 38.9334 | 141.117 | 128.4 | 73.2 | 1.018 | 998.658 | IWT0101103111446.NS |
| 20 | 11 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | KAMAISHI | IWT007 | KNET | 39.2701 | 141.856 | 114.3 | 49 | 0.711 | 697.491 | IWT0071103111446.NS |
| 21 | 12 | usc0001xgp | 9.0 | 1103111446 | 38.297 | 142.372 | KITAKAMI | IWT012 | KNET | 39.3209 | 141.138 | 154 | 79.7 | 0.603 | 591.543 | IWT0121103111446.NS |

## *B-IV Facility-Level Restoration Analysis Model Using DES*

A DES facility-level restoration analysis model in the Anylogic recovery simulation federate aims to analyze facility restoration operations according to estimated facility damage from the facility damage assessment model, which can be useful for facility-level restoration planning. To analyze the effects of unfavorable post-disaster external project conditions as well as diverse damage patterns on facility restoration projects, the descriptions of the SD-DES interactions in the multi-level recovery simulation are provided with a consideration of both damage assessment models. Based on the simulation model as shown in previous figures (Figs. 4.9 and 4.12) in Ch. 4.2.2 and 4.3, this study provides the descriptions of model structures and equations as well as input values.

**[Interface Variables]**

```
/* For general facility restoration in A region */

  /* Restoration condition updates at every time step */
  MaterialSupplyRatio =
     get_Main().MaterialSupplyRatio.sum(General, Aregion);
  WorkforceSupplyRatio =
     get_Main().WorkforceSupplyRatio.sum(General, Aregion);
  DeliveryDelayRatio =
     get_Main().DeliveryDelayRatio.sum(Aregion);
  WorkDelayRatio =
     get_Main().WorkDelayRatio.sum(General, Aregion);;

  MaterialSupplyAmount =
     (int) (Math.ceil(MaterialSupplyRatio * WorkAmountForFloor));

  /* Facility damage updates when an event occurs */
  FDamage =
     (int) (Math.ceil( Math.max(Damage.CoreDamageIntensity,
        Damage.FrameDamageIntensity) * NumberOfUpperFloor));
```

**[Input Values for Activity Durations from Actual Data]**

By investigating actual building reconstruction processes data from the 8-story residential/commercial complex building as an example case, including removal/demolition of interior/external materials of the building, structural reinforcement, deck and frame installation, and curtain wall works, the DES model uses actual activity information as input values, as shown below.

| Variables | Values |
|---|---|
| NormalSiteWorkTime | 1.5 (weeks) |
| NormalDemolishTime | 1 (weeks) |
| NormalReinforceTime | 2 (weeks) |
| NormalDeliveryTime | 0.06 (weeks) |
| NormalColumnInstallTime | 0.8 (weeks) |
| NormalBeamInstallTime | 0.6 (weeks) |
| NormalInspectionTime | 0.2 (weeks) |
| NormalWeldingTime | 0.4 (weeks) |
| NormalFrameFoundationTime | 2 (weeks) |
| NormalDeckInstallTime | 0.4 (weeks) |
| NormalReBarInstallTime | 0.4 (weeks) |
| NormalPouringTime | 0.1 (weeks) |
| NormalCwallInstallTime | 1.9 (weeks) |

**[Definitions of Activities in DES Model]**

| Variables | Resource Input | Durations |
|---|---|---|
| SiteWork | SiteWorkCrew | (triangular(1 * (NormalSiteWorkTime-ActiVariation) , NormalSiteWorkTime, NormalSiteWorkTime * (1+ActiVariation) )) * (1 / (1-WorkDelayRatio)) |
| Demolish | DemolishCrew | (triangular(1 * (NormalDemolishTime-ActiVariation) , NormalDemolishTime, NormalDemolishTime * (1+ActiVariation) )) * (1 / (1-WorkDelayRatio)) |
| Reinforce | ReinforceCrew | (triangular(1 * (NormalReinforceTime-ActiVariation) , NormalReinforceTime, NormalReinforceTime * (1+ActiVariation) )) * (1 / (1-WorkDelayRatio)) |
| Delivery (All materials) | All materials | triangular(NormalDeliveryTime * (1-ActiVariation) , NormalDeliveryTime, NormalDeliveryTime * (1+ActiVariation) ) * (1 / (1-DeliveryDelayRatio)) |
| ColumnInstall | Frame Material FrameCrew | triangular(NormalColumnInstallTime * (1-ActiVariation) , NormalColumnInstallTime, NormalColumnInstallTime * (1+ActiVariation) ) * (1 / WorkforceSupplyRatio) |
| BeamInstall | Frame Material (Column install done) FrameCrew | triangular(NormalBeamInstallTime * (1-ActiVariation) , NormalBeamInstallTime, NormalBeamInstallTime * (1+ActiVariation) ) * (1 / WorkforceSupplyRatio) |
| Inspection | Frame Material (Beam install done) | triangular(NormalInspectionTime * (1-ActiVariation) , NormalInspectionTime, NormalInspectionTime * (1+ActiVariation) ) |
| Welding | Frame Material (Inspection done) FrameCrew | triangular(NormalWeldingTime * (1-ActiVariation) , NormalWeldingTime, NormalWeldingTime * (1+ActiVariation) ) * (1 / WorkforceSupplyRatio) |
| DeckInstall | Deck Material Frame Done Frame Crew | triangular(NormalDeckInstallTime * (1-ActiVariation) , NormalDeckInstallTime, NormalDeckInstallTime * (1+ActiVariation) ) * (1 / WorkforceSupplyRatio) |
| ReBarInstall | Deck (Deck install done) ReBarCrew | triangular(NormalReBarInstallTime * (1-ActiVariation) , NormalReBarInstallTime, NormalReBarInstallTime * (1+ActiVariation) ) * (1 / WorkforceSupplyRatio) |
| RmcPouring | Deck (Rebar install done) RMC RmcCrew | triangular(NormalPouringTime * (1-ActiVariation) , NormalPouringTime, NormalPouringTime * (1+ActiVariation) ) * (1 / WorkforceSupplyRatio) |
| CwallInstall | CurtainWall Material DeckDone CwallCrew | triangular(NormalCwallInstallTime * (1-ActiVariation) , NormalCwallInstallTime, NormalCwallInstallTime * (1+ActiVariation) ) * (1 / WorkforceSupplyRatio) |

* ActiVariation: Range of activity duration variations (±20%)

# Appendix C: HLA Development

## *C-I Importance Methods for HLA Development*

The following are details concerning some of the more important methods within each of the classes of the interactive recovery simulation, based on the previous chapter, Ch. 5. It is not an exhaustive list, for which users of the API should consult the header files. For method calls where the C++ and Java implementations differ significantly, both calls are provided--however, when the differences between implementations are essentially cosmetic (e.g. a C++ type is `const char*` and its Java type is `String`), only the C++ version is shown.

### HLA::Federate

· ***`Federate(const char* p_federateName)`***

The constructor for a federate instance. The argument `p_federateName` attaches a unique name to the federate which can later be referenced.

· ***`void join(const char* p_executionName, const char* p_FEDid, int p_timeoutInSeconds = 5)`***

Join a federation with the name `p_executionName` using the FOM file `p_FEDid`. If the federation does not yet exist, it will be created and the caller will be labeled as the "creator."

· ***void resign()***

Unpublish/unsubscribe all objects and attributes and resign from the federation. If the federate is labeled as the "creator," it will attempt to destroy the federation. If other federates are still joined, the Federate object will persist in memory and periodically check for joined federates. Once all have resigned, the federation and the Federate instance will be destroyed.

· ***Object* createObject(const char* p_className, const char* p_objectName, PubSubType p_subscriptionType)***

· ***Interaction* createInteraction(const char* p_className, PubSubType p_subscriptionType)***

Create an object or interaction with the name `p_className`, which must exactly match the name given in the FOM file. The enumerated type `PubSubType` is passed to indicate whether the object/interaction is published or subscribed. In the case of an object, `p_objectName` is used to provide a unique name for the object instance (this is not necessary for interactions, as only one interaction instance can be created from an interaction definition). A pointer to the newly created object or interaction is returned, which must be stored by the user.

· ***void publishSubscribe()***

Publish and/or subscribe all objects and interactions. This must be called after every object, interaction, attribute, and parameter have been created.

· ***void updateAll(const char* p_tag = "")***

Update all objects and interactions with the federation. This is equivalent to calling `HLA::DataClass::update()` on every object and interaction. The argument `p_tag` is used for debugging purposes--it attaches a unique tag to the particular update.

### HLA::DataClass

`HLA::DataClass` is the superclass for `HLA::Object` and `HLA::Interaction`. Any of these methods can be used with either objects or interactions.

Note that the constructor is not shown, as the user should never instantiate object or interaction instances themselves (instead, use `HLA::Federate::createObject()` or `HLA::Federate::createInteraction()`).

· ***(C++) void addHandler(function<void (const DataClass&)> p_handler)***

· ***(Java) void addHandler(UpdateHandler p_handler)***

Add handler `p_handler` to the object/interaction. The handler will be executed whenever the object or interaction data is updated with the federation. Its signature must have a return type of `void` and a single argument of type `const DataClass&`.

· ***void clearHandlers()***

Remove all handlers from the object/interaction.

· ***void update(const char* p_tag = "")***

Update the object or interaction with the federation.

**HLA::DataClass::Member**

`HLA::DataClass::Member` is the superclass for `HLA::Object::Attribute` and `HLA::Interaction::Parameter`. Any of these methods can be used with either attributes or parameters.

Note that the constructor is not shown, as the user should never instantiate attribute or parameter instances themselves (instead, use `HLA::Object::createAttribute()` or `HLA::Interaction::createParameter()`).

· ***(C++) template <typename T> void setData(const T& p_data)***

· ***(Java) void setString(String p_data)***

Set the data for the attribute/parameter to `p_data` of type `T` (C++). Currently, Java calls must use `setString()` and convert argument `p_data` to `String` (this will be resolved in future releases).

· ***(C++) template <typename T> void setVector(const vector<T>& p_data)***

· ***(Java) void setCustom(CustomTypes.DoubleArray.class)***

Set the data for the attribute/parameter to `p_data` of type `vector<T>` (C++) or `CustomTypes.DoubleArray` which extends `ArrayList<Double>` (Java).

· ***(C++) template <typename T> T getData() const***

· ***(Java) int getInt()***

***byte getByte()***

***long getLong()***

***float getFloat()***

***double getDouble()***

***boolean getBool()***

***String getString()***

Retrieve the data stored in the attribute/parameter as type `T` (C++) or the indicated return type (Java).

· ***(C++) template <typename T> vector<T> getVector() const***

· ***(Java) CustomTypes.DoubleArray getCustom(CustomTypes.DoubleArray.class)***

Retrieve the data stored in the attribute/parameter as type `vector<T>` (C++) or `CustomTypes.DoubleArray` which extends `ArrayList<Double>` (Java).

· ***bool isDataSet()***

Returns true if either `setData<>()` or `setVector<>()` have been successfully called at least once and `false` otherwise.

**HLA::Object**

· ***Attribute* createAttribute(const char* p_attributeName)***

Create an attribute with the name `p_attribute` name (which must exactly match the attribute name defined in the FOM file). A pointer to the newly created attribute is returned, which must be stored by the user.

**HLA::Interaction**

· ***Parameter* createParameter(const char* p_parameterName)***

Create a parameter with the name `p_parameterName` (which must exactly match the parameter name defined in the FOM file). A pointer to the newly created parameter is returned, which must be stored by the user.

# *C-II Federation Development*

In this Appendix, the detailed descriptions of interactive recovery simulation are provided with relevant computational codes for federate development and simulation interactions, based on the previous chapter, Ch. 5.

## [Federation Object Model (FOM)]

```
;; Interactive Recovery Simulation Prototype

(Fed
     (Federation IRSP)
(FedVersion v1.3)
     (Federate "fed" "Public")
     (Spaces)
     (Objects)
     (Interactions
          (Class InteractionRoot BEST_EFFORT RECEIVE
               (Class RTIprivate BEST_EFFORT RECEIVE)
               (Class USGSRequest RELIABLE TIMESTAMP
                    (Sec_Level "Public")
                    (Parameter StartTime)
                    (Parameter EndTime)
                    (Parameter MinMagnitude)
                    (Parameter MinLatitude)
                    (Parameter MaxLatitude)
                    (Parameter MinLongitude)
                    (Parameter MaxLongitude)
               )
               (Class Earthquake RELIABLE TIMESTAMP
                    (Sec_Level "Public")
                    (Parameter EventID)
                    (Parameter Time)
                    (Parameter Longitude)
                    (Parameter Latitude)
                    (Parameter Depth)
                    (Parameter Magnitude)
               )
               (Class GroundMotion RELIABLE TIMESTAMP
                    (Sec_Level "Public")
                    (Parameter ScaleFactor)
                    (Parameter AccelerationFilePath)
               )
               (Class StructuralDisplacement RELIABLE TIMESTAMP
                    (Sec_Level "Public")
                    (Parameter Displacements)
               )
          )
     )
)
```

## [Federation Ambassador Class: Header Files]

## (for the Anylogic Federate)

```
import jhla.*;
import jhla.Federate.*;
import certi.rti.impl.*;
import hla.rti.*;
import hla.rti.jlc.*;

/**
 * FedAmbassador
 */
public class FederationAmbassador extends Federate implements Serializable {

     public class USGSRequestInteraction implements Serializable {

          public Interaction parent;
          public Interaction.Parameter StartTime;
          public Interaction.Parameter EndTime;
          public Interaction.Parameter MinMagnitude;
          public Interaction.Parameter MinLatitude;
          public Interaction.Parameter MaxLatitude;
          public Interaction.Parameter MinLongitude;
          public Interaction.Parameter MaxLongitude;

          private static final long serialVersionUID = 1L;
     }

     public class EarthquakeInteraction implements Serializable {

          public Interaction parent;
          public Interaction.Parameter EventID;
          public Interaction.Parameter Time;
          public Interaction.Parameter Magnitude;
          public Interaction.Parameter Depth;
          public Interaction.Parameter Latitude;
          public Interaction.Parameter Longitude;

          public EarthquakeHandler Handler = new EarthquakeHandler();

          private static final long serialVersionUID = 1L;

          public class EarthquakeHandler implements UpdateHandler, Serializable {

               public void execute(DataClass sender) {
                    owner.OnEarthquakeUpdate(sender);
               }

               private static final long serialVersionUID = 1L;
          }
     }

     public class GroundMotionInteraction implements Serializable {

          public Interaction parent;
          public Interaction.Parameter ScaleFactor;
          public Interaction.Parameter AccelerationFilePath;
```

```
        private static final long serialVersionUID = 1L;
    }

    public class StructuralDisplacementInteraction implements Serializable {

        public Interaction parent;
        public Interaction.Parameter Displacements;

        public StructuralDisplacementHandler Handler =
        new StructuralDisplacementHandler();

        public class StructuralDisplacementHandler implements UpdateHandler,
        Serializable {

            public void execute(DataClass sender) {
                owner.OnStructuralDisplacementUpdate();
            }

            private static final long serialVersionUID = 1L;
        }

        private static final long serialVersionUID = 1L;
    }

    public EarthquakeInteraction Earthquake = new EarthquakeInteraction();
    public USGSRequestInteraction USGSRequest = new USGSRequestInteraction();
    public GroundMotionInteraction GroundMotion = new GroundMotionInteraction();
    public StructuralDisplacementInteraction StructuralDisplacement =
    new StructuralDisplacementInteraction();

    private Damage owner;

  /**
   * Default constructor
   */
  public FederationAmbassador(Damage owningAgent) throws RTIinternalError {
    super(owningAgent.toString());
    owner = owningAgent;

    try
    {
        join("IRSP", "irsp.fed");

        // Construct interaction: USGSRequest
        USGSRequest.parent = createInteraction("USGSRequest", PubSubType.Publish);
        USGSRequest.StartTime = USGSRequest.parent.createParameter("StartTime");
        USGSRequest.EndTime = USGSRequest.parent.createParameter("EndTime");
        USGSRequest.MinMagnitude = USGSRequest.parent.createParameter("MinMagnitude");
        USGSRequest.MinLatitude = USGSRequest.parent.createParameter("MinLatitude");
        USGSRequest.MaxLatitude = USGSRequest.parent.createParameter("MaxLatitude");
        USGSRequest.MinLongitude = USGSRequest.parent.createParameter("MinLongitude");
        USGSRequest.MaxLongitude = USGSRequest.parent.createParameter("MaxLongitude");

        // Construct interaction: Earthquake
        Earthquake.parent = createInteraction("Earthquake", PubSubType.Subscribe);
        Earthquake.EventID = Earthquake.parent.createParameter("EventID");
        Earthquake.Time = Earthquake.parent.createParameter("Time");
        Earthquake.Longitude = Earthquake.parent.createParameter("Longitude");
        Earthquake.Latitude = Earthquake.parent.createParameter("Latitude");
        Earthquake.Depth = Earthquake.parent.createParameter("Depth");
```

```
            Earthquake.Magnitude = Earthquake.parent.createParameter("Magnitude");
            Earthquake.parent.addHandler(Earthquake.Handler);

            // Construct interaction: GroundMotion
            GroundMotion.parent = createInteraction("GroundMotion", PubSubType.Publish);
            GroundMotion.ScaleFactor = GroundMotion.parent.createParameter("ScaleFactor");
            GroundMotion.AccelerationFilePath =
            GroundMotion.parent.createParameter("AccelerationFilePath");

            // Construct interaction: StructuralDisplacement
            StructuralDisplacement.parent = createInteraction("StructuralDisplacement",
            PubSubType.Subscribe);
            StructuralDisplacement.Displacements =
            StructuralDisplacement.parent.createParameter("Displacements");
            StructuralDisplacement.parent.addHandler(StructuralDisplacement.Handler);

            publishSubscribe();
        }
        catch (Exception ex)
        {
            // TODO Add error handling to object/interaction construction
            System.out.println("Error in FederationAmbassador constructor: " +
            ex.getMessage());
            ex.printStackTrace();
        }

    }

        @Override
        public String toString() {
            return super.toString();
        }

        /**
         * This number is here for model snapshot storing purpose<br>
         * It needs to be changed when this class gets changed
         */
        private static final long serialVersionUID = 1L;

}
```

## (for the USGS Federate)

```
ifndef HLA_SEISMIC_SEISMICFED_H
#define HLA_SEISMIC_SEISMICFED_H

#include <string>

#include "Federate.h"
#include "EventQueue.h"

namespace HLA
{
namespace Seismic
{
using std::string;

class SeismicFed
{
```

```
public:
	SeismicFed(Parameters p_eventParameters, time_t p_startTime, const char*
	p_federateName, const char* p_executionName, const char* p_fedFilepath,
	int p_timeoutInSeconds = 5);

	SeismicFed(Parameters p_eventParameters, tm p_startTime, const char* p_federateName,
	const char* p_executionName, const char* p_fedFilepath, int p_timeoutInSeconds = 5);

	SeismicFed(Parameters p_eventParameters, const char* p_federateName, const char*
	p_executionName, const char* p_fedFilepath, int p_timeoutInSeconds = 5);

	~SeismicFed();

	bool sendLatest();

	void handleUSGSRequest(const DataClass& p_sender);

	inline void tick()
	{
		m_fedAmbassador.tick();
	}

	inline void queryServer()
	{
		m_events.fill();
	}

	inline size_t queueCount()
	{
		return m_events.size();
	}


private:
	Federate m_fedAmbassador;
	EventQueue m_events;

	struct
	{
		Interaction* parent;
		Interaction::Parameter* eventID;
		Interaction::Parameter* time;
		Interaction::Parameter* longitude;
		Interaction::Parameter* latitude;
		Interaction::Parameter* depth;
		Interaction::Parameter* magnitude;
	} Earthquake;

	struct
	{
		Interaction* parent;
		Interaction::Parameter* startTime;
		Interaction::Parameter* endTime;
		Interaction::Parameter* minMagnitude;
		Interaction::Parameter* minLatitude;
		Interaction::Parameter* maxLatitude;
		Interaction::Parameter* minLongitude;
		Interaction::Parameter* maxLongitude;
	} USGSRequest;
```

```
    void initAll(const char* p_executionName, const char* p_FEDid,
    int p_timeoutInSeconds);

}; // class SeismicFed


} // namespace Seismic
} // namespace HLA

#endif // HLA_SEISMIC_SEISMICFED_H
```

## (for the OpenSees Federate)

```
#ifndef OPENSEES_FED_H
#define OPENSEES_FED_H

#include "Federate.h"
#include "Model.h"
#include "Analyses.h"

namespace HLA
{
namespace OpenSees
{

class OpenSeesFed : public Federate
{
public:
    OpenSeesFed(const char* p_federateName, const char* p_executionName, const char*
    p_fedFilepath, int p_timeoutInSeconds = 5);

    ~OpenSeesFed() {
        resign();
    }

    void handleGroundMotion(const DataClass& p_sender);

private:
    struct {
        Interaction* parent;
        Interaction::Parameter* displacements;
    } StructuralDisplacement;

    struct {
        Interaction* parent;
        Interaction::Parameter* scaleFactor;
        Interaction::Parameter* accelerationFilePath;
    } GroundMotion;

    Domain* m_domain;
    Model* m_model;

}; // class OpenSeesFed

} // namespace HLA
} // namespace OpenSees

#endif // OPENSEES_FED_H
```

## [Federation Initialization/Update Handlers]

## (for the Anylogic Federate)

```
System.out.println(this.toString());
if (!this.toString().equals("root.damage"))
{
      System.out.println("initHLA");
      try
      {
            FedAmb = new FederationAmbassador(this);

            FedAmb.USGSRequest.MinMagnitude.setString(String.valueOf(Min_Magnitude));
            FedAmb.USGSRequest.MinLatitude.setString(String.valueOf(Min_Latitude));
            FedAmb.USGSRequest.MaxLatitude.setString(String.valueOf(Max_Latitude));
            FedAmb.USGSRequest.MinLongitude.setString(String.valueOf(Min_Longitude));
            FedAmb.USGSRequest.MaxLongitude.setString(String.valueOf(Max_Longitude));
      }
      catch (RTIexception ex)
      {
            System.out.println(ex.getMessage());
            ex.printStackTrace();
      }
```

## (for the USGS Federate)

```
#include "SeismicFed.h"

using namespace HLA::Seismic;


SeismicFed::SeismicFed(Parameters p_eventParameters, time_t p_startTime, const char*
p_federateName, const char* p_executionName, const char* p_fedFilepath,
int p_timeoutInSeconds)
      : m_fedAmbassador(p_federateName), m_events(p_eventParameters, p_startTime)
{
      initAll(p_executionName, p_fedFilepath, p_timeoutInSeconds);
}

SeismicFed::SeismicFed(Parameters p_eventParameters, tm p_startTime, const char*
p_federateName, const char* p_executionName, const char* p_fedFilepath,
int p_timeoutInSeconds)
      : m_fedAmbassador(p_federateName), m_events(p_eventParameters, p_startTime)
{
      initAll(p_executionName, p_fedFilepath, p_timeoutInSeconds);
}

SeismicFed::SeismicFed(Parameters p_eventParameters, const char* p_federateName,
const char* p_executionName, const char* p_fedFilepath, int p_timeoutInSeconds)
      : m_fedAmbassador(p_federateName), m_events(p_eventParameters)
{
      initAll(p_executionName, p_fedFilepath, p_timeoutInSeconds);
}

SeismicFed::~SeismicFed()
{
      m_fedAmbassador.resign();
}
```

```
bool
SeismicFed::sendLatest()
{
	if (m_events.size() > 0)
	{
		Event latestEvent = m_events.front();

		Earthquake.eventID->setData<string>(latestEvent.eventID);
		Earthquake.time->setData<time_t>(latestEvent.time);
		Earthquake.longitude->setData<double>(latestEvent.longitude);
		Earthquake.latitude->setData<double>(latestEvent.latitude);
		Earthquake.depth->setData<double>(latestEvent.depth);
		Earthquake.magnitude->setData<double>(latestEvent.magnitude);

		Earthquake.parent->update();
		m_events.pop();

		return true;
	}
	else
	{
		return false;
	}
}

void
SeismicFed::handleUSGSRequest(const HLA::DataClass& p_sender)
{
	Parameters newParams;

	newParams.startTime.set(USGSRequest.startTime->getData<string>());
	newParams.endTime.set(USGSRequest.endTime->getData<string>());
	newParams.minMagnitude.set(USGSRequest.minMagnitude->getData<double>());
	newParams.minLatitude.set(USGSRequest.minLatitude->getData<double>());
	newParams.maxLatitude.set(USGSRequest.maxLatitude->getData<double>());
	newParams.minLongitude.set(USGSRequest.minLongitude->getData<double>());
	newParams.maxLongitude.set(USGSRequest.maxLongitude->getData<double>());

	Debug("Received USGS request:\n");
	Debug(newParams);

	m_events.resetParameters(newParams);
	m_events.fill();

	//tick();
	while (sendLatest() == true)
	{
		//tick();
	}
}

void
SeismicFed::initAll(const char* p_executionName, const char* p_fedFilpath,
int p_timeoutInSeconds)
{
	m_fedAmbassador.join(p_executionName, p_fedFilpath, p_timeoutInSeconds);

	Earthquake.parent = m_fedAmbassador.createInteraction("Earthquake",
	PubSubType::PUBLISH);
```

```
    Earthquake.eventID = Earthquake.parent->createParameter("EventID");
    Earthquake.time = Earthquake.parent->createParameter("Time");
    Earthquake.longitude = Earthquake.parent->createParameter("Longitude");
    Earthquake.latitude = Earthquake.parent->createParameter("Latitude");
    Earthquake.depth = Earthquake.parent->createParameter("Depth");
    Earthquake.magnitude = Earthquake.parent->createParameter("Magnitude");

    USGSRequest.parent = m_fedAmbassador.createInteraction("USGSRequest",
    PubSubType::SUBSCRIBE);
    USGSRequest.startTime = USGSRequest.parent->createParameter("StartTime");
    USGSRequest.endTime = USGSRequest.parent->createParameter("EndTime");
    USGSRequest.minMagnitude = USGSRequest.parent->createParameter("MinMagnitude");
    USGSRequest.minLatitude = USGSRequest.parent->createParameter("MinLatitude");
    USGSRequest.maxLatitude = USGSRequest.parent->createParameter("MaxLatitude");
    USGSRequest.minLongitude = USGSRequest.parent->createParameter("MinLongitude");
    USGSRequest.maxLongitude = USGSRequest.parent->createParameter("MaxLongitude");

    USGSRequest.parent->addHandler(
        [&] (const DataClass& p_sender) {
            handleUSGSRequest(p_sender);
        } );

    m_fedAmbassador.publishSubscribe();
}
```

## (for the OpenSees Federate)

```
#include "OpenSeesFed.h"

using namespace HLA::OpenSees;

OpenSeesFed::OpenSeesFed(const char* p_federateName, const char* p_executionName,
const char* p_fedFilepath, int p_timeoutInSeconds)
    : Federate(p_federateName)
{
    join(p_executionName, p_fedFilepath, p_timeoutInSeconds);

    GroundMotion.parent = createInteraction("GroundMotion", PubSubType::SUBSCRIBE);
    GroundMotion.scaleFactor = GroundMotion.parent->createParameter("ScaleFactor");
    GroundMotion.accelerationFilePath = GroundMotion.parent->
    createParameter("AccelerationFilePath");
    GroundMotion.parent->addHandler(
        [&] (const DataClass& p_sender) {
            handleGroundMotion(p_sender);
        } );

    StructuralDisplacement.parent = createInteraction("StructuralDisplacement",
    PubSubType::PUBLISH);
    StructuralDisplacement.displacements = StructuralDisplacement.parent->
    createParameter("Displacements");

    publishSubscribe();

    m_domain = new Domain();
    // TO-DO: model specifications via HLA
    m_model = new Model(m_domain,
        4,
        3,
        1968.5,
```

```
        33.5,
        47.7,
        4090,
        5700,
        4724.1,
        27.7,
        47.7,
        2700,
        5700,
        29000,
        0.3);
}

void
OpenSeesFed::handleGroundMotion(const HLA::DataClass& p_sender)
{
    string accelFile = GroundMotion.accelerationFilePath->getData<string>();
    double scaleFactor = GroundMotion.scaleFactor->getData<double>();

    Analyses::gravity(*m_model, 1.0e-8, 10);
    Analyses::groundMotion(*m_model, accelFile, 1, 1.885191592563307E-52, 0.01);

    const Vector* response;
    double responseValue;
    vector<double> nodeResponses;

    Debug("Node responses:" << endl);

    int vRes = m_model->getNumStories() + 1;
    int hRes = m_model->getNumBays() + 1;
    for (int iPier = 1; iPier <= hRes; ++iPier)
    {
        for (int iLevel = 2; iLevel <= vRes; ++iLevel)
        {
            int nodeID = iLevel * 10 + iPier;
            response = m_domain->getNodeResponse(nodeID, NodeResponseType::Disp);
            responseValue = (*response)(0) * -1.27257498306;
            nodeResponses.push_back(responseValue);
            Debug("Level " << iLevel << ", pier " << iPier << ":\t" <<
            responseValue << endl);
        }
    }

    StructuralDisplacement.displacements->setVector<double>(nodeResponses);
    StructuralDisplacement.parent->update();

}
```

## [Simulation Executions]

## (Driver Project: "Main" Functions)

```
#include "Federate.h"

#include <iostream>
#include <vector>

using namespace std;

void receiptHandle(const HLA::DataClass& p_data);
void receiveTest();


void main(int argc, char** argv)
{
     struct
     {
          HLA::Interaction* parent;
          HLA::Interaction::Parameter* eventID;
          HLA::Interaction::Parameter* time;
          HLA::Interaction::Parameter* longitude;
          HLA::Interaction::Parameter* latitude;
          HLA::Interaction::Parameter* depth;
          HLA::Interaction::Parameter* magnitude;
     } EarthquakePub;

     struct
     {
          HLA::Interaction* parent;
          HLA::Interaction::Parameter* eventID;
          HLA::Interaction::Parameter* time;
          HLA::Interaction::Parameter* longitude;
          HLA::Interaction::Parameter* latitude;
          HLA::Interaction::Parameter* depth;
          HLA::Interaction::Parameter* magnitude;
     } EarthquakeSub;


     HLA::Federate pub("pub");
     HLA::Federate sub("sub");

     pub.join("IRSP", "irsp.fed");
     sub.join("IRSP", "irsp.fed");

     EarthquakePub.parent = pub.createInteraction("Earthquake",
     HLA::PubSubType::PUBLISH);
     EarthquakePub.eventID = EarthquakePub.parent->createParameter("EventID");
     EarthquakePub.time = EarthquakePub.parent->createParameter("Time");
     EarthquakePub.longitude = EarthquakePub.parent->createParameter("Longitude");
     EarthquakePub.latitude = EarthquakePub.parent->createParameter("Latitude");
     EarthquakePub.depth = EarthquakePub.parent->createParameter("Depth");
     EarthquakePub.magnitude = EarthquakePub.parent->createParameter("Magnitude");

     EarthquakeSub.parent = sub.createInteraction("Earthquake",
     HLA::PubSubType::SUBSCRIBE);
     EarthquakeSub.eventID = EarthquakeSub.parent->createParameter("EventID");
     EarthquakeSub.time = EarthquakeSub.parent->createParameter("Time");
```

```
	EarthquakeSub.longitude = EarthquakeSub.parent->createParameter("Longitude");
	EarthquakeSub.latitude = EarthquakeSub.parent->createParameter("Latitude");
	EarthquakeSub.depth = EarthquakeSub.parent->createParameter("Depth");
	EarthquakeSub.magnitude = EarthquakeSub.parent->createParameter("Magnitude");

	pub.publishSubscribe();
	sub.publishSubscribe();

	vector<double> vecPub;
	vecPub.push_back(1.2);
	vecPub.push_back(34.5);
	vecPub.push_back(67.8);

	EarthquakePub.eventID->setVector(vecPub);
	EarthquakePub.time->setData(10);
	EarthquakePub.longitude->setData(10);
	EarthquakePub.latitude->setData(10);
	EarthquakePub.depth->setData(10);
	EarthquakePub.magnitude->setData(10);

	pub.updateAll();
	sub.tick();

	vector<double> vecSub = EarthquakeSub.eventID->getVector<double>();

	cout << "Published:\n";
	for (double entry : vecPub)
	{
		cout << entry << endl;
	}

	cout << "\nSubscribed:\n";
	for (double entry : vecSub)
	{
		cout << entry << endl;
	}

	system("pause");
}

void receiptHandle(const HLA::DataClass& p_data)
{
	cout << "Received \"" << p_data.getName() << "\"." << endl;

	for (auto entry : p_data)
	{
		cout << entry.getName() << ": " << entry.getData<string>() << endl;
	}
	cout << endl;
}

void receiveTest()
{
	struct
	{
		HLA::Interaction* parent;
		HLA::Interaction::Parameter* eventID;
		HLA::Interaction::Parameter* time;
		HLA::Interaction::Parameter* longitude;
		HLA::Interaction::Parameter* latitude;
```

```
		HLA::Interaction::Parameter* depth;
		HLA::Interaction::Parameter* magnitude;
	} Earthquake;

	HLA::Federate rec("rec");

	rec.join("IRSP", "irsp.fed");

	Earthquake.parent = rec.createInteraction("Earthquake", HLA::PubSubType::SUBSCRIBE);
	Earthquake.eventID = Earthquake.parent->createParameter("EventID");
	Earthquake.time = Earthquake.parent->createParameter("Time");
	Earthquake.longitude = Earthquake.parent->createParameter("Longitude");
	Earthquake.latitude = Earthquake.parent->createParameter("Latitude");
	Earthquake.depth = Earthquake.parent->createParameter("Depth");
	Earthquake.magnitude = Earthquake.parent->createParameter("Magnitude");

	rec.publishSubscribe();

	Earthquake.parent->addHandler(receiptHandle);

	while (true)
	{
		rec.tick();
	}
}
```

## (USGS: “Main” Functions)

```
#include "EventQueue.h"
#include "Parameters.h"
#include "SeismicFed.h"
#include "Federate.h"

#include <iostream>
#include <exception>
#include <Windows.h>

using namespace std;
using namespace HLA::Seismic;


void fedTest();
void receiptHandle(const HLA::DataClass& p_data);
void receiveTest();


void main(int argc, char** argv)
{
	fedTest();

	system("pause");
}

void fedTest()
{
	Parameters params;
	time_t curTime = time(nullptr);
	curTime -= 100000;
```

```
	SeismicFed seismic(params, curTime, "Seismic", "IRSP", "irsp.fed");

	while (true)
	{
		seismic.tick();
		Wait(500);
	}
}
```

## (USGS: "EventQueue" Header File)

```
#ifndef HLA_SEISMIC_EVENTQUEUE_H
#define HLA_SEISMIC_EVENTQUEUE_H

#include <time.h>
#include <string>
#include <set>

#include <cpprest/http_client.h>
#include <cpprest/json.h>

#include "Parameters.h"
#include "Event.h"
#include "Preprocessor.h"

namespace HLA
{
namespace Seismic
{
using namespace web;
using namespace web::http;
using namespace web::http::client;
using std::string;
using std::ostream;
using std::set;
using std::endl;


class EventQueue
{
public:
	EventQueue(Parameters p_queryParameters)
		: m_parameters(p_queryParameters), m_startTime(time(nullptr)) { }

	EventQueue(Parameters p_queryParameters, time_t p_initialStartTime)
		: m_parameters(p_queryParameters) , m_startTime(p_initialStartTime) { }

	EventQueue(Parameters p_queryParameters, tm p_initialStartTime)
		: m_parameters(p_queryParameters), m_startTime(mktime(&p_initialStartTime)) { }

	void fill();

	inline size_t size() const { return m_data.size(); }

	inline Event front() const { return (m_data.size() > 0 ? *m_data.begin() : *
	(new Event)); }

	inline void pop()
	{
```

```
		if (m_data.size() > 0)
			m_data.erase(m_data.cbegin());
	}

	inline void clear()
	{
		m_data.clear();
	}

	inline void resetParameters(Parameters p_queryParameters) { m_parameters =
	p_queryParameters; }

private:
	uri buildURI() const;

	pplx::task<void> sendGeoJSONQuery(const uri& p_completeURI);

	time_t m_startTime;
	Parameters m_parameters;
	set<Event> m_data;

public:
	typedef set<Event>::iterator iterator;
	typedef set<Event>::const_iterator const_iterator;
	iterator begin() { return m_data.begin(); }
	const_iterator begin() const { return m_data.begin(); }
	const_iterator cbegin() const { return m_data.cbegin(); }
	iterator end() { return m_data.end(); }
	const_iterator end() const { return m_data.end(); }
	const_iterator cend() const { return m_data.cend(); }

}; // class EventQueue


} // namespace Seismic
} // namespace HLA

#endif // HLA_SEISMIC_EVENTQUEUE_H
```

## (USGS: “EventQueue” Functions)

```
#include "EventQueue.h"

using namespace HLA::Seismic;


void
EventQueue::fill()
{
	uri queryURI = buildURI();

	sendGeoJSONQuery(queryURI).wait();

	if (m_data.size() > 0)
	{
		m_startTime = m_data.rbegin()->time + 1;
	}
}
```

```
uri
EventQueue::buildURI() const
{
	uri_builder newURI(URI_USGS_BASE);

	if (m_parameters.startTime.isSet())
	{
		newURI.append_query(
			m_parameters.startTime.method(),
			utility::conversions::to_utf16string(m_parameters.startTime.value())
			);
	}
	else
	{
		char timeBuffer [30];
		tm tmTime;

		gmtime_s(&tmTime, &m_startTime);
		strftime(timeBuffer, 30, "%Y-%m-%dT%H:%M:%S", &tmTime);

		newURI.append_query(L"starttime", timeBuffer);
	}

	if (m_parameters.endTime.isSet())
	{
		newURI.append_query(
			m_parameters.endTime.method(),
			utility::conversions::to_utf16string(m_parameters.endTime.value())
			);
	}

	if (m_parameters.minLongitude.isSet())
	{
		newURI.append_query(
			m_parameters.minLongitude.method(),
			m_parameters.minLongitude.value()
			);
	}

	if (m_parameters.maxLongitude.isSet())
	{
		newURI.append_query(
			m_parameters.maxLongitude.method(),
			m_parameters.maxLongitude.value()
			);
	}

	if (m_parameters.minLatitude.isSet())
	{
		newURI.append_query(
			m_parameters.minLatitude.method(),
			m_parameters.minLatitude.value()
			);
	}

	if (m_parameters.maxLatitude.isSet())
	{
		newURI.append_query(
			m_parameters.maxLatitude.method(),
			m_parameters.maxLatitude.value()
```

```
            );
    }

    if (m_parameters.minMagnitude.isSet())
    {
        newURI.append_query(
            m_parameters.minMagnitude.method(),
            m_parameters.minMagnitude.value()
            );
    }

    newURI.append_query(L"format", L"geojson");

    Debug("\nBuilt new URI:\n");
    Debug(utility::conversions::to_utf8string(newURI.to_string()) << endl);
    return newURI.to_uri();
}

pplx::task<void>
EventQueue::sendGeoJSONQuery(const uri& p_completeURI)
{
    http_client client(p_completeURI);
    http_request request(methods::GET);
    request.headers().set_content_type(U("application/json"));

    return client
        .request(request)
        .then([this](http_response response) -> pplx::task<json::value>
    {
        if (response.status_code() == status_codes::OK)
        {
            return response.extract_json();
        }

        return pplx::task_from_result(json::value());
    })
        .then([this](pplx::task<json::value> previousTask)
    {
        const json::value& v = previousTask.get();
        const json::array& features = v.as_object().at(L"features").as_array();

        for (auto entry : features)
        {
            Event newData;
            const json::value& properties = entry.at(L"properties");
            const json::value& coordinates = entry.at(L"geometry").at(L"coordinates");

            newData.eventID =
            utility::conversions::to_utf8string(entry.at(L"id").as_string());
            newData.time = properties.at(L"time").as_number().to_uint64() / 1000;
            newData.magnitude = properties.at(L"mag").as_double();
            newData.longitude = coordinates.at(0).as_double();
            newData.latitude = coordinates.at(1).as_double();
            newData.depth = (int)(coordinates.at(2).as_double() * 1000);

            m_data.insert(newData);
            Debug("\nAdded event to queue:\n" << newData);
        }
    });
}
```

## (OpenSees: "Main" Functions)

```
#include <iostream>

#include "OpenSeesFed.h"

#include "Model.h"
#include "Analyses.h"

using std::cout;
using std::endl;

using namespace HLA::OpenSees;

int main(int argc, char** argv)
{
     OpenSeesFed osFed("OpenSees", "IRSP", "irsp.fed");

     while (true)
     {
          osFed.tick();
          Wait(500);
     }
     system("pause");
}
```

## (OpenSees: "Model" Using a .tcl File)

```
# ------------------------------------------------------------------------------------------
# 2D frame: Silvia Mazzoni & Frank McKenna, 2006 (Updated 20140225)
# nonlinearBeamColumn element, elastic section
#

# SET UP ------------------------------------------------------------------------------------
wipe;                          # clear opensees model
model basic -ndm 2 -ndf 3;               # 2 dimensions, 3 dof per node
file mkdir Data;                         # create data directory
set dataDir OpenSeesData;           # set up name of data directory
set GMdir GMfiles;                  # ground-motion file directory
source $dataDir/LibUnits.tcl;            # define units
source $dataDir/DisplayPlane.tcl;        # procedure for displaying a plane in model
source $dataDir/DisplayModel2D.tcl;      #  procedure  for  displaying  2D  perspectives  of
model

# define GEOMETRY -------------------------------------------------------------------------
# define structure-geometry paramters
set LCol [expr 6700*$cm];      # column height (parallel to Y axis)
set LBeam [expr 7000*$cm];     # beam length (parallel to X axis)
set NStory 4;                  # number of stories above ground level
set NBay 3;                    # number of bays (max 9)

# define NODAL COORDINATES ------------------------------------------------------------------
--
for {set level 1} {$level <=[expr $NStory+1]} {incr level 1} {
     set Y [expr ($level-1)*$LCol];
     for {set pier 1} {$pier <= [expr $NBay+1]} {incr pier 1} {
          set X [expr ($pier-1)*$LBeam];
          set nodeID [expr $level*10+$pier]
          node $nodeID $X $Y;       # actually define node
```

```
     }
}

# Constraints -- Boundary Conditions -----------------------------------------------------
fix 11 1 1 1;            # node DX DY RZ
fix 12 1 1 1;            # node DX DY RZ
fix 13 1 1 1;            # node DX DY RZ
fix 14 1 1 1;            # node DX DY RZ

# calculated MODEL PARAMETERS, particular to this model
puts "Number of Stories: $NStory Number of bays: $NBay"

# Structural-Steel W-section properties -------------------------------------------------
# material properties:
set Es [expr 29000];     # Steel Young's Modulus
set nu 0.3;
set Gs [expr $Es/2./[expr 1+$nu]];  # Torsional stiffness Modulus

# column sections: W27x114
set AgCol 3600;     # cross-sectional area
set IzCol 1080000;       # moment of Inertia
# beam sections: W24x94
set AgBeam 5760;         # cross-sectional area
set IzBeam 4423680;      # moment of Inertia

set ColSecTag 1
set BeamSecTag 2
section Elastic $ColSecTag $Es $AgCol $IzCol
section Elastic $BeamSecTag $Es $AgBeam $IzBeam

# define ELEMENTS --------------------------------------------------------------------------
# set up geometric transformations of element
#   separate columns and beams, in case of P-Delta analysis for columns
set IDColTransf 1; # all columns
set IDBeamTransf 2; # all beams
set ColTransfType Linear;          # options, Linear PDelta Corotational
geomTransf $ColTransfType $IDColTransf;  # only columns can have PDelta effects (gravity
effects)
geomTransf Linear $IDBeamTransf;

# Define Beam-Column Elements ------------------------------------------------------------
set np 5; # number of Gauss integration points for nonlinear curvature distribution--
np=2 for linear distribution ok
# columns
set N0col 1000; # column element numbers
set level 0;
for {set level 1} {$level <=$NStory} {incr level 1} {
     for {set pier 1} {$pier <= [expr $NBay+1]} {incr pier 1} {
          set elemID [expr $N0col + $level*10 +$pier]
          set nodeI [expr  $level*10 + $pier]
          set nodeJ  [expr  ($level+1)*10 + $pier]
          element nonlinearBeamColumn $elemID $nodeI $nodeJ $np $ColSecTag $IDColTransf;
     }
}

# beams
set N0beam 2000;     # beam element numbers
for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     for {set bay 1} {$bay <= $NBay} {incr bay 1} {
          set elemID [expr $N0beam + $level*10 +$bay]
```

```
          set nodeI [expr  $level*10 + $bay]
          set nodeJ  [expr  $level*10 + $bay+1]
          element   nonlinearBeamColumn   $elemID   $nodeI   $nodeJ   $np   $BeamSecTag
$IDBeamTransf;  # beams
     }
}

# pier 1 free node
for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+1]
     recorder Node -file $dataDir/displacement$nodeID.out -node $nodeID -dof 1 disp;
     }
# pier 2 node
for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+2]
     recorder Node -file $dataDir/displacement$nodeID.out -node $nodeID -dof 1 disp;
     }
# pier 3 node
for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+3]
     recorder Node -file $dataDir/displacement$nodeID.out -node $nodeID -dof 1 disp;
     }
# pier 4 free node
for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+4]
     recorder Node -file $dataDir/displacement$nodeID.out -node $nodeID -dof 1 disp;
     }

# Define GRAVITY LOADS, weight and masses
# calculate dead load of frame, assume this to be an internal frame (do LL in a similar
manner)
# calculate distributed weight along the beam length
set GammaConcrete [expr 150*$pcf];            # Reinforced-Concrete floor slabs
set Tslab [expr 6*$in];             # 6-inch slab
set Lslab [expr 2*$LBeam/2];             # assume slab extends a distance of $LBeam1/2
in/out of plane
set Qslab [expr $GammaConcrete*$Tslab*$Lslab];
set QBeam [expr 94*$lbf/$ft];       # W-section weight per length
set QdlBeam [expr $Qslab + $QBeam]; # deav d load distributed along beam.
set QdlCol [expr 114*$lbf/$ft];     # W-section weight per length
set WeightCol [expr $QdlCol*$LCol];           # total Column weight
set WeightBeam [expr $QdlBeam*$LBeam];   # total Beam weight


# assign masses to the nodes that the columns are connected to
# each connection takes the mass of 1/2 of each element framing into it (mass=weight/$g)
mass 21 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
# level 2
mass 22 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
mass 23 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
mass 24 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
mass 31 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
# level 3
mass 32 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
mass 33 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
mass 34 [expr ($WeightCol/2 + $WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
mass 41 [expr ($WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
# level 4
mass 42 [expr ($WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
mass 43 [expr ($WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
```

```
mass 44 [expr ($WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
mass 51 [expr ($WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
# level 5
mass 52 [expr ($WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
mass 53 [expr ($WeightCol/2 +$WeightBeam/2 +$WeightBeam/2)/$g] 0. 0.;
mass 54 [expr ($WeightCol/2 +$WeightBeam/2)/$g] 0. 0.;
# calculate total Floor Mass
set WeightFloor2 [expr $WeightCol*4/2+$WeightCol*4/2+3*$WeightBeam];
# level 2 weight
set WeightFloor3 [expr $WeightCol*4/2+$WeightCol*4/2+3*$WeightBeam];
set WeightFloor4 [expr $WeightCol*4/2+3*$WeightBeam];
set WeightFloor5 [expr $WeightCol*4/2+3*$WeightBeam];
set WeightTotal [expr $WeightFloor2 + $WeightFloor3 + $WeightFloor4 + $WeightFloor5];
# total frame weight
set MassFloor2 [expr $WeightFloor2/$g];
set MassFloor3 [expr $WeightFloor3/$g];
set MassFloor4 [expr $WeightFloor4/$g];
set MassFloor5 [expr $WeightFloor5/$g];
set MassTotal [expr $MassFloor2+$MassFloor3+$MassFloor4+$MassFloor5];                #
total frame mass

# define GRAVITY -----------------------------------------------------------------------
#pattern Plain 1 Linear {
#   load 51 0. -2000. 0.;           # node#, FX FY MZ --  superstructure-weight
#}
# DYNAMIC ground-motion analysis --------------------------------------------------------
# create load pattern

source $dataDir/AnylogicToOpenSees.tcl;   #GMfilename, ScaleFactor

set GMdirection 1
set inFile $GMdir/$GMfilename
set dt 0.01

puts "GM: $inFile"

set accelSeries "Series -dt $dt -filePath $inFile -factor [expr $SF*$g]"; #[expr $SF/$g]";
#[expr $SF/981*$g]"; # define acceleration vector from file (dt=0.01 is associated with
the input file gm)
pattern UniformExcitation 2 $GMdirection -accel $accelSeries;      # define where and how
(pattern tag, dof) acceleration is applied
#rayleigh 0. 0. 0. [expr 2*0.02/pow([eigen 1],0.5)];     # set damping based on first
eigen mode

# create the analysis
wipeAnalysis;                      # clear previously-define analysis parameters
constraints Plain;                      # how it handles boundary conditions
numberer Plain;                    # renumber dof's to minimize band-width
(optimization), if you want to
system BandGeneral;                     # how to store and solve the system of equations
in the analysis
test NormDispIncr 1.0e-8 10;                 # determine if convergence has been
achieved at the end of an iteration step
algorithm Newton;                       # use Newton's solution algorithm: updates
tangent stiffness at every iteration
integrator Newmark 0.5 0.25 ;           # determine the next time step for an analysis
analysis Transient;                     # define type of analysis: time-dependent
analyze 100 1;                     # apply 1000 0.02-sec time steps in analysis

# write the output file [In one File] ----------------------------------------------------
```

```
set outputfile "OpenSeesToAnylogic_disp.txt"
set Displacements [open $dataDir/$outputfile "w"]

# pier 1 free node -- caculated by unit (m) rather than (in.)
     for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+1]
     set disp [nodeDisp $nodeID 1]
     puts $Displacements [expr $disp*$cm/100]
     }

# pier 2 free node
     for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+2]
     set disp [nodeDisp $nodeID 1]
     puts $Displacements [expr $disp*$cm/100]
     }

# pier 3 free node
     for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+3]
     set disp [nodeDisp $nodeID 1]
     puts $Displacements [expr $disp*$cm/100]
     }

# pier 4 free node
     for {set level 2} {$level <=[expr $NStory+1]} {incr level 1} {
     set nodeID [expr $level*10+4]
     set disp [nodeDisp $nodeID 1]
     puts $Displacements [expr $disp*$cm/100]
     }

close $Displacements


puts "Ground Motion Done. End Time: [getTime]"
```

# 國 文 抄 錄

# 多角的인 施設物 災難復舊管理를 위한 上位體系構造(HLA) 基盤의 相互連動 시뮬레이션 構築과 活用

최근 시뮬레이션 기술은 재난대응과 복구활동을 효율적으로 지원할 수 있어 시설물 재난관리에 널리 활용되고 있다. 그러나 보다 효율적이고 복합적인 재난관리를 위해서는 재난 발생에 따른 다양한 피해 형태를 고려해야 하며, 수많은 재난대응 또는 복구활동에 대한 다각적인 분석이 필요하다. 이러한 상황에서 기 개발된 단일 시스템 혹은 시뮬레이션 기술만으로는 다양한 재난 상황에 대한 복합적인 정보처리의 한계가 있다. 특히, 시설물 복구단계에서는 전 지역차원의 복구정책 수립과 개별 시설물차원의 복구계획 수립이 동시에 요구되는데, 각 계획수립의 주체가 상이할 뿐 아니라 분석 요구사항과 범위 또한 다르기 때문에 서로 다른 시뮬레이션 조합이 가능해야 한다.

이에 본 연구는 미 국방성 (Department of Defense: DOD)에서 제시한 개념인 상위체계구조(HLA) 기반의 분산형 시뮬레이션 아키텍처(IEEE 1516)를 활용하여, 다양한 재난유형, 피해형태, 대응 또는 복구활동에 대한 확장성을 가진 상호연동 시뮬레이션 프레임워크를 구축한다. 이를 바탕으로, 지진 발생 후의 시설물 구조체 피해와 복구상황에 초점을 맞추어 지역차원 또는 개별 시설물차원의 복구계획 수립을 각각 지원할 수 있는 상호연동형 복구관리 시뮬레이션 프로토타입을 개발한다. 본 프로토타입에서는 USGS (미 지질 조사국)에서 제공하는 지진정보 수집기술과 OpenSees 구조체거동 시뮬레이션 기술이 지역전반의 피해와 개별 시설물의 구조체 피해 분석에 활용되며,

시스템 다이내믹스 (SD) 및 이산사건 시뮬레이션 (DES) 모델링 방법을 활용하여 구축되는 지역/시설물 복구 시뮬레이션과 연동된다. 본 상호연동 시뮬레이션은 분석 목적에 따른 시뮬레이션 조합을 통해 다각적인 복구계획 수립에 필요한 서로 다른 분석 요구사항에 대응할 수 있게 한다. 예를 들어, 지역 전반의 복구계획 수립을 뒷받침하기 위한 USGS 지진정보 수집기술과 SD 모델의 상호연동은 지진정보에 따른 전 지역의 피해 산정을 지원하고, 이에 따른 다양한 복구활동에 대한 포괄적인 이해를 제공할 수 있다. 반면, 개별 시설물의 복구계획 수립을 위해서는 상세한 구조체 피해정보와 세부 복구공정 분석이 요구되는데, OpenSees 구조체거동 시뮬레이션과 DES 모델의 상호연동을 통해 이러한 분석 요구사항을 만족시킬 수 있다.

이처럼, 지역과 시설물차원의 복구활동에 초점을 맞춰 개발된 상호연동형 재난관리 시뮬레이션은 분석 목적에 따라 시뮬레이션 조합을 다양화하고 재사용성과 확장성을 제공함으로써 재난관리 분야 전반에 활용될 수 있다. 이를 확인하기 위해, 개발된 프레임워크가 다른 시뮬레이션 기술과 상호연동 하는 경우 가능한 분석의 확장 시나리오(예: 화재 발생 후 피난 시뮬레이션)를 제시하였다. 본 연구결과는 시뮬레이션 모듈 간 상호운용성을 도모하는 재난관리의 기반 기술과 프레임워크를 통해, 향후 태풍과 같은 다양한 재난 유형, 정전 또는 화재와 같은 다양한 피해 형태, 피난 또는 인명구조 등의 다양한 재난대응과 복구활동 관리분야로 확장될 수 있을 것으로 기대된다.

**주요어:** 분산형 시뮬레이션; 상위체계구조 (High Level Architecture); 시설관리; 시설물 복구; 재난대응

**학 번:** 2010-30174