



저작자표시-비영리-동일조건변경허락 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.
- 이차적 저작물을 작성할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



동일조건변경허락. 귀하가 이 저작물을 개작, 변형 또는 가공했을 경우에는, 이 저작물과 동일한 이용허락조건하에서만 배포할 수 있습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사 학위논문

**Ontological Modeling for Process and
Reliability Simulation of LNG FPSO
Liquefaction Cycle based on the DEVS
formalism considering Ship Motion Effect**

선체 운동을 고려한 LNG FPSO 액화 사이클의
온톨로지 모델링 및 DEVS 형식론 기반 공정 및
신뢰도 시뮬레이션

2013년 2월

서울대학교 대학원
조선해양공학과
하 솔

Contents

Abstract.....	1
1. Introduction	4
1.1. LNG FPSO Liquefaction Cycle	4
1.1.1. LNG FPSO Concept.....	4
1.1.2. LNG FPSO Liquefaction Process System	6
1.1.3. Design and Evaluation of the LNG FPSO Liquefaction Cycle.....	7
1.2. Motivation	12
1.3. Scope of This Thesis.....	14
1.4. Related Studies	19
1.4.1. Process Simulation to Determine the Optimal Operating Conditions of the Liquefaction Cycles.....	19
1.4.2. Reliability Analysis of the Liquefaction Cycles	20
1.4.3. Effect of Ship Motion on the Performance of the Heat Exchanger	22
2. Automatic Generation of Logical Models of Liquefaction Cycles Based on Ontological Modeling....	24
2.1. Conventional Liquefaction Cycles Used in Onshore Plants	25
2.1.1. DMR Cycle.....	26
2.1.2. C3MR Cycle	27
2.1.3. Cascade Cycle.....	28
2.1.4. Nitrogen Expander Cycles	29
2.1.5. Niche cycle.....	31
2.1.6. PRICO (Poly-Refrigerant Integrated Cycle Operations) Cycle.....	32
2.2. General Rules for Combining Liquefaction Cycle Equipment.....	34
2.3. Multiple Cooling Cycles.....	35

2.3.1. Basic Stage and Additional Stages	37
2.3.2. Interconnection of Liquefaction Cycle Equipment.....	37
2.4. Ontological Modeling of the Liquefaction Cycle using the SES.....	41
2.4.1. Logical Model of Liquefaction Cycles: Their Structural Aspect	41
2.4.2. SES: An Ontological Framework	47
2.4.3. Procedure for Making the SES for the Liquefaction Cycle.....	57
2.4.4. SES for the Liquefaction Cycle	63
2.4.5. Automatic Generation of the Logical Models for the Liquefaction Cycle by Pruning the SES	73

3. Process Simulation to Determine the Optimal Operating Conditions of the LNG FPSO Liquefaction Cycles Using a Sequential Modular Simulator Based on the DEVS Formalism 83

3.1. Procedure for Determining the Optimal Operating Conditions of the Liquefaction Cycle.....	85
3.2. Process Simulator for Calculating the Operating Conditions of the Liquefaction Cycle.....	89
3.2.1. Process Simulator.....	89
3.2.2. Calculation of the Operating Conditions Using the Sequential Modular Simulator	94
3.2.3. Modeling of Equipment for Sequential Modular Simulator	98
3.2.4. Making of a Process Simulator Using the SES and MB.....	114
3.3. Optimization Methods for Determining the Optimal Operating Conditions.....	118
3.4. Verification of Sequential Modular Simulator by Applying to Single Cooling Cycle (Refrigerator)	119
3.5. Calculation Results of the Optimal Operating Conditions for the DMR Cycle.....	121

4. Reliability Simulation of LNG FPSO Liquefaction Cycles Based on the DEVS Formalism..... 127

4.1. Reliability Analysis Method	128
4.1.1. Fault Tree Analysis.....	128
4.1.2. Reliability Block Diagram	143
4.2. Reliability Simulation Based on the DEVS Formalism	146
4.2.1. Procedure of Reliability Simulation.....	146
4.2.2. Physical Model for Reliability Simulation Based on the DEVS Formalism	149
4.3. Verification of the Reliability Analysis Method Proposed in This Thesis.....	153
4.3.1. Firewater Pumping System.....	153
4.3.2. Power Switching System.....	155
4.3.3. Hypothetical Computer System	158
4.4. Reliability Analysis of the Conventional Liquefaction Cycles	161
4.4.1. Variation of the System Entity Structure for Reliability Analysis.....	161
4.4.2. Reliability Analysis of the DMR Cycle.....	163
4.4.3. Reliability Analysis Results of Other Conventional Liquefaction Cycles	170
5. Effect of Ship Motion in the Heat Exchanger as Determined Using the Lattice Boltzmann Method	173
5.1. Modeling of Cryogenic Heat Exchangers.....	173
5.1.1. Cryogenic Heat Exchangers	173
5.1.2. Simplified Model of the Spiral Wound Heat Exchanger	175
5.2. Lattice Boltzmann Method	178
5.2.1. Concept of the LBM	178
5.2.2. Algorithms to Calculate Using the LBM	180
5.3. Analysis of the Motion Effect on the Heat Exchanger Using LBM ...	193
6. Conclusions.....	194
References	196

APPENDICES201

A. Simulation Kernel based on Discrete Event System Specification (DEVS) and Discrete Time System Specification (DTSS) Formalism202

A.1. DEVS and DTSS Formalism.....	202
A.1.1. DEVS Formalism	203
A.1.2. DTSS Formalism.....	206
A.2. Simulation based on the DEVS Formalism	207
A.2.1. Definition of Simulation Model.....	207
A.2.2. Procedure of Simulation Based on DEVS and DTSS Formalism	209
A.3. Simulation based on the DEVS and DTSS Formalism.....	213
A.4. Structure of the Simulation Kernel	221
A.4.1. Variables Used on Simulation Kernel	221
A.4.2. Simulation Model	224
A.4.3. Simulation Engine.....	225

B. Mathematical Model of Equipment Based on Thermodynamics226

B.1. Compressor.....	226
B.2. Condenser	233
B.3. Expansion Valve	237
B.4. Heat Exchanger.....	240
B.5. Tee	243
B.6. Phase Seperator.....	246
B.7. Common Header.....	249
B.8. Calculation of the Specific Enthalpy for a Pure Substance	251
B.9. Calculation of the Specific Entropy for a Pure Substance	254

C. Methods of Solving Nonlinear Equations	257
D. Comparison of Calculation Results for One-Dimensional Heat Equation Using the Lattice Boltzmann Method and Finite Difference Method.....	260
D.1. Solving the Equation Using the Explicit Method of the FDM.....	261
D.2. Solving the Equation Using the Implicit Method of the FDM	265
D.3. Solving the Equation Using Crank-Nicolson Method of the FDM...	270
D.4. Solving the Equation Using LBM.....	276
D.5. Comparison of the Calculation Results Using the FDM and the LBM	283
국문 초록.....	284
후기.....	287

Figures

Figure 1-1 LNG production in an onshore facility.....	4
Figure 1-2 LNG production via LNG FPSO.....	5
Figure 1-3 LNG FPSO configuration.....	6
Figure 1-4 Typical process component parts used in the process flowsheet (Edgar, 2001) 7	
Figure 1-5 Sample liquefaction cycle: a refrigerator	8
Figure 1-6 Another combination of liquefaction cycles for a refrigerator: addition of equipment to enhance the efficiency of the liquefaction cycle	9
Figure 1-7 Generation of the logical models of the liquefaction cycle based on the rules for combining equipment	14
Figure 1-8 Scope of this thesis	15
Figure 1-9 Configuration of the program that was developed in this thesis	17
Figure 1-10 Experiment for measuring the effect of tilting and movements on the performance of the heat exchanger: (a) test section of the heat exchanger and (b) liquid collection system below the test section (Lex et al., 2007)	23
Figure 2-1 Representation of the logical model of a liquefaction cycle using the SES	24
Figure 2-2 Configuration of the DMR (Dual Mixed-Refrigerant) cycle.....	27
Figure 2-3 Configuration of the C3MR cycle.....	28
Figure 2-4 Configuration of the cascade cycle.....	29
Figure 2-5 Configuration of the nitrogen expander cycle: N ₂ single expander cycle	30
Figure 2-6 Configuration of the nitrogen expander cycle: N ₂ dual expander cycle	31
Figure 2-7 Configuration of the Niche cycle	32
Figure 2-8 Configuration of the PRICO cycle	33
Figure 2-9 Sample rule for combining liquefaction cycle equipment: the compressor and the condenser.....	34
Figure 2-10 Sample rule for combining liquefaction cycle equipment: the valve (expander) and the heat exchanger.....	34
Figure 2-11 High-level representation of the number of cycles in various liquefaction processes (Barclay and Shukri, 2000)	36
Figure 2-12 Regeneration: (a) basic refrigerator and (b) refrigerator with regeneration ..	40
Figure 2-13 Hierarchical representation of the liquefaction cycle configuration using the tree structure.....	42

Figure 2-14 Variation of the logical model for the liquefaction cycle: adding new equipment and rules	43
Figure 2-15 Variation of the logical model for the liquefaction cycle: equipment change	44
Figure 2-16 Variation of the logical model of the liquefaction cycles: making the PRICO cycle from the refrigerator.....	46
Figure 2-17 Representation of a human using the SES.....	48
Figure 2-18 Pruning of the SES that represents a human	50
Figure 2-19 PES after finishing the pruning process	51
Figure 2-20 SES and MB concepts	52
Figure 2-21 DEVS and DTSS formalisms: configuration of an atomic model.....	53
Figure 2-22 Configuration of the coupled model based on the DEVS and DTSS formalisms.....	54
Figure 2-23 Synthesis of the PES and the MB.....	56
Figure 2-24 Logical model of the typical liquefaction cycle: the dual cooling cycle (precooling: three stages of compression refrigeration, and main cooling: three stages of compression and three stages of refrigeration)	57
Figure 2-25 Procedure for making the SES for the liquefaction cycle: rules for multiple cooling cycles.....	58
Figure 2-26 Procedure for making the SES for the liquefaction cycle: basic stage	59
Figure 2-27 Procedure for making the SES for the liquefaction cycle: compression refrigeration stage.....	60
Figure 2-28 Procedure for making the SES for the liquefaction cycle: compression with an intercooler stage.	61
Figure 2-29 Procedure for making the SES for the liquefaction cycle: refrigeration stage	62
Figure 2-30 SES for the liquefaction cycle	63
Figure 2-31 Rules in the SES for the liquefaction cycle: basic stage.....	64
Figure 2-32 Rules in the SES for the liquefaction cycle: selecting multiple cooling cycles	66
Figure 2-33 Rules for interconnecting the equipment in the SES for the liquefaction cycle: selecting the compression refrigeration stage.....	68
Figure 2-34 Rules for interconnecting the equipment in the SES for the liquefaction cycle: selecting “compression with an intercooler stage”	70
Figure 2-35 Rules for interconnecting the equipment in the SES for the liquefaction cycle: selecting “refrigeration stage”	72

Figure 2-36 Pruning the SES to generate the logical model of the DMR cycle.....	73
Figure 2-37 Pruned entity structure for the DMR cycle generated by pruning the SES ...	74
Figure 2-38 Logical model of the DMR cycle generated using the pruned entity structure	75
Figure 2-39 Pruning the SES to generate the logical model of the C3MR cycle.....	76
Figure 2-40 PES for the C3MR cycle generated by pruning the SES.....	76
Figure 2-41 Pruning the SES to generate the logical model of the cascade cycle	77
Figure 2-42 PES for the cascade cycle generated by pruning the SES	77
Figure 2-43 Pruning the SES to generate the logical model of the single N ₂ expander cycle	78
Figure 2-44 Pruning the SES to generate the logical model of the dual N ₂ expander cycle	79
Figure 2-45 Pruning the SES to generate the logical model of the Niche cycle	80
Figure 2-46 Sample logical models of the liquefaction cycle generated by pruning the SES.....	81
Figure 2-47 Sample logical models of the liquefaction cycle generated by pruning the SES.....	81
Figure 2-48 Sample logical models of the liquefaction cycle generated by pruning the SES.....	82
Figure 2-49 Sample logical models of the liquefaction cycle generated by pruning the SES.....	82
Figure 3-1 Efficiency of the liquefaction cycle: a design criterion of LNG FPSO.....	84
Figure 3-2 Procedure for determining the optimal operating conditions of the liquefaction cycle	87
Figure 3-3 Tearing of the stream of the recycle loop	94
Figure 3-4 Tearing of the stream of the recycle loop and the heat exchanger.....	95
Figure 3-5 Sequential calculation of the operating conditions for each equipment.....	96
Figure 3-6 Checking of the convergence of the tear stream.....	97
Figure 3-7 Physical model representing each equipment.....	99
Figure 3-8 General thermodynamic-based mathematical model representing each equipment.....	100
Figure 3-9 Operating conditions of a compressor	101
Figure 3-10 Bisection method.....	106

Figure 3-11 Newton-Raphson method	107
Figure 3-12 Sample DEVS model of a condenser, a compressor, and an expansion valve based on the DEVS formalism	110
Figure 3-13 Sample DEVS model of a heat exchanger based on the DEVS formalism	111
Figure 3-14 Sample DEVS model of a common header based on the DEVS formalism	112
Figure 3-15 Sample DEVS model of a tee and a phase separator based on the DEVS formalism	113
Figure 3-16 Combination of a logical model and a physical model for calculating the operating conditions using the sequential modular simulator	114
Figure 3-17 Making of a process simulator by synthesizing the SES and the MB for the liquefaction cycle	115
Figure 3-18 Function of the operator model that manages the sequential calculation procedure.....	116
Figure 3-19 Operator model based on the DEVS formalism.....	117
Figure 3-20 Algorithm of the sequential quadratic programming.....	118
Figure 3-21 Making of the single cooling cycle by combining the logical model in the SES and the physical model in the MB	119
Figure 3-22 Generation of process simulation to determine the optimal operating conditions of the DMR cycle using the sequential modular simulator	121
Figure 3-23 Tearing the streams of the DMR cycle	123
Figure 3-24 Sequential calculation of the DMR cycle using sequential modular simulator	124
Figure 4-1 Simple FT for a simple serial system with two valves	129
Figure 4-2 Simple FT for a simple parallel system with two valves.....	130
Figure 4-3 Gates of the FT (Stamatelatos, 2002)	130
Figure 4-4 Simple redundant system with two valves and a switch: switch failure after valve A failure	131
Figure 4-5 Simple redundant system with two valves and a switch: valve A failure after switch failure.....	132
Figure 4-6 Gates of the DFT: (a) functional dependency (FDEP) gate, (b) spare (SP) gate, (c) priority-and (PAND) gate, and (d) sequence-enforcing (SEQ) gate.....	133
Figure 4-7 DFT for the simple redundant system with two valves and a switch	134
Figure 4-8 Static FT for the simple redundant system with two valves and a switch	134
Figure 4-9 Calculation of failure probability using BDDs.....	136

Figure 4-10 Calculation of the failure probability: conversion to the equivalent Markov chain	137
Figure 4-11 Calculation of the failure probability using the Markov chain	138
Figure 4-12 Procedure of calculating the failure probability using FTA and the Markov chain	139
Figure 4-13 Simple Bayesian network: wet grass	140
Figure 4-14 Conversion of an FT to the equivalent Bayesian network.....	142
Figure 4-15 Calculation of the failure probability using the Bayesian network	143
Figure 4-16 Reliability analysis method used in this thesis	144
Figure 4-17 Reason for the use of an RBD in this thesis	145
Figure 4-18 PES and the logical model of a refrigerator	146
Figure 4-19 Conversion of the logical model to the equivalent RBD to assess the reliability of a refrigerator	147
Figure 4-20 Procedure of the probability calculation using the RBD	147
Figure 4-21 Procedure of the probability calculation using the RBD	148
Figure 4-22 Failure probability calculation via Monte Carlo simulation, using the RBD	149
Figure 4-23 Physical model for equipment with a single inlet/outlet based on the DEVS formalism	150
Figure 4-24 Physical model for equipment with multiple inlets/outlets based on the DEVS formalism	151
Figure 4-25 Analyzer model based on the DEVS formalism: checking for failure of the whole system.....	152
Figure 4-26 Logical model of the firewater pumping system	153
Figure 4-27 FT of the firewater pumping system	154
Figure 4-28 RBD of the firewater pumping system based on the DEVS formalism	154
Figure 4-29 Logical model of the power switching system	156
Figure 4-30 FT of the power switching system.....	156
Figure 4-31 RBD of the power switching system based on the DEVS formalism	157
Figure 4-32 Logical model of the hypothetical computer system.....	158
Figure 4-33 FT of the hypothetical computer system	159
Figure 4-34 RBD of the hypothetical computer system based on the DEVS formalism	159
Figure 4-35 Variation of the SES for reliability analysis considering redundant equipment	

.....	161
Figure 4-36 Logical model of the DMR cycle with control logics	163
Figure 4-37 Partitioning of the cooling cycle: the precooling and main cooling cycles .	164
Figure 4-38 RBD of the DMR cycle: precooling cycle with a control valve.....	165
Figure 4-39 RBD of the DMR cycle: precooling considering the full redundancies on the condenser and expansion valve	166
Figure 4-40 RBD of the DMR cycle: precooling considering the half-redundancies on the condenser and expansion valve	166
Figure 4-41 RBD of the DMR cycle: precooling cycle	167
Figure 4-42 RBD of the DMR cycle: main cooling cycle.....	167
Figure 4-43 Checking the system failure of the precooling cycle of the DMR cycle	168
Figure 4-44 Reliability analysis of the C3MR cycle.....	170
Figure 4-45 Reliability analysis of the cascade cycle	171
Figure 4-46 Reliability analysis of the single N ₂ expander cycle	171
Figure 4-47 Reliability analysis of the Niche cycle	172
Figure 5-1 Cryogenic heat exchangers: plate fin heat exchanger and spiral wound heat exchanger	174
Figure 5-2 Concept of the spiral wound heat exchanger (SWHE).....	175
Figure 5-3 Simplified model of the heat exchanger.....	176
Figure 5-4 Analysis techniques of physical phenomena	178
Figure 5-5 (a) Velocity vectors of the HPP model. (b) Head-on collision rule for the HPP model.....	181
Figure 5-6 (a) Velocity vectors of the FHP model. (b) Sample head-on collision rule for the FHP model.....	182
Figure 5-7 Concept of the lattice Boltzmann method: application to one-dimensional rod	184
Figure 5-8 Simplified distribution function for an example of one-dimensional rod	185
Figure 5-9 Collision process of the lattice Boltzmann method: (a) before collision, (c) particles of Group R after collision	186
Figure 5-10 Equilibrium distribution function and collision operator of the lattice Boltzmann method	187
Figure 5-11 Lattice arrangements for a 2D problem with nine directions (D2Q9).....	190
Figure 5-12 Particle streaming paths from a corner node in a periodic 2D system in LBM	

.....	191
Figure 5-13 On-grid bounce-back boundary in LBM	192
Figure 5-14 Mid-grid bounce-back boundary in LBM	192
Figure 5-15 Analysis of the motion effect on the heat exchanger using LBM and its application to reliability analysis.....	193
Figure A-1 (a) Concept of torpedo-firing simulation, (b) Configuration of model based on the DEVS formalism	207
Figure A-2 Procedure of torpedo-firing simulation based on the DEVS formalism (1/3)	209
Figure A-3 Procedure of torpedo-firing simulation based on the DEVS formalism (2/3)	211
Figure A-4 Procedure of torpedo-firing simulation based on the DEVS formalism (3/3)	212
Figure A-5 (a) Concept of submarine diving simulation, (b) configuration of models for submarine diving simulation	213
Figure A-6 Procedure of submarine diving simulation based on the DEVS and DTSS formalism (1/3).....	214
Figure A-7 Procedure of submarine diving simulation based on the DEVS and DTSS formalism (2/3).....	217
Figure A-8 Procedure of submarine diving simulation based on the DEVS and DTSS formalism (3/3).....	219
Figure A-9 Class for the definition of variables using on the simulation based on the DEVS and DTSS formalism	221
Figure A-10 Classes for messages in the simulation kernel: defined by inheriting 'CValue' class	222
Figure A-11 Classes for inputs, outputs, and state variables: defined by inheriting 'CState' class	223
Figure A-12 Classes to define an atomic model and a coupled model.....	224
Figure A-13 Classes to define a simulator and a coordinator.....	225
Figure B-1 Design variables of a compressor	226
Figure B-2 Design variables of a condenser	233
Figure B-3 Design variables of an expansion valve.....	237
Figure B-4 Design variables of a heat exchanger	240
Figure B-5 Design variables of a tee.....	243

Figure B-6 Design variables of a phase separator	246
Figure B-7 Common header.....	249
Figure D-1 One-dimensional heat equation for 1-D rod	260
Figure D-2 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 2 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	262
Figure D-3 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 1 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	263
Figure D-4 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 0.5 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	264
Figure D-5 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 0.4 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	264
Figure D-6 Computational molecules demonstrating the fundamental differences between (a) explicit and (b) implicit methods	265
Figure D-7 Applying an implicit method of the FDM to one-dimensional heat equation	266
Figure D-8 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 2 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	268
Figure D-9 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 1 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	268
Figure D-10 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 0.5 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	269
Figure D-11 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 0.4 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	269
Figure D-12 Computational molecules demonstrating the fundamental differences between (a) simple implicit and (b) Crank-Nicolson methods.....	270
Figure D-13 Applying the Crank-Nicolson method of the FDM to one-dimensional heat equation.....	271
Figure D-14 Calculation result of one-dimensional heat equation using the Crank- Nicolson method of the FDM ($\Delta x = 2 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$) .	273
Figure D-15 Calculation result of one-dimensional heat equation using the Crank- Nicolson method of the FDM ($\Delta x = 1 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$) .	274
Figure D-16 Calculation result of one-dimensional heat equation using the Crank- Nicolson method of the FDM ($\Delta x = 0.5 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$)	274
Figure D-17 Calculation result of one-dimensional heat equation using the Crank- Nicolson method of the FDM ($\Delta x = 0.4 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$)	

.....	275
Figure D-18 Lattices for one-dimensional heat equation using the LBM.....	277
Figure D-19 Procedure of the calculation using the LBM for one-dimensional heat equation at step $k = 1$	278
Figure D-20 Consideration of the boundary conditions using the LBM for one-dimensional heat equation at step $k = 1$	278
Figure D-21 LB equations and lattice distributions for one-dimensional heat equation at step $k = 1$	279
Figure D-22 LB equations and lattice distributions for one-dimensional heat equation at step $k = 2$	280
Figure D-23 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 2 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	281
Figure D-24 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 1 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	281
Figure D-25 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 0.5 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	282
Figure D-26 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 0.4 \text{ cm}$, $\Delta t = 0.1 \text{ s}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$).....	282

Tables

Table 2-1 Rules for interconnecting the child entities of the <i>basic stage</i> entity.....	65
Table 2-2 Rules for interconnecting entities when the expansion equipment is chosen ...	65
Table 2-3 Rules for disconnecting and connecting the equipment of multiple cooling cycles.....	66
Table 2-4 Rules for disconnecting the equipment of the <i>basic stage</i> entity when additional stage compression refrigeration is chosen.....	67
Table 2-5 Rules for interconnecting the child entities of the compression refrigeration stage entity	68
Table 2-6 Rules for disconnecting the equipment of the <i>basic stage</i> entity when the additional stage, compression with an intercooler, is chosen.....	69
Table 2-7 Rules for interconnecting the child entities of the <i>compression refrigeration stage</i> entity	70
Table 2-8 Rules for disconnecting the equipment of the <i>basic stage</i> entity when the additional stage “compression with an intercooler” is chosen	71
Table 2-9 Rules for interconnecting the child entities of the <i>refrigeration stage</i> entity....	71
Table 3-1 General optimization problem to determine the optimal operating conditions of liquefaction cycle	85
Table 3-2 Optimization problem of a refrigerator for determining the optimal operating conditions based on the equation-oriented simulator	91
Table 3-3 Rules for the calculation procedure for each equipment.....	108
Table 3-4 Rules for checking the convergence for each equipment.....	109
Table 3-5 Operating conditions of the single cooling cycle obtained using ASPEN HYSYS and of that obtained in this thesis	120
Table 3-6 Given data for the DMR cycle	122
Table 3-7 Resulting optimal operating conditions for the DMR cycle using the process simulator proposed in this thesis	125
Table 4-1 Comparison of reliability analysis results for the firewater pumping system.	155
Table 4-2 Comparison of reliability analysis results for the power switching system....	157
Table 4-3 Comparison of reliability analysis results for the hypothetical computer system	160
Table 4-4 Comparison of reliability analysis results for the DMR cycle.....	169

Table 4-5 Probability convergence according to the number of calculations	169
Table 5-1 Comparison of advantages and disadvantages between numerical analysis methods	177
Table A-1 Components of an atomic model based on DEVS formalism.....	203
Table A-2 Components of a coupled model based on DEVS formalism	205
Table A-3 Components of a atomic model based on DTSS formalism.....	206
Table A-4 Classes for the messages in the simulation kernel based on the DEVS formalism	222
Table A-5 Classes for inputs, outputs, and state variables	223
Table B-1 Values of the parameters for the SRK equation.....	253
Table D-1 Comparison of the calculation result for one-dimensional heat equation: temperature at $t = 10\text{ s}$ at $x = 2\text{ cm}$	283

Abstract

Ontological Modeling for Process and Reliability Simulation of LNG FPSO Liquefaction Cycle based on the DEVS formalism considering Ship Motion Effect

The liquefaction process system is regarded as primary among all topside systems in liquefied natural gas (LNG) floating, production, storage and offloading (FPSO). The liquefaction process system typically accounts for 70 percent of the capital cost of the topside process system and 30-40 percent of the overall LNG FPSO cost, so it is very important to optimize its cycles. Such cycles have seven main equipment parts: the compressor, condenser, expansion valve, evaporator, phase separator, common header and tee. Many types of liquefaction cycles have been determined according to their respective syntheses. Thus, ontological modeling and simulation were carried out in this thesis to determine the optimal LNG FPSO liquefaction cycles.

This thesis has three parts. In the first part, to automatically generate various alternative liquefaction cycles, a generic logical model of the liquefaction cycle is proposed based on ontological modeling. A logical model is a liquefaction cycle configuration that reflects the relationship between the equipment. There are general rules in combining equipment extracted from existing onshore liquefaction cycles like the C3MR (propane pre-cooled mixed-refrigerant) cycle and the DMR (dual mixed-refrigerant) cycle. The generic logical

model that has all these rules is represented by the system entity structure (SES), an ontological framework that hierarchically represents the elements of a system and their relationships. By pruning the SES, various alternative logical models for liquefaction cycles can be automatically generated.

The second part of this thesis presents the results of the process and reliability simulation that was performed using the aforementioned generic logical model to consider two design criteria: effectiveness and reliability. Process simulation was performed to determine the optimal--i.e., the most effective--operating conditions for the liquefaction cycles. There are two kinds of process simulator: the equation-based simulator and the sequential modular simulator. In this thesis, the sequential modular simulator was used because of its advantages of equipment modularization, easy debugging, and convenient modification of the liquefaction cycles. All the equipment for the sequential modular simulator were modeled based on the Discrete Event System Specification (DEVS) formalism and connected with the logical liquefaction cycle models. Reliability simulation was also performed to calculate the failure probability of the liquefaction cycle in terms of its reliability. The fault tree analysis (FTA) with the Markov chain or the Bayesian network is traditionally used to calculate the failure probability of the system. To perform the FTA, however, an expert and high-cost complex system is needed. Therefore, in this thesis, the reliability block diagram (RBD), which is a graphical presentation of a system diagram that connects subsystems of components according to their reliability relationships, was used. The RBD can be easily generated from the logical models of liquefaction cycles. It can be quantitatively analyzed using Monte Carlo simulation, and the equipment for the reliability simulation is modeled based on the DEVS formalism. Process simulation and reliability simulation are performed for conventional liquefaction cycles like the C3MR cycle, DMR cycle, cascade cycle, N₂ dual expander cycle and Niche (N₂-methane) cycle.

Finally, the influence of ship motion on the performance of a heat exchanger was also considered in this thesis. The change in the effectiveness of the heat exchanger due to the ship motion was calculated using the Lattice Boltzmann method (LBM). During the operating time, the failure of the heat exchanger was counted considering this effectiveness, and the result was applied to the reliability simulation.

Keywords: LNG FPSO, Liquefaction Cycle, Ontological Modeling, Process and Reliability Simulation, Ship Motion Effect, DEVS Formalism

Student number: 2003-21035

1. Introduction

1.1. LNG FPSO Liquefaction Cycle

1.1.1. LNG FPSO Concept

Natural gas is expected to play a central role in meeting the world's mounting energy demand in the coming decades because it is readily available and the cleaner energy alternative to oil and coal. Thus, there is greater reliance on it today. This is especially important in view of rising concern with environmental pollution and nuclear power plant hazards. This reliance has brought about the need to develop new liquefied natural gas (LNG) projects and innovate technologies to meet the global demand.

Among the proven natural gas reserves in the world, about one-third is in offshore fields. Until now, natural gas in an offshore production site is transported by a pipeline to the onshore LNG plant, where the natural gas is liquefied. As shown in Figure 1-1, a traditional onshore plant for offshore gas reserves usually includes an offshore platform for dehydration and compression (Figure 1-1 (a)), large pipelines leading to the shore (Figure 1-1 (b)), an onshore liquefaction plant (Figure 1-1 (c) and (d)), and a harbor to accommodate LNG carriers (Figure 1-1 (e)).

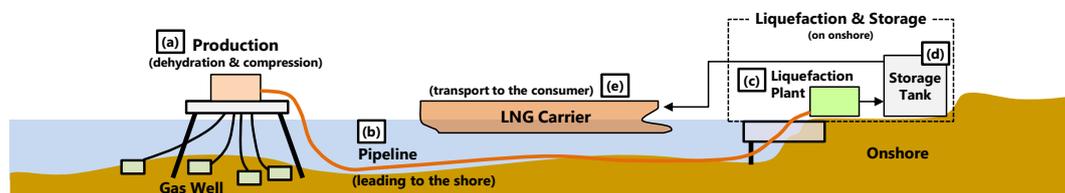


Figure 1-1 LNG production in an onshore facility

Offshore gas reserves are often considered stranded. Much of the world's gas reserves are in offshore fields, though onshore LNG processing is generally favored. As a result, there is growing interest in unlocking and monetizing these reserves with floating facilities that are capable of liquefying and storing natural gas.

An LNG floating production storage and offloading (FPSO) facility, as a stand-alone unit, must have all the systems of an onshore baseload LNG facility, like its own power and heat generation unit and all other necessary utilities (Figure 1-2). An LNG FPSO facility in an offshore field site takes away the need for gas compression platforms, long subsea pipelines for transporting the natural gas to an onshore LNG plant, and onshore construction, including an onshore LNG plant, roads, storage yards and accommodation facilities.

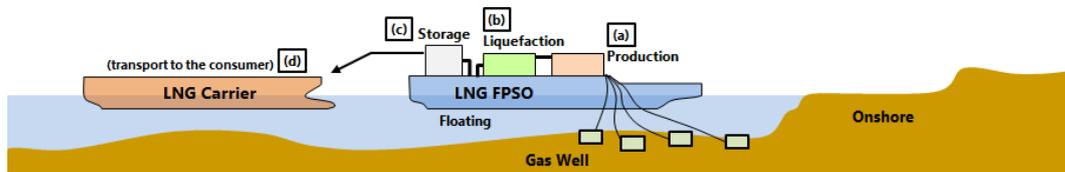


Figure 1-2 LNG production via LNG FPSO

Compared with other exploration technologies such as direct pipeline transfer, LNG FPSO is a more effective and realistic way to exploit and use marginal gas fields and offshore associated gas fields, where a pipeline network is not available, with limited recoverable reserves and few wells (Gu and Ju, 2008). This floating plant can produce and store LNG in the sea and transfer the LNG product through offloading facilities to LNG carriers and to the world market. Moreover, LNG FPSO is considered a major attraction in a supply-short environment, and it consists of a mobile production unit that can be moved to a new location once the existing field is depleted. Therefore, LNG FPSO can accelerate LNG development.

1.1.2. LNG FPSO Liquefaction Process System

An LNG FPSO facility consists of a hull, a turret and a topside. The topside has two parts: a process system and a utility system (Hwang et al., 2013). The process system consists of separation, pretreatment, fractionation and liquefaction, as shown in Figure 1-3.

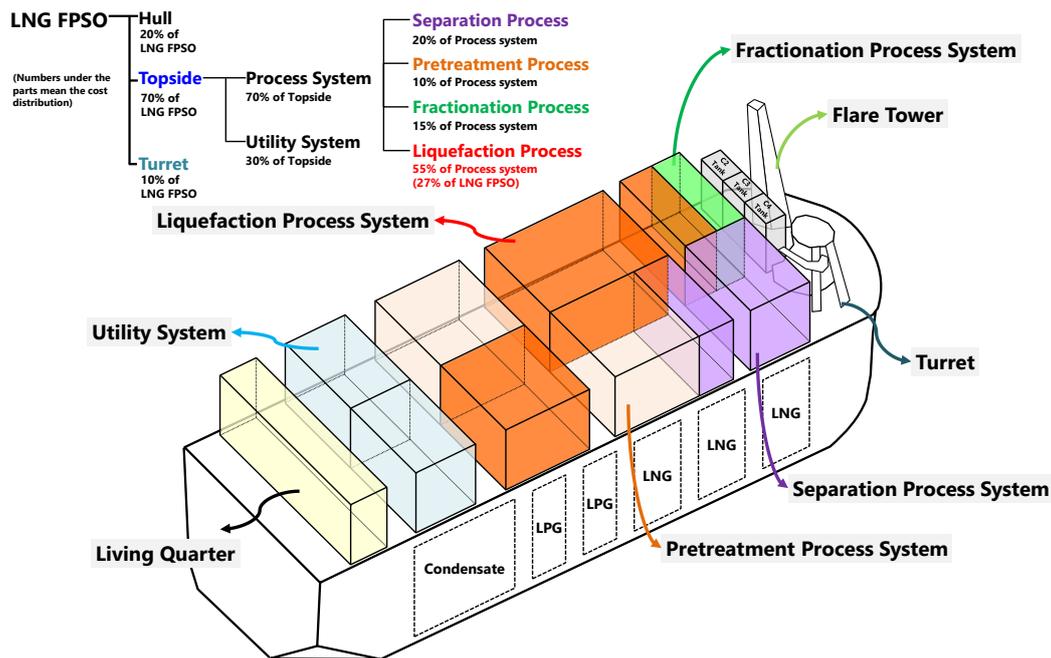


Figure 1-3 LNG FPSO configuration

The liquefaction process system, which liquefies natural gas into LNG, is regarded as primary among all topside systems in LNG FPSO. As shown in Figure 1-3, the liquefaction process system typically accounts for 70 percent of the capital cost of the topside process system and 30-40 percent of the overall LNG FPSO cost, so its optimal liquefaction cycle must be determined. Such cycle has seven main equipment parts: a compressor, condenser, expansion valve, evaporator, phase separator, common header and tee. Many types of liquefaction cycles were determined according to their respective syntheses.

1.1.3. Design and Evaluation of the LNG FPSO Liquefaction Cycle

(1) Synthesis of liquefaction cycles

Figure 1-4 shows typical icons of the component parts (equipment) used in the flowsheet of a steady-state process simulator. The synthesis of the liquefaction cycles is the step in the design wherein the engineer selects the component parts and how to interconnect them to create the flowsheet of the liquefaction process system (Nishida et al., 1981).

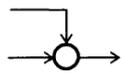
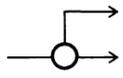
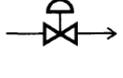
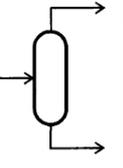
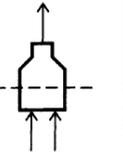
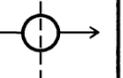
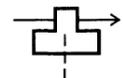
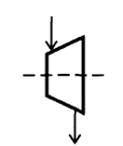
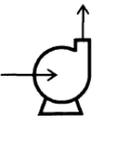
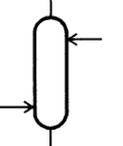
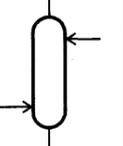
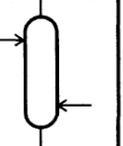
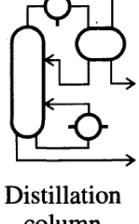
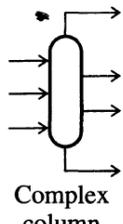
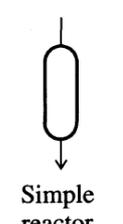
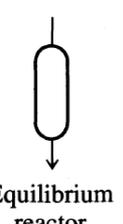
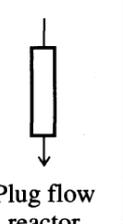
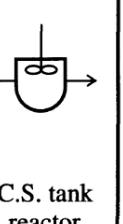
MIXER  Mixer	SPLIT  Splitter	VALVE  Valve	FDRUM  Flash drum	FURNC  Furnace	EXCHR  Heat exchanger
COMPR  Compressor	TURBN  Turbine	PPUMP  Process pump	ABSOR  Absorber	XTRCT  Extractor	STRIP  Stripper
DISTF  Distillation column	CXCOL  Complex column	RSIMP  Simple reactor	REQUL  Equilibrium reactor	RPLUG  Plug flow reactor	RCSTR  C.S. tank reactor

Figure 1-4 Typical process component parts used in the process flowsheet (Edgar, 2001)

Synthesis means the designation of the structure of the plant elements, like the unit operations and equipment, that will meet the designer's goals (Edgar, 2001). The engineers

want to find the optimal liquefaction cycle within the design criteria of the liquefaction process system. They check the flowsheet to identify equipment that can be eliminated or rearranged, alternative separation methods, unnecessary feeds that can be eliminated, unwanted or hazardous products or byproducts that can be deleted, heat integration that can be improved, etc. Even if no new technology will be used, the number of alternatives substantially increases. The engineers evaluate these various alternatives based on the design criteria and choose the optimal design.

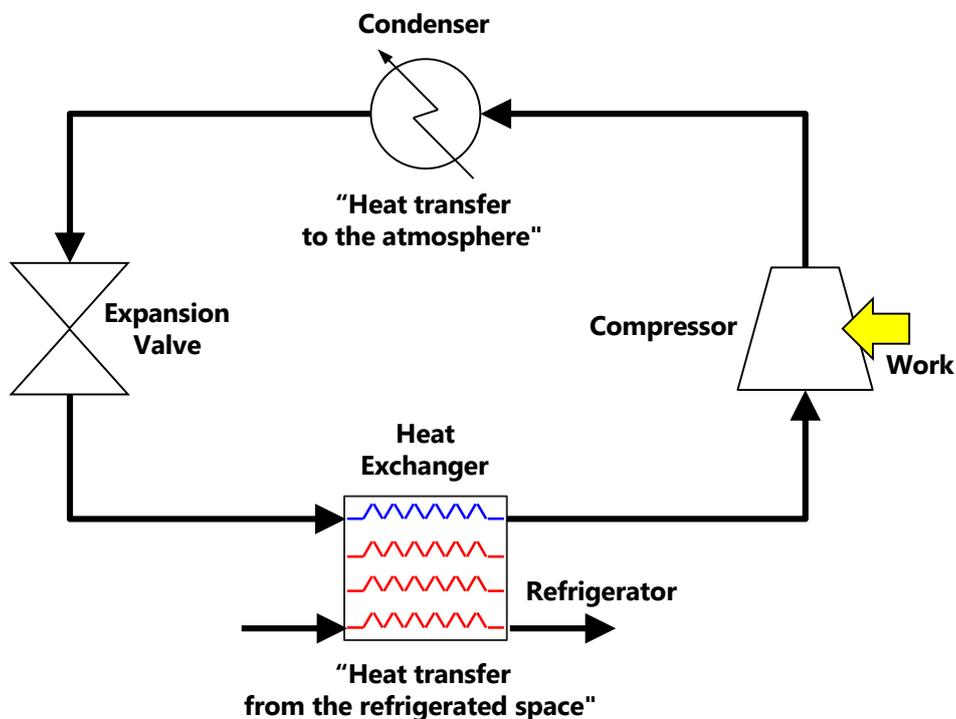


Figure 1-5 Sample liquefaction cycle: a refrigerator

The flowsheet of the liquefaction process system is based on a “cycle.” The cycles of the liquefaction process system require seven main equipment parts: a compressor, condenser, expansion valve (or expander), evaporator, phase separator, common header and tee. Figure 1-5 shows a refrigerator, which has a single liquefaction cycle. In this thesis,

the refrigerator was made by selecting the component parts like the compressor and the condenser, and interconnecting them.

If two compressors are cheaper than a larger one with the same capacity, multiple compressions can be considered using two smaller compressors in a point of the cost. For this reason, the configuration of the single cooling cycle can be varied by adding and rearranging the equipment parts, as shown in Figure 1-6.

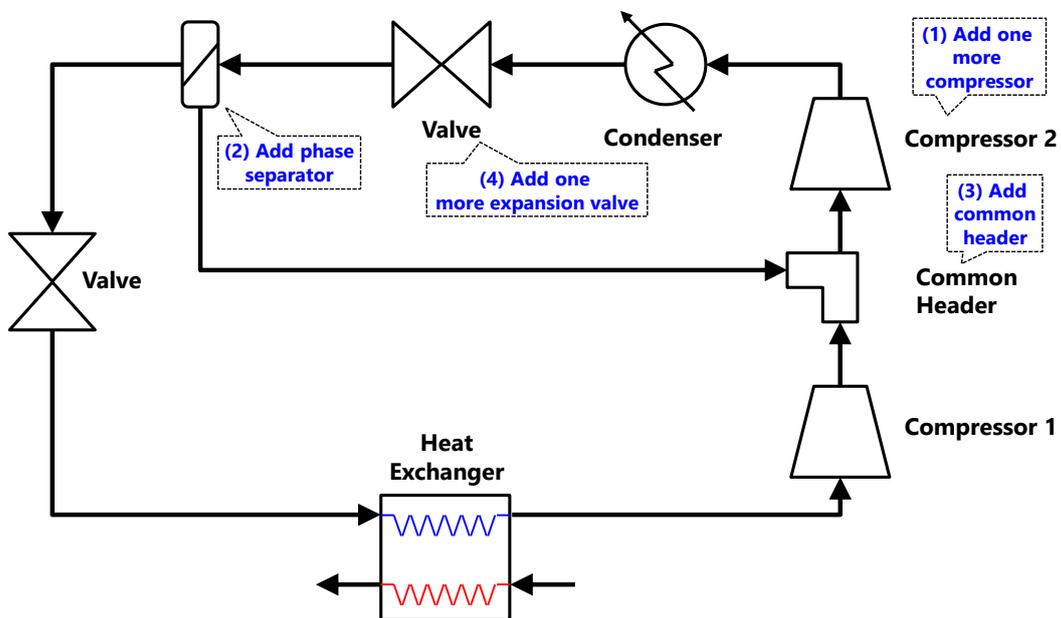


Figure 1-6 Another combination of liquefaction cycles for a refrigerator: addition of equipment to enhance the efficiency of the liquefaction cycle

Since the area available for the liquefaction cycle in offshore applications is smaller than that in an onshore plant, various LNG FPSO alternative liquefaction cycles must be evaluated using process syntheses to select the optimal one. For the optimal synthesis, some specific objectives, which are the design criteria in the LNG FPSO liquefaction cycle, are needed like efficiency and reliability.

(2) LNG FPSO design criteria

LNG plants in LNG FPSO facilities have demands on equipment that differ from those of traditional onshore LNG plants. While thermodynamic efficiency and train capacity are undoubtedly the important performance parameters for onshore natural gas fields, for LNG FPSO facilities, other aspects are important. Offshore plants have considerably smaller footprints, higher demand for process safety, and high flexibility. They should also sometimes be able to handle various gas compositions. Furthermore, the LNG process should be able to cope with varying environmental conditions. Ship motion is known to aggravate the performance of some equipment, like the heat exchanger. A proven and optimized technology remains very important, however, due to the high uptime demand.

① Efficiency: minimizing power consumption

Efficiency is an important criterion in selecting an LNG FPSO liquefaction cycle. The total efficiency is often stated as the ratio of the power consumption to the produced LNG in tons in a day. An exergy analysis usually aims to determine the actual and maximum efficiencies of a process system and the sites of exergy destruction. In thermodynamics, the exergy of a system is the maximum useful work possible during a process that brings the system into equilibrium with a heat reservoir. An exergy analysis is performed in the field of industrial ecology to use energy more efficiently. An important objective of exergy analysis for systems that perform work such as liquefaction of gases is to find the minimum work required for a certain desired result and compare it with the actual work performed. The ratio of these two quantities is often considered the exergy efficiency of such a liquefaction process.

Offshore LNG processes have much smaller available footprints than onshore plants. This is one of the major LNG FPSO parameters for the selection of liquefaction processes. Generally, the more effective the selected liquefaction cycle is, the smaller the required footprint is for the LNG FPSO liquefaction process system.

② Reliability: Failure probability of the system

Reliability is important for the overall economics of the plant. It involves the risks of unplanned production stops and the downtime duration. Repairs and maintenance are relatively time-consuming for cryogenic processes and are difficult to perform offshore. All liquefaction processes are widely applied in onshore plants and have proven reliability. Because of the specific environment of offshore plants, processes that involve rotating equipment or greater heights have higher probabilities of failure and require more frequent maintenance than static equipment.

③ Ship motion effect

One of the major performance parameters of offshore plants is the effect of ship motion. Ship motion leads to more mechanical reliability and safety issues due to sloshing effects, but the main concern is the ship motion aggregate performance of LNG liquefaction process equipment like large heat exchangers. In particular, liquefaction processes that deal with two-phase flow may malfunction as flow maldistribution occurs in the equipment.

1.2. Motivation

With the increasing demand for natural gas, LNG FPSO is an effective and realistic way of exploiting, recovering, storing, transporting and propagating end-use applications of marginal gas fields and offshore associated gas resources. As mentioned in Section 1.1.2, the liquefaction process system, which liquefies natural gas into LNG, is regarded as primary among all topside systems in LNG FPSO. Thus, the design and evaluation of the liquefaction cycle, on which the flowsheet of the liquefaction process system is based, is very important in LNG FPSO design.

To design the LNG FPSO liquefaction cycle, the engineer should review the liquefaction cycles that had been effectively used in onshore plants by adding, rearranging and eliminating equipment parts. During this kind of modification, the engineer should check if this modification is reasonable or not, and he or she may unintentionally miss some of the better alternative liquefaction cycles. Moreover, if beginners in the liquefaction process system field want to understand the modified cycles, comment on an idea and make more modifications, they should have knowledge on the liquefaction cycles.

In this thesis, an ontological modeling that can automatically generate various alternative liquefaction cycles is proposed to help design the optimal liquefaction cycle. The ontology for modeling the liquefaction cycles includes the rules for designing the liquefaction cycle by combining the equipment parts. These rules are extracted from the liquefaction cycles used in onshore plants. The system entity structure (SES), which hierarchically represents the elements of a system and their relationships, is adopted for the creation of an ontological modeling framework.

After the design of the liquefaction cycles, the engineers should review and evaluate

them based on the LNG FPSO design criteria mentioned in Section 1.1.3(2). The SES describes the relationships between the equipment and the hierarchical structures of the liquefaction cycles. In view of the efficiency of the liquefaction cycle, if the behavioral (functional) models for the process simulation are given, the process simulation can be performed by combining the SES with these behavioral models. In the same manner, if the behavioral models for the reliability analysis are given, the reliability (failure probability) of the liquefaction cycle can be calculated. Similarly, a concept of the SES and the model base (MB) proposed by Zeigler (1990) is adopted to make the process and the reliability simulation automatic for the evaluation of the liquefaction cycles.

An LNG FPSO facility, being an offshore plant, is affected by ship motion. Ship motions require greater mechanical reliability due to their sloshing effects. Especially, the heat exchanger, which is the main device in the liquefaction cycle, is very sensitive to ship motion because of its greater height. Thus, the effectiveness of the heat exchanger in tile motions is evaluated in the flow and thermal analysis. Since the heat exchanger has a complex internal structure, it is simplified and analyzed using the lattice Boltzmann method (LBM), which can model simply the complex boundaries.

1.3. Scope of This Thesis

First, to automatically generate various alternative liquefaction cycles, a generic logical model of a liquefaction cycle is proposed based on ontological modeling, as shown in Figure 1-7. A logical model is a configuration of a liquefaction cycle that reflects the relationship between the equipment parts and their hierarchical structures. The rules for interconnecting the equipment parts of the liquefaction cycle are extracted from existing onshore liquefaction cycles like the C3MR (propane pre-cooled mixed-refrigerant) cycle and the DMR (dual mixed-refrigerant) cycle. A generic logical model has all these rules, and it can be used to make a logical model.

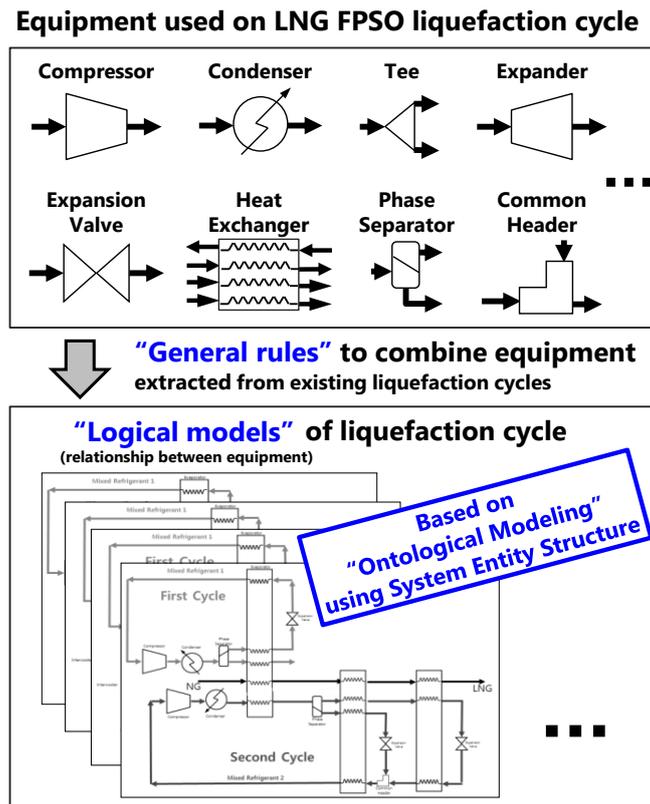


Figure 1-7 Generation of the logical models of the liquefaction cycle based on the rules for combining equipment

The generic logical model that contains all these rules is represented by the SES, which is an ontological framework that hierarchically represents the elements of a system and their relationships. The rules for combining the equipment are included in the SES structure. By pruning the SES, various alternatives to the logical model for liquefaction cycles can be automatically generated. Automatic generation of a logical model for a liquefaction cycle based on the SES is introduced in Chapter 2.

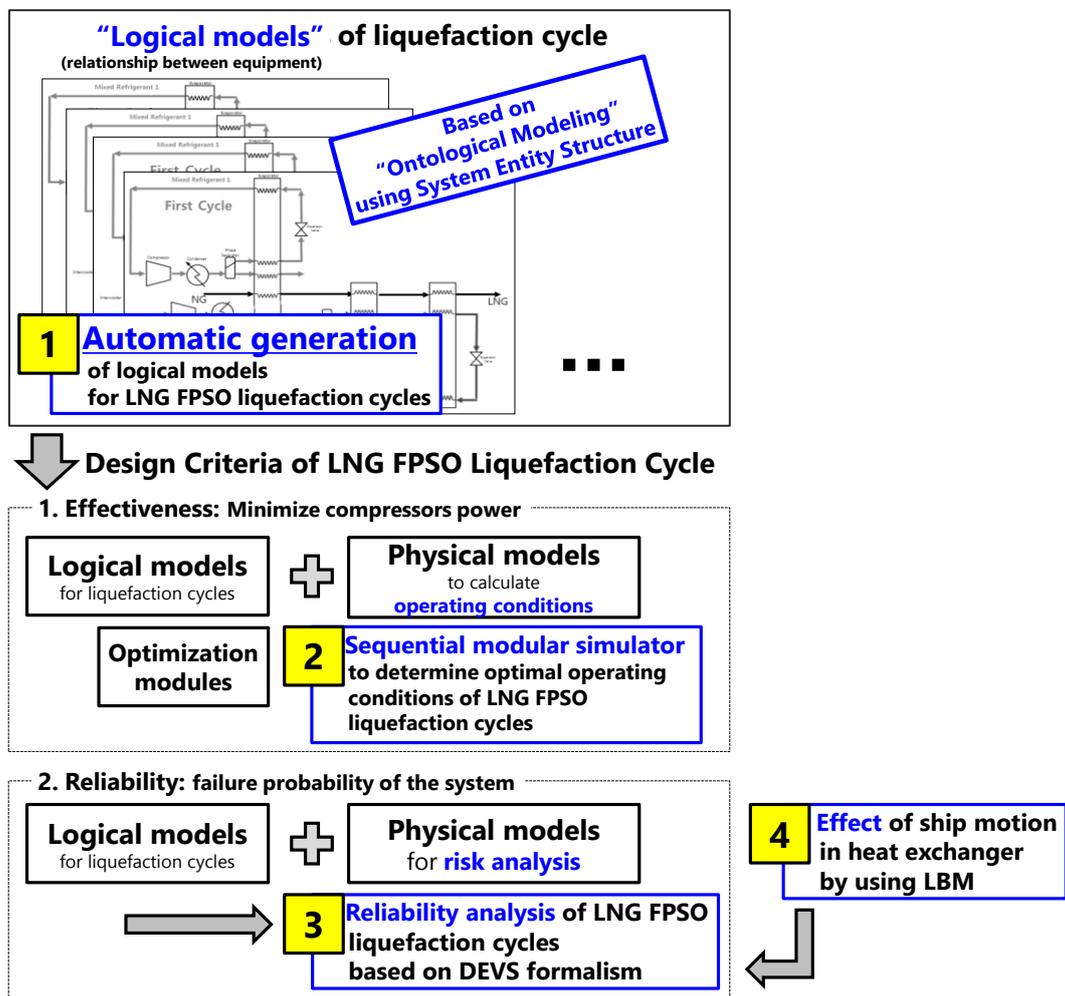


Figure 1-8 Scope of this thesis

In this thesis, using this generic logical model, process and reliability simulation were performed to consider the design criteria of effectiveness and reliability. Process simulation was performed to determine the optimal operating conditions of the liquefaction cycles in terms of their effectiveness. There are two methods of process simulation: equation-based simulation and sequential modular simulation. In this thesis, sequential modular simulation was used for the process simulation because it has some advantages like modularization of equipment, easy debugging and convenient modification of the liquefaction cycles. All the equipment for the sequential modular simulation were modeled based on the DEVS formalism and connected based on the relationships in the logical model of the liquefaction cycle. In Chapter 3 of this thesis, the process simulation to determine the optimal operating conditions of the liquefaction cycle is introduced.

Reliability simulation was also performed to calculate the failure probability of the liquefaction cycle in terms of its reliability. The fault tree analysis (FTA) with the Markov chain or the Bayesian network is traditionally used to calculate the failure probability of the system. To perform the FTA, however, expert experience and a high-cost complex system are needed. Therefore, in this thesis, an RBD, which is a graphical presentation of a system diagram that connects subsystems of components according to their reliability relationships, was used. The RBD can be easily generated from the logical models of liquefaction cycles. The RBD was quantitatively analyzed in this thesis using Monte Carlo simulation, and the equipment for the reliability simulation was modeled based on the DEVS formalism. The reliability simulation of the liquefaction cycle is introduced in Chapter 4 of this thesis.

The influence of the ship motion on the performance of the heat exchanger is also considered here. The change in the effectiveness of the heat exchanger due to the ship

motion was calculated using the lattice Boltzmann method (LBM). During the operation, the duration of the failure of the heat exchanger was measured considering this effectiveness, and the result was applied in the reliability simulation. The ship motion effect is introduced in Chapter 5 of this thesis.

Figure 1-9 shows the configuration of the program that was developed in this thesis. For the ontological modeling, the SES was implemented in the simulation kernel. The DEVS formalism, which is used to model the behavior for the process and reliability simulation, was also implemented in the simulation kernel. The logical models of the liquefaction cycles were based on the SES of the simulation kernel.

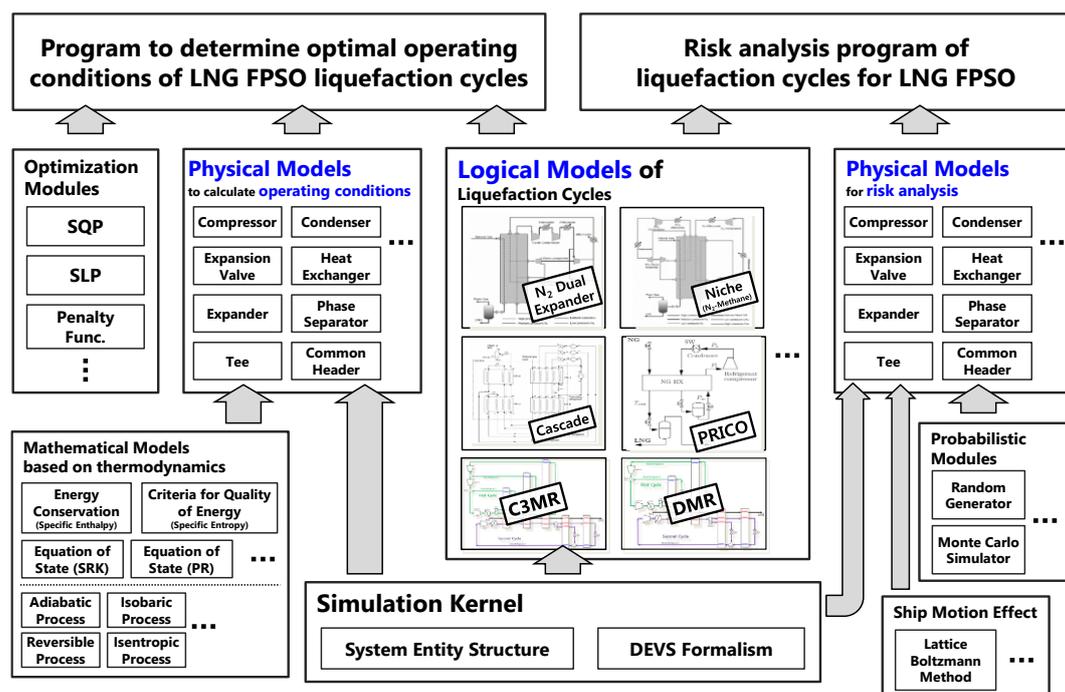


Figure 1-9 Configuration of the program that was developed in this thesis

For the process simulation, mathematical models based on thermodynamics were used to produce the physical model, which includes the behavior for the process simulation,

based on the DEVS formalism of the simulation kernel. EzOptimizer (Lee et al., 2002b) was used for the optimization module. Using the logical model, physical model for the process simulation and optimization module, the program for determining the optimal operating conditions of the liquefaction cycle was configured.

For the reliability simulation, a module for calculating the probability was made and used to produce the physical model for the reliability simulation, which was also modeled based on the DEVS formalism of the simulation kernel. The program for calculating the reliability was configured using the logical model and the physical model for the reliability simulation. In addition, the module for considering the effect of the ship motion on the heat exchanger was made based on LBM, and the result was applied to the reliability simulation.

1.4. Related Studies

1.4.1. Process Simulation to Determine the Optimal Operating Conditions of the Liquefaction Cycles

Process simulation is a widely used technique in the design, analysis and optimization of process plants. There are two types of process simulators based on their architecture: sequential modular simulators and equation-oriented simulators (Edgar, 2001 and Venkatarathnam, 2008). The governing equations in each process unit are solved one at a time and sequentially in the sequential modular approach, whereas the governing equations of all the units are solved simultaneously in the equation-oriented approach.

Using the equation-oriented approach, Lee et al. (2002a) determined the optimal operating conditions for the single mixed-refrigerant (SMR) cycle to minimize the power required for the compressors in the cycle, by formulating a mathematical model of the cycle. Jensen (2008) also determined the optimal operating conditions of the SMR cycle by formulating a mathematical model. While a mathematical model of the SMR cycle was the focus of a study of Lee et al. (2002a), nine cases, including those that were formulated, and the optimal operating conditions in each case were obtained. Venkatarathnam (2008) determined the optimal operating conditions of various cycles, including of the DMR cycle, to maximize their efficiency. Instead of formulating a mathematical model of the cycle, ASPEN Plus was used. Furthermore, the optimization problem was solved using the sequential quadratic programming (SQP) method.

The equation-based approach has several inconveniences in handling errors and modifying the flowsheet, though it has no need for nested iteration loops and better

performance. In contrast, the sequential modular approach has the advantages of easy module construction and understanding, easy addition and deletion of modules to and from a flowsheet, and easier module programming and debugging than with the sets of equations in the equation-oriented approach. Therefore, many commercial process simulators like ASPEN HYSYS and PRO/II use this sequential modular approach. In this thesis, the sequential modular approach was used because of its aforementioned advantages and its easy combination with the logical model based on the SES, so alternative liquefaction cycles were evaluated to choose the optimal process.

1.4.2. Reliability Analysis of the Liquefaction Cycles

A system is a collection of components, subsystems and assemblies that are arranged according to a specific design to achieve acceptable performance and reliability levels (Distefano and Puliafito, 2009). The main goal of system reliability/availability evaluation is the construction of a model (life distribution) that represents the time to the failure of the entire system based on the life distributions and maintenance policies for the components, subassemblies and/or assemblies (black boxes) with which the system was composed.

There are several ways to represent and analyze system reliability. A fault tree (FT) is a compact graphical method of analyzing system reliability (Vesely et al., 1981). FTs use Boolean gates to show how unit failures combine to produce system failure. They do not have any elements or capabilities to model the reliability interactions among the components or subsystems — aspects that are conventionally qualified as dynamic. Common examples of dynamic reliability behavior are standby redundancy, interferences, dependencies and common-cause failures. These arguments awakened the scientific community to the need for new formalisms such as dynamic fault trees (DFTs) (Boyd,

1991). Markov models are widely used to calculate system reliability with DFTs (Stamatelatos, 2002). Chang et al. (2008) used only Markov models to assess the reliability of reliquefaction systems of LNG carriers without DFTs. Markov models do not directly use the structure of the DFTs, however, and they need too many states when the DFTs are more complex. Because of these disadvantages of Markov models, Bayesian networks are used to calculate system reliability (Bobbio et al., 2001). Bayesian networks can use the structure of DFTs without any changes, so they are more compact and fast than Markov models.

All the aforementioned system reliability analysis methods are powerful though not always user-friendly, because it is sometimes very hard to obtain a model directly from the specifications of the system, especially of complex systems. To make the DFTs of the system, an expert's experience and high-cost complex systems are needed. In addition, the results of the analysis differ according to the selection of the logical gate, so they depend mostly on the expert who makes the DFTs.

This fact prompted the definition of specific reliability modeling formalisms as RBDs. An RBD is a graphical presentation of a system diagram that connects subsystems of components according to their function or reliability relationships. It is user-friendly and makes it easy to obtain a model directly from the specifications. Furthermore, from its concept, dynamic RBDs (DRBDs) were proposed in this thesis based on an extension of the existing RBD formalism (Distefano and Puliafito, 2007). The main advantage of a DRBD is its capability to model dependencies among subsystems or components concerning their reliability interactions. A DRBD can be quantitatively analyzed using existing methods like Markov chains and Monte Carlo simulation.

The logical models of the liquefaction cycle, which the SES automatically generated in

this thesis, have related equipment. Since the concept of the logical model is similar to that of RBDs, the logical models were transformed into RBDs without any touches. The causes of system failure are trigger events, so the DEVS formalism, which can represent a model for event-based simulation, was used in this thesis. All the used reliability simulation models were based on the DEVS formalism. The reliability of the liquefaction cycle was calculated using Monte Carlo simulation.

1.4.3. Effect of Ship Motion on the Performance of the Heat Exchanger

Most studies that analyzed the performance of the heat exchanger due to ship motion were based on experiment results. Pekdemir and White (2007) performed the tests on the liquid distribution of the model for a spiral-wound heat exchanger (SWHE). In the study of Pekdemir and White, the liquid distribution was measured but its effect on the SWHE performance could not be determined. Lex et al. (2007) erected two test facilities in a laboratory to study the SWHE performance, as shown in Figure 1-10. In such study, a numerical model for the heat exchanger was developed and later tuned and verified using the data from the internal fluid flow tests and the heat transfer test that were performed in the test facilities.

Hetland et al. (2009) considered the impact of tilting on liquid hold-up and gas-liquid areas. They proposed ways to avoid the disturbance of mass transfers. Cullinane et al. (2011) studied the impact of wave motion on packed tower performance instead of on heat exchanger performance. They reviewed major published studies, both theoretical and experimental, and addressed the effect of the tile and motions in the static and dynamic states.

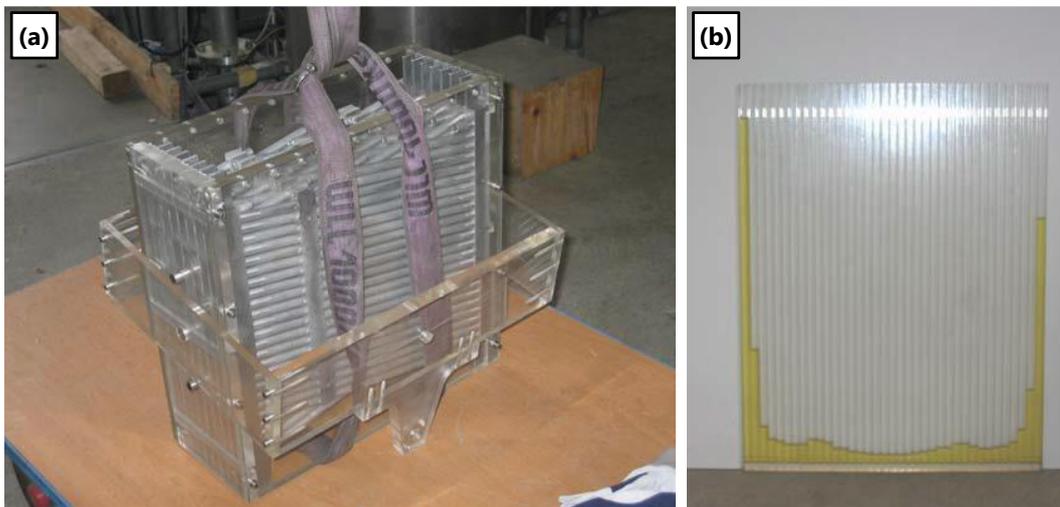


Figure 1-10 Experiment for measuring the effect of tilting and movements on the performance of the heat exchanger: (a) test section of the heat exchanger and (b) liquid collection system below the test section (Lex et al., 2007)

2. Automatic Generation of Logical Models of Liquefaction Cycles Based on Ontological Modeling

The design of the LNG FPSO liquefaction cycle is a procedure for finding the optimal synthesis of the liquefaction cycle given various design criteria. To find the optimal synthesis, various liquefaction cycles should be configured and reviewed vis-a-vis the design criteria. Thus, it is necessary to more simply formulate various configurations of the liquefaction cycles and to easily modify the configurations by adding, rearranging and eliminating equipment until the optimal liquefaction cycle is found. In this chapter, a method based on ontological modeling is proposed for automatic generation of configurations of liquefaction cycles and for their simple modification. The rules for the configuration of the liquefaction cycles were extracted from existing conventional liquefaction cycles that are used in onshore plants, and the configuration of the liquefaction cycles was automatically generated using the SES.

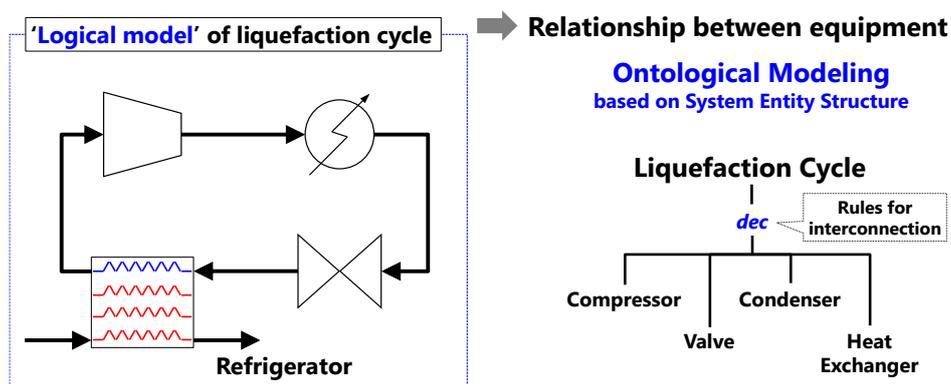


Figure 2-1 Representation of the logical model of a liquefaction cycle using the SES

2.1. Conventional Liquefaction Cycles Used in Onshore

Plants

The LNG industry has a 45-year history, starting with the construction of the Camel plant in Algeria in 1964 (Barclay and Denton, 2005). In the 1970s, the Air Products and Chemicals Inc. (APCI) improved the liquefaction process, and this improved process continues to be used today. Since interest in LNG grew in the last decade, however, there have been considerable developments and diversification of liquefaction processes.

Liquefaction processes can generally be divided into three basic conditions: the number of cycles, the turbine-based cycle and the use of a mixed refrigerant (Hwang, 2013).

The number of cycles is the number of working recycles in which the refrigerant was used for the precooling, liquefaction, and subcooling. If the liquefaction cycle has two recycles for the precooling and liquefaction, it means “two cycles” or “dual cycles.”

The turbine-based cycle is a cycle with an expander. The expander in this cycle generates work that is recovered by the turbo compressor. This cycle can be compensated for by a simple configuration, a low equipment count, and ease of startup and shutdown. The expander can expand less refrigerants than can the expansion valve, however, and cannot expand the liquefied refrigerant. Thus, for large-scale LNG liquefaction, the expansion valve is used instead of the expander.

The use of a mixed refrigerant means the cycle is operated with a mixed refrigerant instead of a pure refrigerant like nitrogen or methane. The mixed refrigerant has combined hydrocarbon components. In most gas industries, refrigerants are mixtures of propane (C_3), ethane (C_2) and methane (C_1), and often have a small amount of nitrogen (N_2). The

proportion of each component can be adjusted according to the composition of each gas field. Such a mixture evaporates over a temperature trajectory rather than at a constant evaporation point. This has significant benefits to the total process. The refrigeration effect will be distributed over a range of temperatures, and accordingly, the overall difference between the temperatures of the natural gas and the mixed refrigerant will be small. This will lead to smaller driving temperature differences and a more efficient liquefaction process.

Many conventional liquefaction cycles are used in onshore plants. In this thesis, six representative liquefaction cycles are introduced in the next section.

2.1.1. DMR Cycle

This cycle is categorized by the following terms.

- Dual cooling cycles
- Without a turbine
- Use of mixed refrigerants

Mixed-refrigerant technology has been assessed for offshore liquefaction based on both the SMR and DMR cycles. The DMR cycle consumes less power but is more complex, as shown in Figure 2-2. It is based on dual cooling cycles with mixed refrigerants. Each cycle has multiple compressions and regenerations to enhance the efficiency of the liquefaction cycle.

Shell, a multinational oil company that is among the largest oil companies in the world, has concluded that the single mixed refrigerant is suitable for smaller LNG production capacities of approximately 2 MTPA, whereas the dual mixed-refrigerant cycle is suitable for capacities of up to 5 MTPA (Bliault, 2001). A dual mixed-refrigerant process forms

part of Shell's onshore Sakhalin Island liquefaction project, with a capacity of 4.8 MTPA per train.

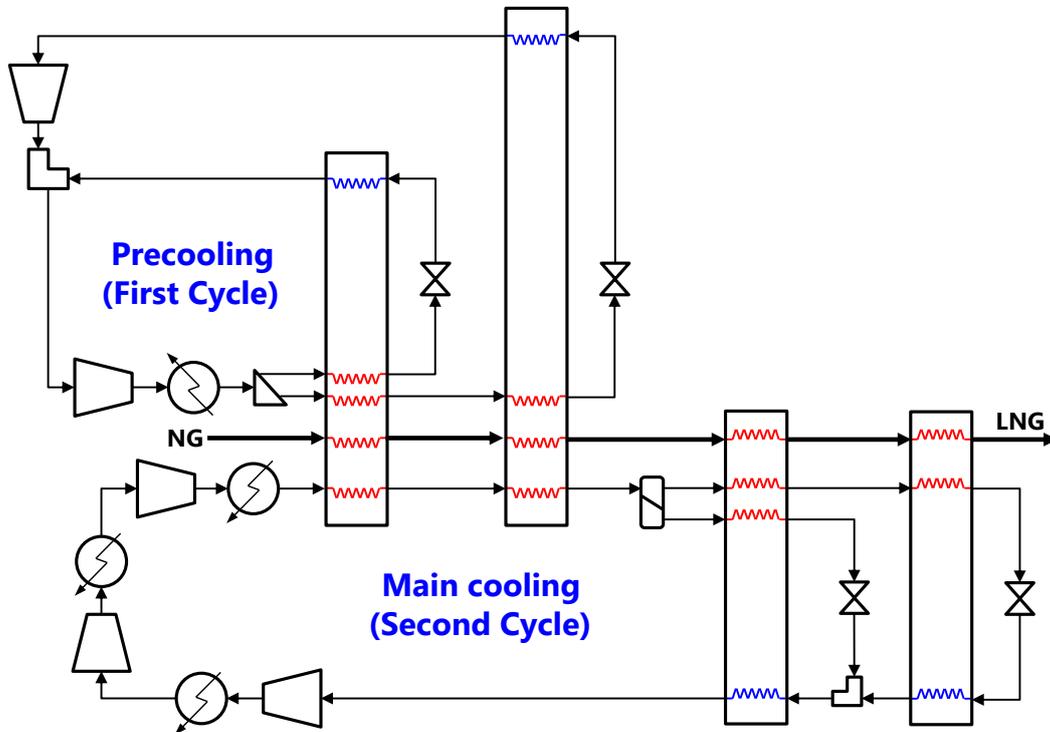


Figure 2-2 Configuration of the DMR (Dual Mixed-Refrigerant) cycle

2.1.2. C3MR Cycle

This cycle is categorized into the following terms.

- Dual cooling cycles
- Without a turbine
- Use of a pure refrigerant for precooling and a mixed refrigerant for liquefaction

The C3MR cycle is the most commonly used process in baseload onshore liquefaction plants. Its configuration is shown in Figure 2-3. As the figure shows, the C3MR cycle is composed of dual cooling cycles. The precooling cycle uses a pure propane refrigerant, and

the liquefaction cycle uses a mixed refrigerant, a combination of hydrocarbon components. The precooling cycle has triple compressions and triple expansions with heat regeneration. The liquefaction cycle has only one compression and dual expansions with heat regeneration.

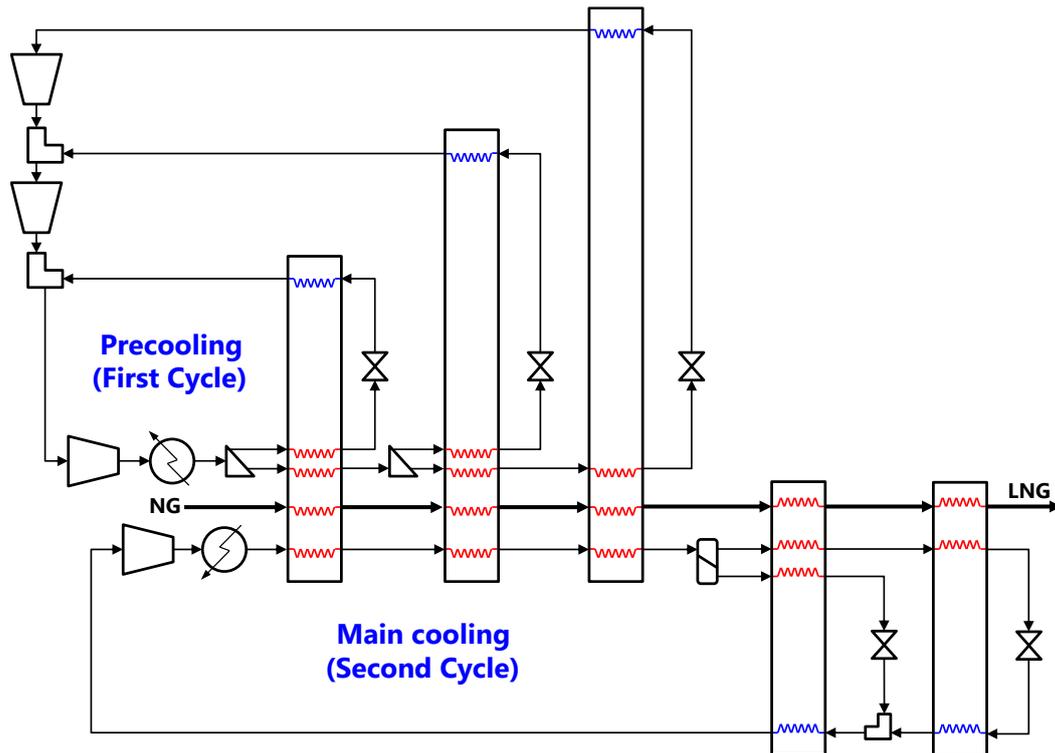


Figure 2-3 Configuration of the C3MR cycle

2.1.3. Cascade Cycle

This cycle is categorized into the following terms.

- Triple cooling cycles
- Without a turbine
- Use of pure refrigerants

In this cycle, natural gas is cooled, condensed and subcooled in heat exchange with

propane, ethylene (or ethane) and finally, methane, in three discrete stages (Wood et al., 2007). Each refrigerant circuit normally has three or four refrigerant expansion and compression stages. After the compression, propane is condensed with cooling water or air; ethylene, with evaporating propane; and methane, with evaporating ethylene. This cycle was typically used in onshore plants before the 1980s, and is currently used for large-capacity LNG production.

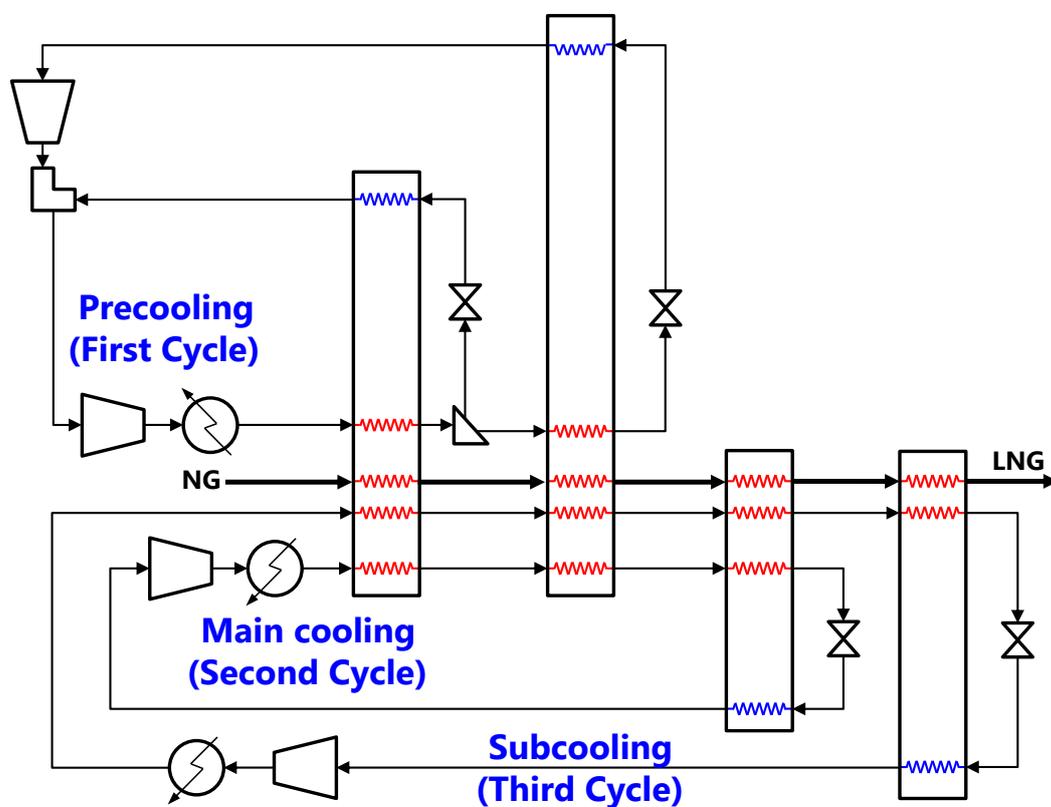


Figure 2-4 Configuration of the cascade cycle

2.1.4. Nitrogen Expander Cycles

This cycle is categorized into the following terms.

- Single cooling cycle

- With a turbine
- Use of a pure refrigerant

In its simplest form, refrigeration is provided through compression and work expansion of a single component gas stream. High-pressure-cycle gas is cooled in counter-current heat exchange with the returning cold-cycle gas. At an appropriate temperature, the cycle gas is work-expanded to reduce its temperature via its expansion through a Joule-Thomson valve. Useful work is generated and usually recovered through a booster-compressor brake, which supplements the main cycle compressor.

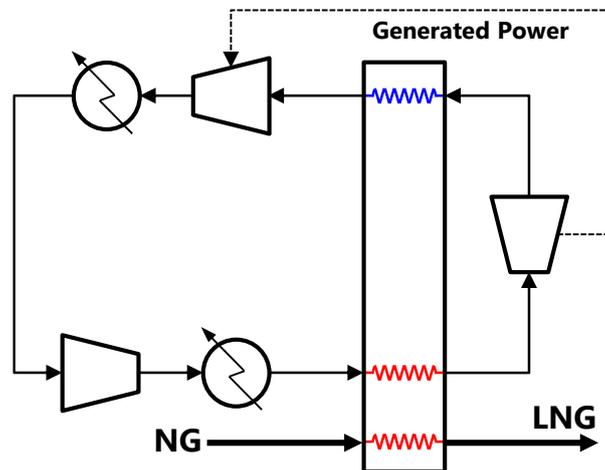


Figure 2-5 Configuration of the nitrogen expander cycle: N_2 single expander cycle

The cold low-pressure gas stream from the expander is returned through various stages of heat exchange where its refrigeration is given up to the incoming natural gas and the incoming cycle gas. The warmed cycle gas is recompressed by the main cycle compressor and the booster-compressor. The refrigerant cycle gas can be either methane or nitrogen. Nitrogen permits subcooling of the natural gas to temperatures low enough to eliminate flashing at the letdown of the LNG. The nitrogen expander cycle is categorized into how many expanders are used in the cycle, as shown in Figure 2-5 and Figure 2-6.

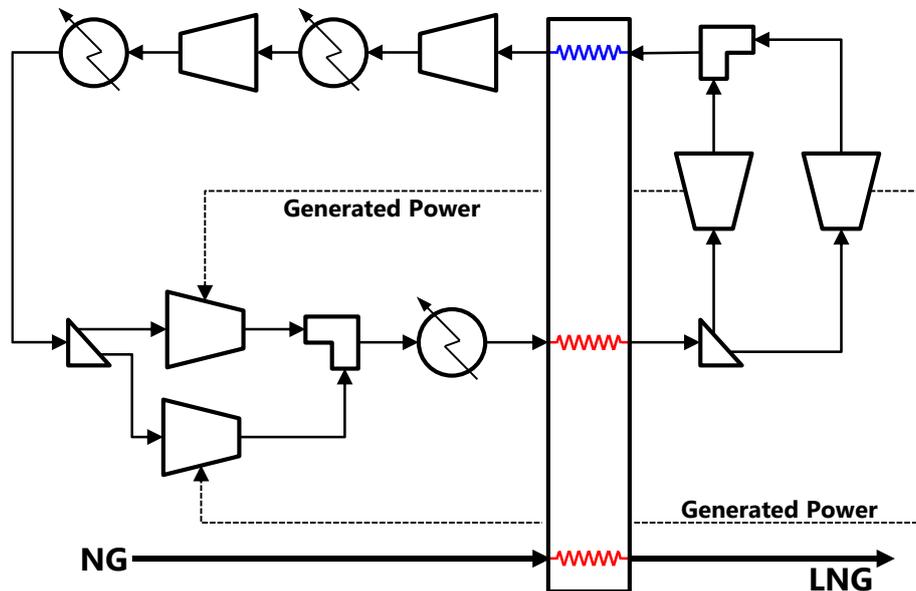


Figure 2-6 Configuration of the nitrogen expander cycle: N₂ dual expander cycle

2.1.5. Niche cycle

This cycle is categorized into the following terms.

- Dual cooling cycles
- With turbines
- Use of pure refrigerants

Using refrigerants other than nitrogen, like methane, or using a mixture of gaseous refrigerants (e.g., nitrogen and helium), can also improve the thermal efficiency. This cycle is similar to the N₂ expander cycle. It has dual cooling cycles with expanders. One cycle uses methane as a refrigerant, and the other, nitrogen.

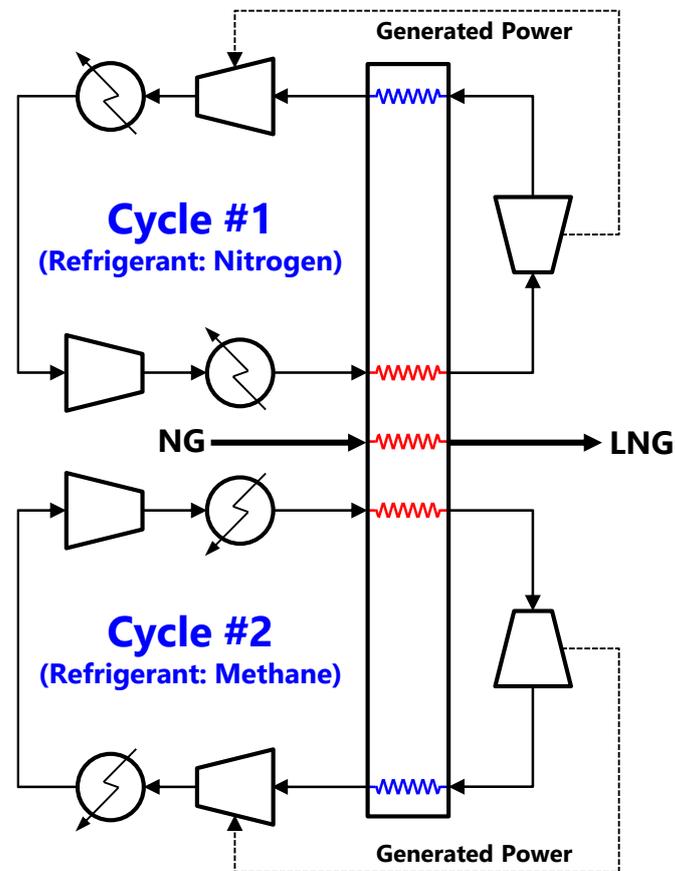


Figure 2-7 Configuration of the Niche cycle

2.1.6. PRICO (Poly-Refrigerant Integrated Cycle Operations) Cycle

This cycle is categorized into the following terms.

- Single cooling cycle
- Without a turbine
- Use of a mixed refrigerant

The PRICO process is a common LNG production process, and is developed and marketed by Black and Veatch Company. Figure 2-8 shows its configuration. It is the simplest form of the mixed-refrigerant cycle and has a single cooling cycle. It consists of

six equipment parts: a compressor, a condenser, a heat exchanger, two expansion valves, and a phase separator. The mixed refrigerant is compressed and passes through the main heat exchanger, where it is condensed. Then it is expanded across an expansion valve (Joule-Thomson valve) and evaporated as it returns counter-currently through the heat exchanger and back to the compressor.

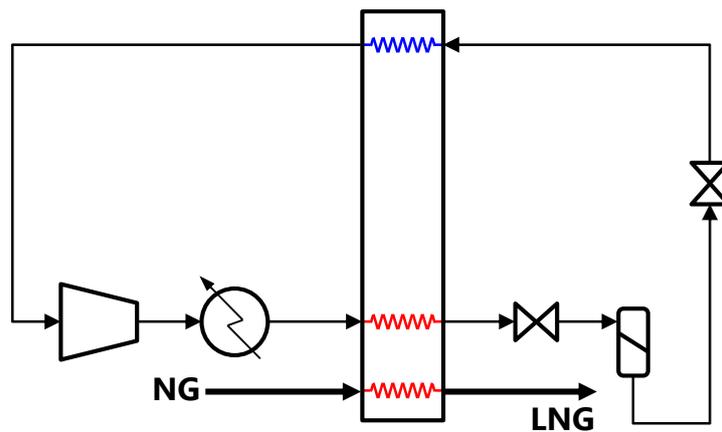


Figure 2-8 Configuration of the PRICO cycle

2.2. General Rules for Combining Liquefaction Cycle

Equipment

An expert making a flowsheet of the liquefaction process should follow certain rules. For example, after the refrigerant is compressed using the compressor, as shown in Figure 2-9, the temperature of the refrigerant from the outlet will depend on the type of process in which the temperature is changed. In adiabatic change, where no change in entropy occurs, the temperature of the refrigerant from the outlet will increase, unlike the temperature of the refrigerant in the inlet. Thus, the condenser may follow the compressor to cool the refrigerant.

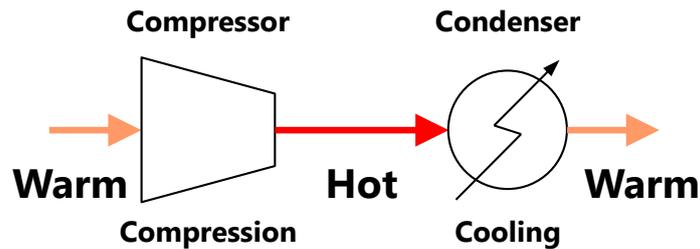


Figure 2-9 Sample rule for combining liquefaction cycle equipment: the compressor and the condenser

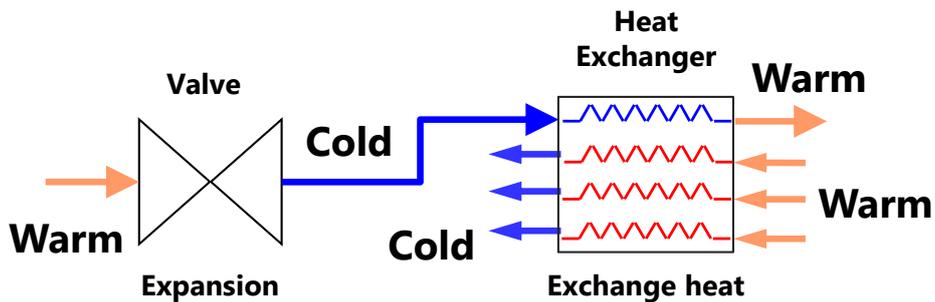


Figure 2-10 Sample rule for combining liquefaction cycle equipment: the valve (expander) and the heat exchanger

Similarly, after the refrigerant is expanded using an expansion valve, as shown in Figure 2-10, the temperature of the refrigerant from the outlet of the expansion valve will be lower in isobaric change, wherein the pressure is constant. Thus, the refrigerant that was cooled after the expansion can be used to cool the warm streams in the heat exchanger.

These rules are based on the experience of the expert, and are the specifications of liquefaction cycles that can be used when the engineer synthesizes a liquefaction cycle. Thus, some rules for synthesizing the liquefaction cycle were extracted from existing onshore liquefaction cycles.

2.3. Multiple Cooling Cycles

As mentioned in Section 2.1, the liquefaction cycle can have multiple cooling cycles (Barclay and Shukri, 2000). The number of cycles is a key factor in the success of the liquefaction process. The cycle takes warm pretreated feed gas and cools and condenses it into an LNG product. All modern baseload liquefaction facilities use either two or three cycles, or rarely one cycle. A high-level representation of the number of cycles in various liquefaction processes is shown in Figure 2-11.

Each cycle has its own refrigerant. There are two kinds of refrigerants: pure refrigerants and mixed refrigerants. Pure refrigerants like nitrogen and propane are composed of only one material, but mixed refrigerants are combinations of hydrocarbon and nitrogen.

Mixed refrigerants evaporates over a temperature trajectory instead of at a constant evaporating point of pure refrigerants. This refrigeration effect will be distributed over a range of temperatures and accordingly the overall temperature difference between the natural gas and mixed refrigerant is small. This leads to smaller driving temperature

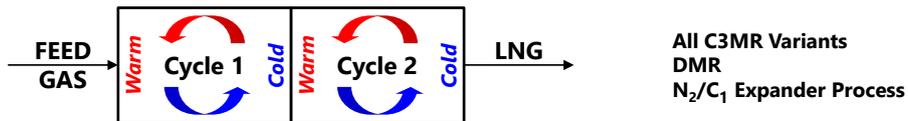
differences and gives a more efficient liquefaction process than pure refrigerants.

On the other hand, the foremost advantage of pure refrigerants compared to mixed refrigerants is simplicity. In the cycle using pure refrigerants, the equipment count is low, the configuration is simple and there is no phase change of the refrigerant. Disadvantages include a higher energy demand per produced tons and a large footprint for a large production.

(a) Single Cycle



(b) Dual Cycle



(c) Triple Cycle

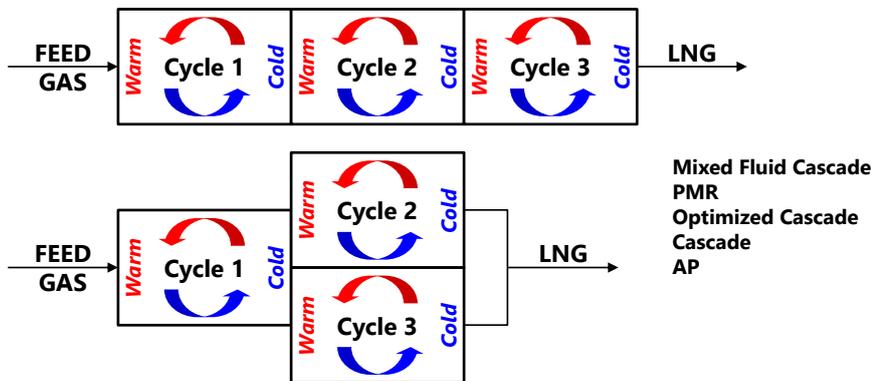


Figure 2-11 High-level representation of the number of cycles in various liquefaction processes (Barclay and Shukri, 2000)

2.3.1. Basic Stage and Additional Stages

To cool and condense natural gas, all the cycles should have four basic equipment: a compressor, a condenser, an equipment for expansion (an expansion valve or an expander) and a heat exchanger. This combination is called the *basic stage*, and it is necessary for all cooling cycles.

To enhance the efficiency of the liquefaction cycle, the engineer can add equipment to it or modify its configuration. There are many types of additional formal stages for enhancing such efficiency, such as *compression refrigeration*, *compression with an intercooler* and *refrigeration*. The compression refrigeration stage is based on multiple expansions and multiple compressions, so five pieces of equipment must be added in this stage. The compression with an intercooler stage is based on multiple compressions and multiple cooling, and the refrigeration stage is based on multiple expansions with phase separation of the refrigerant.

2.3.2. Interconnection of Liquefaction Cycle Equipment

To cool the refrigerant and the natural gas, the liquefaction cycle equipment must be interconnected. The liquefaction cycle is extended by interconnection laws, which describe typical physico-chemical phenomena in liquefaction process systems. In this section, the interconnection laws are extracted from existing conventional liquefaction cycles and categorized based on thermodynamic phenomena such as compression and expansion.

(1) Compression

Since a compressor with a large compression capacity is expensive, the combination of smaller-capacity multiple compressors is often used in the liquefaction process system.

Compressions can be connected in parallel or in series. If the compressors are placed in parallel, a tee or a phase separator may separate the inlet flow, and a common header may merge the outlet flows from these compressors (see the Niche cycle in Figure 2-7). In case of serial connection, a condenser should follow the compressor.

(2) Equipment after compression

Compression reduces volume. Based on thermodynamics, if there is a change in pressure, the volume and temperature of the material will also change. During compression, the temperature of the refrigerant rises, so a mechanism for decreasing the temperature is needed. To directly cool the temperature, the condenser can be placed next to the compressor. Another option for cooling the temperature is mixing the stream with a lower temperature. Thus, a common header can be placed next to the compressor, and this equipment will cool the temperature by mixing the streams with different temperatures. If there are parallel compression like the Niche cycle, a common header can be also placed next to these compressors to merge multiple flows from the compressors.

The rules related to compression are as follows.

- A condenser can be placed after a compressor.
- A common header can be placed after a compressor, but the stream with lower temperature from a heat exchanger or other expander should be joined with the stream from the compressor at the common header.

(3) Equipment after cooling

Even if a material is cooled using a cooling equipment, it cannot be used as a refrigerant because its temperature is higher than the boiling point of natural gas. Thus, the expansion procedure is needed to lower the temperature, which is why the compressor is always

placed in front of the condensing equipment. There are two kinds of cooling equipment: a condenser and a heat exchanger. The expansion procedure typically follows the use of the heat exchanger. In contrast, after the refrigerator is cooled using a condenser, another compression procedure using a compressor (see the DMR cycle in Figure 2-2) or a heat exchanger (see the N₂ dual expander cycle in Figure 2-6) can follow. In addition, after the mixed refrigerant is condensed using a condenser, it can have two material phases: the liquid and vapor phases. In that case, a phase separator can be used after the condenser. If the pure refrigerant is used for the cooling cycle, there is no two-phase flow, so a tee can be used instead of a phase separator (see the Niche cycle in Figure 2-7).

The rules related to cooling are as follows.

- The expansion equipment can follow the heat exchanger.
- A compressor can be used next if it is necessary to compress the refrigerant.
- A heat exchanger can be used next if a pure refrigerant is needed.
- If the refrigerant is a mixed refrigerant, a phase separator must be used next to separate the flows in the two phases.

(4) Equipment after the refrigerant expansion

After the refrigerant is expanded, its temperature is enough to liquefy the natural gas. Thus, the refrigerant from the expansion equipment enters into a heat exchanger, and it will cool the natural gas and warm the refrigerant of the other flows in the heat exchanger.

If the expander is used for the expansion, the expansion work is extracted from the fluid as shaft power and may be used to drive a centrifugal compressor elsewhere in the process. Thus, the expander is connected to the compressor.

The rules related to expansion are as follows.

- The heat exchanger follows the expansion equipment.
- The expander is coupled with the compressor.
- If there is another flow from the heat exchanger, the flows can be merged using the common header.

(5) Regeneration: Self-cooling

The refrigerant is used to cool not only the natural gas but also the warm refrigerants in the other flows. Regeneration means the refrigerant cools itself in other flows. Thus, in the cooling cycle with regeneration, the refrigerant returns counter-currently through the heat exchanger after it expands, as shown in Figure 2-12.

The rule for regeneration is as follows.

- After the expansion, the refrigerant cools itself by flowing through the heat exchanger.

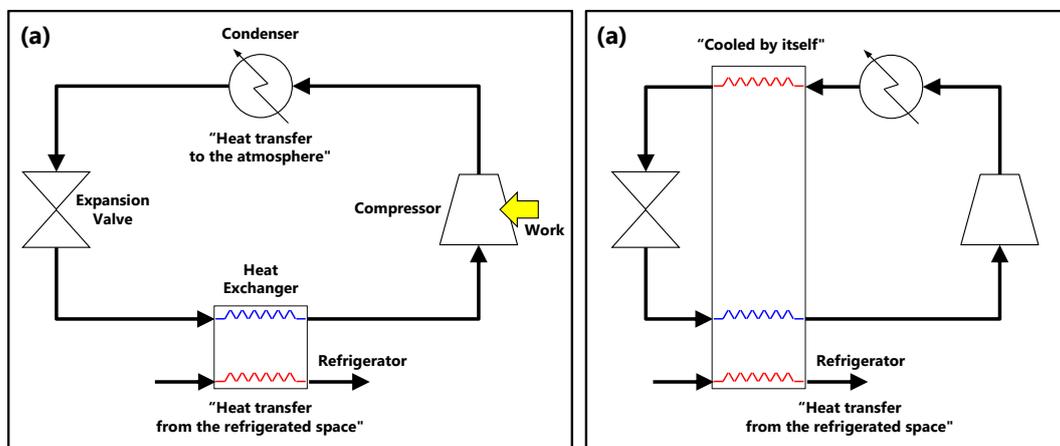


Figure 2-12 Regeneration: (a) basic refrigerator and (b) refrigerator with regeneration

2.4. Ontological Modeling of the Liquefaction Cycle using the SES

Ontology involves explicit specification of a concept. It aims to generally capture consensual knowledge that may be reused and shared by software applications and groups of people. It can be represented with various languages and with different degrees of formality (Morbach et al., 2007). In this thesis, the synthesis of the LNG FPSO liquefaction cycles was focused on as a small part of ontology in process simulation. The SES was adopted and used to express the ontology for the synthesis of the liquefaction cycles.

2.4.1. Logical Model of Liquefaction Cycles: Their Structural Aspect

(1) Hierarchical representation of the liquefaction cycle with the rules for equipment interconnection

Based on the rules in Section 2.2, the engineer can make the liquefaction cycles by interconnecting the equipment. Figure 2-13 shows a hierarchical representation of the liquefaction cycle configuration using the tree structure. The logical model on the left side of the figure has related liquefaction cycle equipment. These can be expressed as the tree structure shown on the right side of the Figure 2-13.

The parent entity, called the liquefaction cycle, has four child entities: the compressor, valve, condenser, and heat exchanger. The interconnection rules are defined in the *dec* node between the parent and the child. This is a simple example of hierarchical representation, but all liquefaction cycles can be expressed like this, using the tree structure and the interconnection rules. Moreover, this method of representing the logical model of liquefaction cycles is similar to the Entity Relationship (ER) model, which is an abstract of

how to describe a database in software engineering.

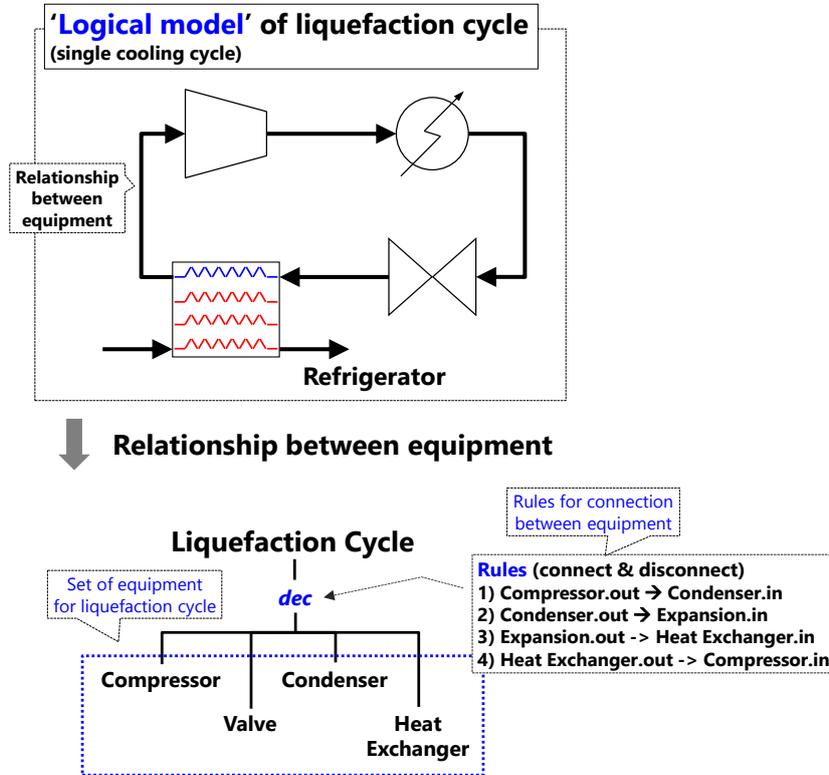


Figure 2-13 Hierarchical representation of the liquefaction cycle configuration using the tree structure

(2) Variation of the liquefaction cycle by adding, rearranging and eliminating equipment

① Adding new equipment

As mentioned in the previous paragraph, the logical model of liquefaction cycles can be represented by the tree structure with the interconnection rules. This method is similar to the ER model, and the logical model can be easily varied by adding, rearranging and eliminating equipment and by modifying the rules in the tree structure.

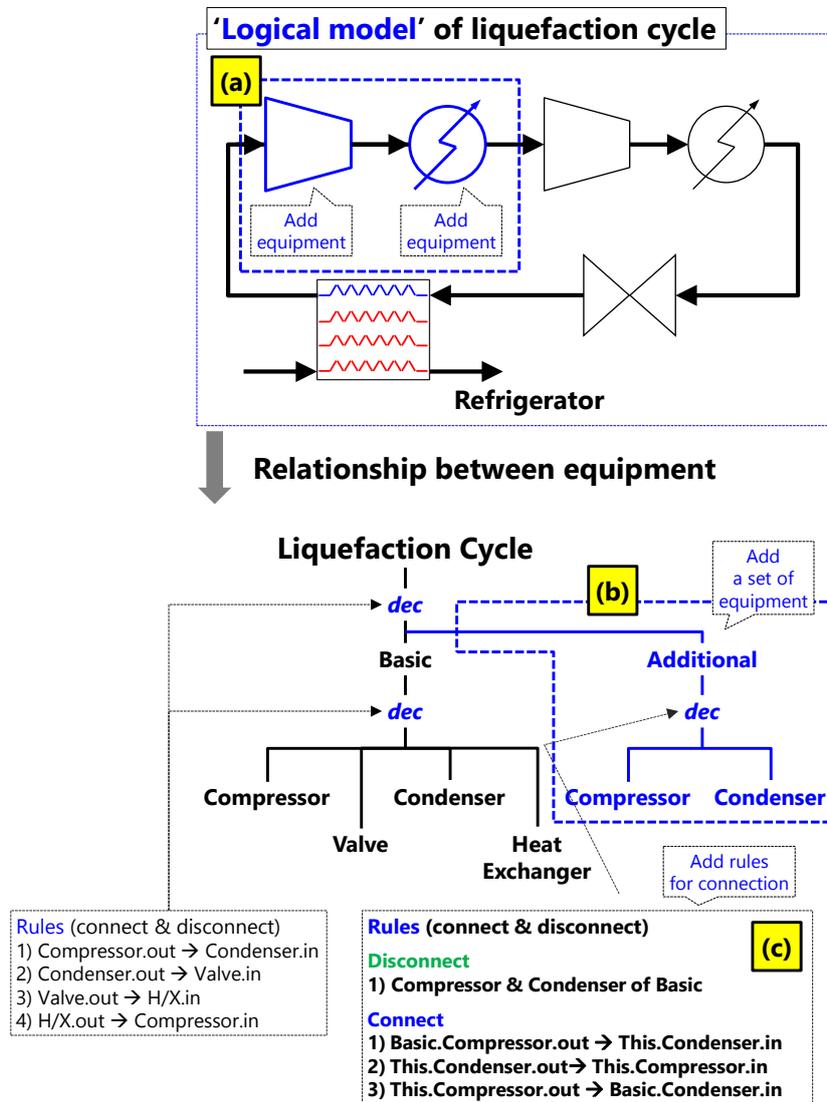


Figure 2-14 Variation of the logical model for the liquefaction cycle: adding new equipment and rules

Figure 2-14 shows an example of the variation of the logical model by adding new equipment and rules. Compared with the logical model in Figure 2-13, the logical model in Figure 2-14 (a) has a compressor and a condenser added in front of the existing compressor. Thus, the compressor and the condenser are also added to the tree structure in

Figure 2-14 (b). Before adding the two equipment parts, they were divided into two groups: the existing and the new. The group of existing equipment was named *Basic*, and the group of new equipment, *additional*.

The rules for interconnecting equipment were also added to the *dec* node below the *additional* node. As shown in Figure 2-14 (a), the existing compressor and the heat exchanger should be disconnected before adding new equipment. Thus, this procedure is added to the rules and called *disconnect* in Figure 2-14 (c). Then the rules for interconnecting new and existing equipment are added to the rules and called *connect*.

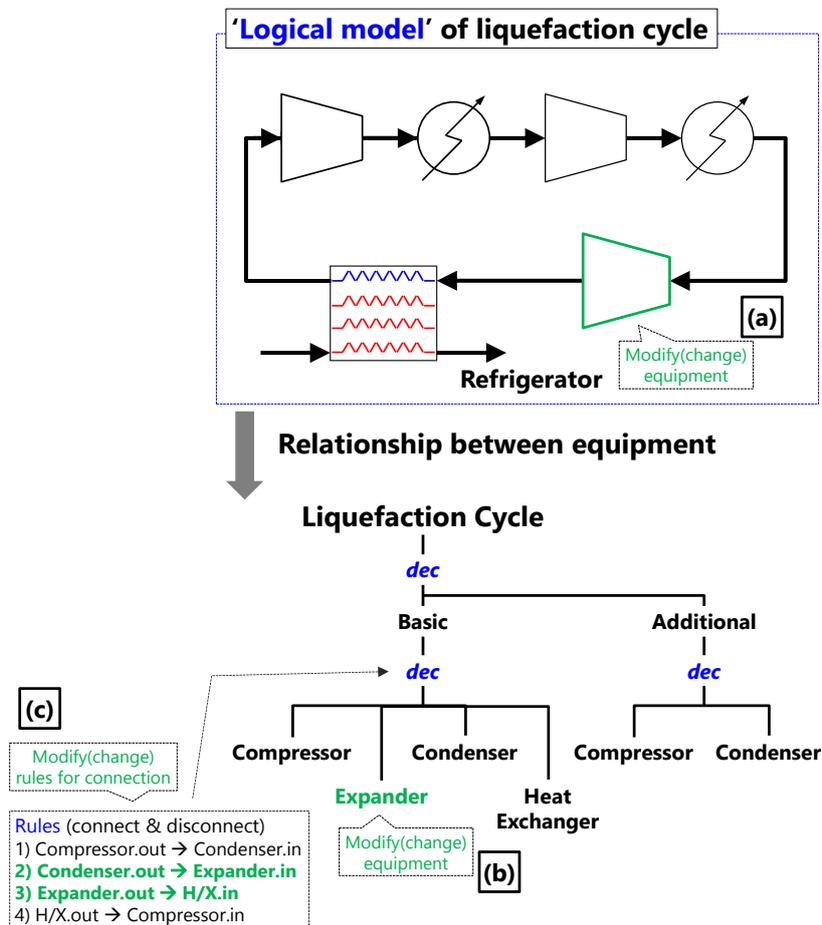


Figure 2-15 Variation of the logical model for the liquefaction cycle: equipment change

② Modifying (rearranging) the existing equipment

The logical model of the liquefaction cycle can be modified by exchanging it with other equipment and rearranging the existing equipment. Figure 2-15 (a) shows an example of the variation of the logical model in Figure 2-13 by exchanging the expansion valve with the expander. To do so, the expansion valve entity is merely exchanged with the expander in the tree structure as shown in Figure 2-15 (b). Then the interconnection rules in the *dec* node should be changed to modify the interconnection, as shown in Figure 2-15 (c).

③ Adding and rearranging equipment

Figure 2-16 shows the sequence of the steps in making the PRICO cycle from the refrigerator by adding and rearranging the equipment. To make the PRICO cycle, an expansion valve and a phase separator are first added to the liquefaction cycle (Figure 2-16 (a)). Then the condenser and the expansion valve, and the compressor and the heat exchanger, of the refrigerator are disconnected (Figure 2-16 (b)). By connecting the newly added equipment to the existing equipment (Figure 2-16 (c)), the logical model of the PRICO cycle can be made, as shown in Figure 2-16.

This procedure is reflected in the tree structure, which represents the hierarchical relationships of the equipment. The *additional* node, which has an expansion valve and a phase separator, is added to the tree structure of the liquefaction cycle. Then the disconnection and connection rules should be added to the *dec* node below the *additional* node, as shown on the right side of Figure 2-16 (d).

The procedure for eliminating equipment is similar to that for adding equipment. The existing node like the *additional* node, which has a set of child equipment, is eliminated,

and the rules for this node are automatically deleted (Figure 2-16 (e)).

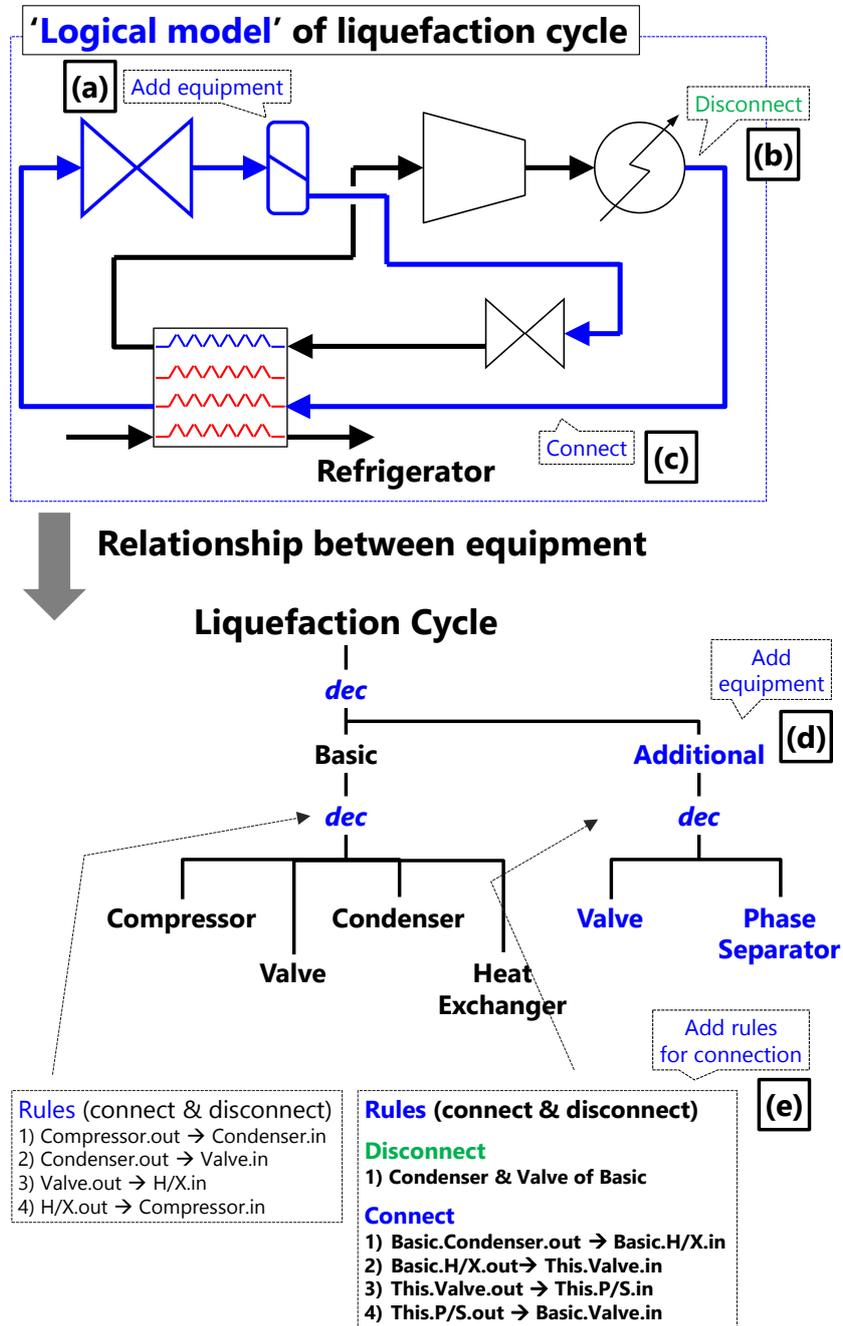


Figure 2-16 Variation of the logical model of the liquefaction cycles: making the PRICO cycle from the refrigerator

2.4.2. SES: An Ontological Framework

Ontology is a study concerned with the nature of existence of things and their relationships. It contains elements, attributes of the elements and relationships between elements with which to represent or model knowledge of a certain domain. There are many methods of representing the ontology of domain knowledge. In this thesis, the system entity structure (SES) is adopted to express the ontology for the liquefaction cycle.

(1) SES

The SES is a formal ontology framework that hierarchically represents the elements of a system and their relationships (Lee and Zeigler, 2010). It provides a model for structurally describing knowledge of a domain. Since it starts with the representation of the model structure, it is easily accommodated in modeling and simulation for automation. While it rigorously represents complex data, it has the flexibility and efficiency needed to change the model structure according to various choices.

Figure 2-17 shows an example of the representation of a human using the SES. Entities such as *Human*, *Head* and *Legs* represent things that exist in a certain domain. They can be independently identified or postulated as components of the decomposition of other real-world objects (entities). They can also have variables to which values can be assigned within a given range and with certain types.

Decomposition, noted as *dec* in Figure 2-17, represents ways of taking things apart into more detailed components and labeling the decomposition relationship of the parent to its children. The children of decomposition are entities that represent components of the decomposition of their parent. For example, *Human* is decomposed into *Sex*, *Body*, *Mind*,

etc. These are necessary components of humans, and they mean that the entities, as parts of the decomposition, are necessary subsystems of the parent. The decomposition entities represent distinct components of the decomposition. A model can be constructed by interconnecting some or all of these components.

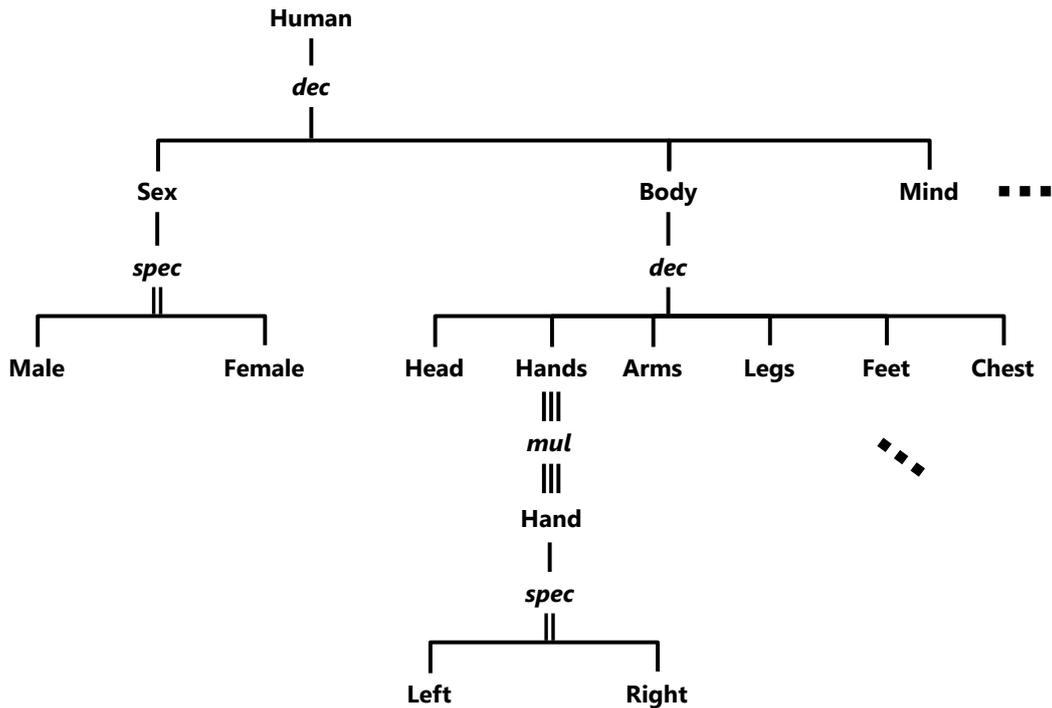


Figure 2-17 Representation of a human using the SES

Multiple decomposition, which is noted as *mul* in Figure 2-17, is a kind of decomposition with the same type of components. It is used to represent entities whose number in a system may vary. For instance, *Human* can have less than two hands, so the entity *Hands* can be decomposed into individual *Hands* using multiple decomposition, as shown in Figure 2-17.

Specialization, noted as *spec* in Figure 2-17, categorizes things into the specific forms that they can assume. It is a labeled relationship that expresses alternative choices of a

system entity. The children of a specialization are entities that represent variants of their parent. For example, as shown in Figure 2-17, the entity *Sex* could have specializations like *male* and *female*. These are variants of the *Sex* of a *Human*.

Decomposition and multiple decomposition express the subsystem of a system, so they have the rules for relationships between their subsystems. In Figure 2-17, *Human* is decomposed of *Sex*, *Body*, *Mind*, etc. Thus, the *dec* node below a *Human* entity has some rules for connecting the child entity, like *Sex*, *Body* and *Mind*. On the other hand, specialization represents variants of the parent, so it does not have rules for interconnecting the children. The children have their own specific values, however, to distinguish them from other specialization variants. In this thesis, decomposition, multiple decomposition, and specialization are called *nodes*.

(2) Pruning the SES

A pure entity structure is one with no specializations and at most, one decomposition hanging from each entity. It is created via pruning, which is the SES operation that changes the model structure to extract specific information. It cuts off the unnecessary structure in the SES based on the specification of a pragmatic frame. It includes these processes: (a) assignment of particular values to the entity variables, and (b) trimming of the SES and determination of the minimal SES for end-users by picking specific elements from multiple choices. The pruning process reduces the choices in specialization. After its completion, there should be no choice left in the aforementioned relations. Moreover, at the implementation level, a pragmatic frame can choose anything in the ontology. The result of pruning is a pruned entity structure (PES) that contains no specialization but the original SES and therefore, specifies a smaller family of alternative models.

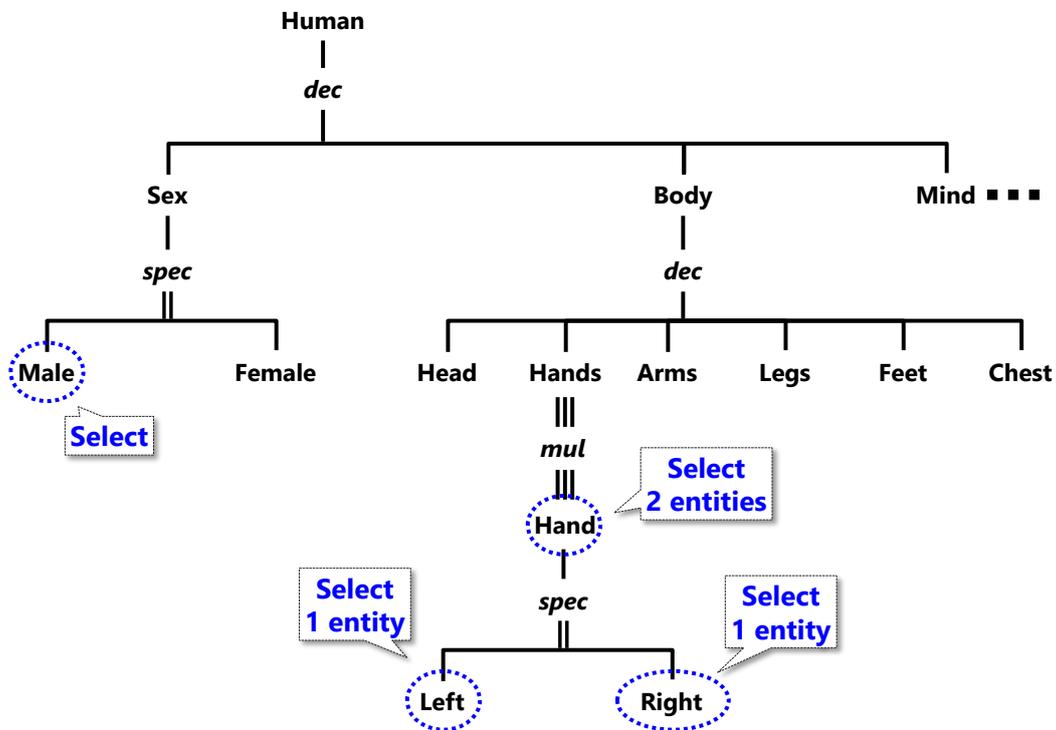


Figure 2-18 Pruning of the SES that represents a human

Figure 2-18 shows the pruning procedure. Since the entity *Sex* was categorized into two types, a kind of *Sex* can be chosen, so the entity *Male* is chosen in this example. The entity *Hands* can have multiple *Hand* entities because it has the multiple decomposition node. Thus, two *Hand* entities can also be selected as children of the entity *Hands*. Each *Hand* can be assigned to the *Left* or *Right* via specialization. After the choices shown in Figure 2-18 are completed, the PES shown in Figure 2-19 is achieved.

Because of the nodes like the specialization and multiple decomposition nodes, the pruning process can make various alternative models from the SES of a certain domain. In addition, if there is a method of choosing the child entities of the specialization and multiple decomposition nodes, the pruning process can automatically generate alternative models.

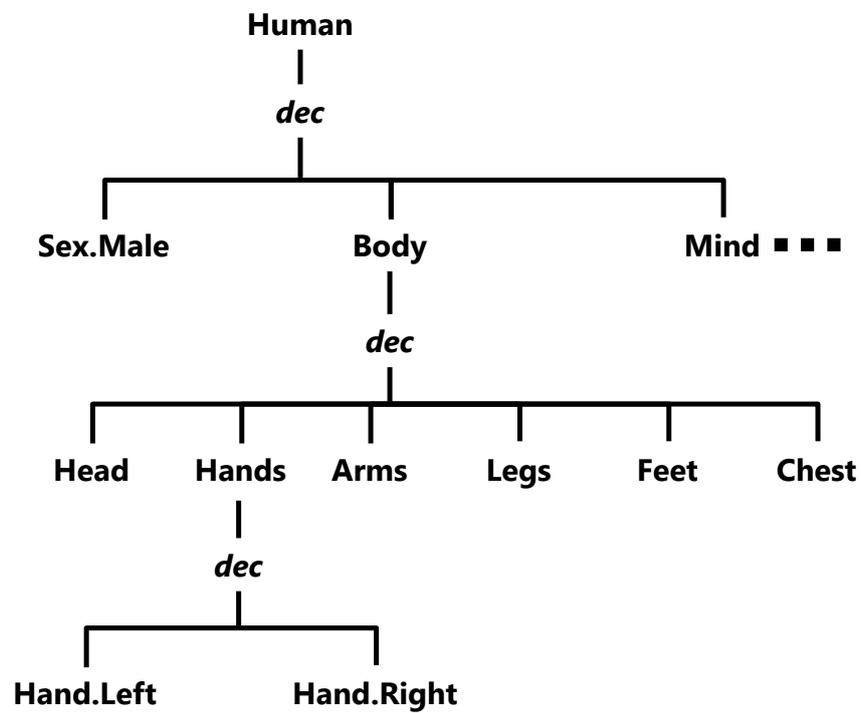


Figure 2-19 PES after finishing the pruning process

(3) Model base (MB)

Each system is described by two aspects: its structural and functional (behavioral) aspects (Dahmani and Hamri, 2011). The SES represents the structural aspect, and the model base (MB), the behavioral aspect of the entities in the SES. Let the hierarchical composition structures of the hierarchical and modular models first be extracted from their implementation. Then the structures of a system and the implementation of each subsystem can be saved separately in organized libraries. Figure 2-20 shows the concept of the SES and the MB.

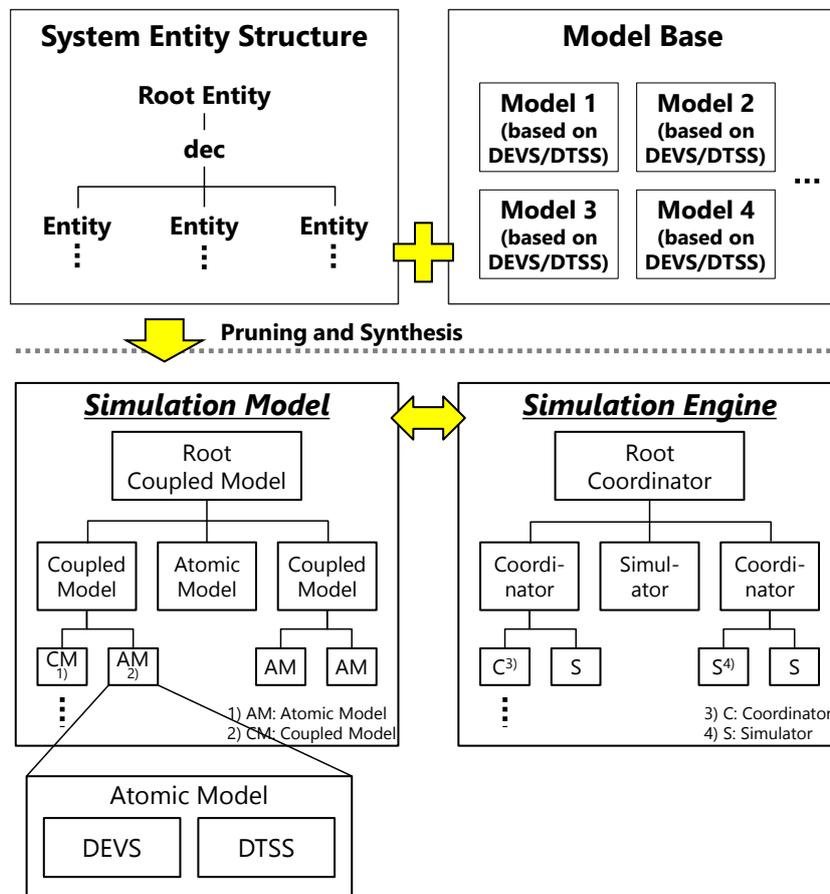
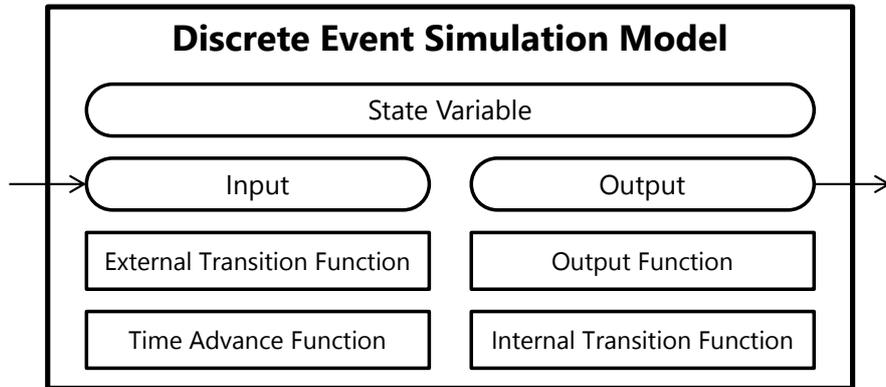


Figure 2-20 SES and MB concepts

The behavioral knowledge of the entities in the MB is modeled based on the DEVS and DTSS formalisms that can handle simulation models of a discrete event and a discrete time (Zeigler et al., 2000). Using these formalisms, a simulation engine can perform a simulation by changing the state of a model according to the events and by calculating the state variables of the model every unit time when the model is in the specific state. These formalisms are widely used as standard formalisms for modeling and simulation.

DEVS (*Discrete Event System specification*)

:model template for discrete event simulation



DTSS (*Discrete Time System Specification*)

:model template for discrete time simulation

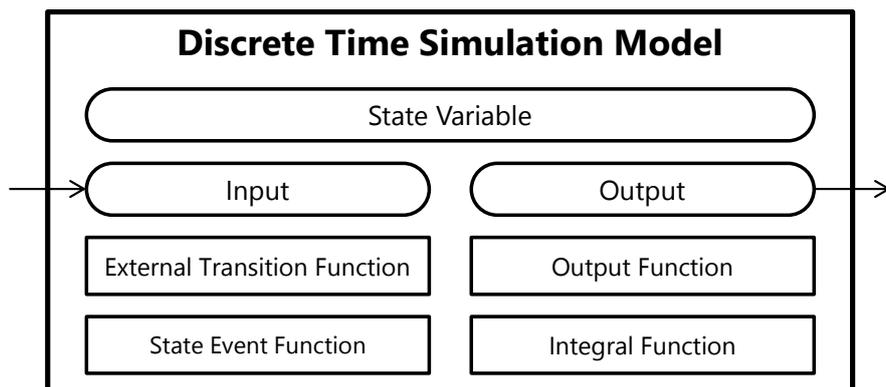


Figure 2-21 DEVSS and DTSS formalisms: configuration of an atomic model

The DEVSS formalism consists of two models: the atomic model and the coupled model. The atomic model is the basic component of the formalism and describes the behavior of the system. Each atomic model has input values, output values, output functions and state variables. The atomic model for discrete event simulation is based on the DEVSS formalism and has the internal transition, external transition, and time advance functions to change its state according to external and internal events. The atomic model for discrete time

simulation is based on the DTSS model and has the integral function of calculating the state variables of the model at each unit time and a state event function to check if the discrete event simulation was completed.

On the other hand, the coupled model provides the method of assembly of several atomic and/or coupled models to build a complex system hierarchy using the SES. Figure 2-22 shows an example of a DEVS coupled model with two atomic models. These basic models are interconnected through the corresponding input (the set X in Figure 2-22) and output (the set Y in Figure 2-22) ports. The models are connected to the external coupled model through the External Input Coupling (EIC) and External Output Coupling (EOC) connectors.

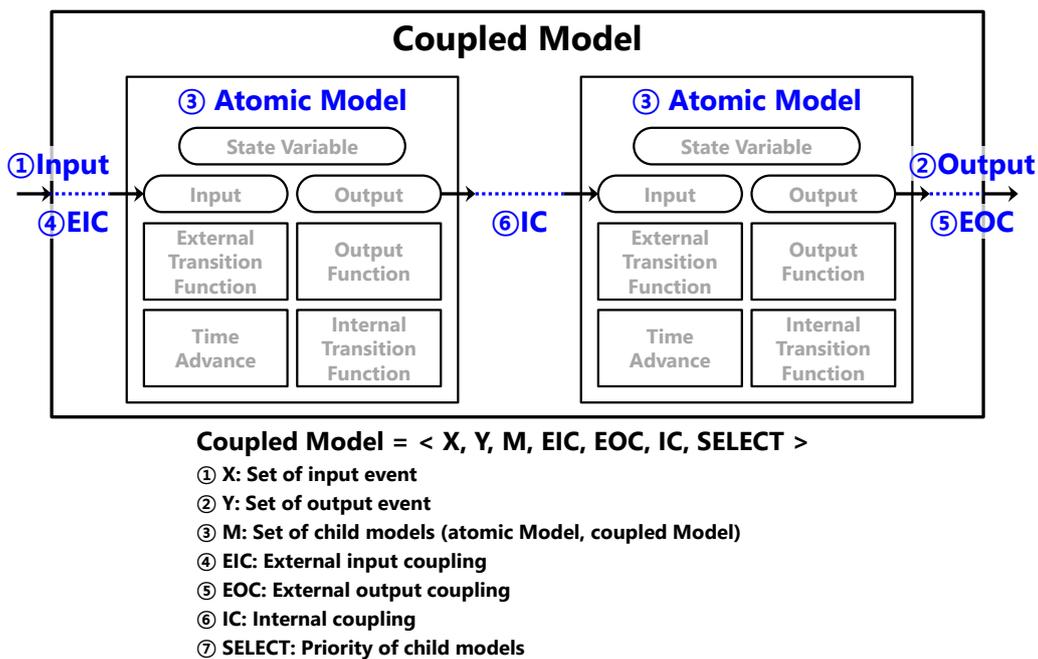


Figure 2-22 Configuration of the coupled model based on the DEVS and DTSS formalisms

The coupled model definition has the specifications of the components (the set M in Figure 2-22), their internal coupling (the set IC in Figure 2-22), and their external couplings

(the sets *EIC* and *EOC* in Figure 2-22). Coupled models group several DEVS into a composite model that can be regarded, due to the closure property, as a new DEVS model. The closure property guarantees that the coupling of several class instances will result in a model of the same class, which will allow hierarchical construction.

Because multiple subcomponents can be scheduled for internal transition at the same time, ambiguity could arise. The *SELECT* function provides a simple way to dispel this ambiguity. The function defines the order of all the components of the coupled model so that only the first model that must be executed in the case of simultaneous internal events can be chosen.

Bang developed a simulation framework based on the combined DEVS/DTSS concepts (Bang, 2006). To evaluate the efficiency and applicability of this simulation framework, Cha et al. (2009) and Ha et al. (2012a and 2012b) applied it to the block erection process in shipbuilding, the dive of a submarine, and an analysis of evacuation of a passenger ship.

(4) Creation of a simulation model by combining the PES and MB

The PES has relationships among its entities and their hierarchies. The entities in the MB represent their behavioral aspect. By synthesizing the PES and the MB, a model of the whole system that represents all the functions in the system can be made. Figure 2-23 shows a male human with two hands generated by the synthesis of the PES and the MB.

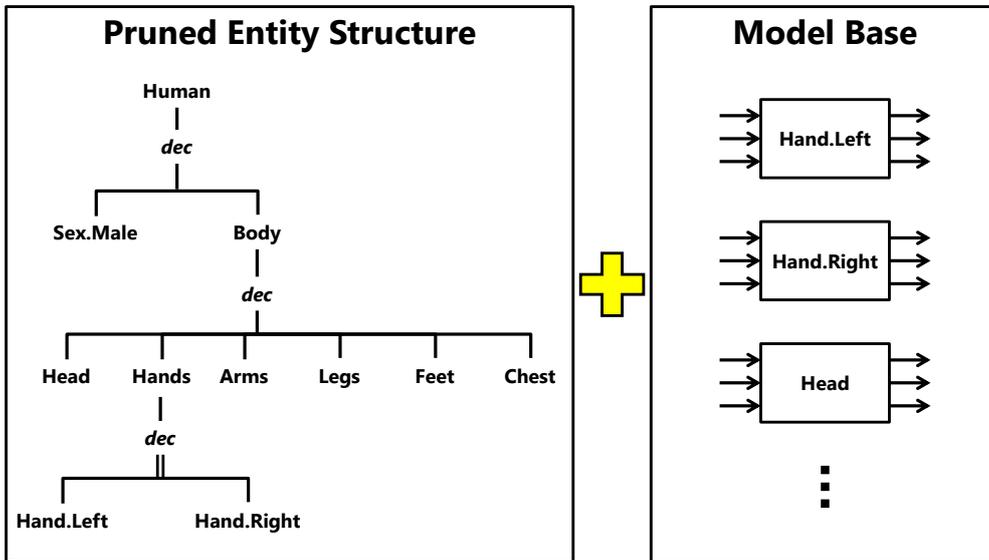


Figure 2-23 Synthesis of the PES and the MB

If the SES for the liquefaction cycle is configured, the logical models of various alternative liquefaction cycles, which have related equipment, can be automatically generated by pruning the SES. Thus, by combining the PES and the MB, all the possible alternatives for the LNG FPSO liquefaction cycle can be presented and evaluated.

2.4.3. Procedure for Making the SES for the Liquefaction Cycle

Based on the rules mentioned in section 2.2, the SES for the liquefaction cycle is proposed in this thesis. The procedure for making this SES is as follows.

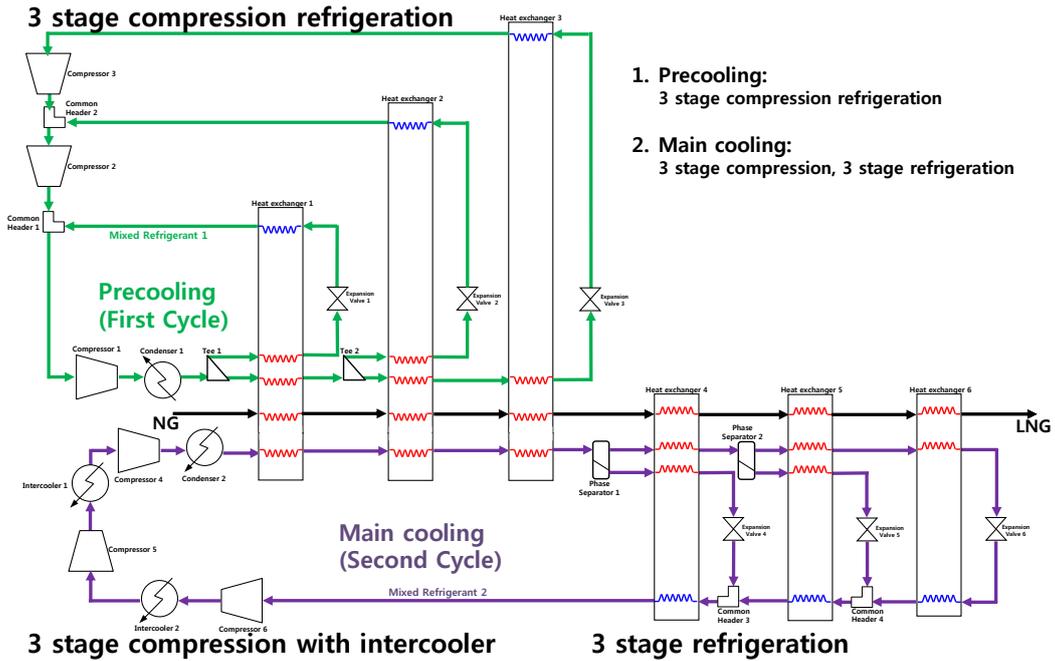


Figure 2-24 Logical model of the typical liquefaction cycle: the dual cooling cycle (precooling: three stages of compression refrigeration, and main cooling: three stages of compression and three stages of refrigeration)

(1) Multiple cooling cycles

Figure 2-24 shows an example of a logical model of liquefaction with dual cooling cycles. As mentioned in Section 2.3, the liquefaction cycle is composed of multiple cooling cycles. These can be expressed using the multiple decomposition of the SES, as shown in Figure 2-25. The entity liquefaction cycle entity is decomposed by the multiple cooling cycles using the multiple decomposition node, which is noted as *mul*.

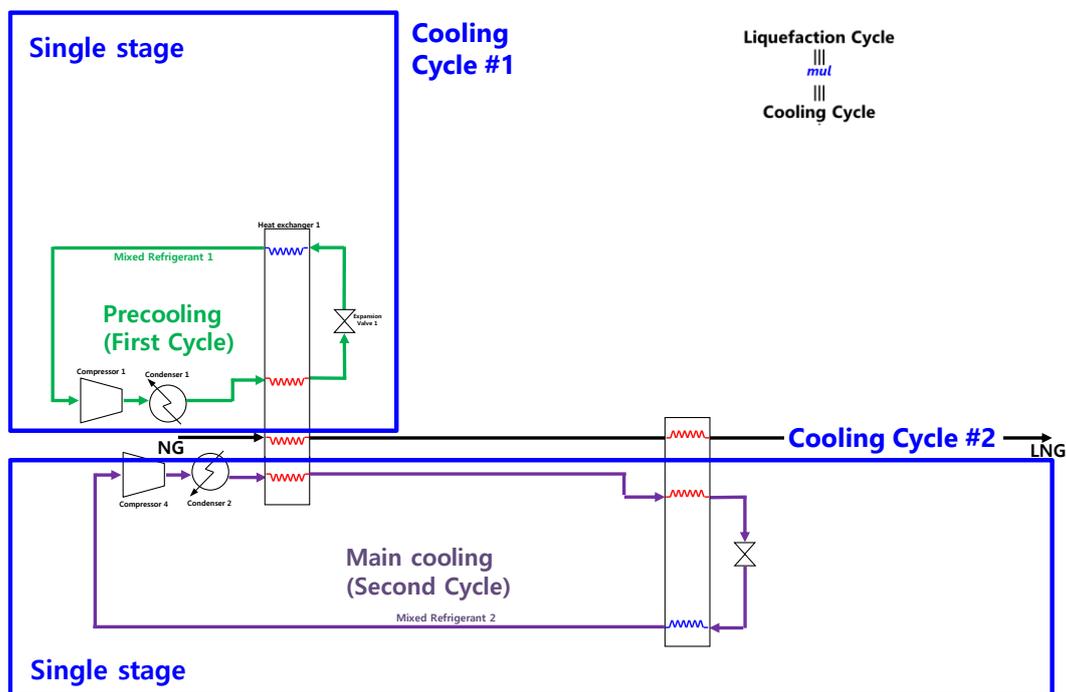


Figure 2-25 Procedure for making the SES for the liquefaction cycle: rules for multiple cooling cycles

(2) Basic stage and additional stages

① Basic stage

Each cooling cycle has four basic equipment: a compressor, a condenser, an expansion equipment (an expansion valve or an expander), and a heat exchanger. This combination is called the *basic stage*, and is necessary for all cooling cycles. Thus, the *basic stage* entity is a child of the *cooling cycle* entity and is decomposed by the four aforementioned equipment. This is expressed using the *dec* (decomposition) node, as shown in Figure 2-26. This node also has the following rules for interconnecting the four basic equipment.

- *Compressor.outlet* is connected to *Condenser.inlet*.
- *Condenser.outlet* is connected to *Expansion Valve.inlet*.

- *Expansion Valve.outlet* is connected to *Heat Exchanger.inlet1*.
- *Heat Exchanger.outlet1* is connected to *Compressor.Inlet*.

If there is a regeneration sequence in the *basic stage*, as shown in Figure 2-26, the connection and disconnection rules should be modified. This will be explained in the next section.

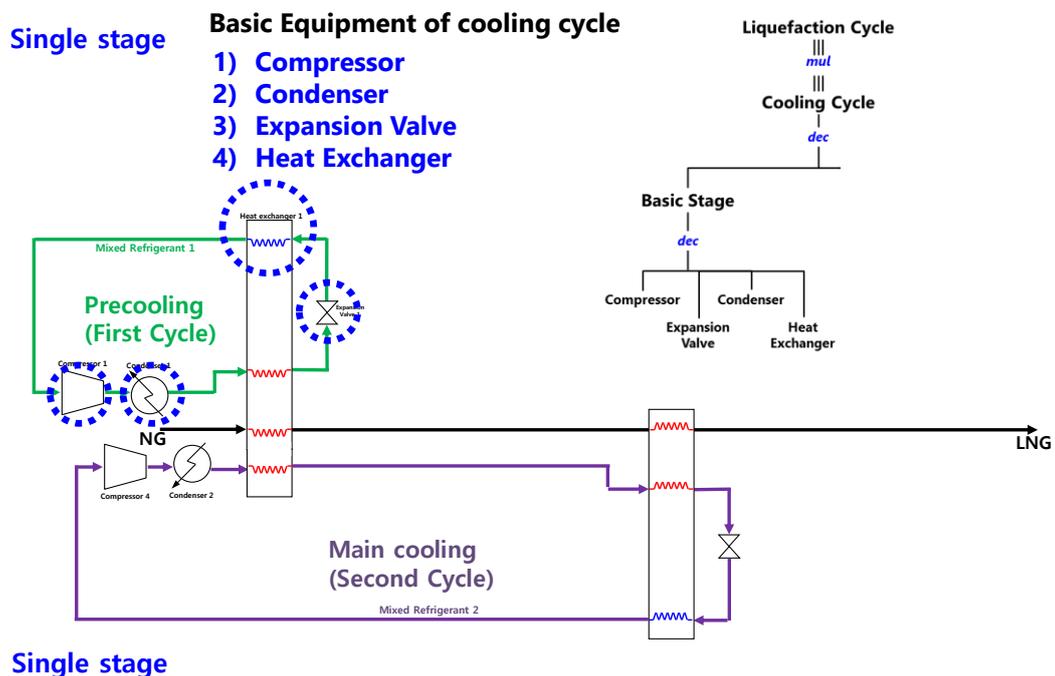


Figure 2-26 Procedure for making the SES for the liquefaction cycle: basic stage

② Additional stages

To enhance the efficiency of the liquefaction cycle, the engineer can add equipment to it or modify its configuration. There are many types of additional formal stages for enhancing such efficiency, such as *compression refrigeration*, *compression with an intercooler*, and *refrigeration*, as mentioned in Section 2.3.1. Thus, the *cooling cycle* entity has the *additional stages* entity as a part of the decomposition, and the *additional stages*

entity has multiple *additional stages* using the *mul* (multiple decomposition) node. In addition, the *additional stage* entity can specialize in *compression refrigeration*, *compression with an intercooler*, or *refrigeration*, and is expressed using the *spec* (specialization) node.

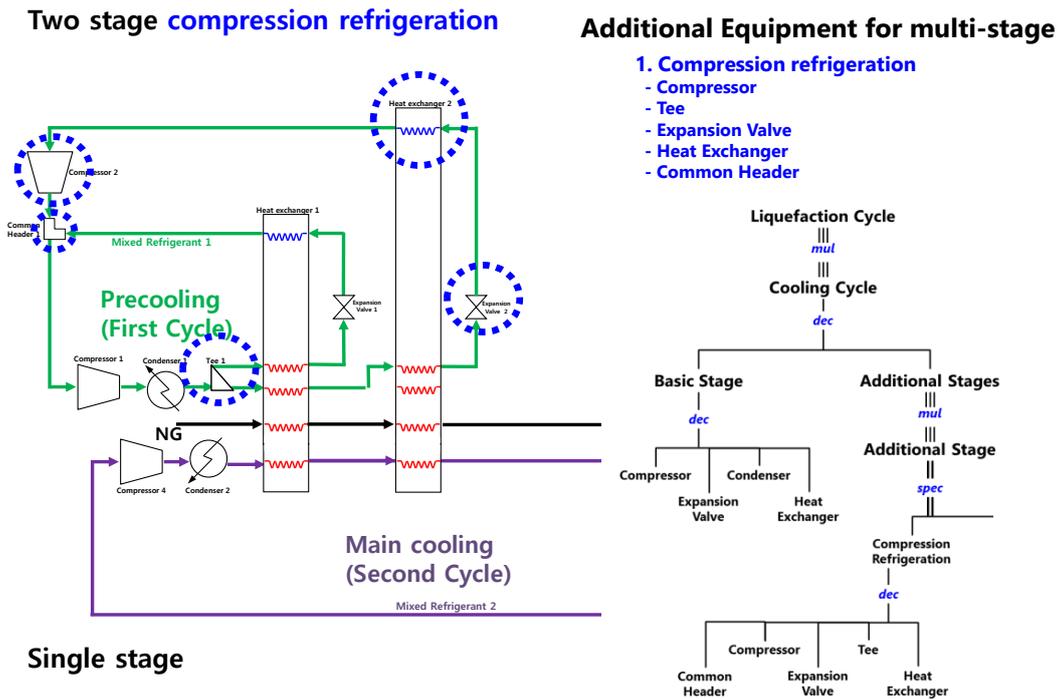


Figure 2-27 Procedure for making the SES for the liquefaction cycle: compression refrigeration stage

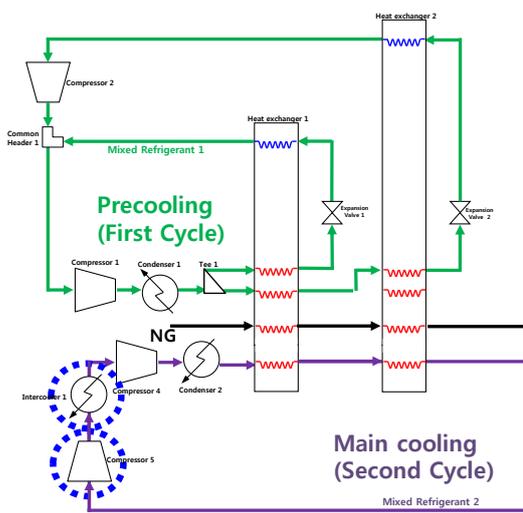
Since a compressor with a large compression capacity is expensive, the combination of smaller-capacity multiple compressors is often used in the liquefaction process system. Moreover, the efficiency of the liquefaction cycle can be enhanced when multiple heat exchangers are used in the cycle. The compression refrigeration stage is based on multiple expansions and multiple compressions to enhance the cycle efficiency.

This stage is decomposed into five equipment: a compressor, a tee, an expansion valve

(or expander), a heat exchanger and a common header. Thus, as shown in Figure 2-27, the compression refrigeration entity is decomposed into five entities using the *dec* node. A tee can sometimes be replaced by a phase separator when the refrigerant of the liquefaction cycle is a mixed refrigerant.

After adding new equipment in the compression refrigeration stage, the equipment will be disconnected and connected based on the rules in section 2.2. The detailed rules defined in the *dec* node are also explained in the next section.

Two stage compression refrigeration



Two stage compression with intercooler

Additional Equipment for multi-stage

2. Compression with intercooler

- Compressor
- Condenser

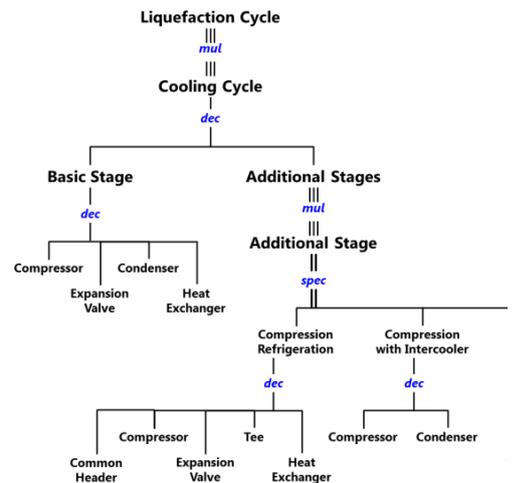


Figure 2-28 Procedure for making the SES for the liquefaction cycle: compression with an intercooler stage.

The compression refrigeration stage has no direct cooling mechanism but multiple compressions in the cycle. The common header supplies the cooling mechanism to the compression refrigeration stage by mixing a cool refrigerant and the heat exchanger. On the other hand, the compression with an intercooler stage is based on multiple compressions

and multiple direct cooling instances using condensers. This stage is composed of two equipment, so the compression refrigeration entity is decomposed into two entities using the *dec* node, as shown in Figure 2-28. The equipment is located in front of the existing compressor of the cycle, and the interconnection rules are also defined in the *dec* node.

The refrigeration stage is based on multiple expansions with phase separation of the refrigerant. This stage has four equipment: a phase separator, a heat exchanger, an expansion valve and a common header. Thus, as shown in Figure 2-29, the refrigeration entity is decomposed into four entities using the *dec* node.

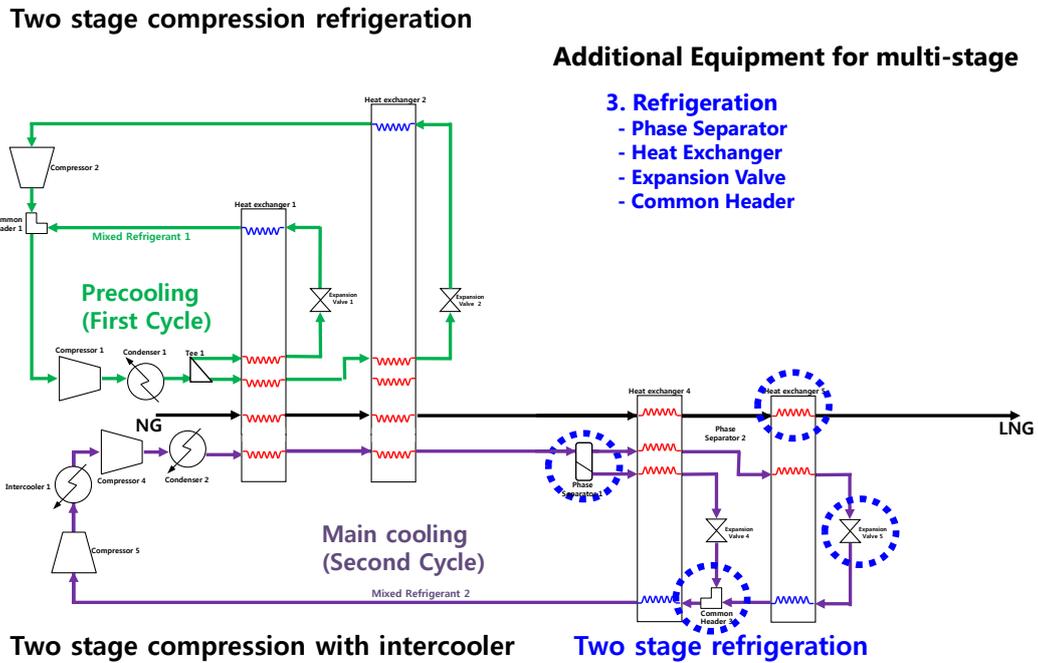


Figure 2-29 Procedure for making the SES for the liquefaction cycle: refrigeration stage

2.4.4. SES for the Liquefaction Cycle

Figure 2-30 shows the SES for the liquefaction cycle. As mentioned in the previous section, the liquefaction cycle is decomposed into multiple cooling cycles. The cooling cycle is decomposed into the basic stage, additional stages, the refrigerant, regeneration, and other options (or entities). The refrigerant entity is used to select the refrigerant for each cycle. The regeneration entity is also used to select the usage of regeneration in the basic stage.

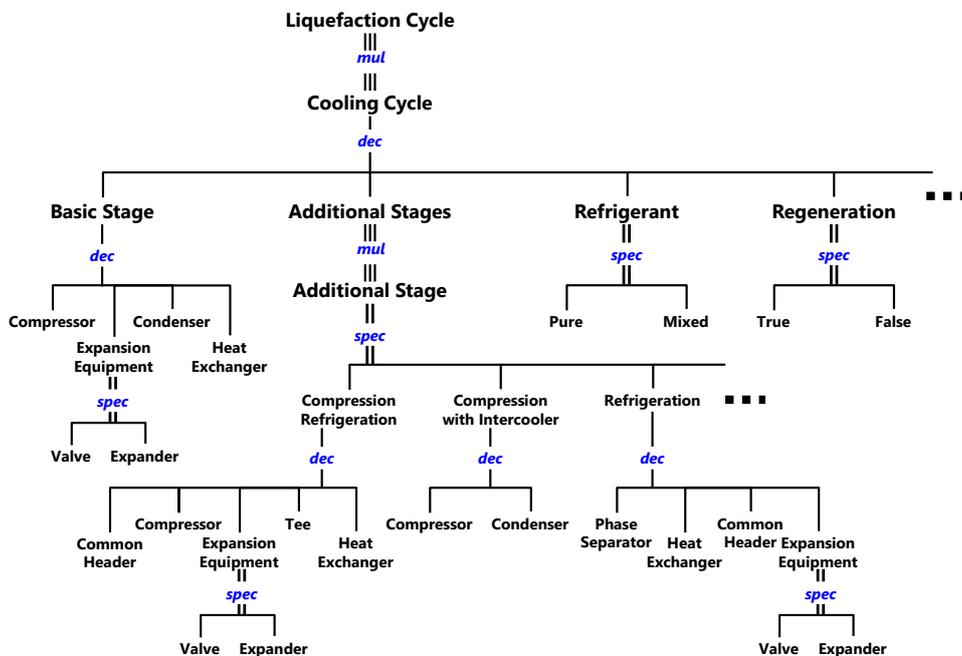


Figure 2-30 SES for the liquefaction cycle

As shown in Figure 2-30, the SES has the hierarchical structure of a system or domain. The information on the interconnection of the child entities is stored in the *dec*, *mul*, and *spec* (specialization) nodes. The *spec* node can also have some information on the properties of each specialization or alternative. The detailed information for the interconnection in the SES follows.

(1) Rules for the basic cycle entity

Figure 2-31 shows an example of the rules for interconnecting the equipment defined in the SES. The basic stage is composed of a compressor, an expansion equipment, a condenser, and a heat exchanger. These are interconnected, and the information on their interconnection is defined in the *dec* node of the *basic stage* entity. The rules defined in the *dec* node of the *basic stage* entity are shown in Table 2-1.

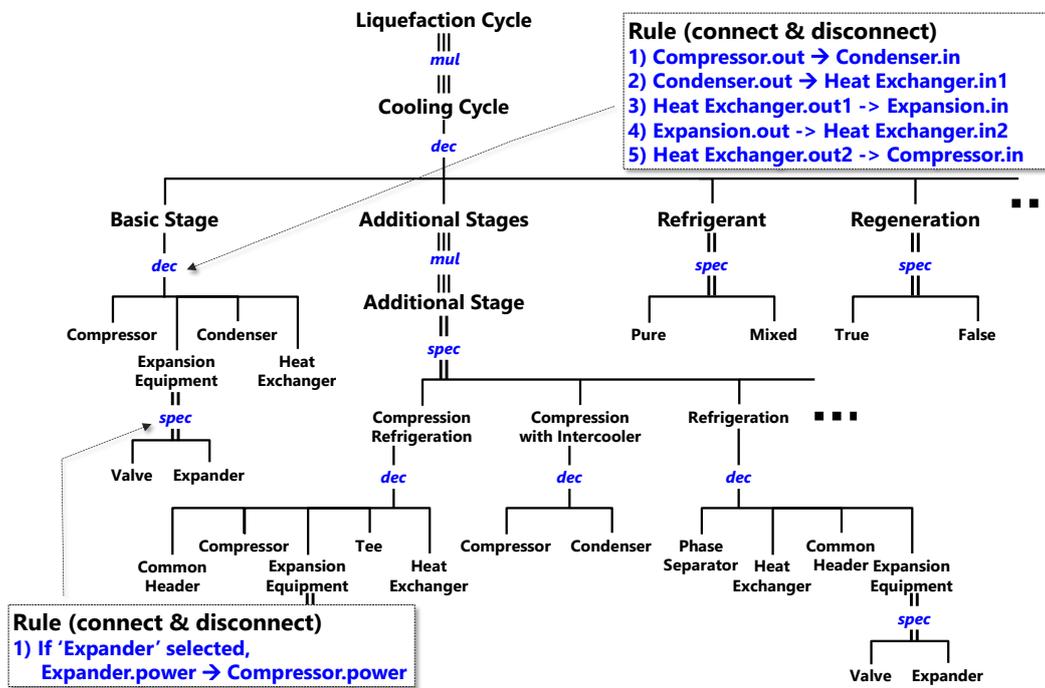


Figure 2-31 Rules in the SES for the liquefaction cycle: basic stage

In the *spec* node of the *expansion equipment* entity, there is a conditional rule for interconnecting equipment. If the *expander* is used for the liquefaction cycle, it generates work that the turbo compressor recovers. Thus, if the expander is selected in the *spec* node, the expander and the compressor should be interconnected. This rule is defined in the *spec* node, as shown in Table 2-2.

Table 2-1 Rules for interconnecting the child entities of the *basic stage* entity

Category	Output		Input	
	Entity	Port	Entity	Port
Connection	Compressor	Outlet	Condenser	Inlet
	Condenser	Outlet	Expansion Equipment	Inlet
	Expansion Equipment	Outlet	Heat Exchanger	Inlet 1
	Heat Exchanger	Outlet 1	Compressor	Inlet

Table 2-2 Rules for interconnecting entities when the expansion equipment is chosen

Category	Output		Input	
	Entity	Port	Entity	Port
Condition	Condition: If the <i>expander</i> entity is chosen			
	Expander	Generated Power	Compressor	Supported Power

(2) Rules for selecting multiple cooling cycles

If the cooling cycle is chosen for more than one cycle, the heat exchanger of each cooling cycle, except that of the precooling cycle, should be connected to those of the other cycles. Thus, the compressor and the heat exchanger of the cooling cycle, except that of the precooling cycle, are disconnected, and the connection of the cooling cycles is made in such a way that the refrigerant of the cooling cycle flows to the heat exchanger of the pre-cycle. The pre-cycle is the previous natural-gas cooling cycle. For example, the pre-cycle of the main cooling cycle is the precooling cycle, and the pre-cycle of the subcooling cycle is the main cooling cycle. The rules for interconnecting equipment after selecting multiple cooling cycles are defined in Table 2-3.

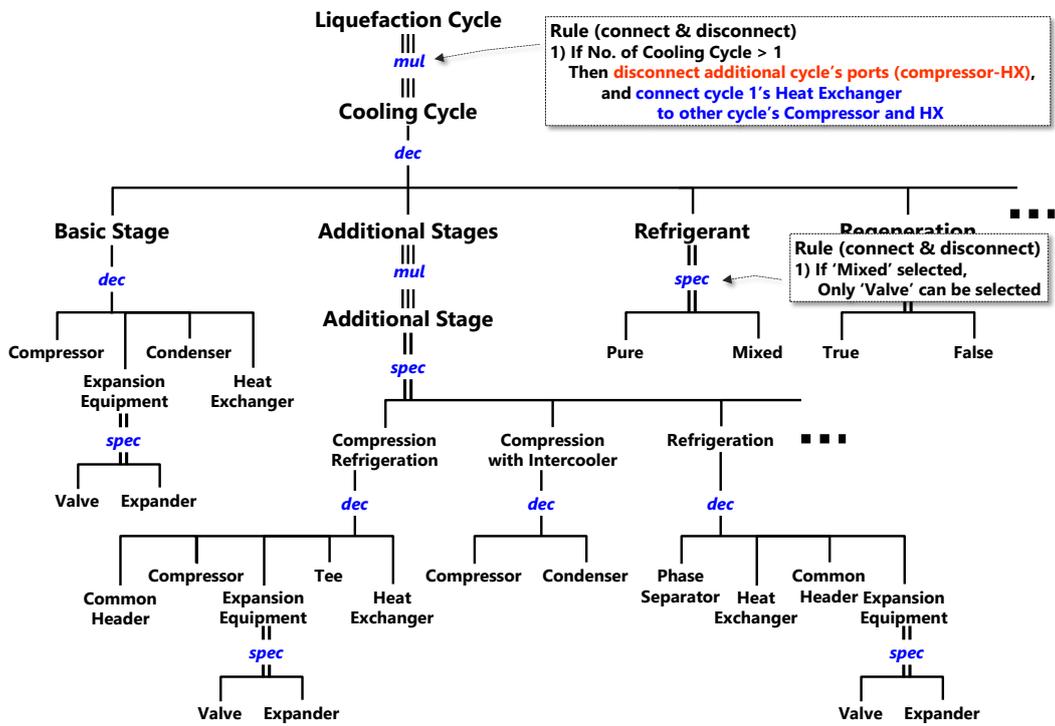


Figure 2-32 Rules in the SES for the liquefaction cycle: selecting multiple cooling cycles

Table 2-3 Rules for disconnecting and connecting the equipment of multiple cooling cycles

Category	Output		Input	
	Entity	Port	Entity	Port
Condition: If a cooling cycle is added				
Disconnection	Condenser	Outlet	Heat Exchanger	Inlet 1
Connection	Condenser	Outlet	Heat Exchanger in the pre-cycle	Inlet (newly added)
	Interconnection of the heat exchangers for the refrigerant of the cycle flow			
	Heat Exchanger (last of the pre-cycle)	Outlet (newly added)	Heat Exchanger	Inlet (first in the added cycle)

(3) Rules for selecting additional stages

① Rules for selecting the compression refrigeration stage

To add an additional stage to the basic stage, the *additional stage* entity should be chosen in the *mul* node of the *additional stage* entity. If an additional stage is added, the equipment in the *basic stage* should be disconnected before being connected to the newly added equipment in the additional stage. Thus, the disconnection rules should be defined in the *mul* node of the *additional stage* entity, as shown in Table 2-4.

Table 2-4 Rules for disconnecting the equipment of the *basic stage* entity when additional stage compression refrigeration is chosen

Category	Output		Input	
	Entity	Port	Entity	Port
Disconnection	Heat Exchanger (last)	Outlet	Compressor (first)	Inlet
	Condenser (last)	Outlet	Heat Exchanger (first)	Inlet

In Figure 2-33, the *compression refrigeration* entity is decomposed into five equipment. The interconnection rules are defined in the *dec* node, as shown in

Table 2-5. The rules of the *spec* node of the *expansion equipment* entity are the same as those in Table 2-2.

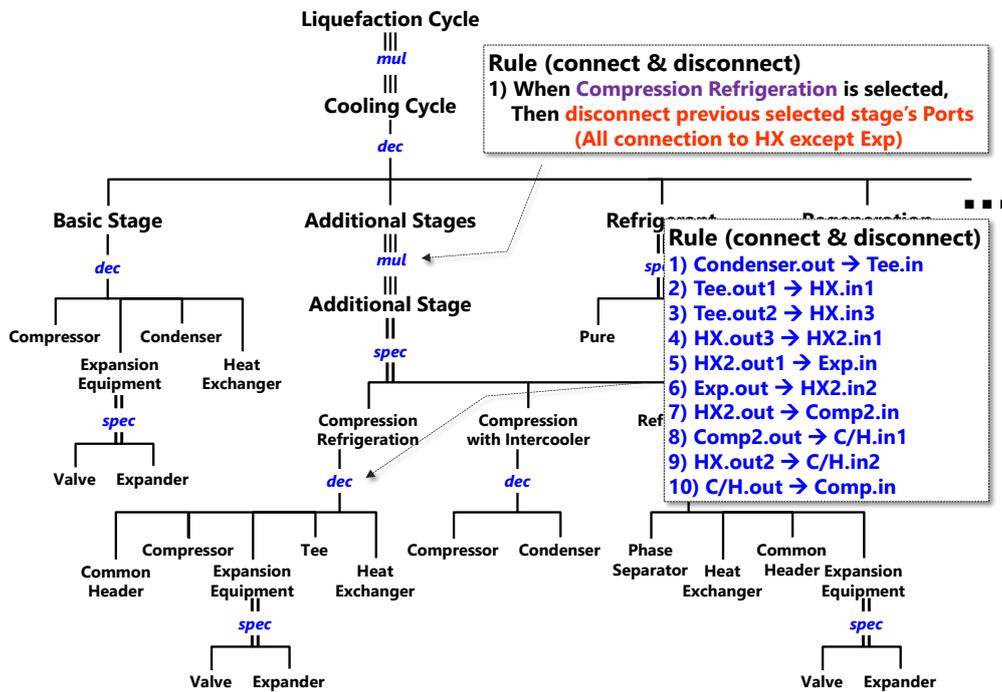


Figure 2-33 Rules for interconnecting the equipment in the SES for the liquefaction cycle: selecting the compression refrigeration stage.

Table 2-5 Rules for interconnecting the child entities of the compression refrigeration stage entity

Category	Output		Input	
	Entity	Port	Entity	Port
Connection	Condenser (last)	Outlet	Tee (newly added)	Inlet
	Tee (newly added)	Outlet 1	Heat Exchanger (newly added)	Inlet 1
	Tee (newly added)	Outlet 1	Heat Exchanger (newly added)	Inlet 3
	Heat Exchanger (newly added)	Outlet 3	Heat Exchanger (newly added)	Inlet 1
	Heat Exchanger (newly added)	Outlet 1	Expansion Equipment (newly added)	Inlet

	Expansion Equipment (newly added)	Outlet	Heat Exchanger (newly added)	Inlet 2
	Heat Exchanger (newly added)	Outlet 2	Compressor (newly added)	Inlet
	Compressor (newly added)	Outlet	Common Header (newly added)	Inlet 1
	Heat Exchanger (last)	Outlet 2	Common Header (newly added)	Inlet 2
	Common Header (newly added)	Outlet	Compressor (first)	Inlet

② Rules for selecting compression with an intercooler stage

If this additional stage is added, the equipment in the basic stage should be disconnected before being connected with the newly added equipment in the additional stage. Thus, the disconnection rules, according to the choice of compression with an intercooler stage, are defined in the *mul* node of the *additional stages* entity, as shown in Table 2-6.

Table 2-6 Rules for disconnecting the equipment of the *basic stage* entity when the additional stage, compression with an intercooler, is chosen

Category	Output		Input	
	Entity	Port	Entity	Port
Disconnection	Heat Exchanger (last)	Outlet	Compressor (first)	Inlet

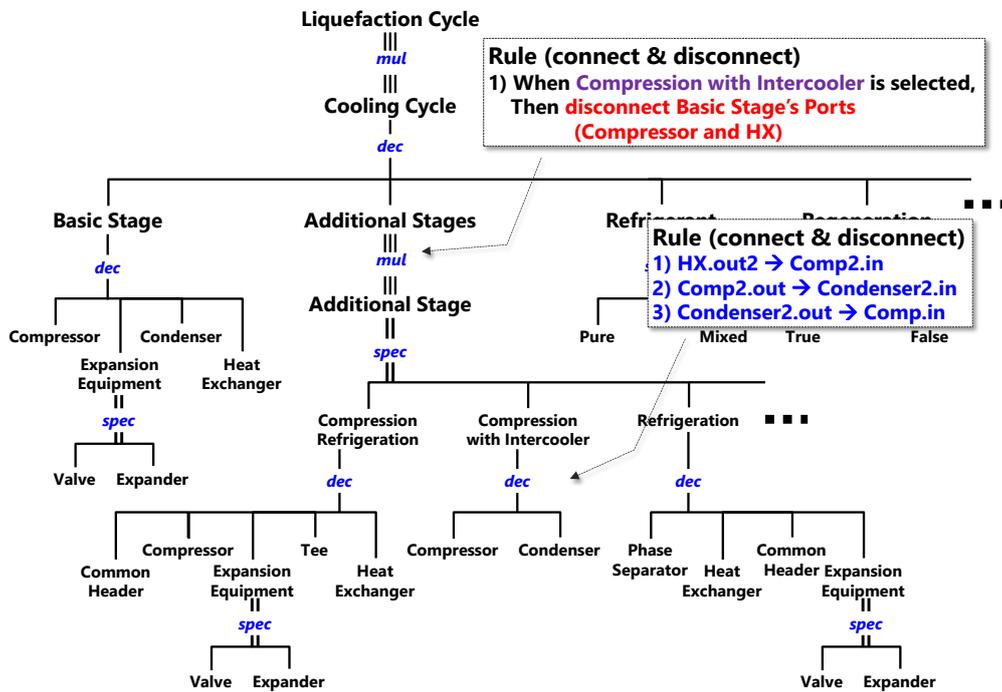


Figure 2-34 Rules for interconnecting the equipment in the SES for the liquefaction cycle: selecting “compression with an intercooler stage”

In Figure 2-34, the *compression with an intercooler* entity is decomposed into two equipment. The interconnection rules are defined in the *dec* node, as shown in Table 2-7.

Table 2-7 Rules for interconnecting the child entities of the *compression refrigeration stage* entity

Category	Output		Input	
	Entity	Port	Entity	Port
Connection	Heat Exchanger (last)	Outlet	Compressor (newly added)	Inlet
	Compressor (newly added)	Outlet	Condenser (newly added)	Inlet
	Condenser (newly added)	Outlet	Heat Exchanger (first)	Inlet

③ Rules for selecting the refrigeration stage

To add the additional stage “refrigeration,” the interconnection between the equipment of the basic stage should be disconnected before being connected with the newly added equipment of the additional stage. Thus, the rules for the disconnection according to the choice of the “refrigeration” stage are defined in the *mul* node of the *additional stages* entity, as shown in Table 2-6.

Table 2-8 Rules for disconnecting the equipment of the *basic stage* entity when the additional stage “compression with an intercooler” is chosen

Category	Output		Input	
	Entity	Port	Entity	Port
Disconnection	Condenser (last)	Outlet	Heat Exchanger (first)	Inlet
	Expansion Valve (last one)	Outlet	Heat Exchanger (first)	Inlet

As shown in Figure 2-35, the *refrigeration stage* entity is decomposed into four equipment. The interconnection rules are defined in the *dec* node, as shown in Table 2-9.

Table 2-9 Rules for interconnecting the child entities of the *refrigeration stage* entity

Category	Output		Input	
	Entity	Port	Entity	Port
Connection	Condenser (last)	Outlet	Phase Separator (newly added)	Inlet
	Phase Separator (newly added)	Outlet 1	Heat Exchanger (newly added)	Inlet1
	Phase Separator (newly added)	Outlet 2	Heat Exchanger (newly added)	Inlet2
	Heat Exchanger (newly added)	Outlet 1	Heat Exchanger (first)	Inlet
	Heat Exchanger	Outlet 2	Expansion	Inlet

	(newly added)		Valve (newly added)	
	Expansion Valve (newly added)	Outlet	Common Header (newly added)	Inlet 1
	Heat Exchanger (first)	Outlet	Common Header (newly added)	Inlet 2
	Common Header (newly added)	Outlet	Heat Exchanger (newly added)	Inlet 3
	Heat Exchanger (newly added)	Outlet 3	Compressor (first)	Inlet

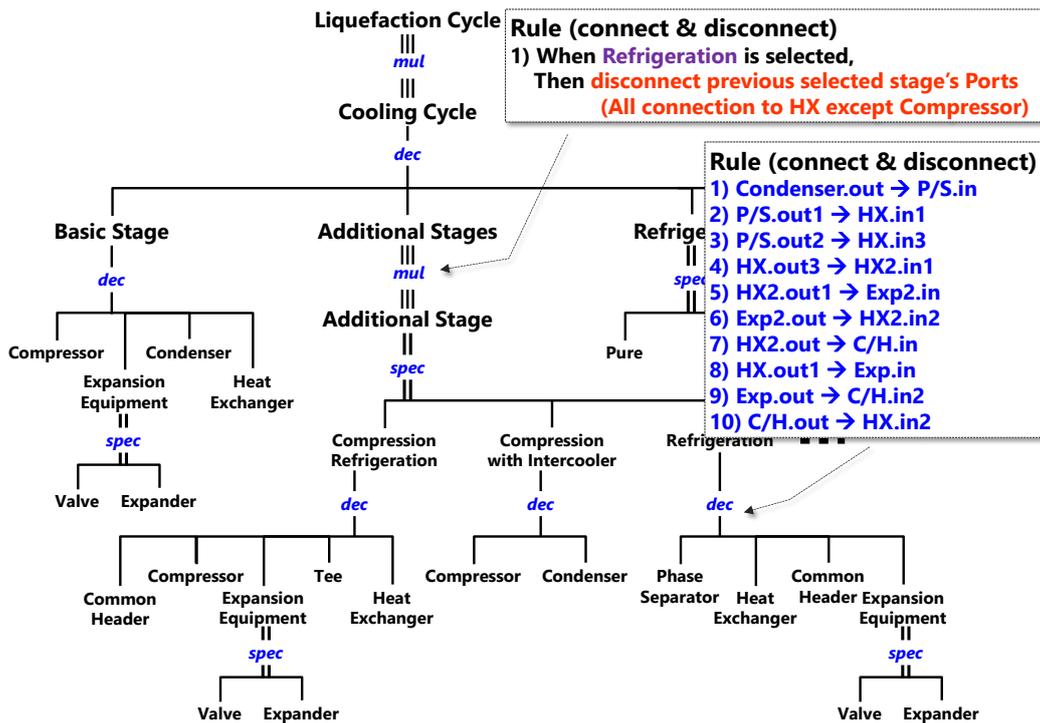


Figure 2-35 Rules for interconnecting the equipment in the SES for the liquefaction cycle: selecting “refrigeration stage”

condenser, an equipment for expansion, and a heat exchanger. In addition, each cycle has an additional stage to enhance its efficiency. In the precooling cycle of the DMR cycle, there is an additional stage for expansion and compression. This is considered by selecting one more additional stage and by specializing it as *compression refrigeration*. In the same manner, the main cooling cycle of the DMR cycle has two additional stages: one for compression and cooling and another for the expansion and phase separation of the refrigerant. Thus, two additional stages are selected and specialized as *compression with an intercooler* and *refrigeration*. As the DMR cycle uses a mixed refrigerant, for all the cycles, the *mixed* entity was selected for the refrigerant.

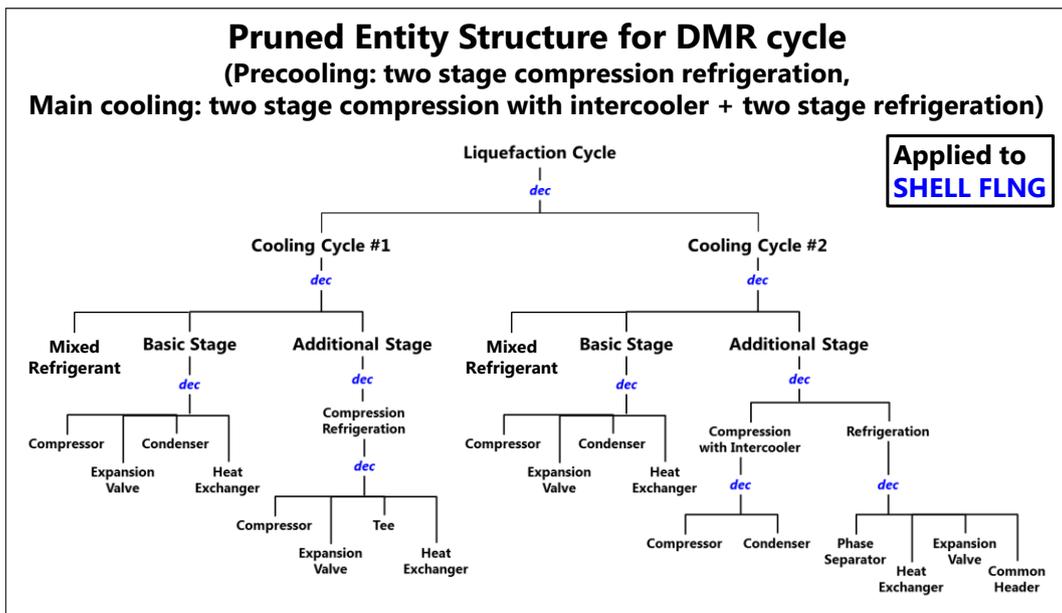


Figure 2-37 Pruned entity structure for the DMR cycle generated by pruning the SES

Figure 2-37 shows the pruned entity structure (PES) for the DMR cycle generated by pruning. PES is a kind of pure entity structure that does not have any selectable node, such as the *mul* and *spec* nodes. In Figure 2-37, the DMR cycle is decomposed into two *cooling cycles*. Each cooling cycle in the DMR cycle has a *basic stage* and *additional stages*. The

precooling cycle, Cooling Cycle #1 in Figure 2-37, has “compression refrigeration,” and the main cooling cycle, Cooling Cycle #2 in Figure 2-37, has “compression with an intercooler” and “refrigeration” stages. All the *dec* nodes in this PES inherited the interconnection rules from the nodes of the SES.

Figure 2-38 shows the logical model of the DMR cycle generated using the PES in Figure 2-37. All the equipment in this logical model are members of the entities in PES, and the equipment are connected with other equipment based on the rules in the nodes of PES.

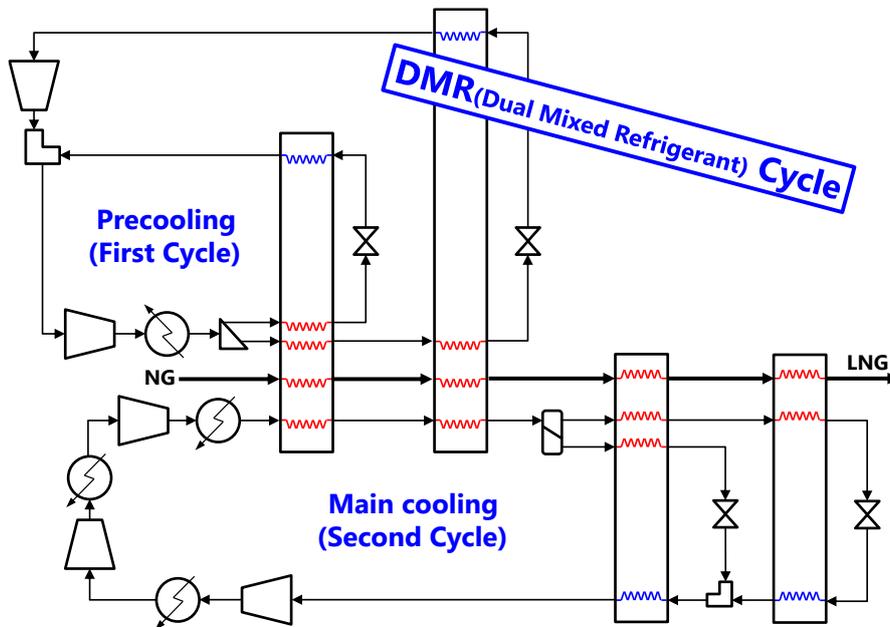


Figure 2-38 Logical model of the DMR cycle generated using the pruned entity structure

(2) Generation of the C3MR cycle by pruning the SES

Figure 2-39 shows the sequence of the SES pruning for the C3MR cycle. The C3MR cycle has two cooling cycles: precooling and main cooling. Thus, two entities are chosen at the *mul* node below the cooling cycle. Each cooling cycle has additional stages to

enhance the efficiency of the cycle. The C3MR cycle uses propane as a refrigerant of the precooling cycle, so the *pure* entity is chosen for the refrigerant of the precooling cycle.

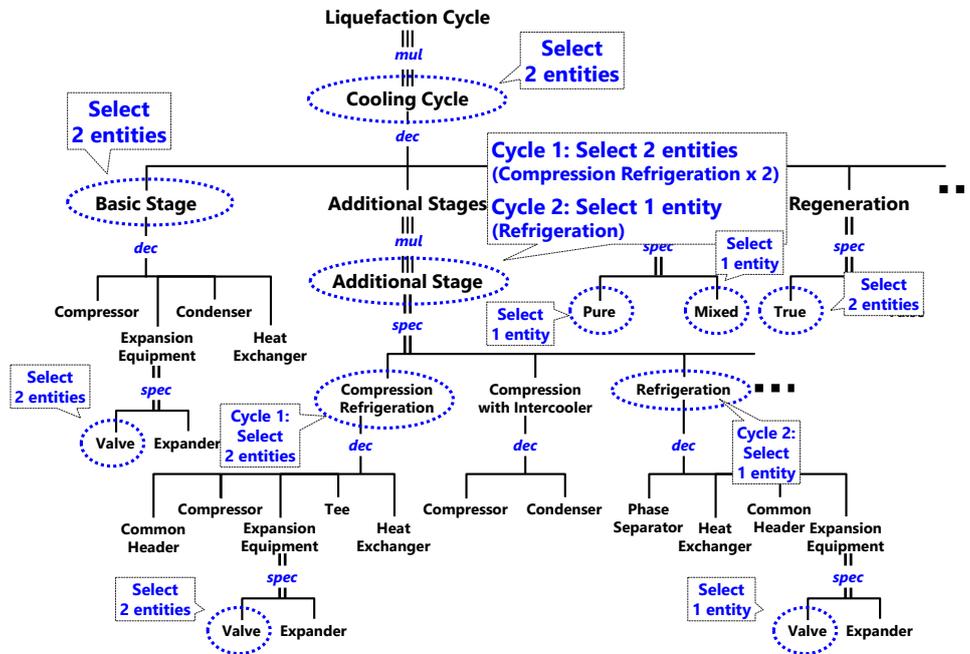


Figure 2-39 Pruning the SES to generate the logical model of the C3MR cycle

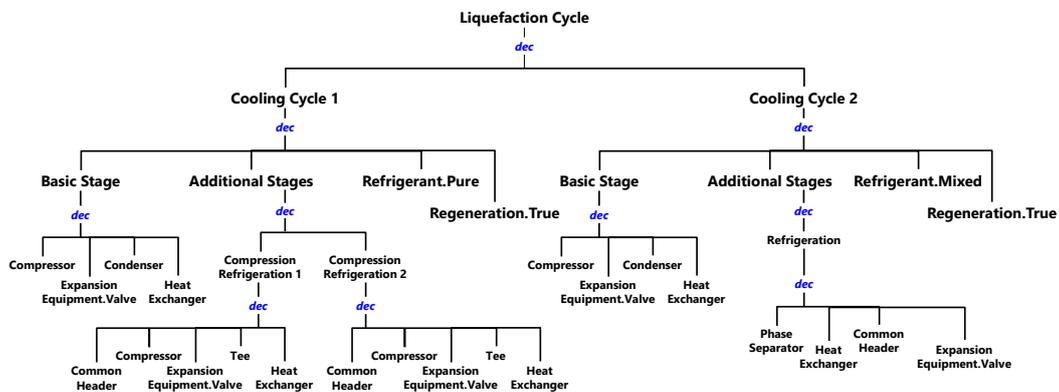


Figure 2-40 PES for the C3MR cycle generated by pruning the SES

Figure 2-40 shows the PES for the C3MR cycle generated by pruning. The PES for the C3MR cycle has two cooling cycles, and each cycle has a *basic stage* and *additional stages* that are in accord with the configuration of the C3MR cycle, as shown in Figure 2-3.

(3) Generation of a cascade cycle by pruning the SES

Figure 2-41 shows the sequence of the SES pruning for the cascade cycle. The cascade cycle has three cooling cycles, and all the cycles use a pure refrigerant. As shown in Figure 2-41, the precooling cycle has only one additional stage: *compression refrigeration*.

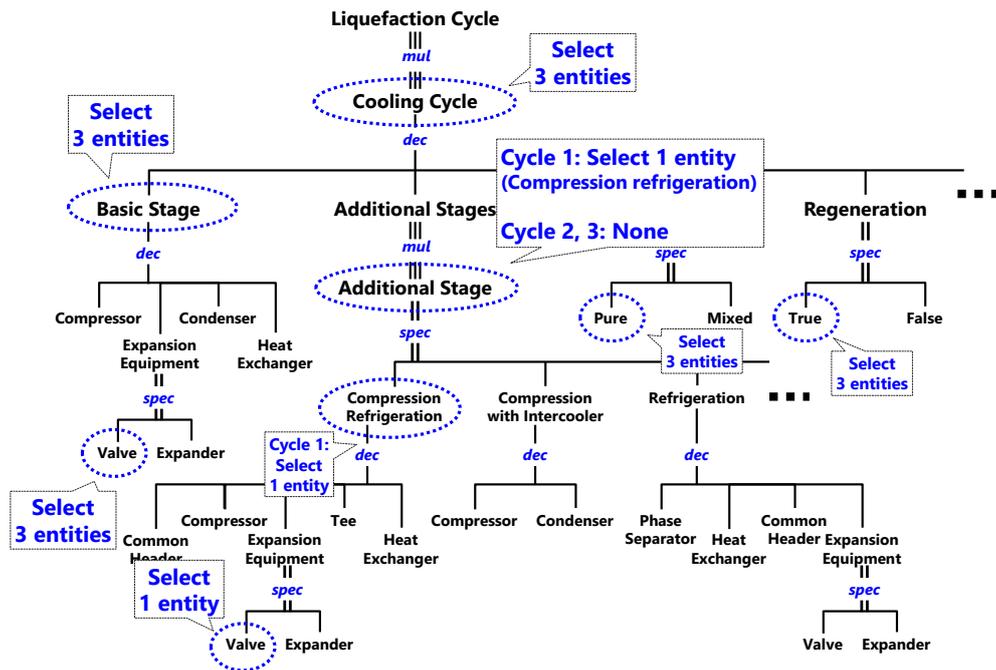


Figure 2-41 Pruning the SES to generate the logical model of the cascade cycle

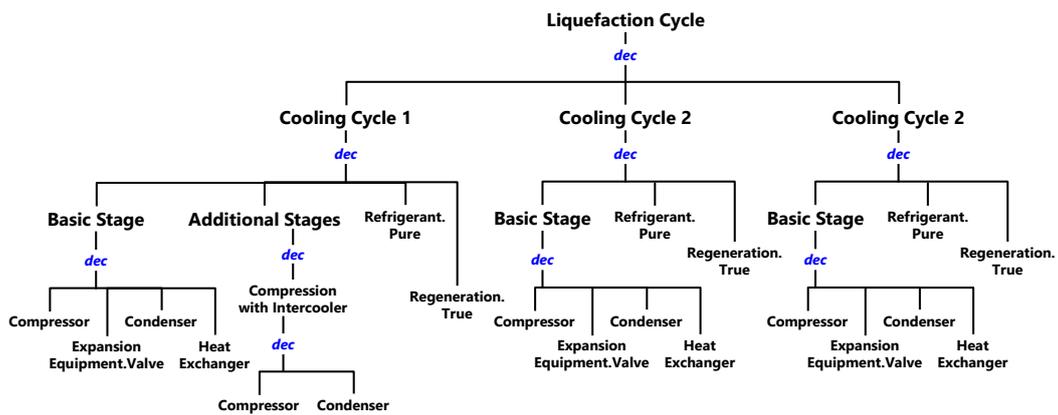


Figure 2-42 PES for the cascade cycle generated by pruning the SES

Figure 2-42 shows the PES for the cascade cycle generated by pruning. As all the *dec* nodes in the PES inherited the interconnection rules from the SES, the PES in Figure 2-42 can be transformed into the logical model of the cascade cycle.

(4) Generation of the N₂ expander cycle

Figure 2-43 shows the sequence of the SES pruning for the single N₂ expander cycle. In the N₂ expander cycle, an expander is used for the expansion equipment instead of a valve, and the expander is chosen in the sequence of pruning. As the single N₂ expander cycle has dual compressions, the “compression with an intercooler” stage was chosen for the additional stage.

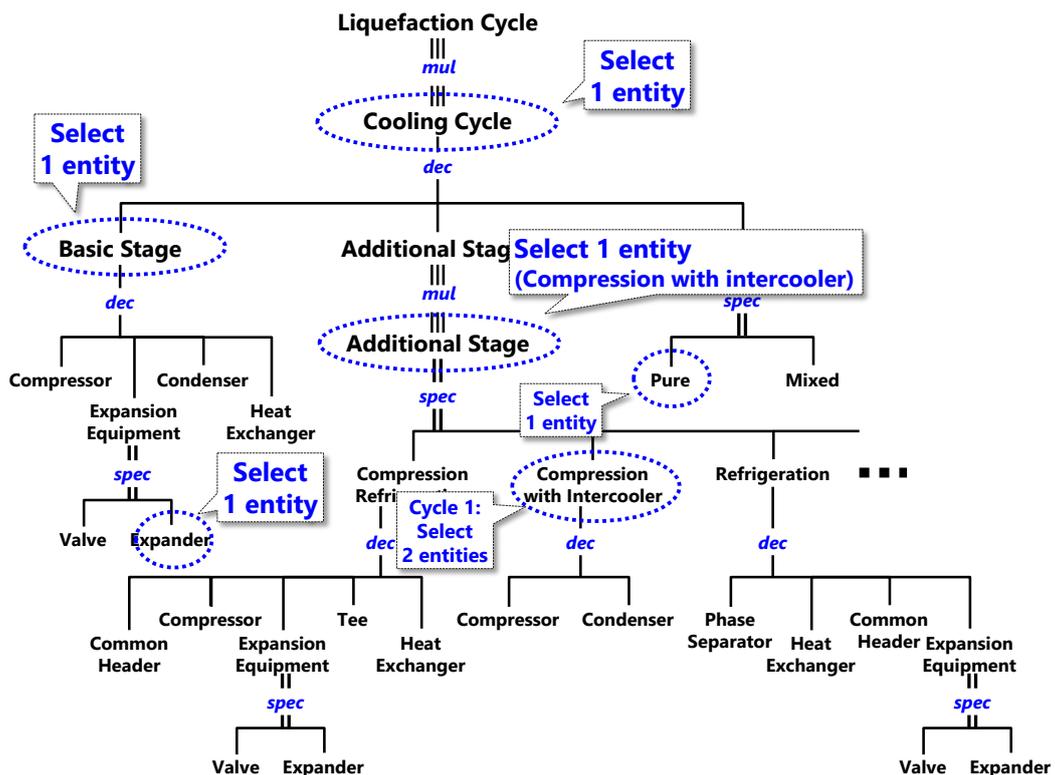


Figure 2-43 Pruning the SES to generate the logical model of the single N₂ expander cycle

Figure 2-44 shows the sequence of the SES pruning for the dual N₂ expander cycle. The dual N₂ expander cycle has two expanders; one is chosen from the *spec* node of the basic stage, and the other from the *spec* node of the “compression refrigeration” stage. For multiple compressions, the “compression with an intercooler” stages are selected, as shown in Figure 2-44.

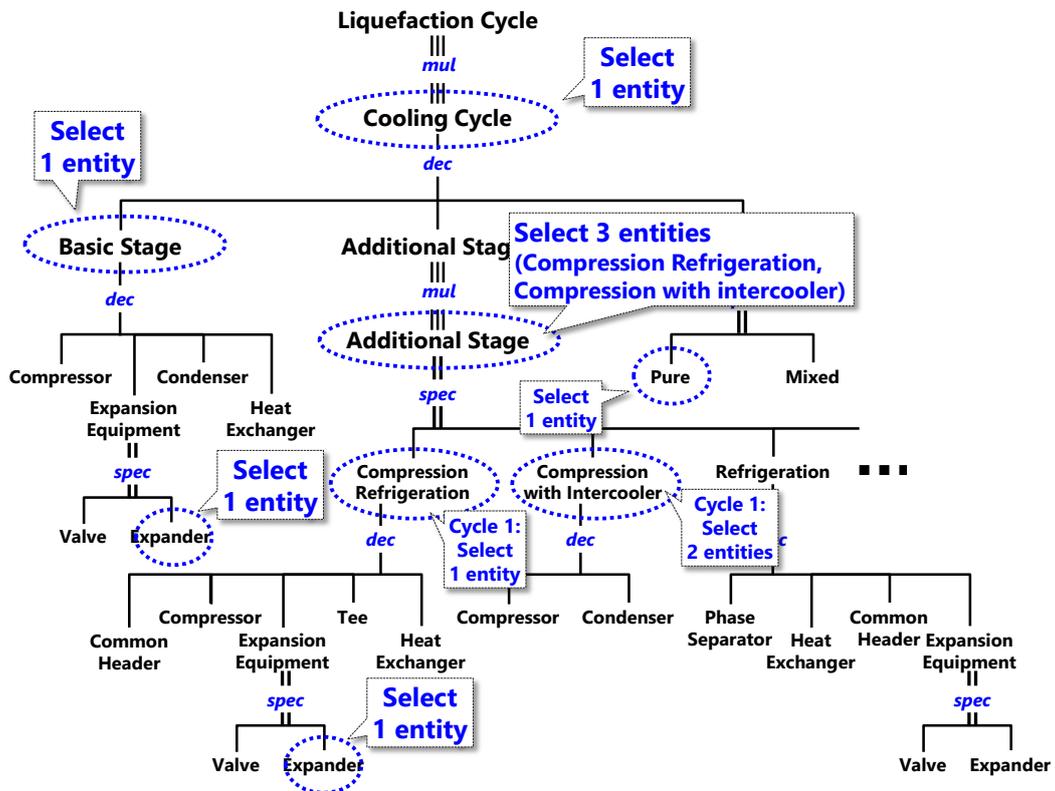


Figure 2-44 Pruning the SES to generate the logical model of the dual N₂ expander cycle

(5) Generation of the Niche cycle

The Niche cycle has two cooling cycles, one using methane as the refrigerant, and another, nitrogen. All the cycles in the Niche cycle use an expander. These reasons are reflected in the pruning sequence, as shown in Figure 2-45.

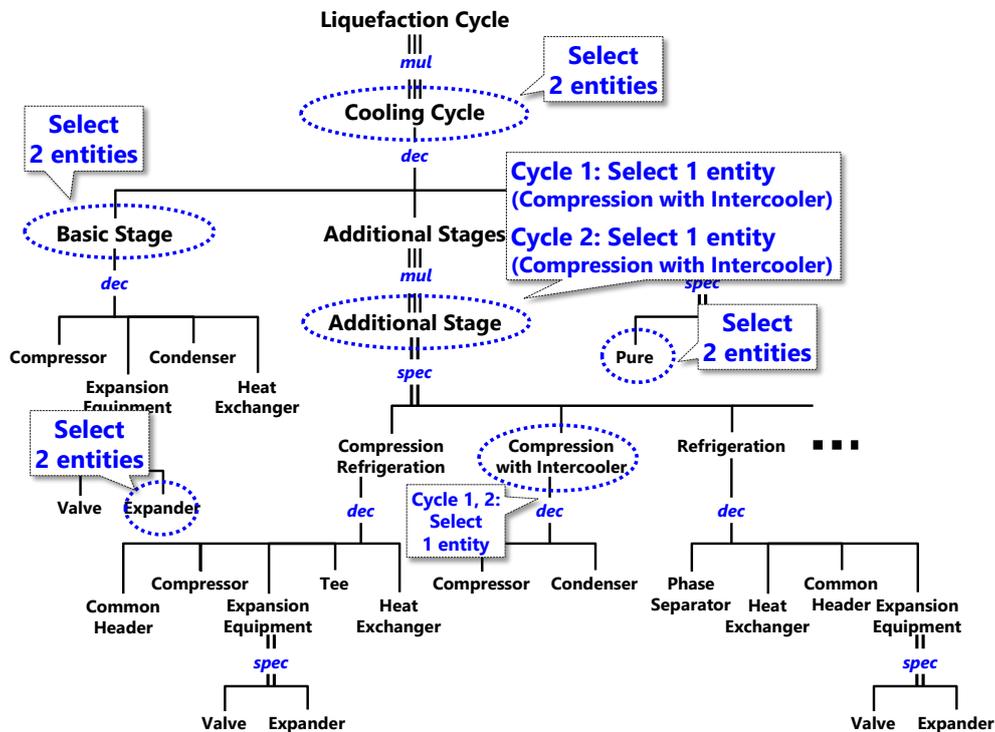


Figure 2-45 Pruning the SES to generate the logical model of the Niche cycle

(6) Various alternative logical models of the liquefaction cycle

Figure 2-46, Figure 2-47, Figure 2-48, and Figure 2-49 show various alternative logical models of the liquefaction cycle generated by the resultant PES after finishing the pruning process. As mentioned in this section, by pruning the proposed SES, the logical models of the liquefaction cycle can be automatically generated, including the conventional liquefaction cycles, such as the DMR, C3MR, cascade, expander, Niche, and other cycles. These logical models can be used for process evaluation, such as process simulation and reliability analysis. Moreover, during the pruning process and the automatic generation of various logical models of the liquefaction cycle, the logical models that are not yet under consideration can be generated and evaluated, and they may have new, improved designs of the liquefaction cycle compared with the conventional liquefaction cycles.

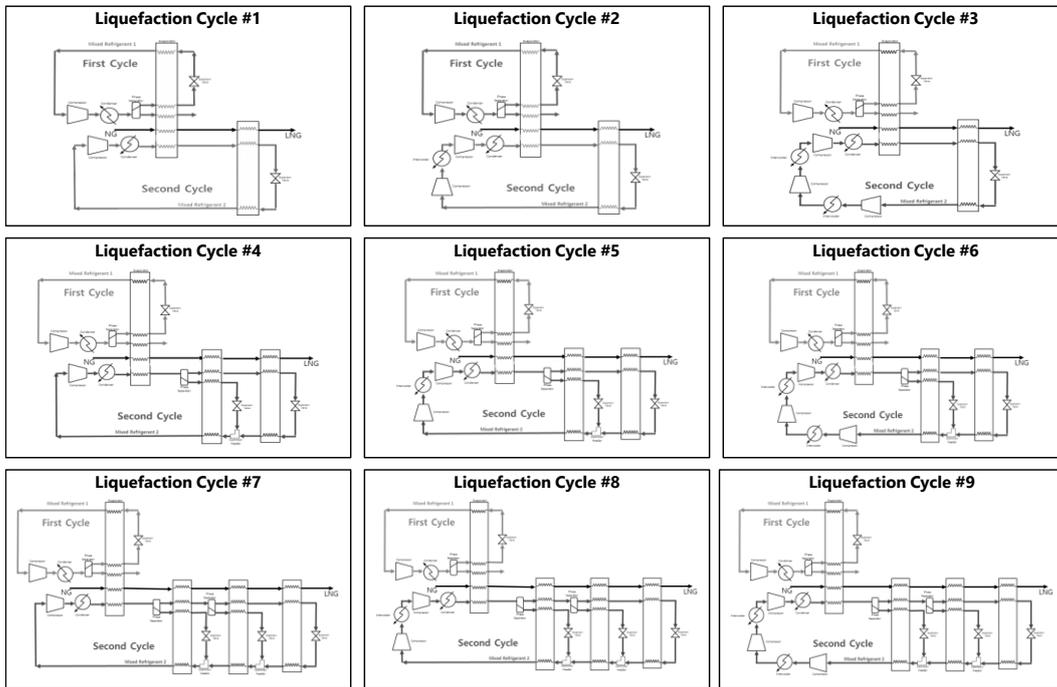


Figure 2-46 Sample logical models of the liquefaction cycle generated by pruning the SES

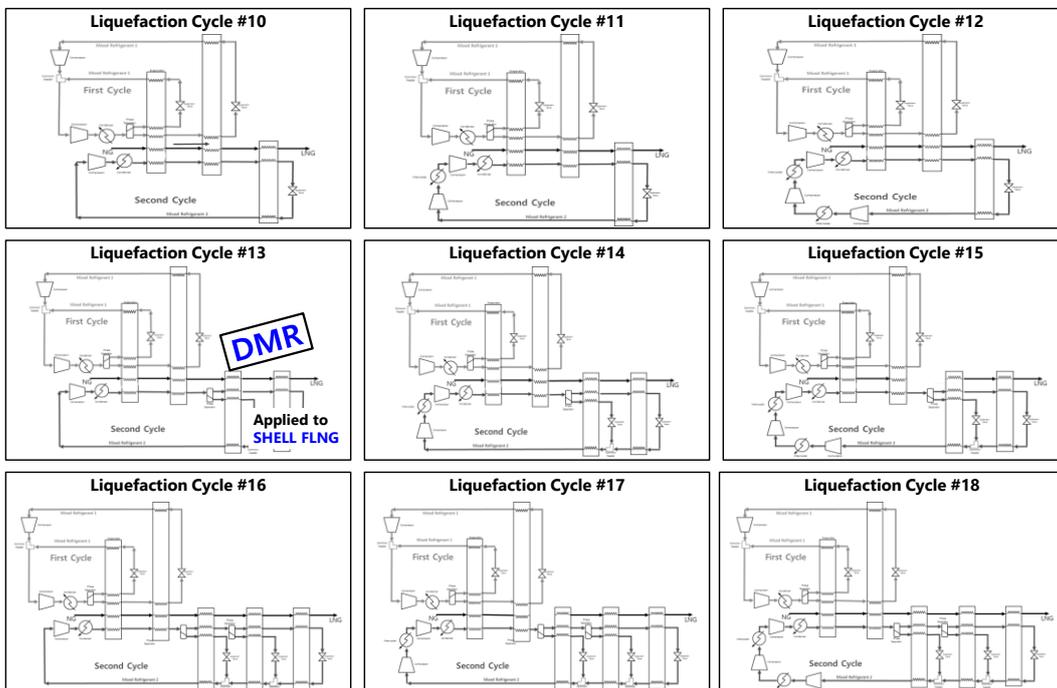


Figure 2-47 Sample logical models of the liquefaction cycle generated by pruning the SES

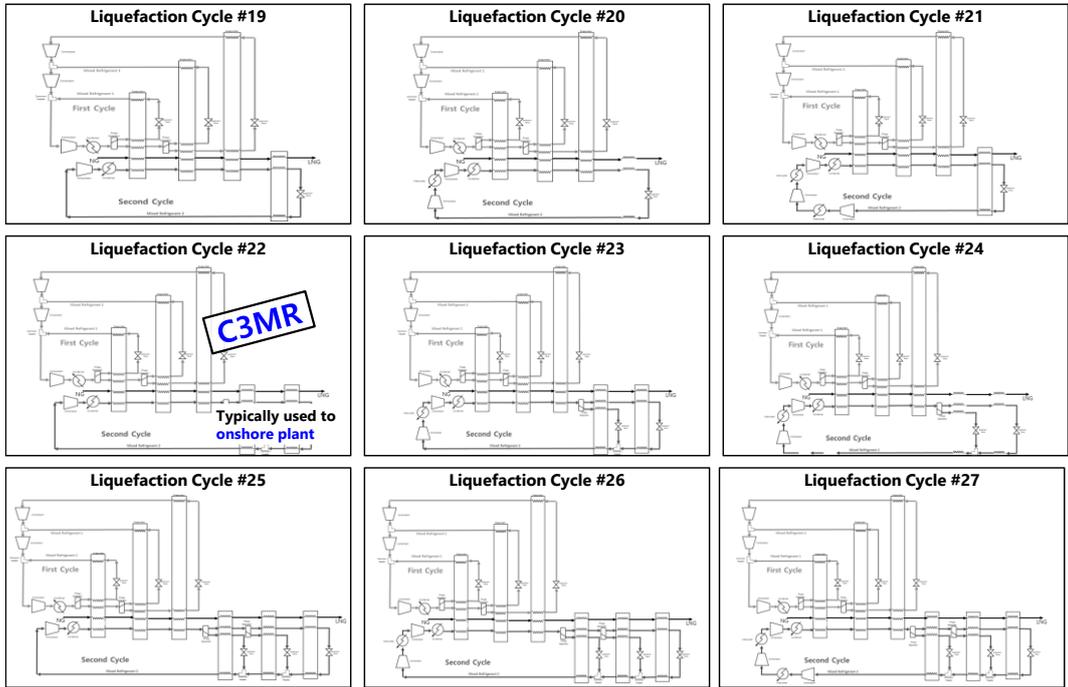


Figure 2-48 Sample logical models of the liquefaction cycle generated by pruning the SES

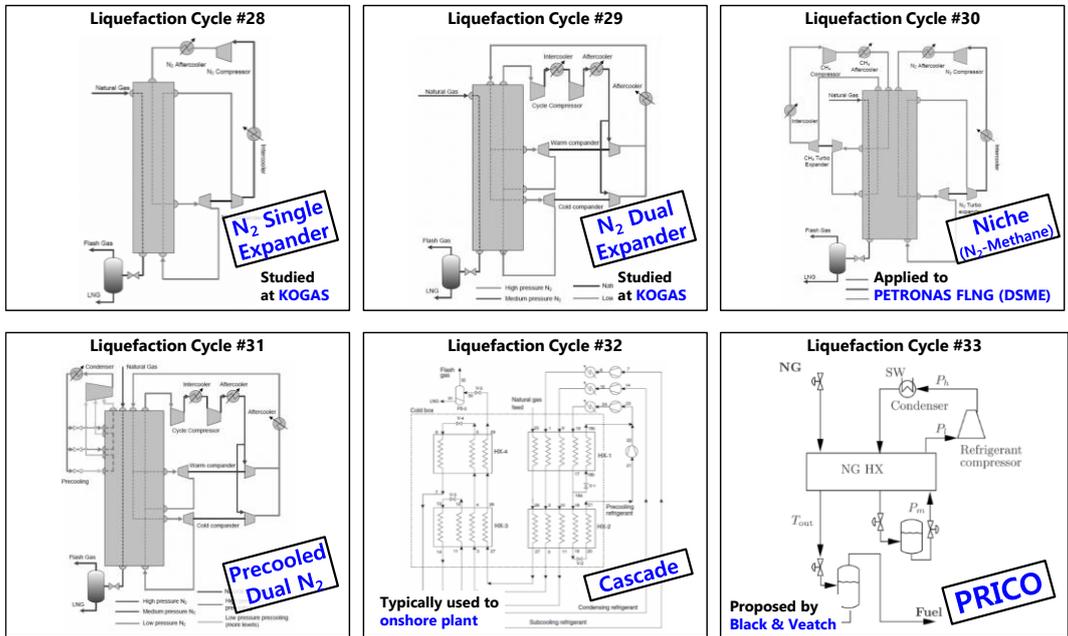


Figure 2-49 Sample logical models of the liquefaction cycle generated by pruning the SES

3. Process Simulation to Determine the Optimal Operating Conditions of the LNG FPSO Liquefaction Cycles Using a Sequential Modular Simulator Based on the DEVS Formalism

As mentioned in chapter 2, various alternative liquefaction cycles can be generated automatically by pruning the proposed SES. Considering the limited spaces in offshore applications, the engineer should consider the design criteria of the LNG FPSO liquefaction cycle, such as efficiency, reliability, ship motion, and compactness (Hwang et al., 2013). As the available area for the liquefaction cycle for offshore applications is smaller than that for onshore plants, the efficiency and compactness of the liquefaction cycle is more important than that of onshore plants.

The efficiency is an important parameter in configuring a liquefaction cycle for offshore applications. It is generally expressed as the ratio of power consumption to the produced LNG in tons in one day. To determine the optimal operating conditions of the liquefaction cycle, the total required power must be minimized, which will result in equipment size reduction. Moreover, the flow rate of the refrigeration will also be reduced by optimizing the operating conditions, and as such, the overall size of the liquefaction process system, including the equipment, instruments, and pipe size, will be reduced. Therefore, the

efficiency should be considered for offshore applications from the viewpoint of the limited space.

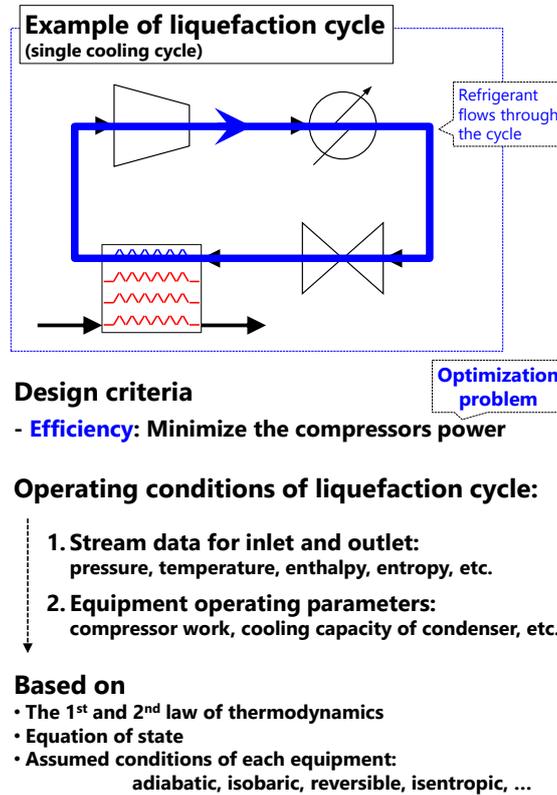


Figure 3-1 Efficiency of the liquefaction cycle: a design criterion of LNG FPSO

The reminder of this chapter is as follows. Section 3.1 describes an optimization problem to determine the optimal operating conditions of the liquefaction cycle, and the procedure of optimization. Section 3.2 presents a process simulator which can calculate the operating conditions by solving the system of nonlinear equations in the optimization problem. Section 3.3 briefly introduces the optimization method used in this thesis. Section 3.4 presents the verification of developed sequential modular simulator by applying to a single cooling cycle (refrigerator). Finally, section 3.5 presents the calculation results of the optimal operating conditions for the DMR cycle.

3.1. Procedure for Determining the Optimal Operating Conditions of the Liquefaction Cycle

During the process simulation, the refrigerant flows through the cycle with its operating conditions. The operating conditions of the liquefaction cycle are the stream data for the inlet and outlet of the equipment, and the operating parameters of the equipment. To determine the optimal operating conditions of the liquefaction cycle, its mathematical model based on thermodynamics should be formulated. A general formulation of optimization problem to determine the optimal operating conditions of liquefaction cycle is shown in Table 3-1.

Table 3-1 General optimization problem to determine the optimal operating conditions of liquefaction cycle

<p><u>Design variables</u> (No.: n)</p> <p>Stream data for inlet and outlet: Pressure (P), Temperature (T), specific volume (v), ... ,</p> <p>Equipment operating parameters: Specific heat transfer (q_H), ... ,</p> <p>Vapor fraction of two-phase flow (v_f),</p> <p>Mass flow rate of a refrigerant for each cooling cycle,</p> <p>Mass fraction of a refrigerant for each cooling cycle, ...</p>
<p><u>Equality constraints</u> (No.: m)</p> <p>The first law of thermodynamics (energy conservation),</p> <p>The second law of thermodynamics,</p> <p>Equations of state,</p> <p>Thermodynamical conditions: adiabatic, reversible, isentropic, isobaric, ... ,</p> <p>Saturated pressure and temperature,</p> <p>Conservation condition of the output temperatures in a heat exchanger, ...</p>

<p><u>Inequality constraints</u> (No.: l)</p> <p>Flow limit, Inlet pressure of a compressor (lower than the dew point), Outlet temperature of a condenser, Maximum temperature difference in a heat exchanger, ...</p>
<p><u>Objective function</u></p> <p>Minimize the total required compressor power:</p> $\text{Minimize } W = \dot{m} \cdot w,$ <p>where \dot{m} is mass flow rate of a refrigerant [kg/s], and w is work input to the compressor per mass [J/kg].</p>

As the design variables are more than the equality constraints (so-called *equations*) of the liquefaction cycle ($n < m$ in Table 3-1), this problem is a kind of optimization problem. For example, in case of a refrigerator with single cooling cycle, the number of design variables (n) is 24, the number of equality constraints (m) is 19, and the number of inequality constraints (l) is 1, so it is a kind of constrained optimization problem (Lee, 2012).

A constrained optimization problem can be solved by using the optimization method such as *sequential linear programming* (SLP) and *sequential quadratic programming* (SQP). Figure 3-2 shows the procedure for determining the optimal operating conditions of the liquefaction cycle using these optimization methods.

Since the number of variables is less than the number of equality constraints, the number of free variables is $m - n$. To calculate the dependent variables of the problem, at first, some free variables should be reasonably assumed (step (a) of Figure 3-2). After assuming the free variables, the dependent variables are calculated by solving the system of nonlinear

equations, which is a combination of the equality constraints in Table 3-1 (step (b) of Figure 3-2).

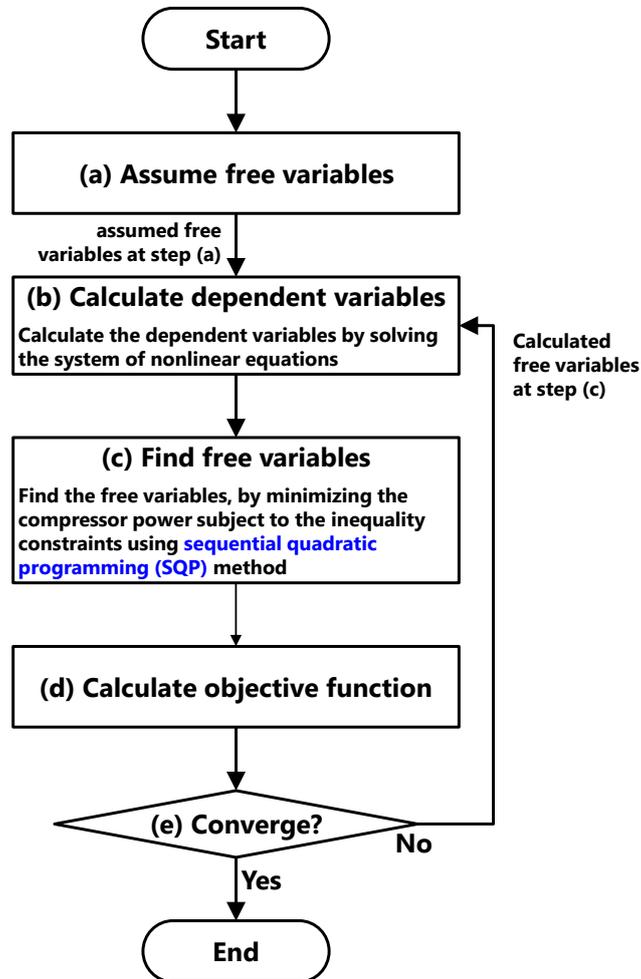


Figure 3-2 Procedure for determining the optimal operating conditions of the liquefaction cycle

After the calculation of the dependent variables, the free variables are found again by minimizing the objective function subject to the inequality constraints (so-called *inequalities*), using an optimization method (step (c) of Figure 3-2). The objective function is given by minimizing the total required power to produce LNG as shown in Table 3-1. The sequential quadratic programming (SQP) method is used for the optimization. After

finishing the optimization at step (c), the value of the objective function is calculated (step (d) of Figure 3-2), and the convergence of the value is checked (step (e) of Figure 3-2). The procedure is carried out until the objective function converges.

In the procedure for determining the optimal operating conditions of the liquefaction cycle, the dependent variables are calculated by solving the system of nonlinear equations. A process simulator such as *equation-oriented simulator* and *sequential modular simulator* performs this procedure and this will be described in the next section.

3.2. Process Simulator for Calculating the Operating Conditions of the Liquefaction Cycle

3.2.1. Process Simulator

Process simulators contain the model of the process and thus contain the bulk of the constraints in an optimization problem. Based on thermodynamics, the equality constraints (so-called *equations*) include all the mathematical relations that constitute the material and energy balances, the rate equations, the phase relations, the connecting variables, and the methods of computing the physical properties used in any of the relations in the model. The inequality constraints (so-called *inequalities*) include the material flow limits and the upper and lower bounds of the operating conditions, such as the pressure and temperature. A *physical model* is a model of individual equipment in a flowsheet that can be coded, analyzed, and debugged by itself. If some data for inlet/outlet flows and equipment operating parameters are given, the physical model can calculate the rest operating conditions (dependent variables).

Two kinds of process simulators are widely used in the process simulator software. In one of these methods, the process model consists of a set of equations that serve as the optimization constraints. This representation is known as *equation-oriented simulator*. The equations can be solved simultaneously using Newton's method.

In the other method, the process can be represented on a flowsheet by a collection of physical models with coded equations to isolate them from one another. As the equipment in the flowsheet is computed using sequential calculations, this process simulator is called *sequential modular simulator*. Sequential modular simulator refers to the process simulator

based on the physical models, and the physical models solved in a sequential precedence order imposed by the flowsheet information flow. Each physical model contains the equipment sizes, the material and energy balance relations, the component flow rates, the temperatures, and the pressures.

(1) Sequential modular simulator

Sequential modular simulators are the most widely used simulators in the industry. The mathematical models that represent individual units are developed separately so that the output stream data (e.g., pressure, temperature, enthalpy, entropy) can be calculated for the given input stream data and equipment operating parameters (e.g., pressure ratio, outlet pressure). The computation proceeds by unit, through the product streams. When there are recycle loops in the process, they are torn at suitable points, and estimated values are assigned to these streams. The recycle loops are sequentially solved until the assumed values of the tear streams match the computed stream information.

The sequential modular method offers the following advantages:

- The process computations follow the material flow through the process. It is therefore easy to debug convergence failures.
- The mathematical models of different units can be developed and coded separately with different solution procedures for different equipment modules.
- New types of equipment modules can be easily added. The only criterion that needs to be kept in mind while developing the new module is that the output stream information should be calculated for the given input stream information and operating conditions of the equipment.
- The overall solution procedure is not affected by the complexities incorporated in each module.

(2) Equation-oriented simulator

The governing equations of each process unit is solved one at a time while the governing equations of all the units are solved together simultaneously in an equation-oriented simulator. In the equation-oriented simulator, each equipment module contributes the governing equations to be solved.

Table 3-2 shows an example of the equation-oriented simulator for a refrigerator. All the equality constraints are defined based on thermodynamics and on the conditions of the equipment. Although a refrigerator is a simple example of the liquefaction cycle, not a few equality constraints are used for this approach. If the number of equipment increases, the number of equality constraints may increase considerably. Thus, the equation-oriented simulator is based on coupled nonlinear equations with high complexity, and it is hard to handle errors when there is inconsistency. The addition of new equipment is not simple, and it will increase in complexity

Table 3-2 Optimization problem of a refrigerator for determining the optimal operating conditions based on the equation-oriented simulator

<p>1. Objective Function: Minimize the power provided to the compressor.</p> $f = \dot{m} \cdot w$
<p>2. Equality Constraints (so-called <i>equations</i>)</p> <p>Compressor:</p> $\dot{m} \cdot h_1(P_1, v_1, T_1) + \dot{m} \cdot w = \dot{m} \cdot h_2(P_2, v_2, T_2) \quad [1^{\text{st}} \text{ law of thermodynamics}]$ $\eta = \frac{h_S(P_2, v_S, T_S) - h_1(P_1, v_1, T_1)}{h_2(P_2, v_2, T_2) - h_1(P_1, v_1, T_1)} \quad [2^{\text{nd}} \text{ law of thermodynamics}]$ $s_1(P_1, v_1, T_1) = s_2(P_2, v_S, T_S)$ $v_1 = \frac{RT_1}{P_1} + b - \frac{a(T_1)}{P_1} \frac{v_1 - b}{(v_1 - \varepsilon b)(v_1 - \sigma b)} \quad [\text{Equation of state}]$

$$v_2 = \frac{RT_2}{P_2} + b - \frac{a(T_2)}{P_2} \frac{v_2 - b}{(v_2 - \varepsilon b)(v_2 - \sigma b)}$$

$$v_S = \frac{RT_S}{P_2} + b - \frac{a(T_S)}{P_2} \frac{v_S - b}{(v_S - \varepsilon b)(v_S - \sigma b)}$$

Condenser:

$$h_2(P_2, v_2, T_2) = q_H + h_3(P_3, v_3, T_3) \quad [1^{\text{st}} \text{ law of thermodynamics}]$$

$$v_3 = \frac{RT_3}{P_3} + b - \frac{a(T_3)}{P_3} \frac{v_3 - b}{(v_3 - \varepsilon b)(v_3 - \sigma b)} \quad [\text{Equation of state}]$$

$$P_2 = P_3 \quad [\text{Isobaric process}]$$

$$\frac{P_3}{10^5} = 10^{A - \frac{B}{T_3 + C - 273.15}} \quad [\text{Saturated pressure and temperature}]$$

Expansion Valve:

$$h_3(P_3, v_3, T_3) = (1 - v_f) \cdot h_{4,l}(P_4, v_{4,l}, T_4) + v_f \cdot h_{4,v}(P_4, v_{4,v}, T_4)$$

$$\frac{P_4}{10^5} = 10^{A - \frac{B}{T_4 + C - 273.15}}$$

$$v_{4,l} = \frac{RT_4}{P_4} + b - \frac{a(T_4)}{P_4} \frac{v_{4,l} - b}{(v_{4,l} - \varepsilon b)(v_{4,l} - \sigma b)}$$

$$v_{4,v} = \frac{RT_4}{P_4} + b - \frac{a(T_4)}{P_4} \frac{v_{4,v} - b}{(v_{4,v} - \varepsilon b)(v_{4,v} - \sigma b)}$$

$$v_4 = (1 - v_f) \cdot v_{4,l} + v_f \cdot v_{4,v}$$

Heat Exchanger:

$$\dot{m}(1 - v_f)h_{4,l}(P_4, v_{4,l}, T_4) + \dot{m}v_f h_{4,v}(P_4, v_{4,v}, T_4) + \dot{m}q_L = \dot{m}h_1(P_1, v_1, T_1)$$

$$\frac{P_1}{10^5} = 10^{A - \frac{B}{T_1 + C - 273.15}}$$

$$P_4 = P_1$$

$$\dot{m} \cdot q_L = 20 [kJ/s]$$

3. Inequality Constraints (so-called *inequalities*)

$$T_3 > T_{amb} + \Delta T_{\min} \quad [\text{Outlet temperature of the condenser}]$$

The equation-oriented simulator offers the following advantages and disadvantages over the sequential modular simulator:

- As all equations are solved simultaneously, there is no need for nested iteration loops, which makes it suitable for the simulation of strongly interconnected processes with many recycle loops.
- Good initial estimates are required for all the variables for convergence.
- It is hard to handle errors when there is inconsistency in the specifications.
- The addition of new equipment modules is not simple.
- A general-purpose, robust, nonlinear equation solver is required.
- Inequality constraints involving design variables are harder to implement in design optimization studies compared to sequential and simultaneous modular approaches.

3.2.2. Calculation of the Operating Conditions Using the Sequential Modular Simulator

Modular Simulator

(1) Tearing stream

To sequentially calculate the operating conditions of the liquefaction cycle, it should be torn at a certain stream. Tearing in connection involves decoupling the interconnections between the equipment so that sequential information flow can take place. Tearing is required because of the loops of information created by the recycle stream.

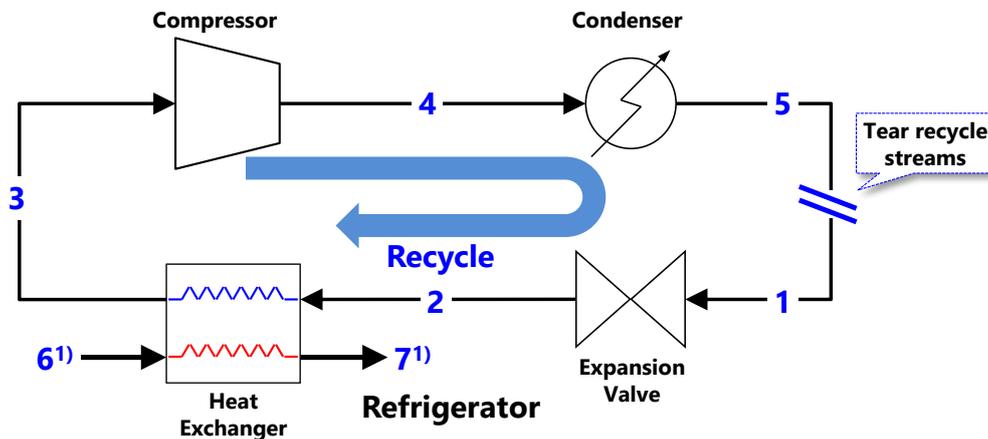


Figure 3-3 Tearing of the stream of the recycle loop

Figure 3-3 shows a simple example of tearing the stream of the recycle loop in a single cooling cycle. The conditions of outlet stream 2 can be determined from those of inlet stream 1 and from the operating characteristics of the expansion valve. The operating conditions of input stream 1 are not known but are dependent on stream 5 from the condenser, and on stream 1. Thus, the stream between the condenser and the expansion valve is torn, and the conditions of stream 1 are assumed. With assumed conditions in stream 1, the conditions of stream 2 can be calculated.

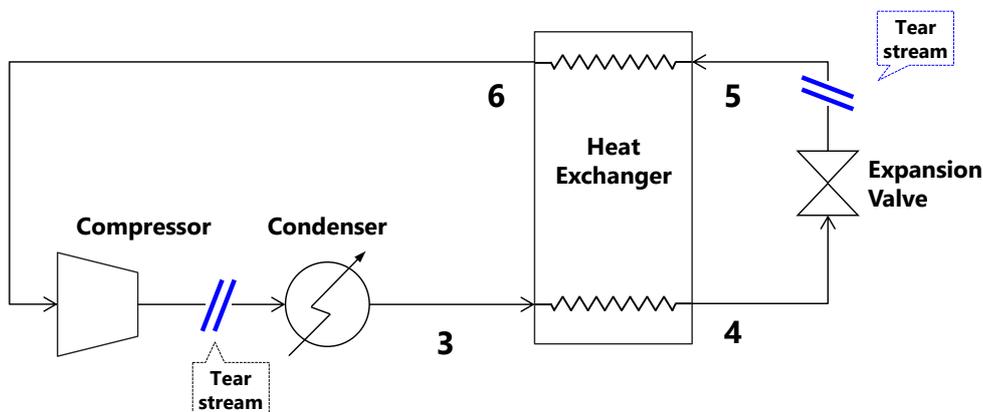


Figure 3-4 Tearing of the stream of the recycle loop and the heat exchanger

The stream should be occasionally torn on the recycle stream and the heat exchanger with multiple inlets and outlets. Figure 3-4 shows an example of tearing the stream of a simple cooling cycle with regeneration. As this cycle is a kind of recycle loop, the stream is torn in front of the condenser. If the inlet conditions of the condenser are assumed, the conditions of stream 3 can be calculated. At the heat exchanger, the conditions of outlet streams 4 and 6 can be determined from those of inlet streams 3 and 5 and from the operating characteristics of the heat exchanger. The temperature and flow rate of one of the two input streams of the heat exchanger (stream 3) are known, but the temperature and flow rate of the second input stream, stream 5 in Figure 3-4, are not known. The temperatures of streams 3 and 5 can be determined only when the temperature of stream 5 is known. The flow rate of stream 5, however, is dependent on the temperature of stream 4 as it enters the expansion valve. Thus, the conditions of streams 4 and 5 are interdependent. Streams that are interdependent are known as *recycle streams*. In such cases, the recycle is broken by tearing a recycle stream, as shown in Figure 3-4. The flow rate, pressure, and temperature of stream 5 are assumed. As the operating conditions at the torn stream should be assumed at the start of the process simulation, it is important to minimize the number of torn streams (Edgar, 2001).

(2) Sequential calculation of the operating conditions

After the tearing of the stream, the operating conditions are calculated at the torn point of the cycle. For example, the calculation will be started at stream 1 in Figure 3-5, and the conditions of stream 1 should be assumed at the start. All the equipment are modeled as physical (mathematical) models containing the equations and inequalities based on thermodynamics. They also store the parameters and variables for the iteration of the calculation.

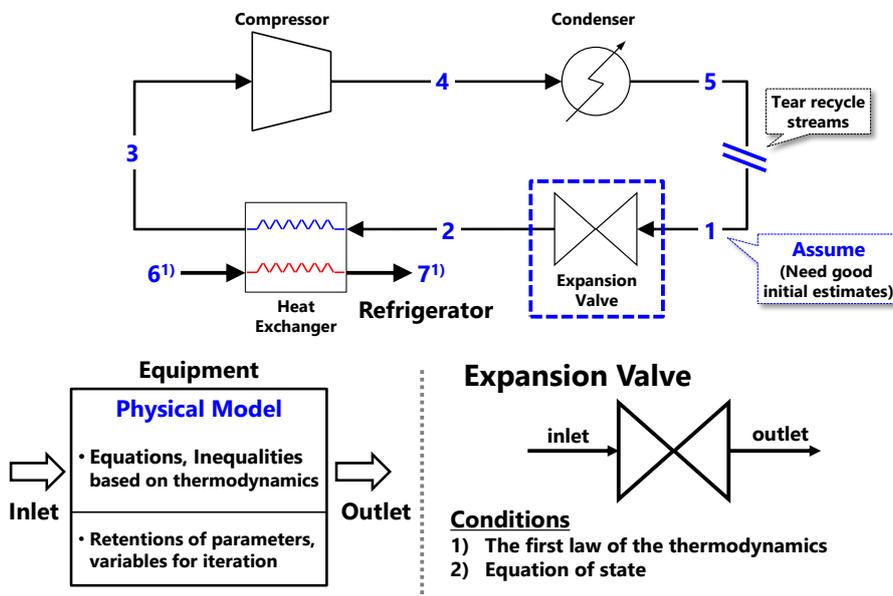


Figure 3-5 Sequential calculation of the operating conditions for each equipment

For example, an expansion valve has the equations based on thermodynamics: the first law of thermodynamics and the equation of state (Hwang, 2013). It also involves some assumptions: (a) that there is insufficient time to transfer much heat from the refrigerant, for which reason such process is assumed to be an adiabatic one; and (b) that by restricting the flow of the refrigerant, its pressure will decrease, for which reason this process is assumed to be an irreversible one. Using these equations and assumptions, the operating

conditions of the outlet at stream 2 can be calculated for the given operating conditions of the inlet at stream 1. Similarly, the operating conditions of all the streams are calculated sequentially by following through with the process flow.

(3) Checking of the convergence of the tear stream

As shown in Figure 3-6, the calculated operating conditions at stream 5 will differ from the operating conditions assumed at stream 1. The Wegstein method or a gradient method can be used to revise the operating conditions of stream 1 (Venkatarathnam, 2008). Any tear streams are assumed to have converged when their operating conditions are almost the same.

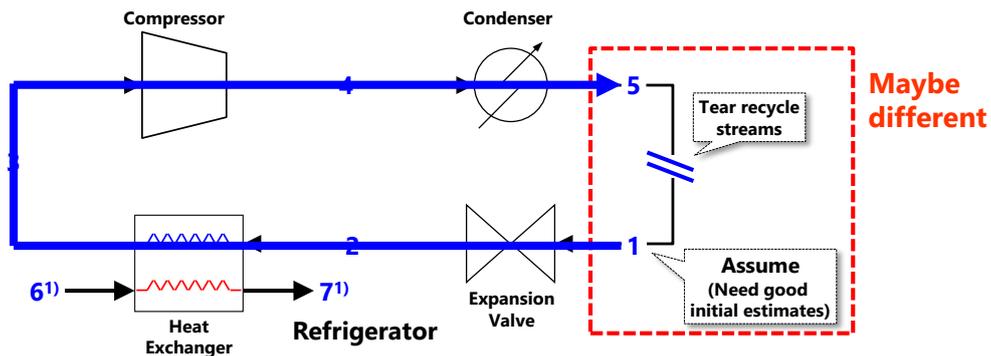


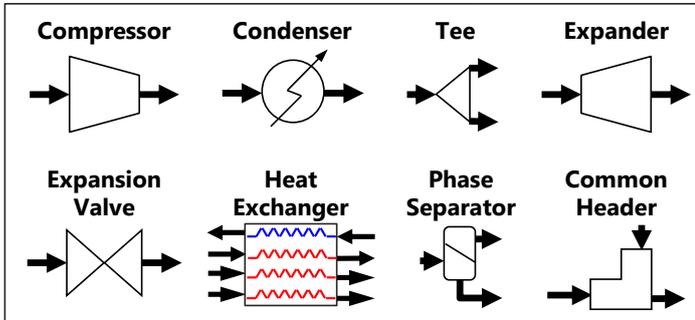
Figure 3-6 Checking of the convergence of the tear stream

3.2.3. Modeling of Equipment for Sequential Modular Simulator

For the calculation of the operating conditions using sequential modular simulator, each equipment shown in Figure 3-7 should be able to calculate the remained operating conditions (dependent variables) for given operating conditions (free variables). A *physical model* in this thesis represents a model of individual equipment in a liquefaction cycle that can be analyzed by itself. If some data for inlet/outlet flows and operating parameters of equipment are given or assumed, the physical model can calculate the rest operating conditions (dependent variables).

The physical model representing each equipment consists of two models: a mathematical model and a DEVS model. A *mathematical model* has the design variables, equations, inequalities, and assumptions of the equipment based on thermodynamics (see Figure 3-7(a)). At each thermodynamic-based mathematical model, the operating conditions of the outlet stream can be calculated from the given conditions of the inlet and from the operating characteristics of the equipment. On the other extreme, to sequentially calculate the operating conditions of the cycle according to the method of sequential modular simulator, it is necessary to manage the calculation procedure and data. A *DEVS model* manages the calculation procedure and data (see Figure 3-7(b)). In the DEVS model, the rules for sequential calculation are defined based on the DEVS formalism.

Equipment of liquefaction cycle



↓ Each equipment is modeled as a physical model

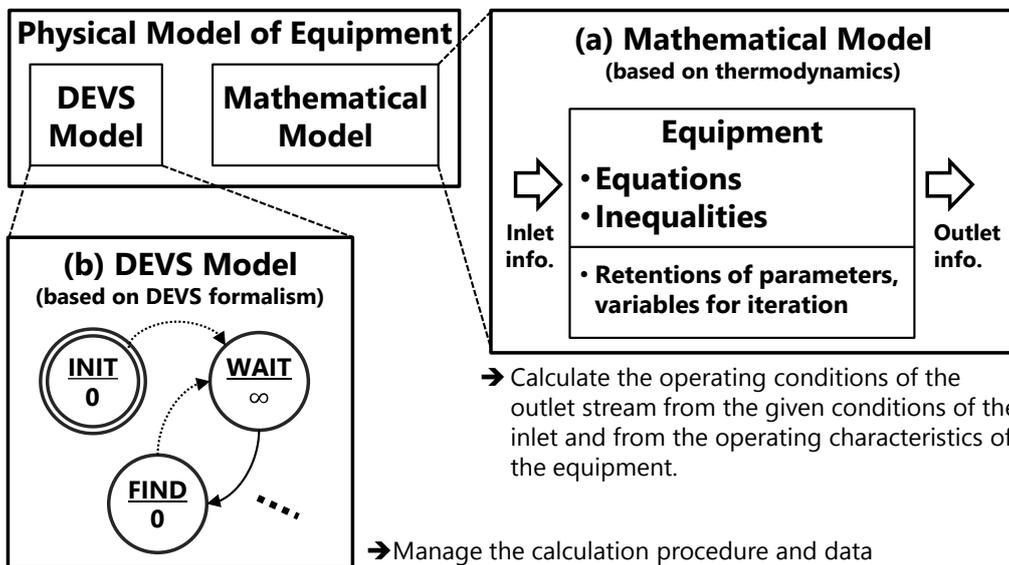


Figure 3-7 Physical model representing each equipment

(1) Mathematical model of equipment based on thermodynamics

The cycles have seven main equipment parts: the compressor, condenser, expansion valve, evaporator, phase separator, common header and tee. Many types of liquefaction cycles have been determined according to their respective syntheses. In this thesis, thermodynamic-based mathematical models of seven main equipment parts are derived to calculate the operating conditions.

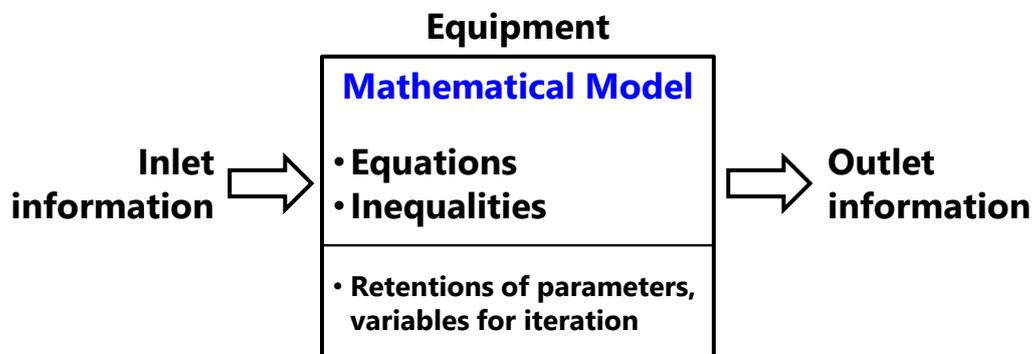


Figure 3-8 General thermodynamic-based mathematical model representing each equipment

Figure 3-8 shows a general thermodynamic-based mathematical model which represents each equipment of the cycle. This is a modular-based model of an individual element in a cycle that can be coded, analyzed, debugged, and interpreted by itself. The model comprises a set of equations and inequalities so that the equations form the constraints of the optimization problem. The operating conditions of each equipment are the stream data for the inlet and outlet, and the operating parameters of the equipment. These variables are used for the design variables of the optimization problem, and some of them are given in the problem.

① Operating conditions as design variables

Figure 3-9 shows the operating conditions of a compressor. The operating conditions are defined as the design variables, including the pressure (P , Pa), temperature (T , K), specific volume (v , m³/g) at the inlet and outlet of the compressor, and the specific work provided to the compressor (w , J/g). The operating conditions of the inlet and outlet are classified by the subscript 'i' and 'o'.

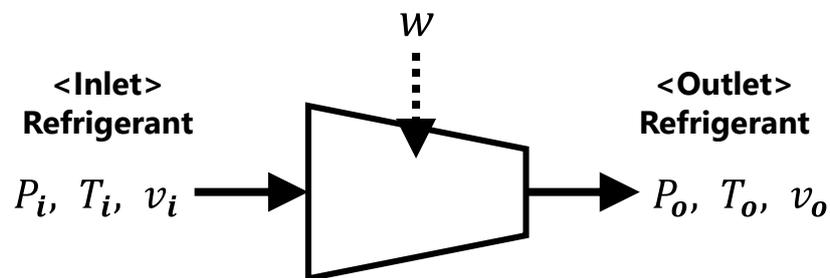


Figure 3-9 Operating conditions of a compressor

② Equations

The equations include all the mathematical relations that constitute the material and energy balances, the rate equations, the phase relations, the controls, connecting variables, and methods of computing the physical properties used in any of the relations in the model.

The first and second law of thermodynamics represent the energy balances in the model. The first law of thermodynamics is a version of the law of conservations of energy, specialized for thermodynamic systems. It is usually formulated by stating that the change in the internal energy of a closed system is equal to the amount of heat supplied to the system, minus the amount of work done by the system on its surroundings. The law of conservation of energy can be stated: the energy of an isolated system is constant.

For an open system at a steady state, the rates of energy transfer by heat and work are constant with time, and the time rate of change for the energy contained within the control volume at any time is equal to zero, so the energy balance, or first law, for the open system is as follows (Cengel, 2007):

$$\sum_{out} \dot{m} \left(h + \frac{V^2}{2} + gz \right) - \sum_{in} \dot{m} \left(h + \frac{V^2}{2} + gz \right) = \dot{Q} + \dot{W}_s, \quad (3-1)$$

where m is the mass flow rate of the refrigerant, h is the specific enthalpy of the refrigerant, V is the speed of the refrigerant in the stream, g is the gravitational acceleration, z is the height of the refrigerant above the ground, Q is the rate of heat transfer between the equipment and the surrounding environment, and W_s is the power provided from the surrounding environment to the equipment.

The second law of thermodynamics states that the entropy of an isolated system never decreases, because isolated systems spontaneously evolve towards thermodynamic equilibrium-the state of maximum entropy. For an open system at a steady state, the rates of energy transfer by heat and work are constant with time, and at a steady state, the time rate of change of the entropy contained within the control volume at any time is equal to zero. Thus, the entropy balance, or second law, for open systems is as follows (Moran & Shapiro, 2008):

$$\sum_{in} \dot{m} \cdot s - \sum_{out} \dot{m} \cdot s + \sum_j \frac{\dot{Q}_j}{T_j} + \dot{S}_{gen} = 0, \quad (3-2)$$

where S is the specific entropy, Q_j is the heat transfer between the surrounding environments at temperature T_j , and S_{gen} is the entropy generated by the irreversible process.

In thermodynamics, an equation of state is a relation between state variables. More specifically, an equation of state is a thermodynamic equation describing the state of matter under a given set of physical conditions. It is a constitutive equation which provides a mathematical relationship between two or more state functions associated with the matter, such as its temperature (T), pressure (P), volume (v), or internal energy. At very low densities, the average distance between molecules is so large that there is intermolecular independence, a situation referred to as an *ideal gas*. Under this approximation, it has been experimentally observed that to a close approximation, a very-low-density gas behaves according to the ideal-gas equation of state, as follows:

$$Pv = RT, \quad (3-3)$$

where R is the universal gas constant, whose value, for any gas, is

$$R = 8.314 \frac{N \cdot m}{mol \cdot K} = 8.314 \frac{J}{mol \cdot K}, \quad (3-4)$$

and where T is the absolute temperature in kelvins (K) (Borgnakke & Sonntag, 2009).

To improve the equation of state for liquids and vapors, the equation of state for an ideal gas is modified via experimentation and experience. As an example, consider the Soave-Redlich-Kwong (SRK) equation, which is a type of cubic equation of state, as follows:

$$v = \frac{RT}{P} + b - \frac{a(T)}{P} \frac{v-b}{(v-\varepsilon b)(v-\sigma b)}, \quad (3-5)$$

where

$$b = \Omega \frac{RT_c}{P_c},$$

$\Omega = 0.08664$ for SRK equation,

T_c : critical temperature of the refrigerant ($T_{c,Ammonia} = 405.7$ K),

P_c : critical pressure of the refrigerant ($P_{c,Ammonia} = 112.80$ bar),

$$a(T) = \psi \frac{\alpha(T_r, \omega) R^2 T_c^2}{P_c},$$

$\Psi = 0.42748$ for SRK equation,

$$\alpha(T_r, \omega) = [1 + (0.480 + 1.574\omega - 0.176\omega^2)(1 - (T_r)^{0.5})]^2,$$

$T_r = \frac{T}{T_c}$, ω : acentric factor ($\omega_{Ammonia} = 0.253$), and

$\varepsilon = 0, \sigma = 1$ for SRK equation.

To make the optimization problem simpler, there are some thermodynamic assumptions such as a reversible, adiabatic, isotropic, and isentropic condition. These are reflected to the first and second law of thermodynamics and conditional equations between inlet and outlet.

The mixed refrigerant has combined hydrocarbon components. Since the components in the mixed refrigerant have different boiling points, the refrigerant can flow gas-liquid two-phase state according to its temperature. The equations reflect this by using a vapor fraction (v_f). Equation (3-6) shows an equation of an expansion valve based on the first law of thermodynamics considering gas-liquid two-phase flow.

$$\dot{m} \cdot h_i(P_i, v_i, T_i) = \dot{m} [(1 - v_f) h_{o,l}(P_o, v_{o,l}, T_o) + v_f \cdot h_{o,v}(P_o, v_{o,v}, T_o)]. \quad (3-6)$$

③ Inequalities

The inequalities include material flow limits; pressure, temperature, and concentration upper and lower bounds; and so on. For example, to avoid damage to the compressor, the pressure of the mixed refrigerant at the inlet streams of the compressor is decreased to below the pressure of the dew point at the given temperature. This can be expressed as following inequality:

$$P_{12} \leq P_{Dew,12}. \quad (3-7)$$

In case of a condenser, with an infinite transfer area for the condenser, the temperature at the condenser outlet is equal to the ambient temperature of 25°C. Considering this practical aspect, the minimum value of the difference between the ambient temperature and the outlet temperature of the condenser can be assumed as follows:

$$T_3 > T_{amb} + \Delta T_{min}, \quad (3-8)$$

where T_{amb} is an ambient temperature and ΔT_{min} is minimum value of the difference between the ambient temperature and the outlet temperature.

The detailed descriptions of design variables, equations, and inequalities for each equipment are introduced in appendix B (Lee, 2012).

④ Methods of Solving Nonlinear Equations

To calculate the operating conditions of each equipment, the system of nonlinear equations should be solved. In this thesis, the bisection method and the Newton-Raphson method was adopted, and the values of the operating conditions (design variables) were calculated.

The bisection method is a root-finding method which repeatedly bisects an interval and then selects a subinterval in which a root must lie for further processing (Richard et al., 2008). It is a very simple and robust method, but it is also relatively slow. Because of this, it is often used to obtain a rough approximation to a solution which is then used as a starting point for more rapidly converging methods (Burden and Faires, 2010).

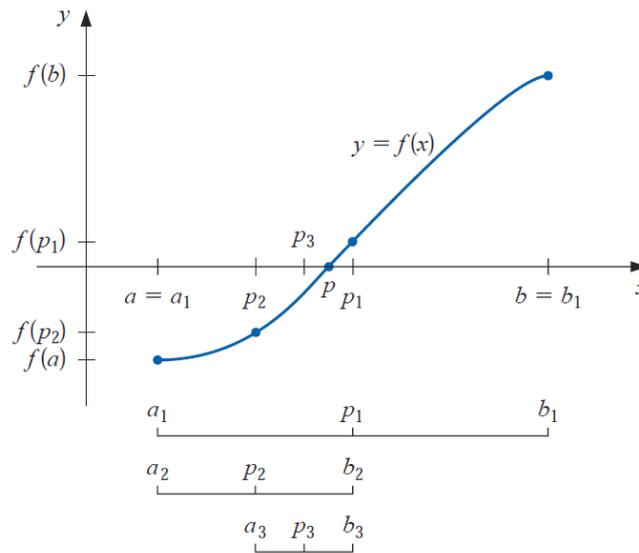


Figure 3-10 Bisection method

Newton's (or the *Newton-Raphson*) method is one of the most powerful and well-known numerical methods for solving a root-finding problem. The idea of the method is as follows: one starts with an initial guess which is reasonably close to the true root, then the function is approximated by its tangent line (which can be computed using the tools of calculus), and one computes the x -intercept of this tangent line (which is easily done with elementary algebra). This x -intercept will typically be a better approximation to the function's root than the original guess, and the method can be iterated.

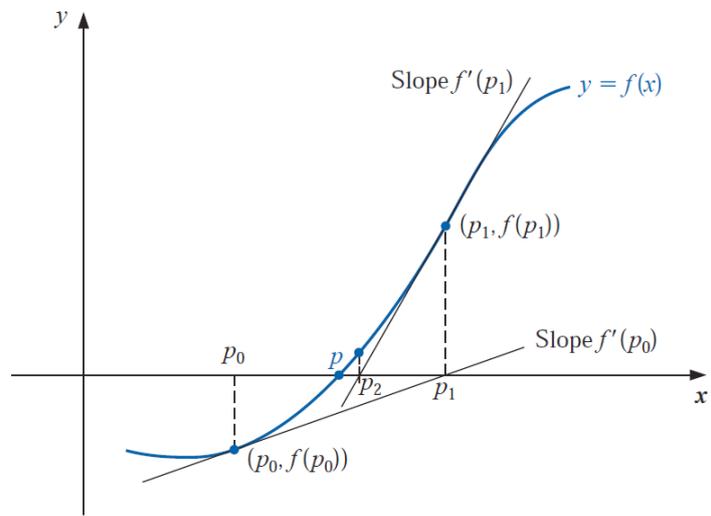


Figure 3-11 Newton-Raphson method

The detailed descriptions for these two method is explained in appendix C.

(2) DEVS model of equipment based on the rules for sequential calculation

To sequentially calculate the operating conditions of the cycle according to the method of sequential modular simulator, it is necessary to manage the calculation order and data. This can be expressed by the rules as follows.

Rules when the inlet data arrived

If the inlet data was given by previous equipment, the procedure to calculate the operating conditions of the outlet should be started. Thus, the input of the inlet data is used as a trigger. Before starting the calculation, it is needed to assume some operating conditions. In addition, if the equipment has multiple inlet, the calculation should be started after all inlet data arrived. This procedure is expressed as following rules.

Table 3-3 Rules for the calculation procedure for each equipment

IF the inlet data arrived, IF all inlet data arrived, assume and calculate the operating conditions of the outlet ELSE wait until others arrived
--

Rules for checking convergence

Each equipment calculates their own operating conditions based on thermodynamics. Because of torn stream and assumed data, there must be needed to check the convergence of torn stream and assumed data. Thus, all equipment should keep its own operating conditions in previous step, and check the convergence between previous and current data. A trigger is needed to start checking the convergence, and it is given by *Operator* model that will be explained in section 3.2.4. This procedure is expressed as following rules.

Table 3-4 Rules for checking the convergence for each equipment

IF a trigger arrived, check convergence and send a signal to the next equipment
--

DEVS formalism, a formal structure used to model the system, was adopted for equipment modeling. According to the aforementioned rules, the DEVS model is defined to manage the calculation procedure. To calculate the operating conditions of the equipment, the functions of the mathematical model are used in the DEVS model when the state has changed.

The procedure for calculating the operating conditions is divided into two steps: calculation and checking the convergence. The DEVS model in physical models have their own states to cover these two steps.

The DEVS model is categorized based on the number of flows. A compressor, a condenser, and an expansion valve have single inlets and single outlets. On the other hand, a common header has dual inlets and a single outlet, and a phase separator has a single inlet and dual outlets. A heat exchanger has multiple inlets and multiple outlets. Thus, there are three groups of the DEVS models.

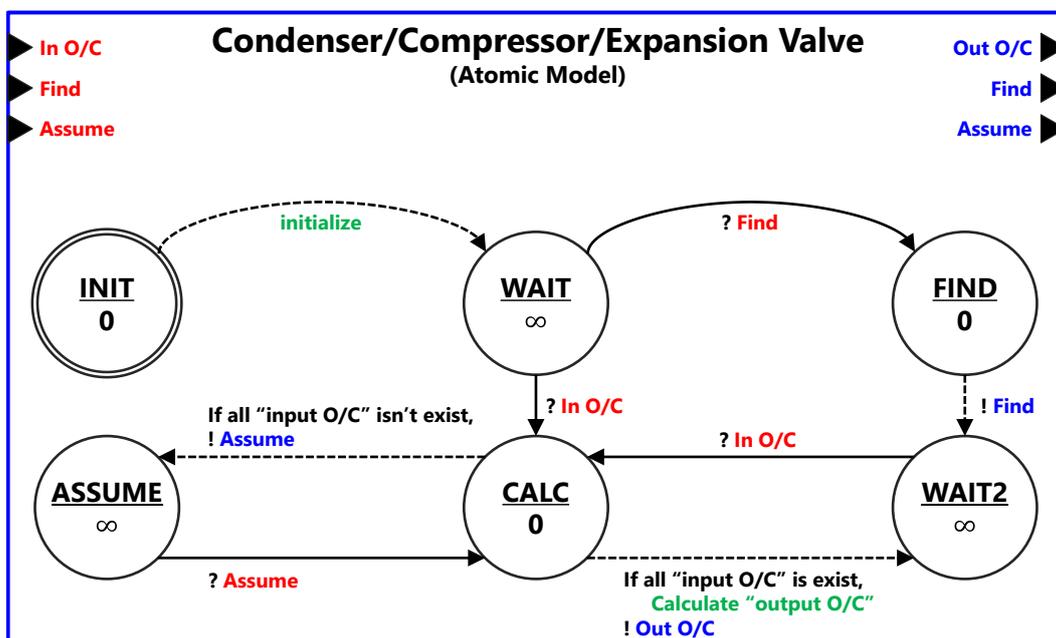


Figure 3-12 Sample DEVS model of a condenser, a compressor, and an expansion valve based on the DEVS formalism

Figure 3-12 shows a DEVS model of a condenser, a compressor, and an expansion valve that have single inlets and single outlets. Each DEVS model is characterized by variable states: INIT, WAIT, FIND, WAIT2, CALC, and ASSUME. The INIT and FIND states correspond to the procedure of tearing recycle loops. The WAIT, WAIT2, and CALC states are concerned with the procedure of calculating the operating conditions, and the ASSUME state is related with the procedures of assuming the operating conditions of the inlet and of checking the convergence. Especially, as there may be a torn stream in front of the condenser, the operating conditions of a torn stream should be assumed, and the ASSUME state performed this kind of function. The DEVS model uses the functions of the mathematical model when the state of the DEVS model changes.

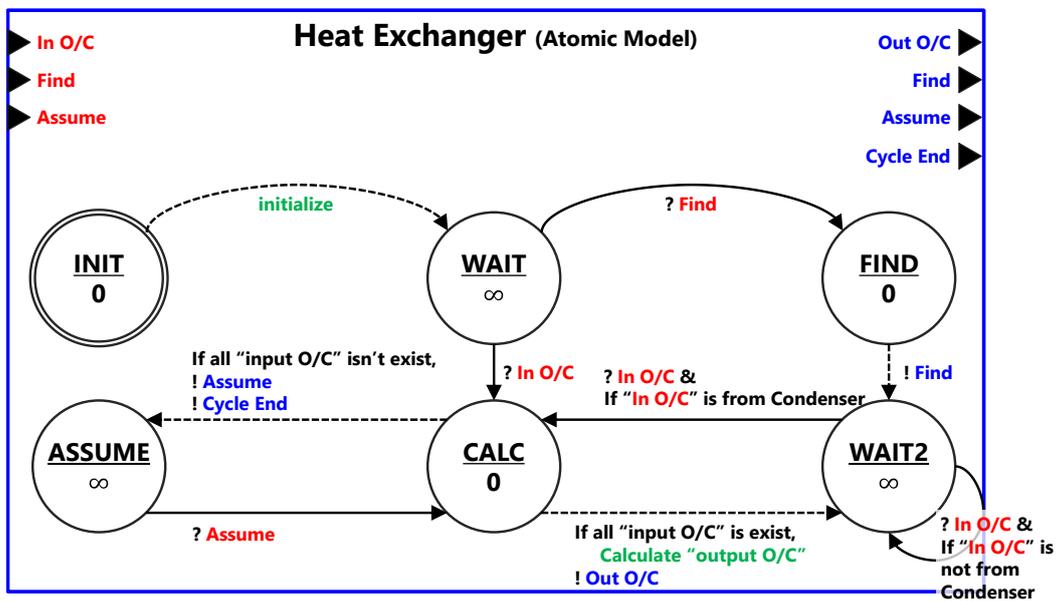


Figure 3-13 Sample DEVS model of a heat exchanger based on the DEVS formalism

Figure 3-13 shows a DEVS model of a heat exchanger based on the DEVS formalism. It is characterized by variable states: INIT, WAIT, FIND, WAIT2, CALC, and ASSUME. All the states in this DEVS model have a function similar to those of the states of a DEVS model with a single inlet and a single outlet. A heat exchanger has multiple inlets and outlets and is concerned with the recycle loops of the liquefaction cycles. Thus, there should be a torn stream at the inlet of the heat exchanger, and it must be assumed to calculate the operating conditions of the heat exchanger. This function is performed in the ASSUME state.

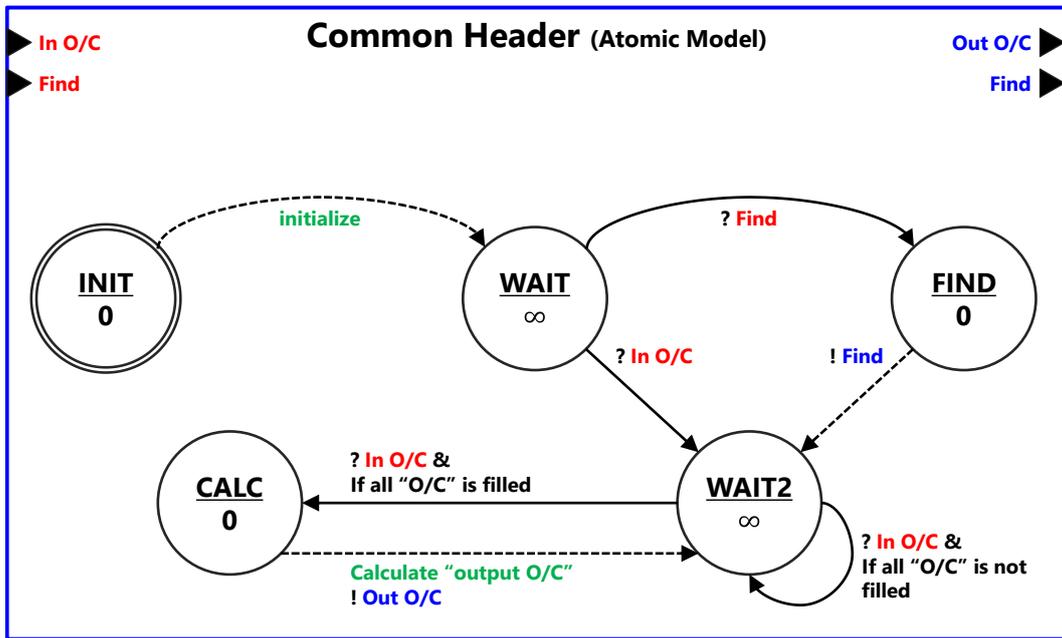


Figure 3-14 Sample DEVS model of a common header based on the DEVS formalism

Figure 3-14 shows a DEVS model of a common header based on the DEVS formalism. The common header has dual inlets and a single outlet, and the dual inlets are always given from the previous equipment (DEVS model). Thus, the DEVS model of the common header does not have the ASSUME state. The other functions in the other states are similar to those of the DEVS model of the other equipment.

Figure 3-15 shows a DEVS model of a tee and a phase separator based on the DEVS formalism. These two DEVS models have single inlets and dual outlets. They are characterized by variable states: INIT, WAIT, FIND, WAIT2, CALC, and ASSUME. All the states in this DEVS model have a function similar to those of the states of a DEVS model with a single inlet and outlet.

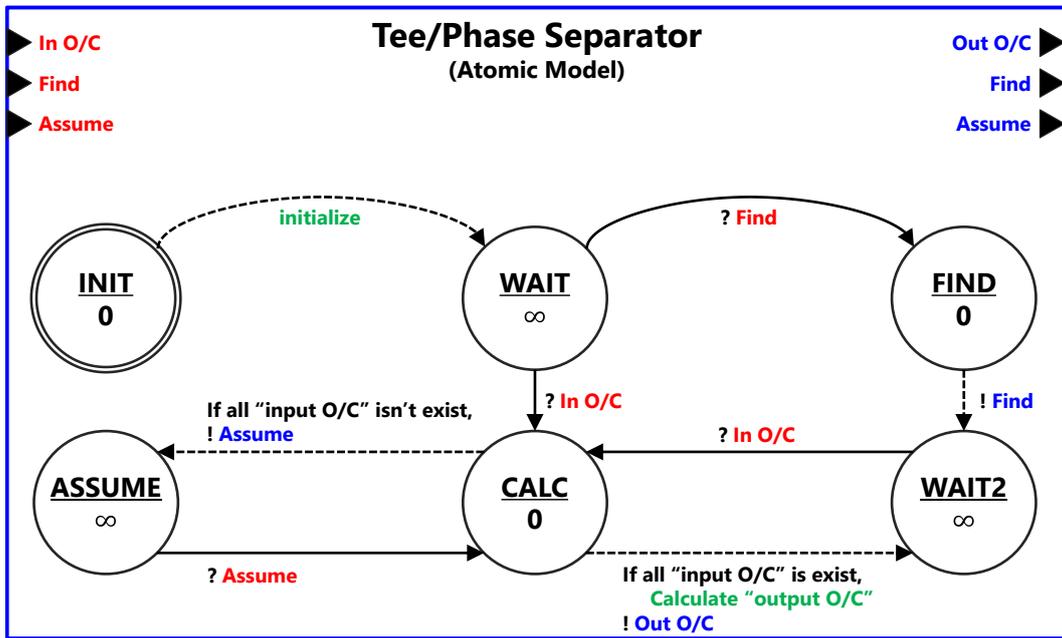
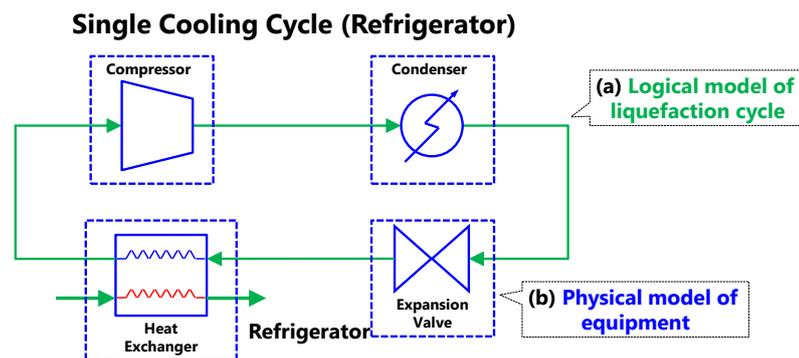


Figure 3-15 Sample DEVS model of a tee and a phase separator based on the DEVS formalism

The DEVS models manage the calculation procedure by changing their own states according to the sequential modular simulator concept, and calculate the operating conditions using the modules based on thermodynamics.

3.2.4. Making of a Process Simulator Using the SES and MB

All the physical models for equipment were stored in a specific set, named *model base* (MB). As mentioned in chapter 2, the SES has related equipment for the liquefaction cycle. For instance, the configuration of a refrigerator in Figure 3-16 can be generated from the SES. The four equipment in Figure 3-16 (compressor, condenser, expansion valve, and heat exchanger) are physical models that have the functions of calculating their operating conditions from those of the given inlet. Thus, the process simulator for calculating the operating conditions can be made by combining the logical model in the SES, which contains the relationships, and the physical model in the MB.



(a) **Logical model:** configuration of liquefaction cycle (relationship between equipment)
(automatically generated by using the system entity structure)

(b) **Physical model:** model of each equipment
for calculating the operating conditions of equipment
(Model base is a set of physical model based on DEVS formalism)

Figure 3-16 Combination of a logical model and a physical model for calculating the operating conditions using the sequential modular simulator

Figure 3-17 shows the typical process of making a process simulator by combining the SES and the MB. By pruning the SES, various alternative logical models of the liquefaction cycle can be automatically generated. The physical model for calculating the operating

conditions is configured and stored in the set of MBs.

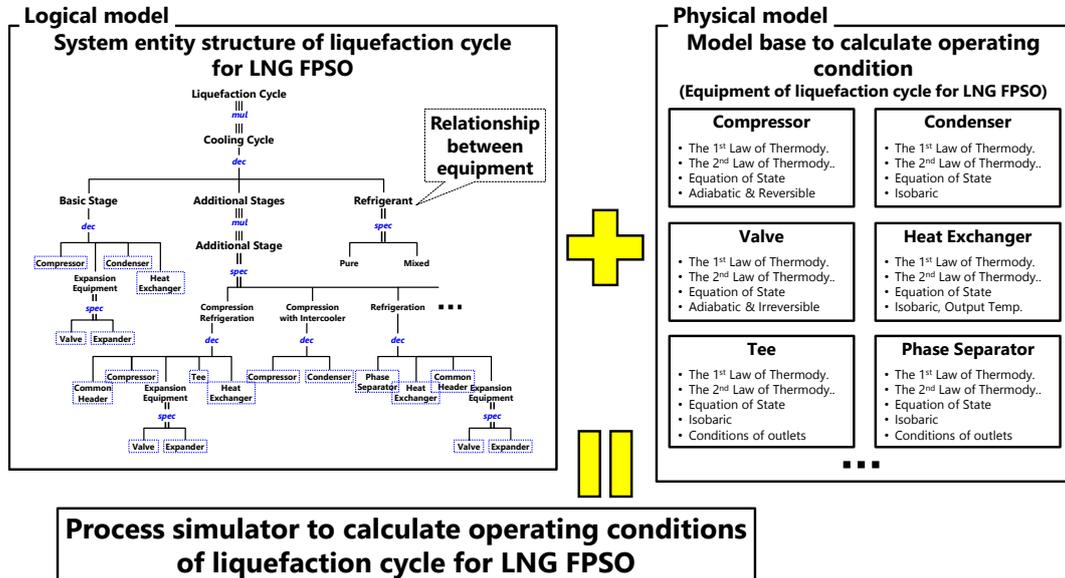


Figure 3-17 Making of a process simulator by synthesizing the SES and the MB for the liquefaction cycle

After the making of the process simulator model, it is necessary to manage the sequential calculation of the operating conditions for the equipment through the process flow. Thus, the *operator* model is attached to the process simulator, and this model manages the calculation procedure.

Figure 3-18 shows the functions of the operator model that manages the sequential calculation procedure and the convergence check. In this thesis, it is assumed that the calculation is started at the next equipment of the last condenser. Thus, at first, the operator model tries to find the last condenser in each cooling cycle. After finding the last condenser, the operator model assumes the operating conditions of the outlet of the last condenser. Then the operator model follows the cycle by calculating the operating conditions of the equipment until it meets the heat exchanger.

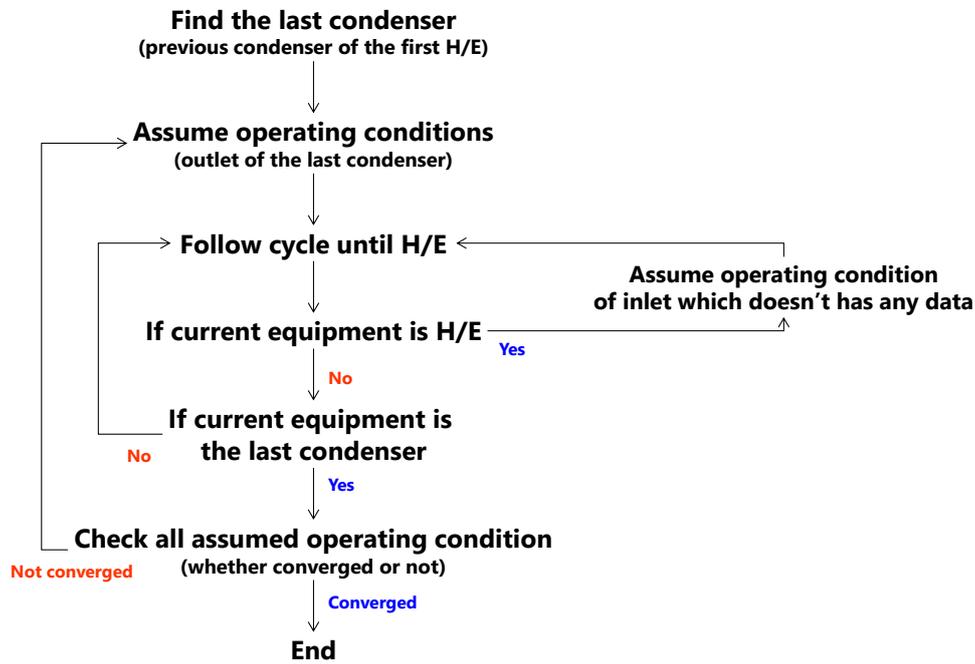


Figure 3-18 Function of the operator model that manages the sequential calculation procedure

When the operator model meets the heat exchanger, it may be necessary to assume the operating conditions of its other inlets to calculate those of its outlets. Thus, the operator model assumes the operating conditions of its other inlets which do not have any data. The operator model follows the cooling cycle again, manages the calculation of the equipment, and assumes the operating conditions, if necessary.

Figure 3-19 shows the state diagram of the operator model based on the DEVS formalism. It is characterized by variable states: INIT, FIND, FINDING, START, ASSUME, WAITING, CHECK, and END. The INIT, FIND, and FINDING states are used to find the last condenser. After finding the last condenser, the calculation may be started with the START state. The WAITING and ASSUME states are used for assuming the operating conditions of the torn streams and the inlet of the heat exchanger. The CHECK and END states are used for checking the convergence.

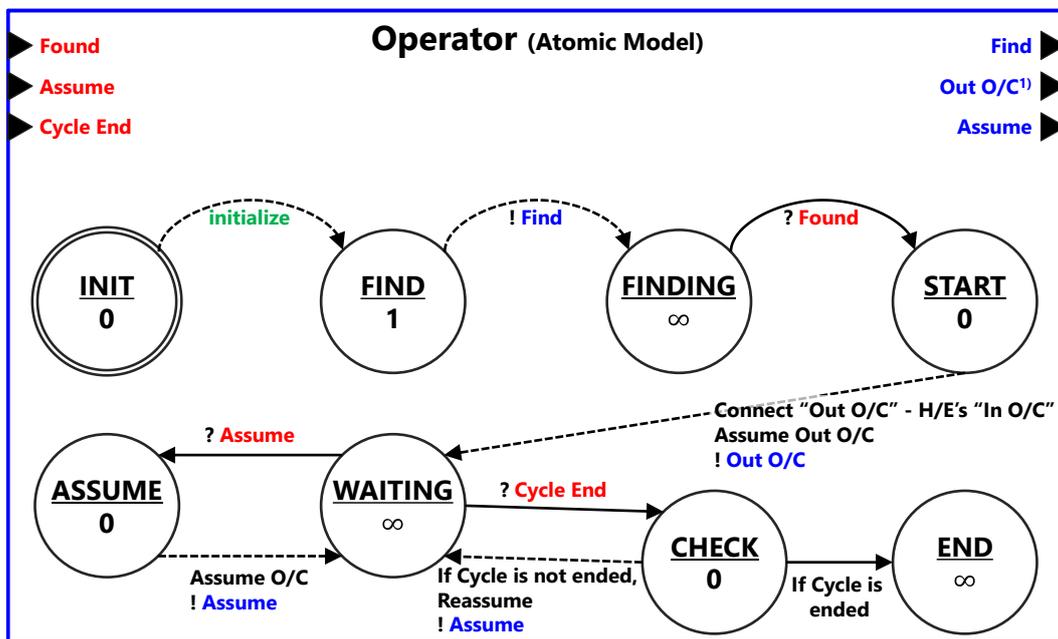


Figure 3-19 Operator model based on the DEVS formalism

All the models that correspond to process simulation are based on the DEVS formalism. As the relation between these models is given by the SES and all the models have the same formalism, the process simulator can be configured much more simply.

3.3. Optimization Methods for Determining the Optimal Operating Conditions

To solve the optimization problem, EzOptimizer (Lee et al., 2002), a program developed by these authors to solve the optimization problem using a hybrid optimization method consisting of a SQP, was used in this thesis. In the SQP, a search direction in the design space is calculated by utilizing the values and gradients of the problem functions; a quadratic programming subproblem is defined and solved. A step size along the search direction is calculated to minimize the descent function; a step size calculation subproblem is defined and solved. This method can effectively find the optimum state near the starting point. As such, SQP is used for local optimization.

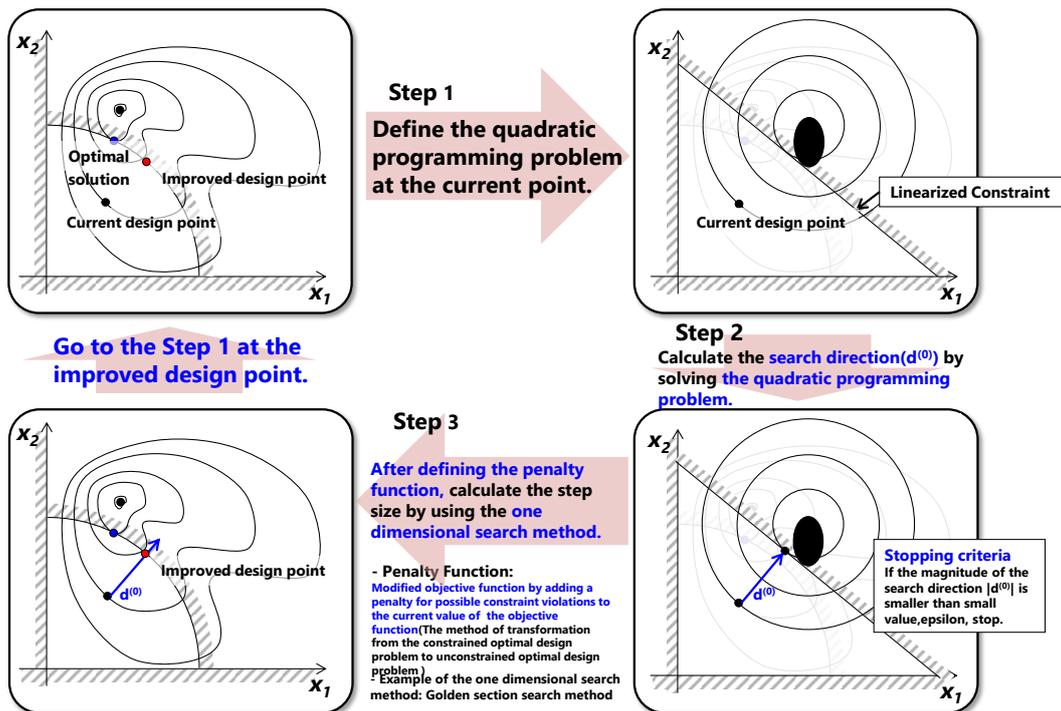


Figure 3-20 Algorithm of the sequential quadratic programming

3.4. Verification of Sequential Modular Simulator by Applying to Single Cooling Cycle (Refrigerator)

The single cooling cycle process simulation was performed to verify the mathematical models (modules) inside the physical model of the equipment. As mentioned in section 3.2.4, the single cooling cycle can be made by combining the logical model pruned from the SES and the physical model in the MB (see Figure 3-21). To calculate the operating conditions, the operator, which manages the calculation procedure and the convergence check, is attached to the process simulator model.

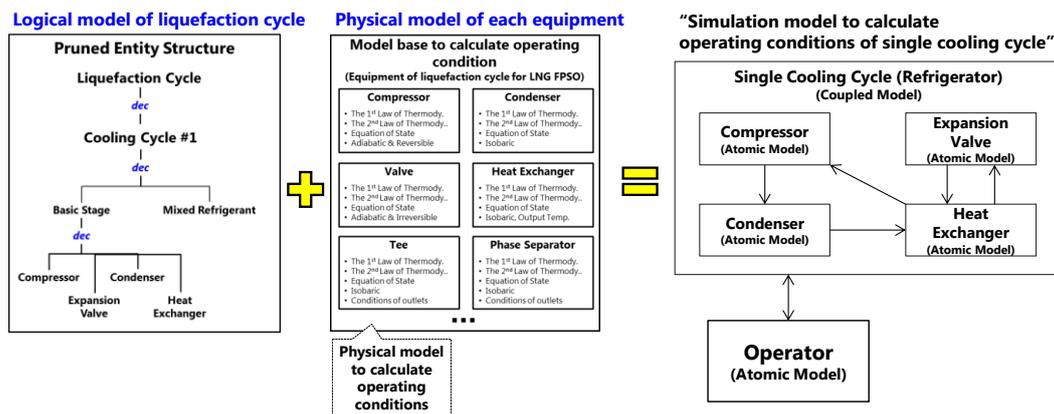


Figure 3-21 Making of the single cooling cycle by combining the logical model in the SES and the physical model in the MB

The calculation procedure using sequential modular simulator is described in detail in section 3.2.2. To verify the mathematical model of the physical model, the operating conditions were calculated, and the results were compared with those of Aspen HYSYS (Aspentech), a process simulation program that solves simultaneous equations by considering the equality constraints related to thermodynamics. As a result, the values of the other design variables that were obtained using the mathematical model in this paper

were almost the same as those obtained with Aspen HYSYS, with a maximum difference of only 0.11%, as shown in Table 3-5.

Table 3-5 Operating conditions of the single cooling cycle obtained using ASPEN HYSYS and of that obtained in this thesis

Content	HYSYS	This thesis	Difference
P ₁	3.26	3.26	-0.07%
T ₁	241.80	241.92	0.05%
V ₁	1603000.00	1601602.78	-0.09%
P ₂	33.66	33.69	0.07%
T ₂	558.10	558.31	0.04%
V ₂	1603000.00	1602190.55	-0.05%
P ₃	32.86	32.86	0.00%
T ₃	297.10	297.10	0.00%
V ₃	1603000.00	1604259.97	0.08%
P ₅	22.36	22.34	-0.11%
T ₅	218.10	218.18	0.04%
V ₅	1603000.00	1602426.93	-0.04%
P ₆	3.56	3.56	0.03%
T ₆	225.53	225.32	-0.09%
V ₆	1603000.00	1602468.50	-0.03%
Z _{Methane}	0.0075	0.0075	0.00%
Z _{Ethane}	0.745493	0.745493	-0.00%
Z _{Propane}	0.244508	0.244508	0.00%
Z _{i-Butane}	0.00225	0.00225	0.00%
Z _{n-Butane}	0.00025	0.00025	-0.00%

3.5. Calculation Results of the Optimal Operating Conditions for the DMR Cycle

The process simulation of the DMR cycle was performed to verify the physical model and the optimization module. As shown in Figure 3-22, the process simulator for the DMR cycle can be made by combining the logical model pruned from the SES (see section 2.4.5(1)) and the physical model in the MB (see section 3.2.3). To calculate the operating conditions, the operator, which manages the calculation procedure and the convergence check, is attached to the process simulator model. The *optimizer model* is added to determine the optimal operating conditions of the DMR cycle.

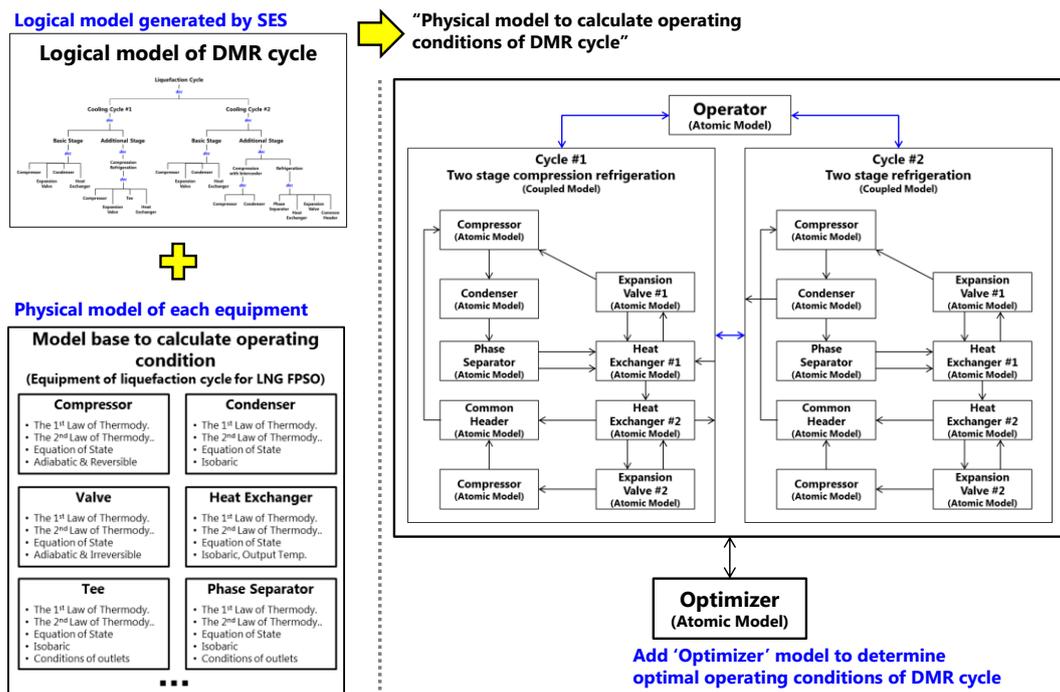


Figure 3-22 Generation of process simulation to determine the optimal operating conditions of the DMR cycle using the sequential modular simulator

(1) Given data

Some of the data for the DMR cycle are shown in Table 3-6.

Table 3-6 Given data for the DMR cycle

Category	Content	Value (given)
Natural gas feed	Pressure	65 [bar]
	Temperature	300 [K]
	Flow rate	0.748 [kg/h]
	Composition	Nitrogen = 0.04 Methane = 0.875 Ethane = 0.055 Propane = 0.021 n-Butane = 0.005 i-Butane = 0.003 i-Propane = 0.001
Compressor efficiency	$\eta_1 = \eta_2 = \eta_3 = 80\%$	
Temperature of the refrigerant leaving the seawater cooler	$T_2 = 310 [K]$ $T_{14} = 305 [K]$	

(2) Procedure of calculation

① Tearing stream

Before the calculation of the operating condition, the recycle streams should be torn because of the loops of information created by them. Figure 3-23 shows the torn result before the calculation of the operating conditions of the DMR cycle using sequential modular simulator. At first, the recycle loops should be torn. The DMR cycle has two cooling cycles: precooling and main cooling. The streams in the next of the last condenser were torn as shown in Figure 3-23 (a) and (b). The stream should be occasionally torn on the heat exchanger with multiple inlets and outlets. Thus, the inlet streams of the heat

exchangers, which are from the connected expansion valves, were torn as shown in Figure 3-23 (c), (d), (e), and (f).

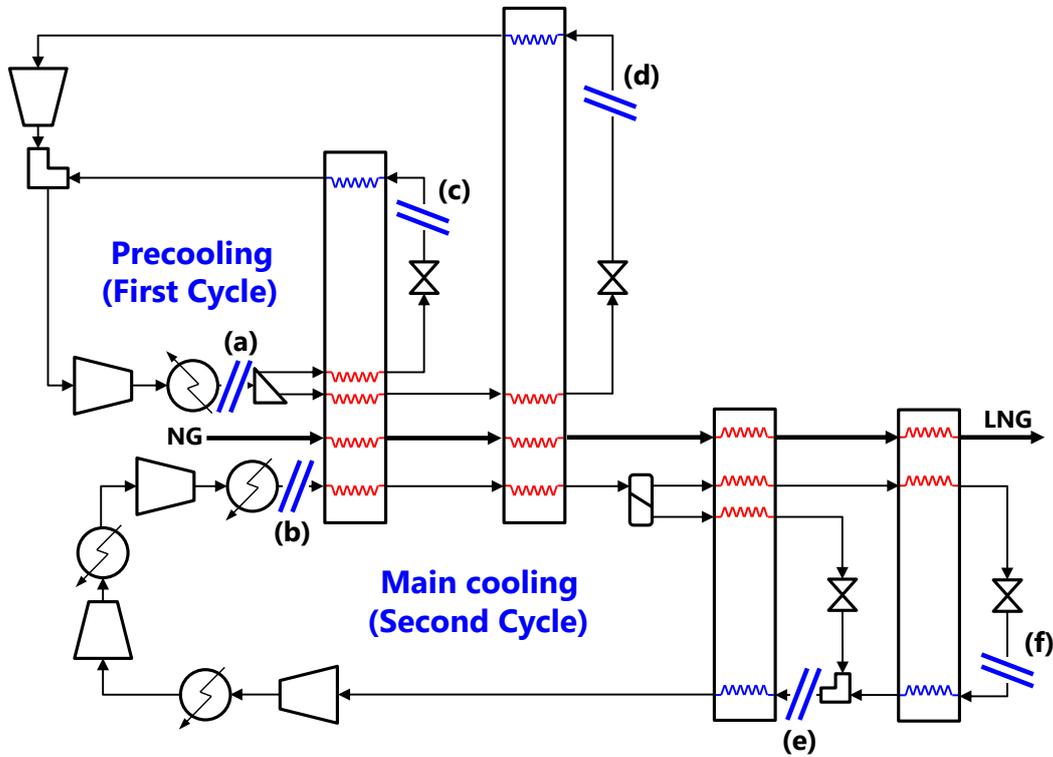


Figure 3-23 Tearing the streams of the DMR cycle

② Sequential calculation of the operating conditions

After the tearing of the streams, the calculation of the operating conditions is started at the torn point of the last condenser of each cooling cycle. For example, the calculation will be started at stream (a) and (b) in Figure 3-23. The operating conditions of stream (a) and (b) in Figure 3-23 should be reasonably assumed at the start of the calculation.

When the calculation reaches at a heat exchanger, all operating conditions of the inlets of the heat exchanger may not be given. If there are some torn streams, its operating conditions should be also assumed reasonably. If the previous equipment is still calculating

the operating conditions, it is necessary to wait until other operating conditions arrives. This calculation procedure is managed by the DEVS model in the physical model of each equipment.

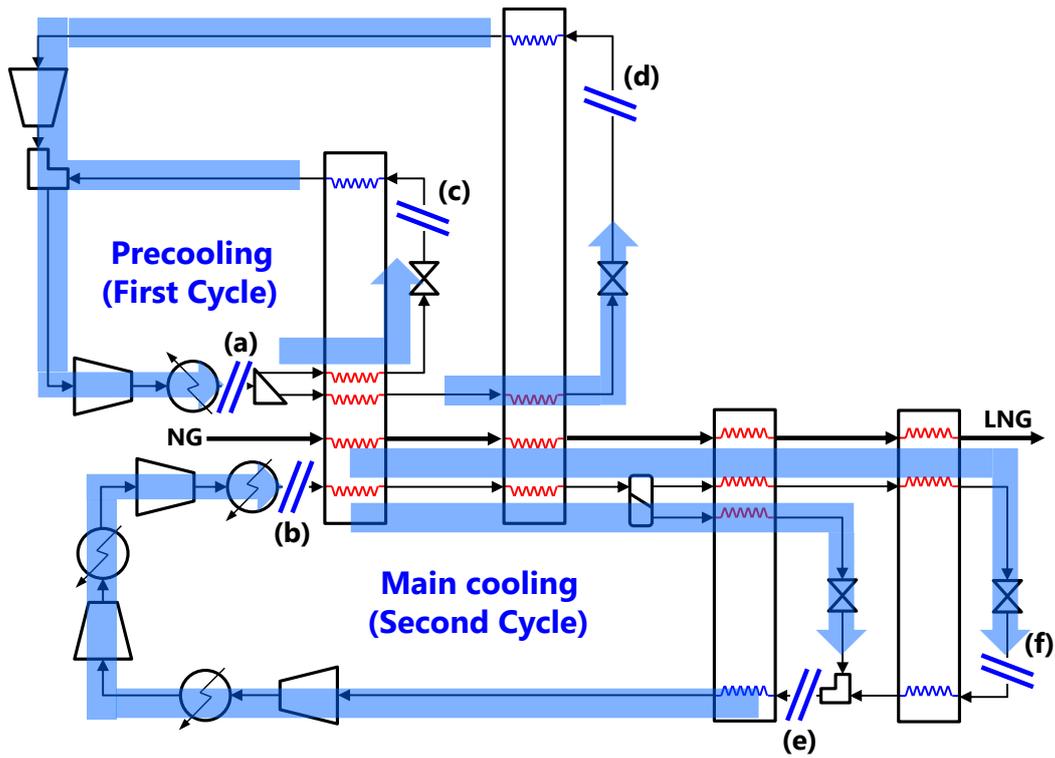


Figure 3-24 Sequential calculation of the DMR cycle using sequential modular simulator

Figure 3-24 shows the calculation of the DMR cycle using sequential modular simulator. Each arrow in Figure 3-24 means that the calculation is started with the arrow and ended with the arrow. To start the calculation for each arrow, the operating conditions of equipment located at the start of each arrow should be assumed or used the value of previous step. After the end of the calculation for all arrows, the convergence at the torn stream should be checked.

(3) Calculation results

The optimal operating conditions of the DMR cycle were obtained using the mathematical model of the cycle, and the obtained operating conditions were compared with those obtained in a past relevant study (Venkatarathnam, 2008).

To solve the optimization problem, EzOptimizer (Lee et al., 2002), a program that solves the optimization problem using SQP, was used. The results are shown in Table 3-7. The results showed a 1.57% decrease in the required power obtained in this paper (11,788 W) for the compressor in the cycle to liquefy the natural gas at the optimal operating conditions, compared to the results of a previous relevant study by Venkatarathnam (2008).

Table 3-7 Resulting optimal operating conditions for the DMR cycle using the process simulator proposed in this thesis

Content	Value	Content	Value	Content	Value	Content	Value
P ₁	19.64	P ₁₁	8.19	P ₂₁	48.92	T _{S1}	346.97
T ₁	352.31	T ₁₁	313.47	T ₂₁	140.36	T _{S11}	306.24
v ₁	0.001208	v ₁₁	0.002846	v ₂₁	0.000043	T _{S13}	395.94
P ₂	19.64	P ₁₂	8.19	P ₂₂	48.92	v _{S1}	0.001467
T ₂	310.00	T ₁₂	307.89	T ₂₂	113.00	v _{S11}	0.003234
v ₂	0.000092	v ₁₂	0.002774	v ₂₂	0.000039	v _{S13}	0.000625
P ₃	19.64	P ₁₃	48.92	P ₂₃	2.79	w ₁	2505.86
T ₃	275.01	T ₁₃	422.24	T ₂₃	105.80	w ₂	1187.98
v ₃	0.000081	v ₁₃	0.000669	v ₂₃	0.000360	w ₃	8746.96
P ₄	19.64	P ₁₄	48.92	P ₂₄	2.79	c	0.584643
T ₄	275.01	T ₁₄	305.00	T ₂₄	137.74	M _{pre}	0.932866
v ₄	0.000081	v ₁₄	0.000379	v ₂₄	0.003016	M _{main}	0.957021
P ₅	8.19	P ₁₅	48.92	P ₂₅	2.79	Z _{pre_Ethane}	0.253895
T ₅	272.01	T ₁₅	275.01	T ₂₅	137.41	Z _{pre_Propane}	0.63883
v ₅	0.000142	v ₁₅	0.000242	v ₂₅	0.001016	Z _{pre_n-Butane}	0.107275
P ₆	8.19	P ₁₆	48.92	P ₂₆	2.79	Z _{main_Nitrogen}	0.069317
T ₆	303.97	T ₁₆	239.64	T ₂₆	237.65	Z _{main_Methane}	0.405874
v ₆	0.002722	v ₁₆	0.000131	v ₂₆	0.006835	Z _{main_Ethane}	0.2964

P ₇	19.64	P ₁₇	48.92	P ₂₈	65.00	Z _{main_Propaane}	0.228409
T ₇	275.01	T ₁₇	239.64	T ₂₈	275.01	Work	11.788
v ₇	0.000081	v ₁₇	0.000068	v ₂₈	0.000290		
P ₈	19.64	P ₁₈	48.92	P ₂₉	65.00		
T ₈	239.64	T ₁₈	140.36	T ₂₉	239.64		
v ₈	0.000074	v ₁₈	0.000050	v ₂₉	0.000209		
P ₉	2.86	P ₁₉	2.79	P ₃₀	65.00		
T ₉	236.58	T ₁₉	136.08	T ₃₀	140.36		
v ₉	0.000232	v ₁₉	0.000344	v ₃₀	0.000042		
P ₁₀	2.86	P ₂₀	48.92				
T ₁₀	265.92	T ₂₀	239.64				
v ₁₀	0.007301	v ₂₀	0.000311				

In the same manner, the process simulations of other liquefaction cycles such as the C3MR cycle, DMR cycle, cascade cycle, N₂ dual expander cycle and Niche (N₂-methane) cycle can be performed. The process simulation of these liquefaction cycle is performing and they are remained as a future work.

4. Reliability Simulation of LNG FPSO

Liquefaction Cycles Based on the DEVS Formalism

All the liquefaction processes are widely applied in onshore plants and have proven reliability. The liquefaction process system for offshore plants is expected to have a higher frequency of failure than a similar land-based liquefaction process system. Further, in most cases, it will take a longer time to repair the failure at sea. The failure of the liquefaction process system may pose a serious risk both to the offshore plant and to the environment. The failure of the liquefaction process system will result in both loss of natural gas and venting of the hydrocarbon gas to the atmosphere, or gas flaring.

To ensure a high operational availability of the liquefaction process system at the lowest possible cost, the following have to be considered: (a) the configuration of the liquefaction process system that can obtain optimal redundancy; (b) the preventive maintenance (PM) program when the offshore plant is at sea; and (c) the repair strategies when the plant is at sea, and the spare parts available at the plant.

The LNG industry, including the ship owners, shipbuilders, and class societies, is interested in deciding which configuration of the liquefaction process system can best ensure high operational availability and at the same time a high safety level, without excessive investment and maintenance costs. To reduce the investment cost, attempts have been made to combine the redundant equipment for the liquefaction process system and to consider the failure rate of the equipment.

It was mentioned in chapter 3 that various alternative liquefaction cycles can be generated automatically by pruning the proposed SES. As all the alternative logical models are automatically generated by the SES, the physical model for assessing the reliability of the liquefaction cycle should be configured in a way similar to the configuration of the process simulator mentioned in chapter 3.

In the first section of this chapter, conventional reliability analysis methods are introduced. Section 2 gives a detailed description of the reliability analysis method used in this thesis. The verification of the reliability analysis method proposed in this thesis is detailed in section 3. In section 4, the reliability analysis of the conventional liquefaction cycles is described.

4.1. Reliability Analysis Method

4.1.1. Fault Tree Analysis

(1) Static fault tree analysis

There are several approaches to representing and analyzing system reliability. In tradition, fault tree (FT) analysis is a compact graphical method for analyzing system reliability (Vesely et al., 1981). Fault tree analysis (FTA) can be simply described as an analytical technique whereby an undesired state of the system is specified (usually a state that is critical from a safety or reliability standpoint), and the system is then analyzed in the context of its environment and operation to find all the realistic ways in which the undesired event (top event) can occur. The FT itself is a graphical model of the various parallel and sequential combinations of faults that will result in the occurrence of the predefined undesired event. The faults can be events associated with component hardware failures,

human errors, software errors, or any other pertinent events that can lead to the undesired event. An FT thus depicts the logical interrelationships of basic events that lead to the undesired event, the top event of the FT.

Intrinsic to an FT is the concept that an outcome is a binary event (i.e., to either success or failure). An FT is composed of a complex of entities known as “gates” that serve to permit or inhibit the passage of fault logic up the tree. The gates show the relationships of the events needed for the occurrence of a “higher” event. The “higher” event is the output of the gate; the “lower” events are the “inputs” to the gate. The gate symbol denotes the type of relationship of the input events required for the output event.

Figure 4-1 shows a simple FT for a simple serial system with two valves. The system in Figure 4-1 has two valves connected serially. If a certain valve is broken for some reason, the flow cannot follow the cycle. Thus, the system failure can be represented by using the “OR” gate, as shown in Figure 4-1.

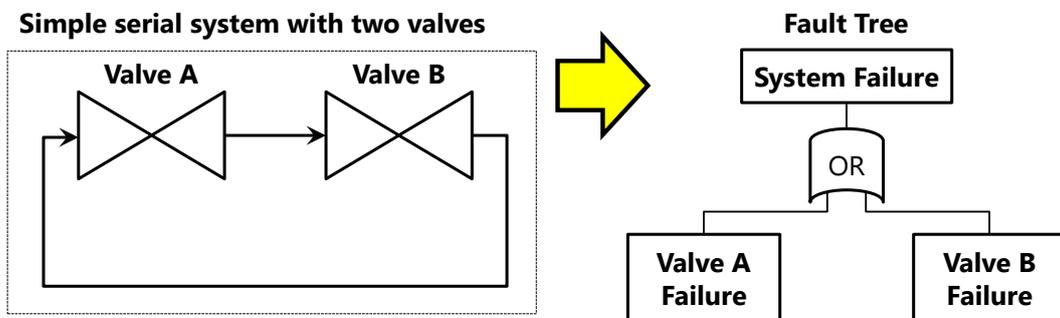


Figure 4-1 Simple FT for a simple serial system with two valves

Figure 4-2 shows a simple FT for a simple parallel system with two valves. The system in Figure 4-2 has two valves connected parallel to each other. As a certain valve is broken for some reason, the flow can follow the cycle by using another valve. Thus, the system failure can be represented by using the “AND” gate, as shown in Figure 4-2.

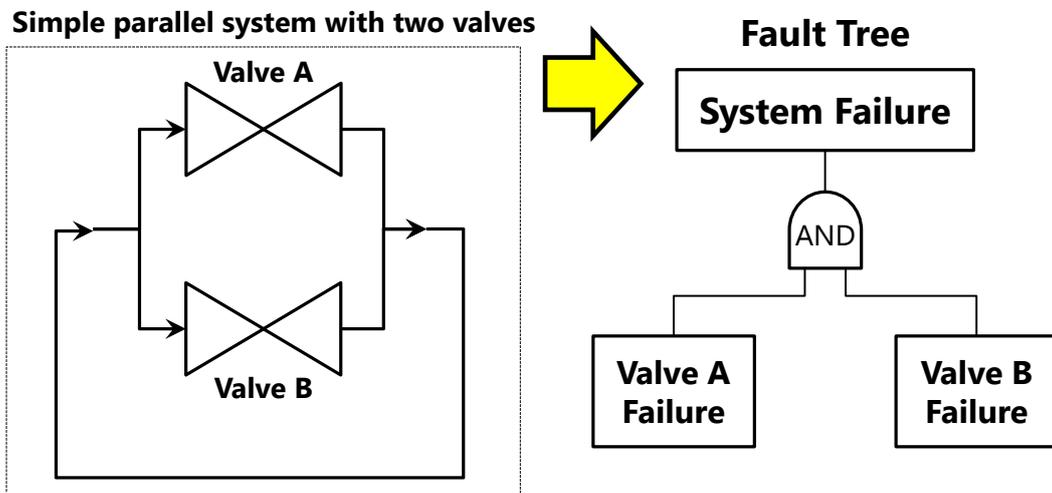


Figure 4-2 Simple FT for a simple parallel system with two valves

As shown in Figure 4-1 and Figure 4-2, the system failure is determined from the combination of the failure events of the equipment in the system. The combination of the failure events is based on the logical operator, such as “and” or “or,” and this is called *gate*. There are two basic types of FT gates: the OR gate and the AND gate. All the other gates in the FT (normally the static FT) are special cases of these two basic types. Figure 4-3 shows the gates used in the static FT.

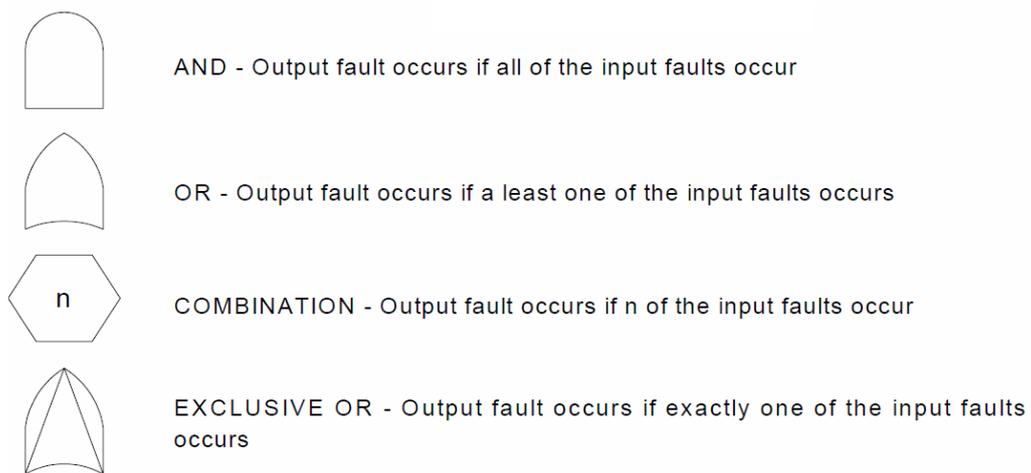


Figure 4-3 Gates of the FT (Stamatelatos, 2002)

As mentioned in the previous paragraph, the gates in the FT are based on the OR gate and the AND gate. Thus, the failure probability of the system can be determined by logical operations.

(2) Dynamic FT analysis

FTs do not provide any element or capability to model the reliability interactions among components or subsystems, aspects that are conventionally qualified as dynamic. Common examples of dynamic reliability behaviors are standby redundancy, interferences, dependencies, and common-cause failures. These arguments awakened the scientific community to the need for new formalisms, such as dynamic fault trees (DFTs) (Boyd, 1991).

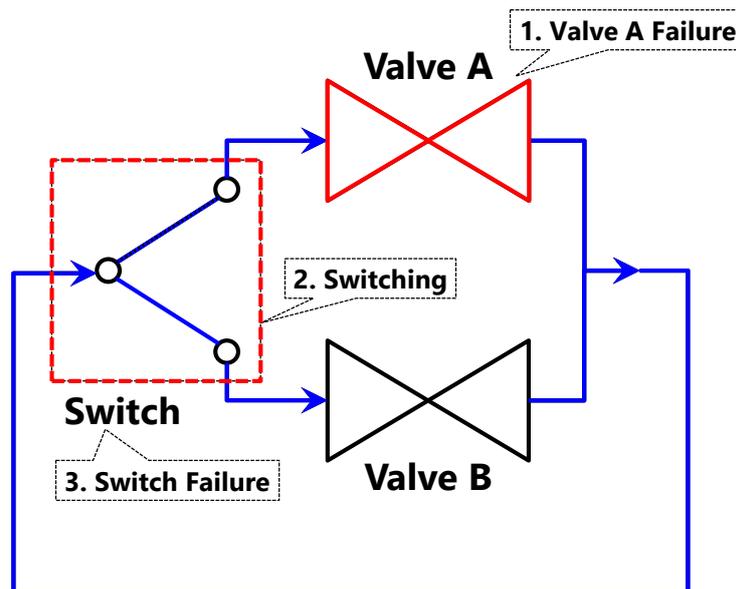


Figure 4-4 Simple redundant system with two valves and a switch: switch failure after valve A failure

Figure 4-4 shows a simple redundant system with two valves and a switch. In this system, the order of the equipment failure affects the system failure. Suppose that the failure mode

of the switch is such that when it fails, it is unable to switch between two valves. The failure of the switch only matters if switching from valve A to valve B is of interest. As shown in Figure 4-4, if the switch fails after valve A fails (and thus, valve B is already in use), then the system can continue to operate.

If the switch fails before valve A fails, however, as shown in Figure 4-5, then valve B cannot be switched into active operation, and the system fails when valve A fails. The order in which the primary and switch fail determines if the system will continue to operate.

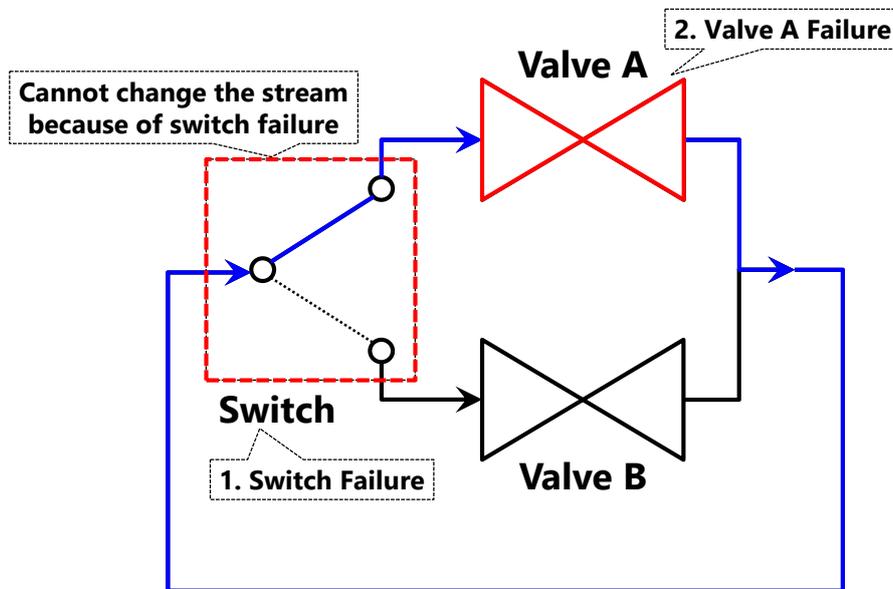


Figure 4-5 Simple redundant system with two valves and a switch: valve A failure after switch failure

This example shows that the order of the equipment failure influences the failure of the whole system. As in the static FT, there is no gate to express these dynamic reliability behaviors, gates are added to consider the dynamic behaviors, such as standby redundancy, interferences, dependencies, and common-cause failures. Figure 4-6 shows the gates used for the DFTs.

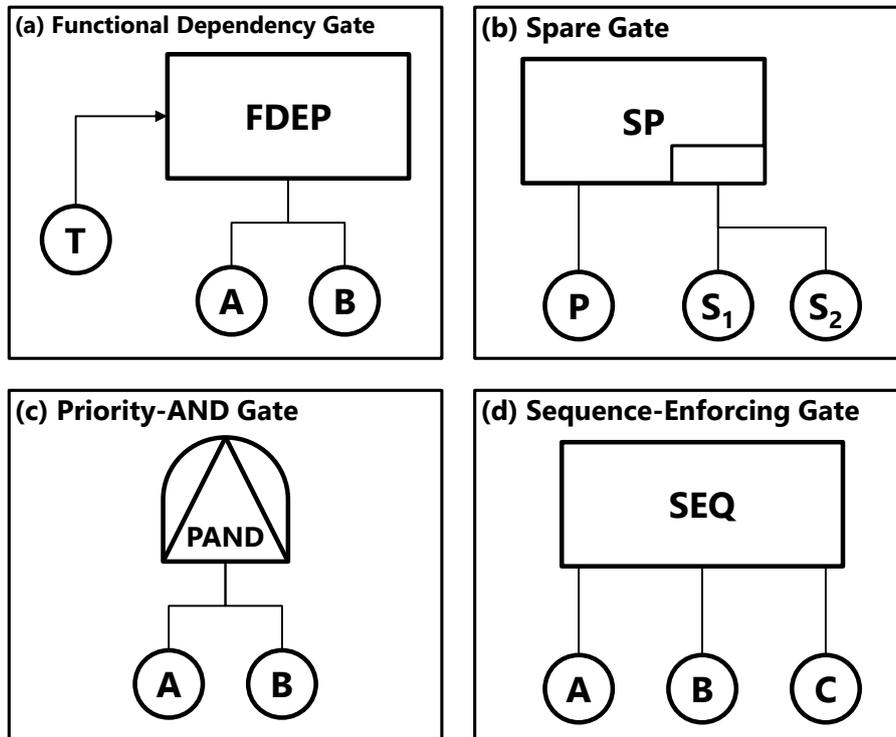


Figure 4-6 Gates of the DFT: (a) functional dependency (FDEP) gate, (b) spare (SP) gate, (c) priority-and (PAND) gate, and (d) sequence-enforcing (SEQ) gate

Figure 4-7 shows the DFT for the simple redundant system in Figure 4-4, where the PAND gate captures the sequence-dependent failure of valve A and the switch. The order of the equipment failure is considered by using the PAND gate on the “switch fails before valve A fails” node.

In the DFTs, the sequence of the failure events affects the failure of the whole system. Thus, the failure probability cannot be calculated by using a logical operation. In the DFTs, the Markov chain or the Bayesian network is used to calculate the failure probability of the system.

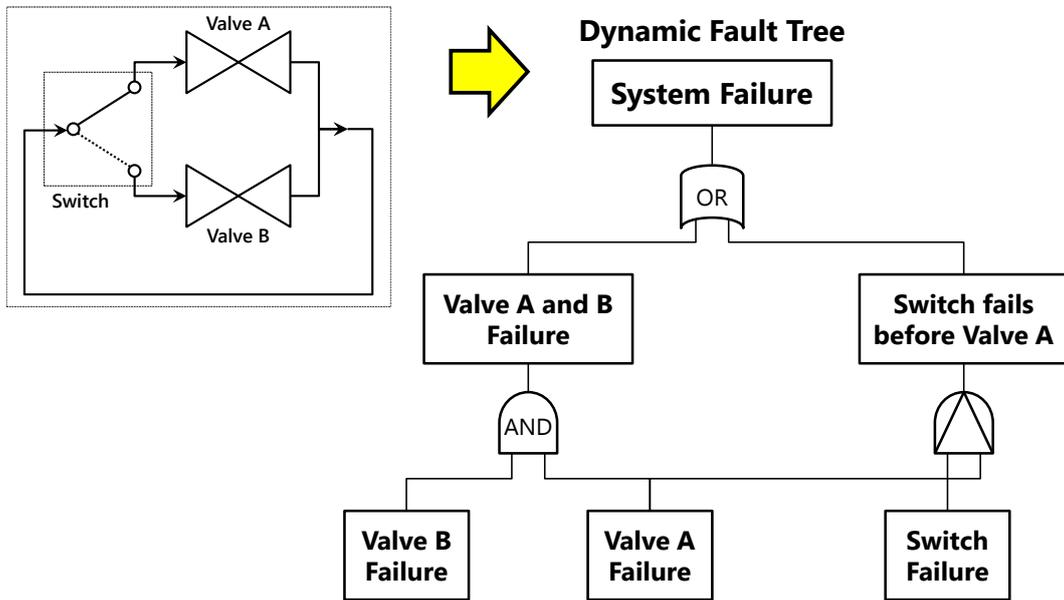


Figure 4-7 DFT for the simple redundant system with two valves and a switch

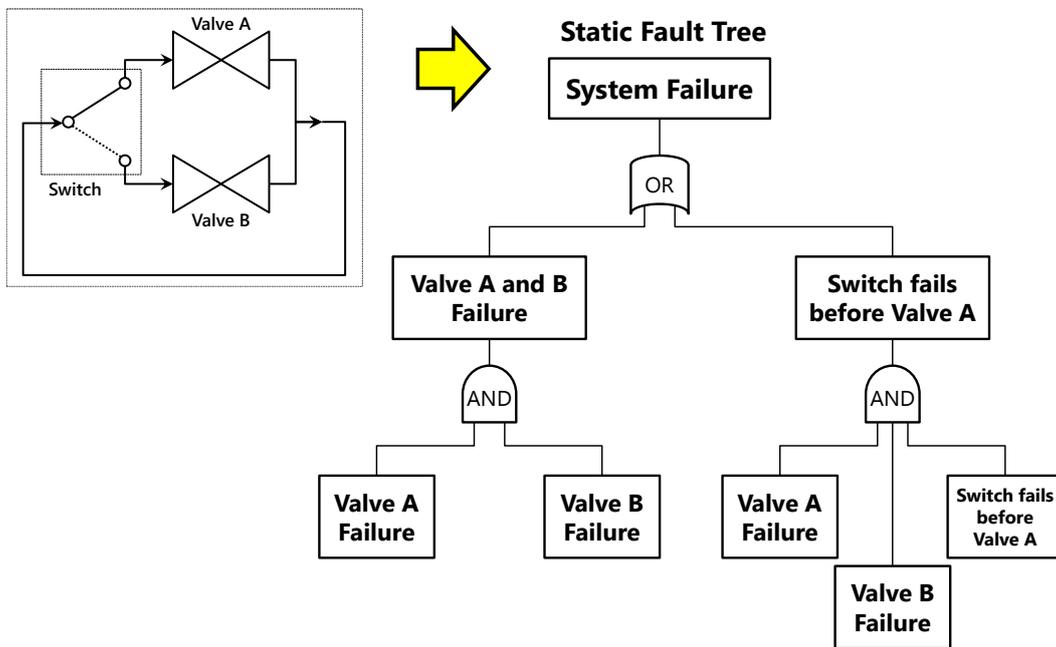


Figure 4-8 Static FT for the simple redundant system with two valves and a switch

The PAND gate can be replaced with the AND gate if the sequence conditions are stated explicitly. Thus, the DFT in Figure 4-7 can be replaced with the one in Figure 4-8. To use the static FT in Figure 4-8, the analyst must calculate the probability associated with the “switch fails before valve A fails” event.

Although many systems with sequence dependencies can be analyzed using traditional FTA, as shown in Figure 4-8, the use of DFT constructs offers an interesting alternative. The DFT, in addition to supporting an exact solution of the PAND gate, facilitates the analysis of other sequence dependencies.

(3) Calculation of failure probability using FTs

There are several methods of calculating the failure probability using FTs. In the static FT, the simplest method for the calculation involves the use of binary decision diagrams (BDDs), which is based on logical operations. The failure probability of the DFT, however, cannot be calculated by this operation because of the sequence dependencies. The DFTs are solved by conversion to the equivalent Markov chain. Monte Carlo simulation can also be used to evaluate the DFTs without conversion to a Markov chain. In an attempt to address the limitations of the solutions using the Markov chain, a solution method using conversion to the equivalent Bayesian networks was recently proposed.

① Binary decision diagrams

Binary decision diagram (BDD) is a simple method of solving static FTs based on logical operations. Figure 4-9 shows an example of the calculation of the failure probability using BDDs. λ_A and λ_B are the failure rate of the equipment. The failure rate is the frequency with which a system fails, expressed, for example, in failures per hour.

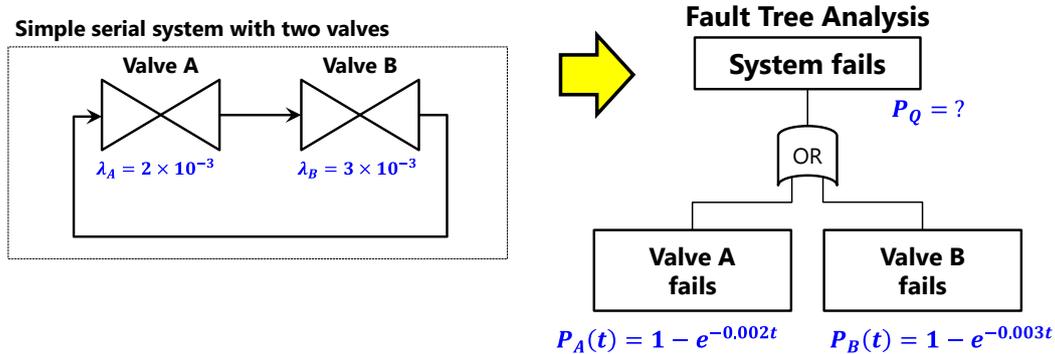


Figure 4-9 Calculation of failure probability using BDDs

Many probability distributions can be used to model the failure distribution. A common model is the exponential failure distribution, and the failure probability $F(t)$ is denoted as

$$F(t) = 1 - e^{-\lambda t}, \quad (4-1)$$

As shown in Figure 4-9, the failure probability of each valve is calculated using equation (4-1). In the FT in Figure 4-9, the OR gate is used. The failure probability of the system can thus be calculated as follows:

$$P_Q(t) = P_A(t) + P_B(t) - P_A(t) \cdot P_B(t). \quad (4-2)$$

The failure probability of system failure $P_Q(t)$ is thus calculated as $1 - e^{-0.005t}$. This means that the failure probability is dependent on time t , and that the system will fail in time (infinitely).

The BDD method is used only for static FTA. It cannot be used for the DFTs because of the sequence dependencies.

② Markov chain

The DFTs are solved by conversion to the equivalent Markov models. Figure 4-10 shows a simple example of the calculation by converting to the equivalent Markov chain.

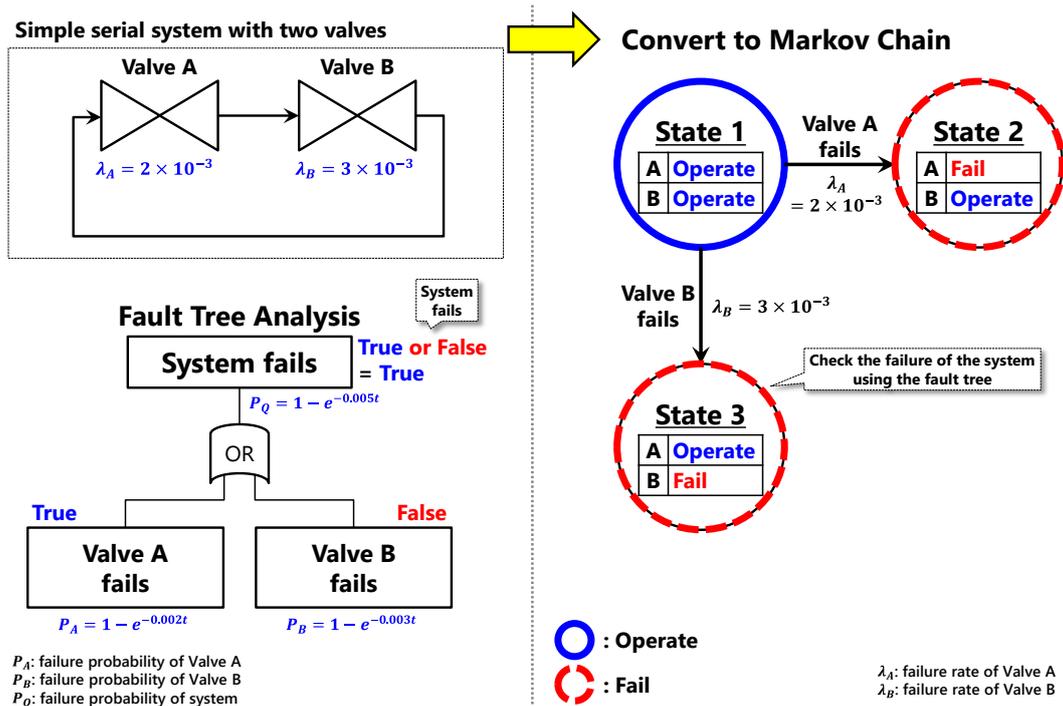
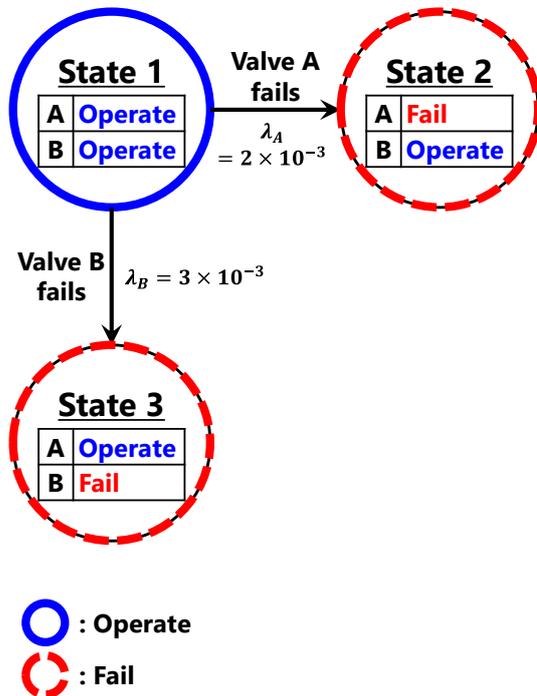


Figure 4-10 Calculation of the failure probability: conversion to the equivalent Markov chain

As shown in Figure 4-10, the starting point of the Markov chain is state 1, in which all the equipment are operating normally. From state 1, a new state is generated by certain failing equipment. If valve A fails, the new state “state 2” is added to the Markov chain, and the rate of the transition per unit time is the same as the failure rate of valve A. In the same manner, if valve B fails, the new state “state 3” is added to the Markov chain, and the rate of the transition per unit time is the same as the failure rate of valve B. For all states, the failure of the system is checked via FTA.



State 1: $\dot{P}_1(t) = -(\lambda_A + \lambda_B)P_1(t)$

State 2: $\dot{P}_2(t) = \lambda_A P_1(t)$

State 3: $\dot{P}_3(t) = \lambda_B P_1(t)$



Markov differential equations

$$\begin{bmatrix} \dot{P}_1(t) \\ \dot{P}_2(t) \\ \dot{P}_3(t) \end{bmatrix} = \begin{bmatrix} -(\lambda_A + \lambda_B) & 0 & 0 \\ \lambda_A & 0 & 0 \\ \lambda_B & 0 & 0 \end{bmatrix} \begin{bmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{bmatrix}$$

Initial Condition:

All equipment is operating at $t = 0$.

→ $P_1(0) = 1, P_2(0) = P_3(0) = 0$

Solution:

$$P_1(t) = e^{-(\lambda_A + \lambda_B)t}$$

$$P_2(t) = \frac{\lambda_A}{\lambda_A + \lambda_B} [1 - e^{-(\lambda_A + \lambda_B)t}]$$

$$P_3(t) = \frac{\lambda_B}{\lambda_A + \lambda_B} [1 - e^{-(\lambda_A + \lambda_B)t}]$$

Failure probability

= Summation of probability of all failure states

$$P_2(t) + P_3(t) = 1 - e^{-(\lambda_A + \lambda_B)t} = 1 - e^{-0.005t}$$

Figure 4-11 Calculation of the failure probability using the Markov chain

The calculation procedure using the Markov chain is shown in Figure 4-11. After converting to the equivalent Markov chain, the equation for each state is denoted as shown in Figure 4-11. From these equations, the Markov differential equation is derived, and it can be solved using the Laplace transform and the inverse Laplace transform. Finally, the failure probability of the system is calculated by summarizing the probability of the failure state in the Markov chain. As can be seen in Figure 4-11, the calculation result is the same as the result using the BDD.

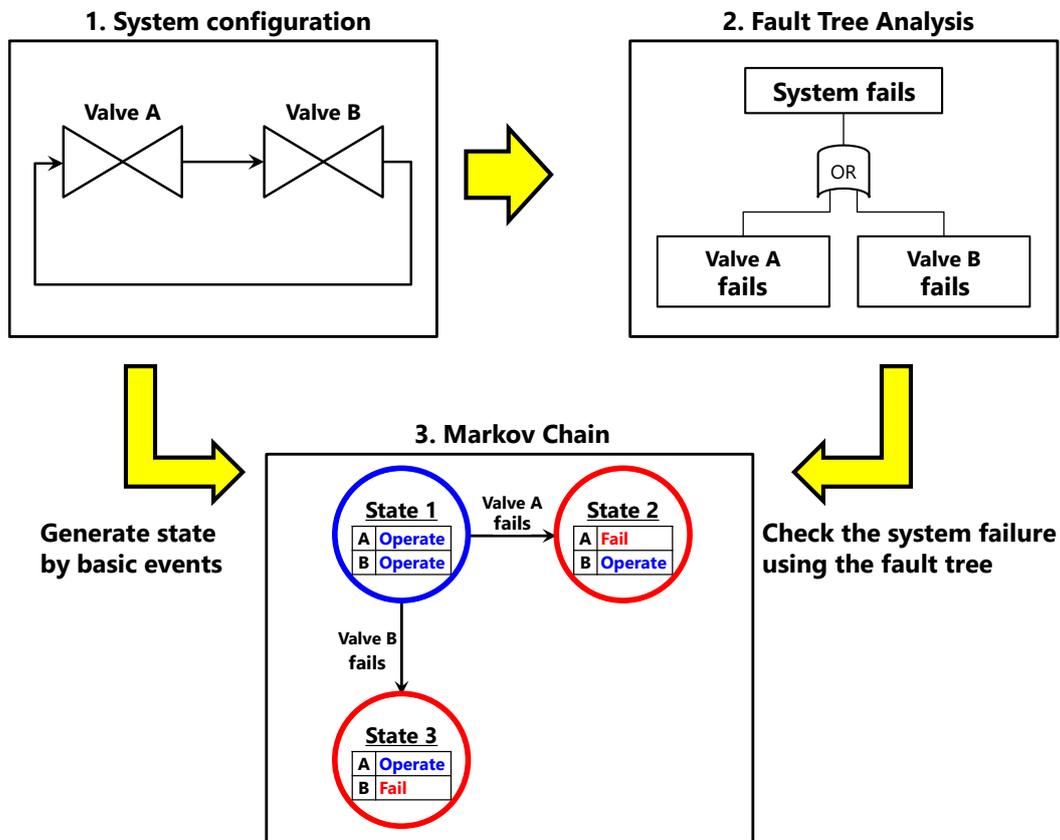


Figure 4-12 Procedure of calculating the failure probability using FTA and the Markov chain

It should be noted that the conversion to a Markov chain generally induces a state space explosion problem as the number of basic events increases. If the number of basic events is n and each event has two conditions, then the total number of states in the Markov chain is 2^n . Recently, the modular approach was proposed to minimize the state space explosion problem.

③ Bayesian network

The Bayesian network is a probabilistic graphical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph. The edges

represent the conditional dependencies, and the nodes that are not connected represent the variables, which are conditionally independent of one another. Each node is associated with a probability function that takes as an input a particular set of values for the node's parent variables, and that gives the probability of the variable represented by the node.

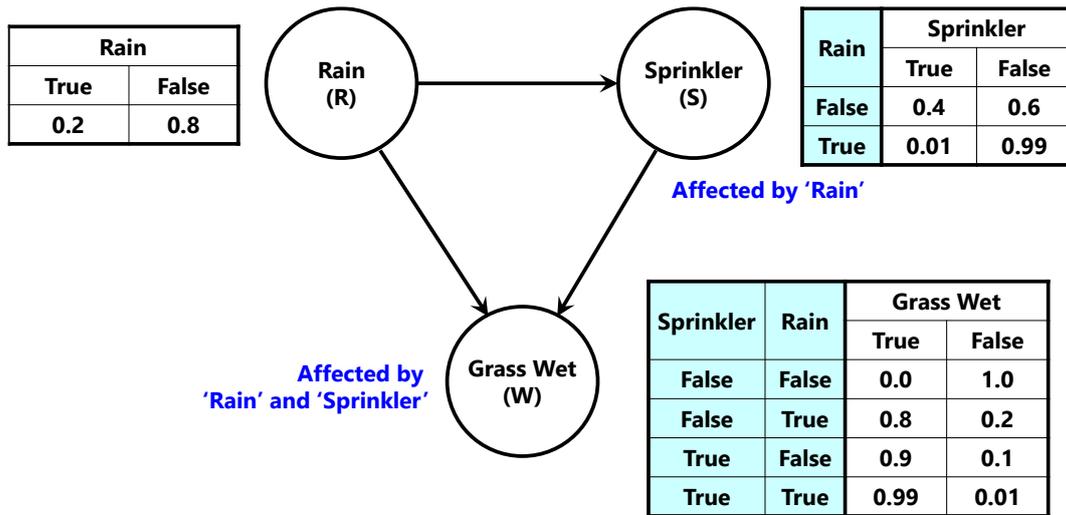


Figure 4-13 Simple Bayesian network: wet grass

Figure 4-13 shows an example of a simple Bayesian network. Suppose that there are two events that could cause grass to become wet: either the sprinkler is on or it is raining. Also, suppose that the rain has a direct effect on the use of the sprinkler (i.e., when it rains, the sprinkler is usually not turned on). Then the situation can be modeled with a Bayesian network (shown). All three variables have two possible values: T (for true) and F (for false).

The model can answer questions like “What is the probability that the grass is wet?” by using the conditional probability formula and summing up all the nuisance variables:

$$P(W = T) = \sum_{S,R \in \{T,F\}} P(W = T, S, R). \quad (4-3)$$

This can be calculated using equation (4-4).

$$\begin{aligned}
 & \sum_{S,R \in \{T,F\}} P(W = T, S, R) \\
 &= \frac{(0.99 \cdot 0.01 \cdot 0.2)}{(W=T, S=T, R=T)} + \frac{(0.9 \cdot 0.4 \cdot 0.8)}{(W=T, S=T, R=F)} + \frac{(0.8 \cdot 0.99 \cdot 0.2)}{(W=T, S=F, R=T)} + \frac{(0.0 \cdot 0.6 \cdot 0.8)}{(W=T, S=F, R=F)} \quad (4-4) \\
 &\therefore P(W = T) = 0.4484
 \end{aligned}$$

As pointed out explicitly in the example, the probability can be calculated by each iteration of the summation.

In an attempt to address the limitations of the solutions using the Markov model, a solution method using conversion to the equivalent Bayesian networks was recently proposed (Bobbio et al., 2001). The converted Bayesian network can be solved as shown in the previous paragraph.

Figure 4-14 shows the conversion of an FT to the equivalent Bayesian network. A simple system with two valves can be represented by the FT, as shown in the left side of Figure 4-14. The OR gate of the FT is converted to the equivalent Bayesian network, as shown in the right side of Figure 4-14. The probability table of the “system fails” node is also automatically generated by pre-defined conversion rules.

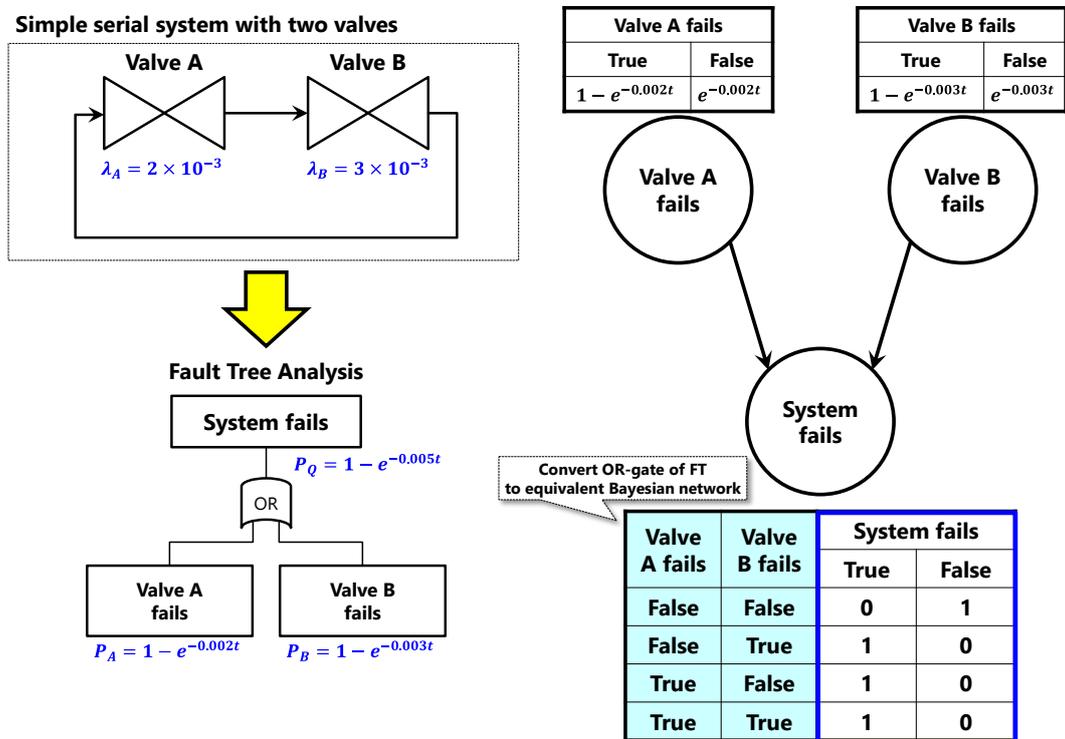
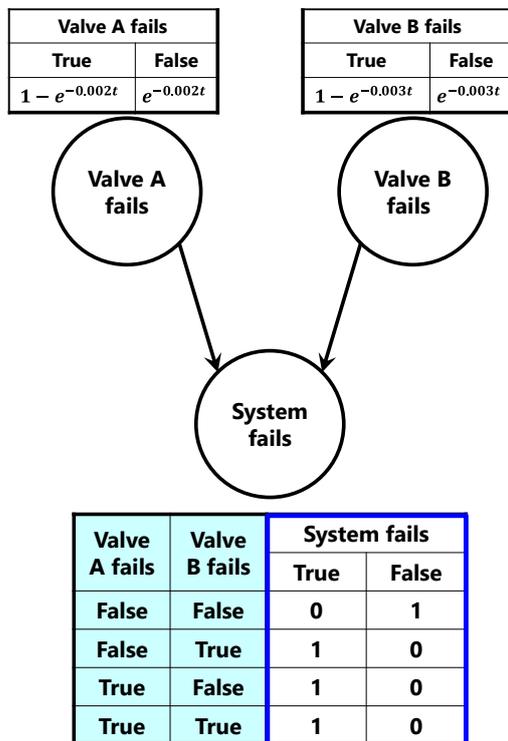


Figure 4-14 Conversion of an FT to the equivalent Bayesian network

Figure 4-15 shows the calculation result of the failure probability using the Bayesian network. The failure probability is calculated by using the conditional probability formula and summing up all the nuisance variables.

As shown in Figure 4-14, the structure of the Bayesian network is equivalent to that of the FT of the same system. Thus, the calculation procedure is much simpler than the Markov chain when the basic event increases. The Bayesian network methods noted above transform the DFT to the Bayesian network as a similar shape. In contrast with those methods, an approach was proposed to convert to a dynamic Bayesian network that has sets of nodes classified by a time axis.



Failure probability:

$$P(Q = T) = \sum_{A,B \in \{T,F\}} P(Q = T, A, B)$$

$$= P(Q = T, A = T, B = T) + P(Q = T, A = T, B = F) + P(Q = T, A = F, B = T) + P(Q = T, A = F, B = F)$$

$$= 1 \cdot (1 - e^{-0.002t}) \cdot (1 - e^{-0.003t}) + 1 \cdot (1 - e^{-0.002t}) \cdot e^{-0.003t} + 1 \cdot e^{-0.002t} \cdot (1 - e^{-0.003t}) + 0 \cdot e^{-0.002t} \cdot e^{-0.003t}$$

$$\therefore P(Q = T) = 1 - e^{-0.005t}$$

Same as the result of the fault tree

Figure 4-15 Calculation of the failure probability using the Bayesian network

4.1.2. Reliability Block Diagram

In the previous section, three methods of calculating the system failure probability were introduced: BDDs, the Markov chain, and the Bayesian network. BDD can be used only for the static FT, but the Markov chain can calculate the failure probability of the DFT. As the Markov chain has the state space explosion problem as the number of basic events increases, the Bayesian network is proposed for probability calculation.

All these methods need FTs for calculating the failure probability of the system. An expert is needed to make the FT of a certain system. It also entails a high cost and has the risk of producing different results depending on the selection of the logical gate of the FT, which is dependent on the experience of the expert.

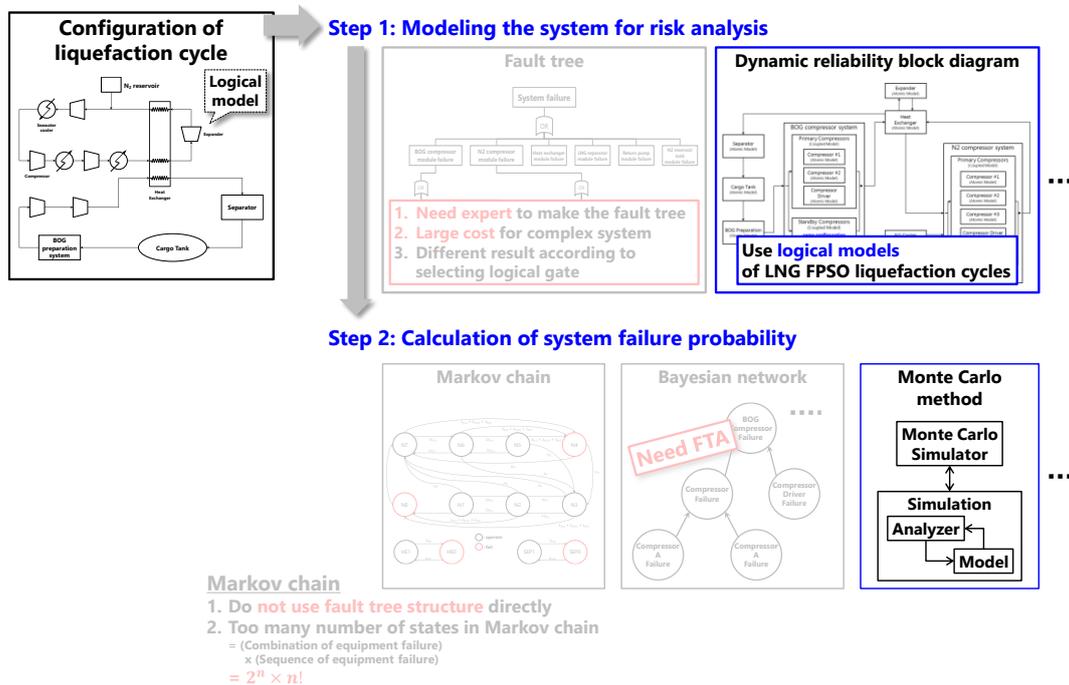


Figure 4-16 Reliability analysis method used in this thesis

Thus, the key idea in this thesis came from the use of the system configuration to assess the reliability instead of making an FT; this is the reliability block diagram (RBD) method. The RBD method is a graphical presentation of a system diagram connecting the subsystems of components according to their functions or reliability relationships. It is a user-friendly method, and it is easy to obtain a model directly from the specifications. From this concept, a new dynamic modeling method involving the use of dynamic reliability block diagrams (DRBDs) was proposed based on an extension of the existing RBD formalism.

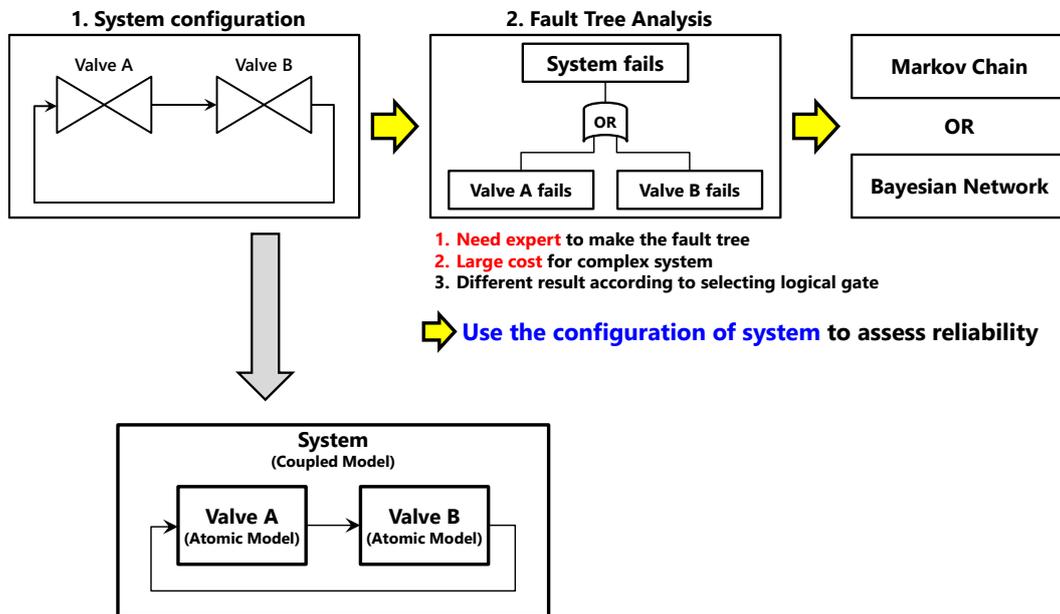


Figure 4-17 Reason for the use of an RBD in this thesis

The main advantage of DRBD is the capability to model dependencies among subsystems or components concerning their reliability interactions. As mentioned in chapter 2, the configuration of the liquefaction cycle is expressed as the SES. Furthermore, various alternative logical models for the liquefaction cycle can be automatically generated by the pruning process. As these logical models have the relationships between the equipment, they can be naturally converted to the equivalent RBD.

A quantitative analysis of DRBD can be conducted using the existing methods, such as the Markov chain and Monte Carlo simulation. In this thesis, Monte Carlo simulation is used to calculate the failure probability.

4.2. Reliability Simulation Based on the DEVS Formalism

In this thesis, the RBD method is used to assess the reliability of the liquefaction cycles. The logical models automatically generated by pruning the SES can be converted to the equivalent RBD because of the similar structures. All the physical models are configured to determine the failure of the equipment, and based on the DEVS formalism.

4.2.1. Procedure of Reliability Simulation

Figure 4-18 shows the PES and the logical model of a refrigerator. The PES has all the equipment and their relationships and thus represents the logical model of the liquefaction cycle. It is also similar to the concept of the RBD and can thus be simply converted to RBD.

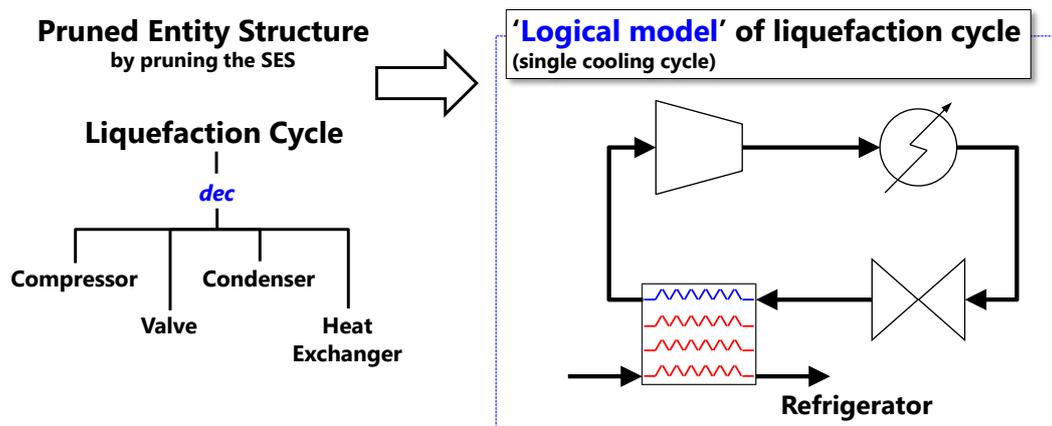


Figure 4-18 PES and the logical model of a refrigerator

Figure 4-19 shows an example of the conversion of the logical model to the equivalent RBD to assess the reliability of a refrigerator. As shown in the right side of Figure 4-19, the equipment and interconnection of the logical model are converted to the RBD as they are.

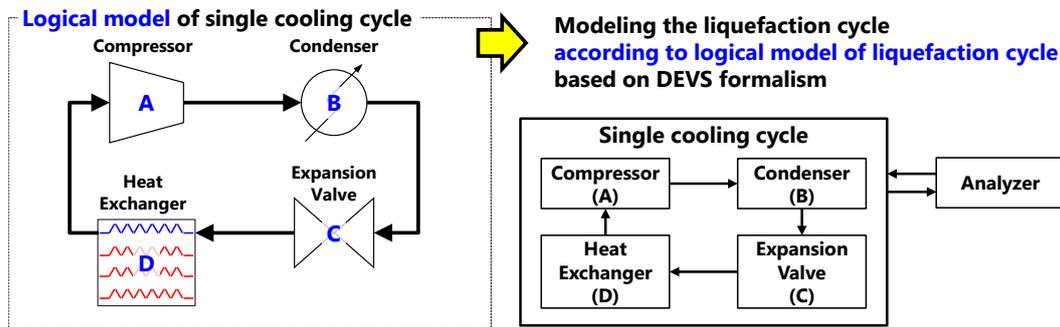


Figure 4-19 Conversion of the logical model to the equivalent RBD to assess the reliability of a refrigerator

Figure 4-20 shows the procedure of the probability calculation using the RBD. Suppose that the operating time of the liquefaction cycle is 20 hours and that the failure rate of the equipment is given as shown in Figure 4-20.

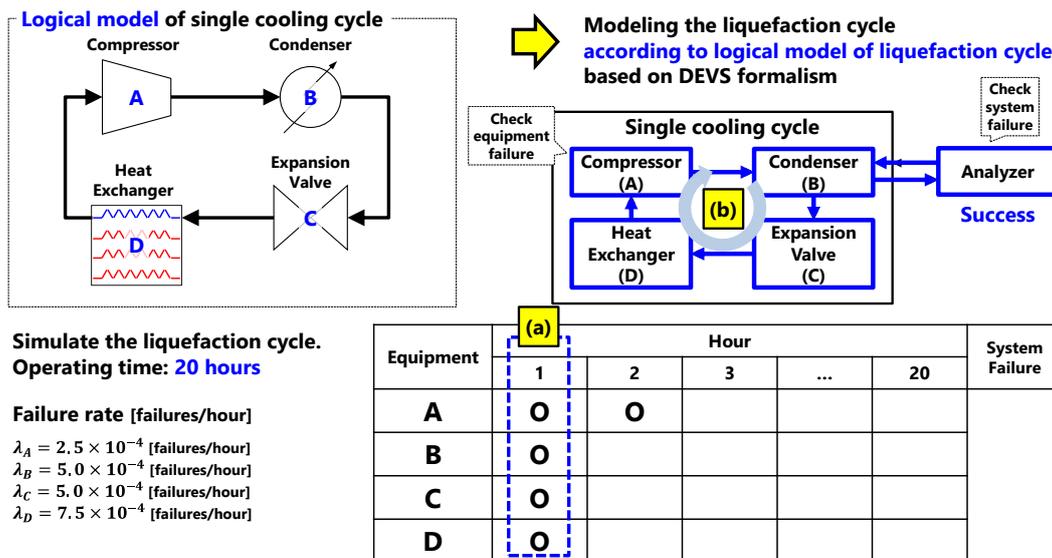


Figure 4-20 Procedure of the probability calculation using the RBD

First, check each equipment for equipment failure. To do this, a random value between 0 and 1 is generated for each equipment. If the generated value is less than the failure rate of the equipment, there is equipment failure. All the equipment in the liquefaction cycle

are to be checked for equipment failure, as shown in Figure 4-20(a). Thereafter, the *analyzer* model checks for system failure. The analysis model sends a signal to the connected model, a condenser in Figure 4-20, and this signal follows the system. If the signal returns to the analyzer model, the system is successfully operating during the simulation.

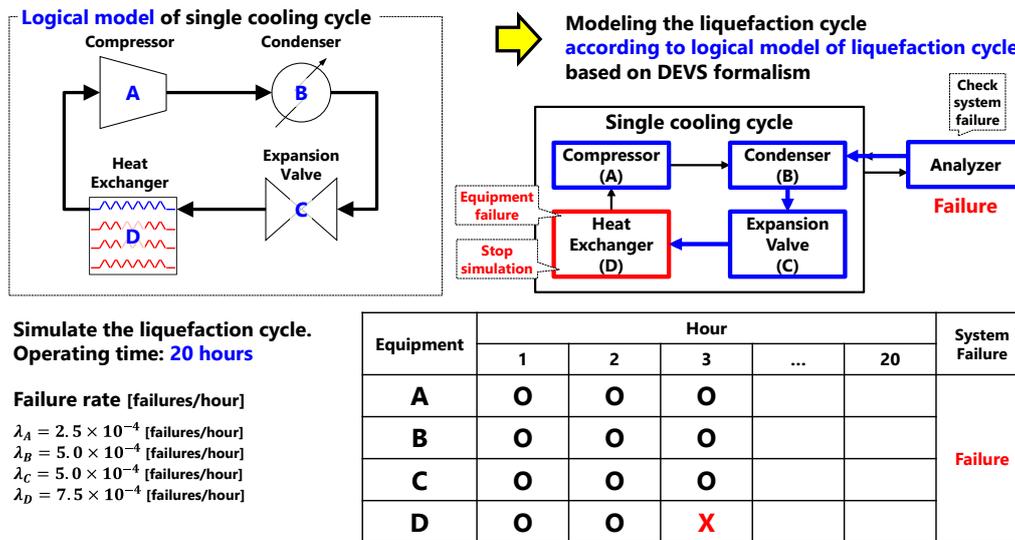


Figure 4-21 Procedure of the probability calculation using the RBD

Figure 4-21 shows an example of system failure during probability calculation. As mentioned in the previous paragraph, equipment failure is first determined. In this example, the heat exchanger fails and is marked as such. The analyzer model sends a signal to check for system failure, and the signal will be stopped at the heat exchanger. Thus, the analyzer determines that the system has failed.

This operation is repeated until the operating time. Figure 4-22 shows the calculation of the failure probability by performing the simulation repeatedly. The failure probability is simply calculated based on the ratio of the number of system failures to the total number of simulations.

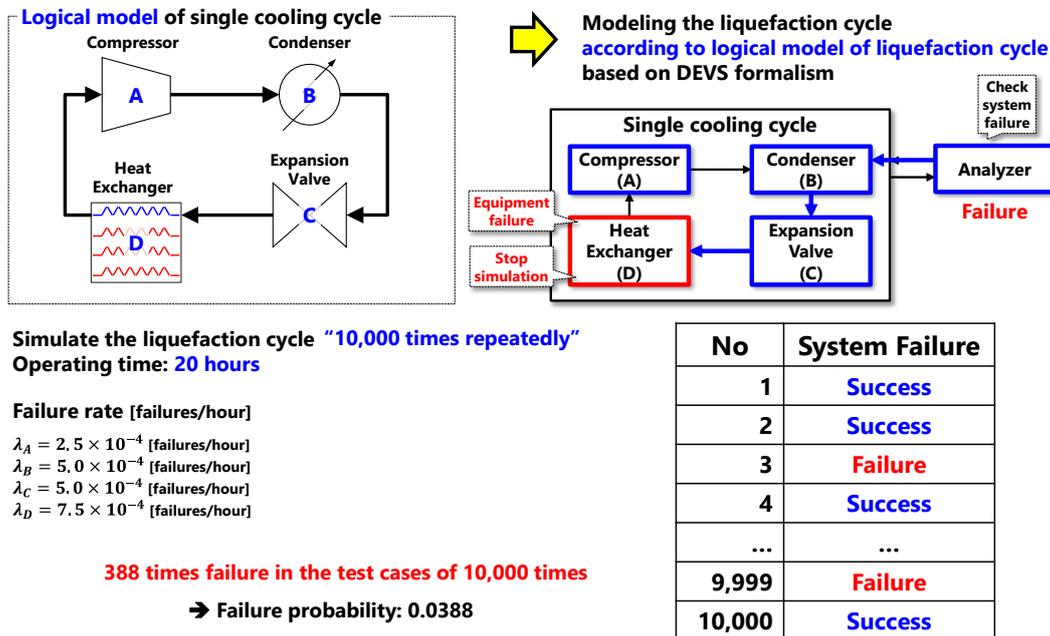


Figure 4-22 Failure probability calculation via Monte Carlo simulation, using the RBD

4.2.2. Physical Model for Reliability Simulation Based on the DEVS Formalism

The RBD is automatically generated by converting the logical models of the liquefaction cycle. A physical model used to determine equipment failure for every unit time is defined based on the DEVS formalism. Similar to the configuration of the process simulator in chapter 3, a simulation model for the reliability assessment is configured by synthesizing a logical model, which is generated by pruning the SES, and a physical model in the MB.

In the reliability simulation, the physical models for equipment are categorized into two groups: single-flow equipment and multiple-flow equipment. This section gives a detailed description of the physical models in the MB.

(1) Single-flow equipment

A compressor, a condenser, an expansion valve, and an expander are all equipment with a single inlet and a single outlet. As shown in Figure 4-23, the physical model for equipment with a single inlet and a single outlet is configured based on the DEVS formalism. It is characterized by variable states: THROW, WAITING, ANALYZE, and FAIL. The THROW state is used to determine equipment failure.

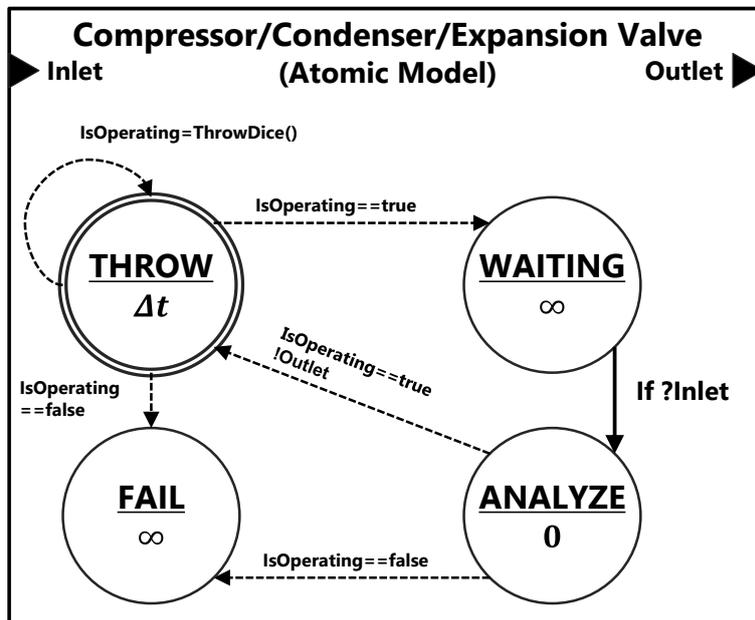


Figure 4-23 Physical model for equipment with a single inlet/outlet based on the DEVS formalism

The model at the **THROW** state throws a dice to generate a random value between 0 and 1 every unit time, and checks for equipment failure according to the generated value and the failure rate. If the equipment is operating after a dice is thrown at the **THROW** state, the state will be changed to the **WAITING** state. In the **WAITING** state, the equipment waits until the signal from the previous equipment is sent. When the equipment receives the signal from the previous equipment, the current state will be changed to the **ANALYZE**

state. In the ANALYZE state, the physical model confirms the equipment failure. If the equipment is operating, the current state will be changed to the THROW state. When the state is changed from ANALYZE to THROW, the signal will be sent to the outlet of this model. On the other hand, if the equipment fails, the current state will be changed to the FAIL state, and the equipment will stay in such state until the simulation is stopped.

(2) Multiple-flow equipment

A common header, tee, phase separator, and heat exchanger are all equipment with multiple inlets and outlets. As shown in Figure 4-24, the physical model for equipment with multiple inlets and outlets is configured based on the DEVS formalism.

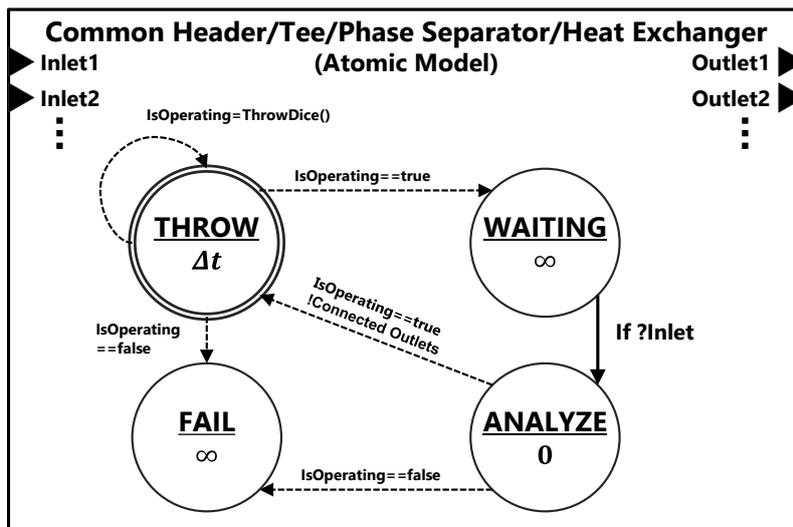


Figure 4-24 Physical model for equipment with multiple inlets/outlets based on the DEVS formalism

It is also characterized by four states: THROW, WAITING, ANALYZE, and FAIL. All operations using these states are similar to that of the physical model for single-flow equipment. The difference is that the signal will be sent to multiple outlets when the state is changed from ANALYZE to THROW.

(3) Analyzer for checking system failure

To check for the failure of the whole system for every unit time, the analyzer model is configured based on the DEVS formalism, as shown in Figure 4-25. The analyzer model has two states: CHECK and CHECKING. At the end of each unit time, the analyzer model changes its state from CHECK to CHECKING. At the same time, a signal to check for system failure is sent to the “check” port. At the CHECKING state, the analyzer model waits for the success signal from the last equipment of the liquefaction cycle. If the success signal is received, the state will be changed to the CHECK state. If not, the simulation will be stopped by the execution rules of the DEVS formalism. If the liquefaction cycle has multiple cooling cycles, the model in the CHECKING state will wait until all the signals from each cycle arrive.

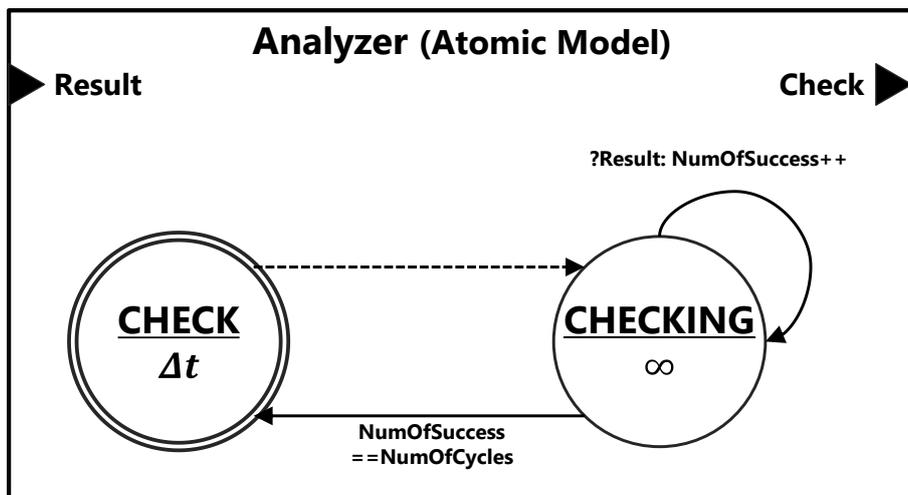


Figure 4-25 Analyzer model based on the DEVS formalism: checking for failure of the whole system

4.3. Verification of the Reliability Analysis Method

Proposed in This Thesis

To verify the reliability analysis method proposed in this thesis, the following three examples are simulated to assess the reliability of each system:

- firewater pumping system;
- power switching system; and
- hypothetical computer system.

4.3.1. Firewater Pumping System

Figure 4-26 shows the logical model of a firewater pumping system. The system has two fire pumps, one engine, and one valve. The engine is necessary to run the fire pump. If the valve or the engine fails, the system will fail. The failure of one fire pump is allowed because the other fire pump will still be running.

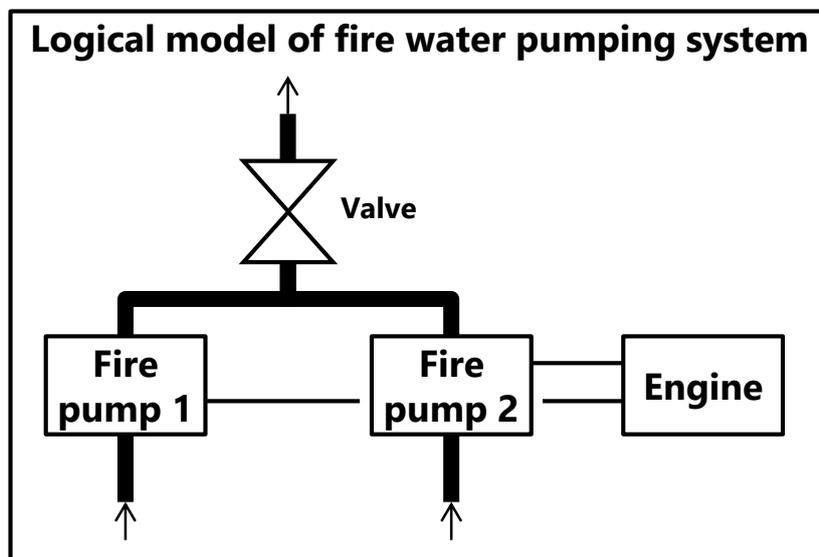


Figure 4-26 Logical model of the firewater pumping system

The FT of the firewater pumping system is configured as shown in Figure 4-27. The failure of the valve directly leads to system failure. The failure of the engine contributes to system failure by combining the fire pumps.

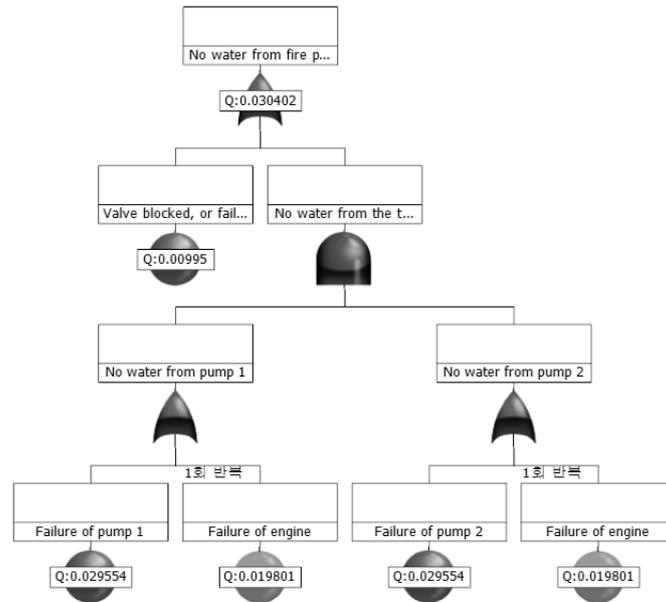


Figure 4-27 FT of the firewater pumping system

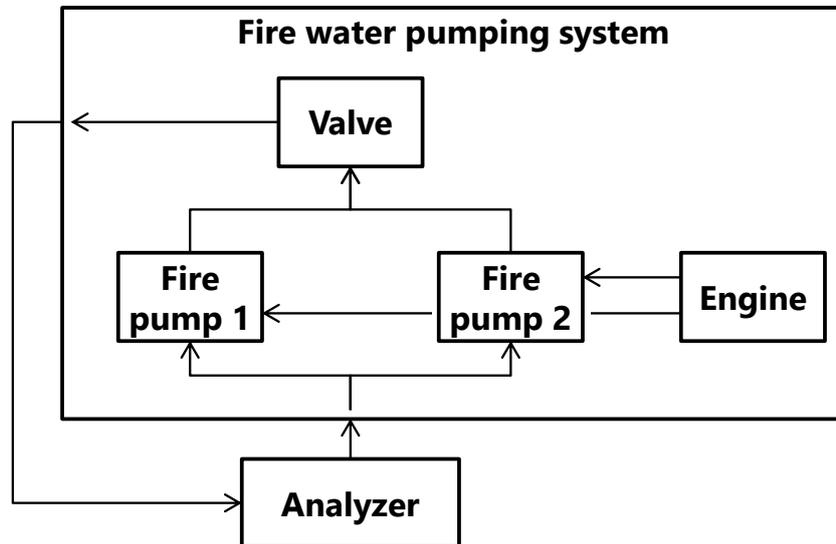


Figure 4-28 RBD of the firewater pumping system based on the DEVS formalism

Figure 4-28 shows the RBD of the firewater pumping system. All the equipment in this diagram are defined based on the DEVS formalism. As can be seen, the configuration of the RBD is the same as that of the logical model of the system, except for the analyzer model.

Table 4-1 compares the reliability analysis results for the firewater pumping system. The operating time of the system is assumed to be 1,000 hours, and the simulation is performed 50,000 times. The probability of system failure is similar to the result using the FT. The execution time using the RBD based on the DEVS formalism was shown to be the longest among the three methods.

Table 4-1 Comparison of reliability analysis results for the firewater pumping system

Reliability analysis method	Probability of system failure	Execution time [s]
FTA and Markov chain	3.040%	0.49 [sec]
FTA and the Bayesian network	3.025%	0.23 [sec]
RBD based on DEVS	3.016%	4 [sec]

4.3.2. Power Switching System

Figure 4-29 shows the logical model of a power switching system. The system has two powers, two relays with one controller, and a motor. Power 1 is the primary system, and power 2 is the standby system. Each power is dependent on each relay, and all the relays are dependent on the controller. At least one power is necessary to run the motor.

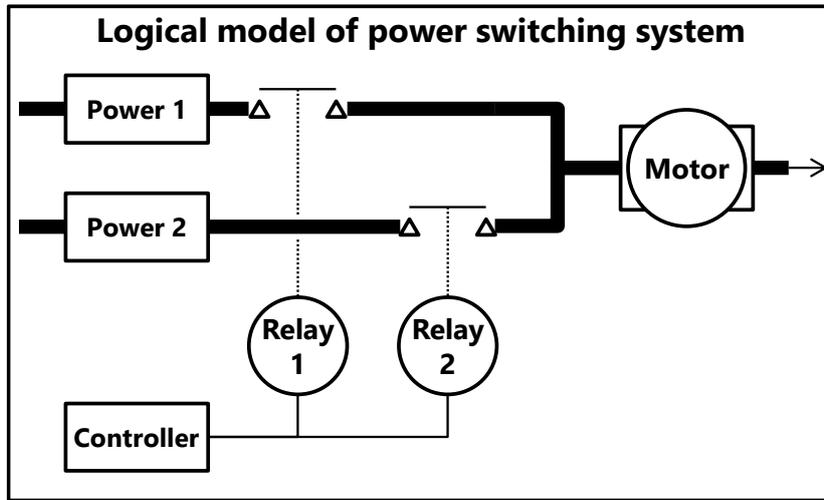


Figure 4-29 Logical model of the power switching system

The FT of the power switching system is configured as shown in Figure 4-30. If both powers fail, the system will fail. As the relays are dependent on the controller, the failure sequence of the relays and the controller leads to system failure. This is expressed by the use of the PAND gate in the FT.

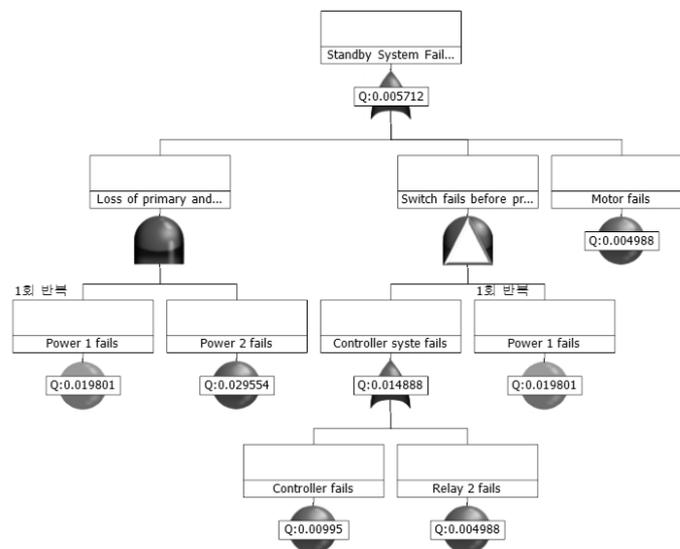


Figure 4-30 FT of the power switching system

Figure 4-31 shows the RBD of the power switching system. All the equipment in this diagram are defined based on the DEVS formalism. As can be seen, the configuration of the RBD is the same as that of the logical model of the system, except for the analyzer model.

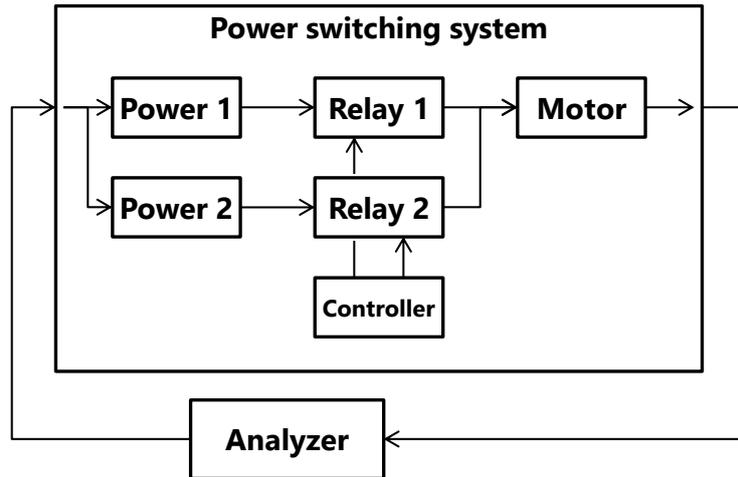


Figure 4-31 RBD of the power switching system based on the DEVS formalism

Table 4-2 Comparison of reliability analysis results for the power switching system

Reliability analysis method	Probability of system failure	Execution time [s]
FTA and Markov chain	0.571%	11.6 [sec]
FTA and the Bayesian network	0.568%	2.1 [sec]
RBD based on DEVS	0.562%	6.9 [sec]

Table 4-2 compares the reliability analysis results for the power switching system. The operating time of the system is assumed to be 1,000 hours, and the simulation is performed 50,000 times. The probability of system failure is similar to the result using the FT. The execution time using the RBD based on the DEVS formalism is better than that using the Markov chain because the number of equipment increases, resulting in the state space explosion of the Markov chain.

4.3.3. Hypothetical Computer System

Figure 4-32 shows the logical model of a hypothetical computer system. The system has three subsystems: the process, memory, and application systems. The process system has two primary processes and one standby process. The memory system has five memories and two memory interface units (MIUs). The application system has an application and a graphical user interface (GUI).

The standby process is used when one process fails. Each MIU in the memory system manages two memories. If the MIU fails, the child memories also fail. The memory system needs three running memories.

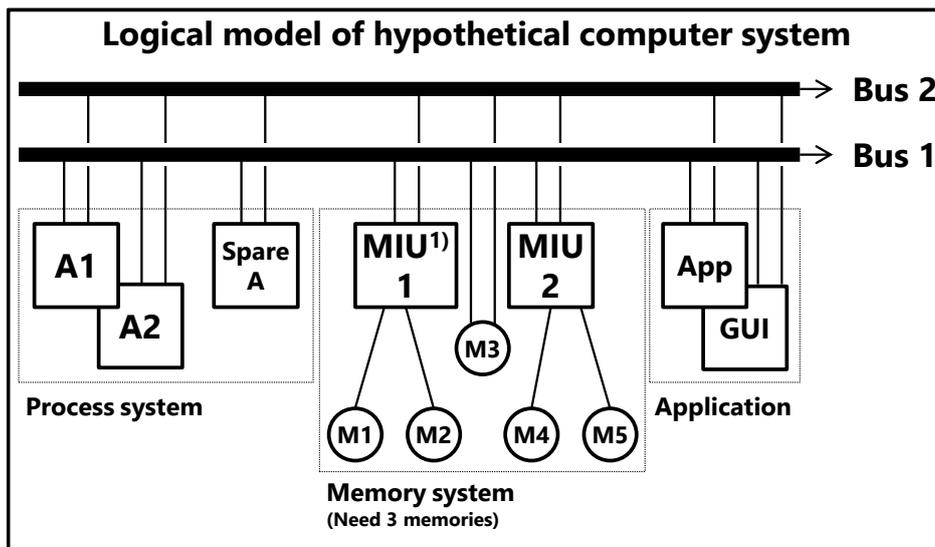


Figure 4-32 Logical model of the hypothetical computer system

The FT of the hypothetical computer system is configured as shown in Figure 4-33. The standby process is considered by using the SPARE gate in the DFT. The MIU is also considered by using the FDEP gate in the DFT.

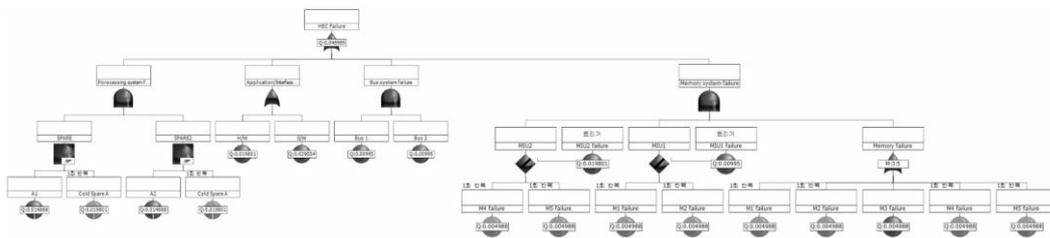


Figure 4-33 FT of the hypothetical computer system

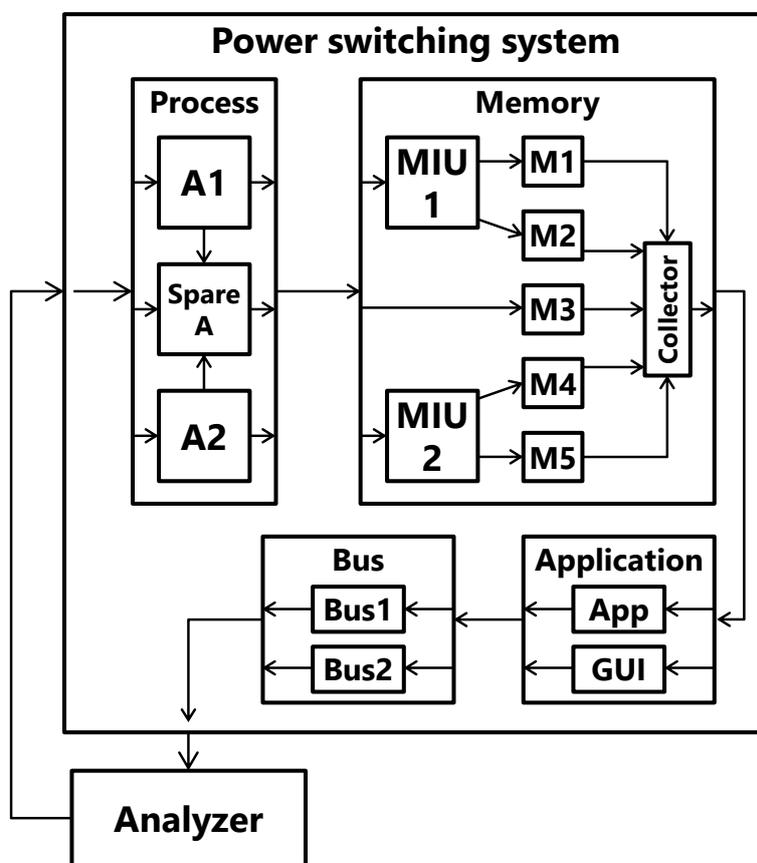


Figure 4-34 RBD of the hypothetical computer system based on the DEVS formalism

Figure 4-34 shows the RBD of the hypothetical computer system. All the equipment in this diagram are defined based on the DEVS formalism. The configuration of the RBD is the same as that of the logical model of the system, except for the analyzer model.

Table 4-3 Comparison of reliability analysis results for the hypothetical computer system

Reliability analysis method	Probability of system failure	Execution time [s]
FTA and Markov chain	4.899%	35 [min]
FTA and the Bayesian network	4.879%	6.3 [sec]
RBD based on DEVS	4.854%	14.1 [sec]

Table 4-3 compares the reliability analysis results for the hypothetical computer system. The operating time of the system is assumed to be 1,000 hours, and the simulation is performed 50,000 times. The probability of system failure is similar to the result using the FT. The execution time using the Markov chain increases exponentially because the number of equipment increases, resulting in the state space explosion of the Markov chain.

4.4. Reliability Analysis of the Conventional Liquefaction Cycles

In the previous section, the RBD method based on the DEVS formalism was verified using three examples. In this section, the method is applied to the conventional liquefaction cycles. Before the application, the SES of the liquefaction was modified for the reliability analysis.

4.4.1. Variation of the System Entity Structure for Reliability Analysis

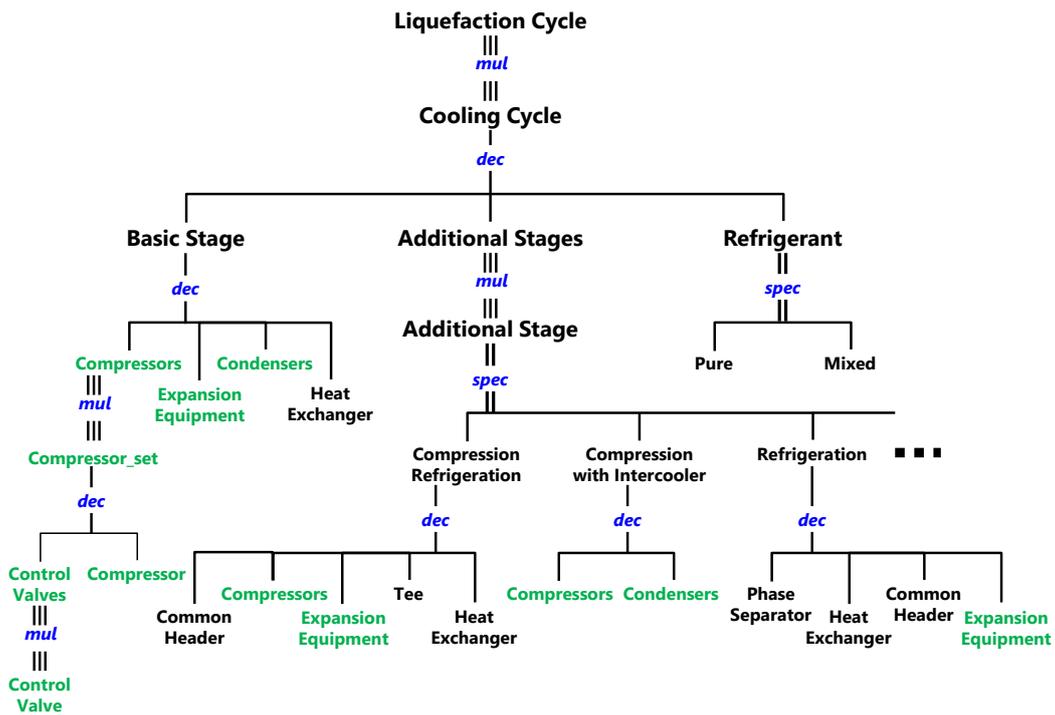


Figure 4-35 Variation of the SES for reliability analysis considering redundant equipment

Figure 4-35 shows the variation of the SES for reliability analysis considering

equipment redundancy and control logic. All the equipment, except for the common header, tee, and heat exchanger, have a control valve to control the flow rate of the equipment. Thus, the control valve is added to the equipment while considering the redundancy. A compressor, a condenser, and an expansion valve can also have redundant equipment for itself, and this is also reflected on the SES.

By using the SES in Figure 4-35, various alternative logical models for reliability analysis can be automatically generated. Moreover, it can be automatically converted to the equivalent RBD, which can assess the reliability by synthesizing with the physical models in the MB. The next sections give a detailed description of this.

4.4.2. Reliability Analysis of the DMR Cycle

(1) Logical model of the DMR cycle

Figure 4-36 shows the logical model of the DMR cycle with control logics for equipment. This logical model is generated by pruning the SES in Figure 4-35, and it has the relationships between equipment, as shown in Figure 4-36.

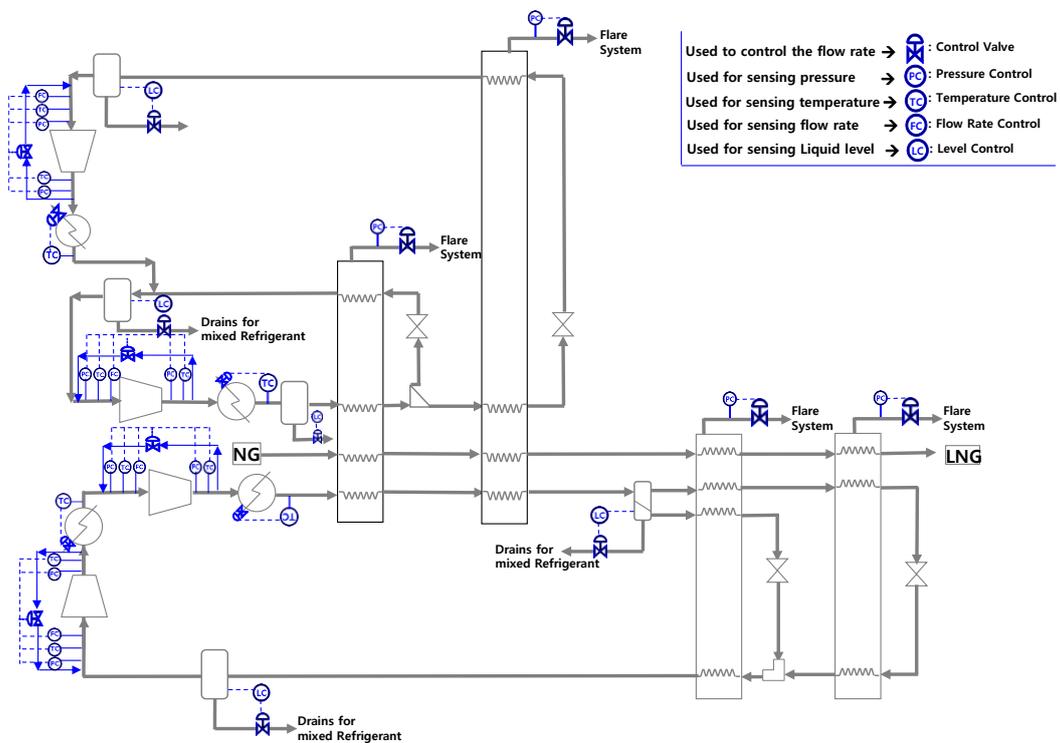


Figure 4-36 Logical model of the DMR cycle with control logics

(2) Conversion to RBD based on the DEVS formalism

The logical model in Figure 4-36 has its child equipment and their interconnections and can thus be simply converted to an RBD. The procedure of the conversion to an RBD is as follows:

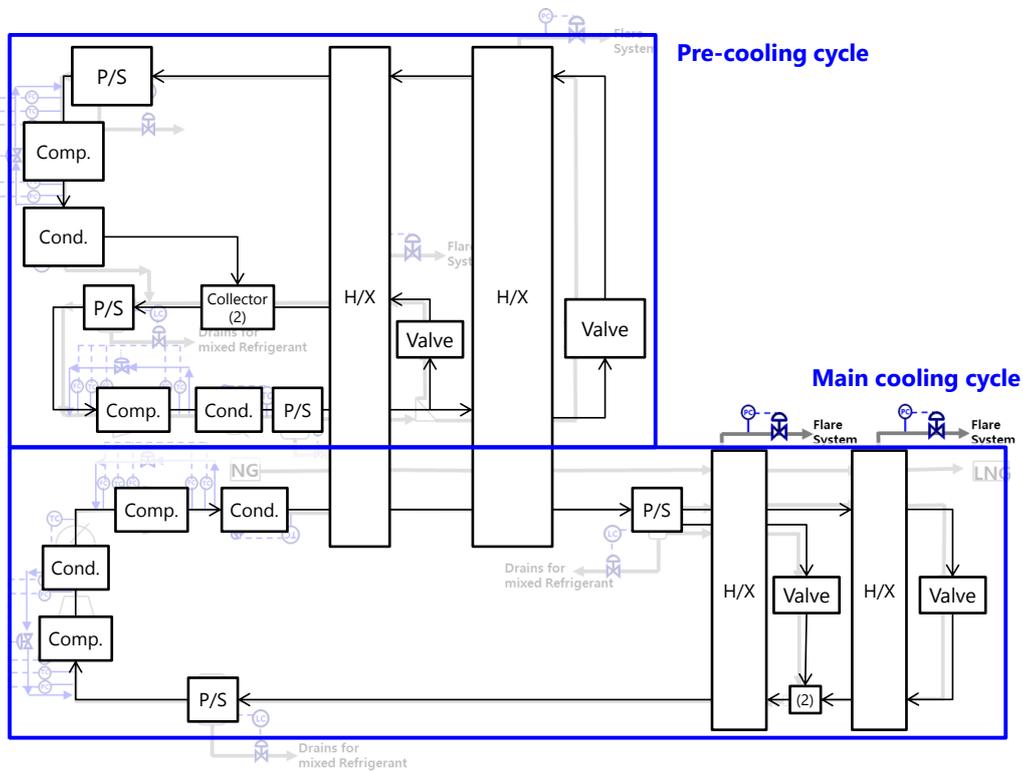


Figure 4-37 Partitioning of the cooling cycle: the precooling and main cooling cycles

First, the cooling cycles of the DMR cycle were divided into the precooling and main cooling cycles. All the equipment in the cooling cycle are defined as coupled models based on the DEVS formalism. In this example, a tee and a common header are not considered equipment, but the common header is used to replace the *collector* model, which performs the function of sending a signal after all the multiple inlets arrive.

As shown in Figure 4-38, it is assumed that the equipment have two control valves, except for the expansion valve and heat exchanger. One control valve is the primary control valve, and the other is the standby. If all the control valves fail, the equipment will also fail. This assumption is configured as a coupled model based on the DEVS formalism, as shown in Figure 4-38. The coupled model of equipment is composed of two control valves, one

collector, and an equipment physical model. The collector model performs the same function as the SPARE gate of the FT.

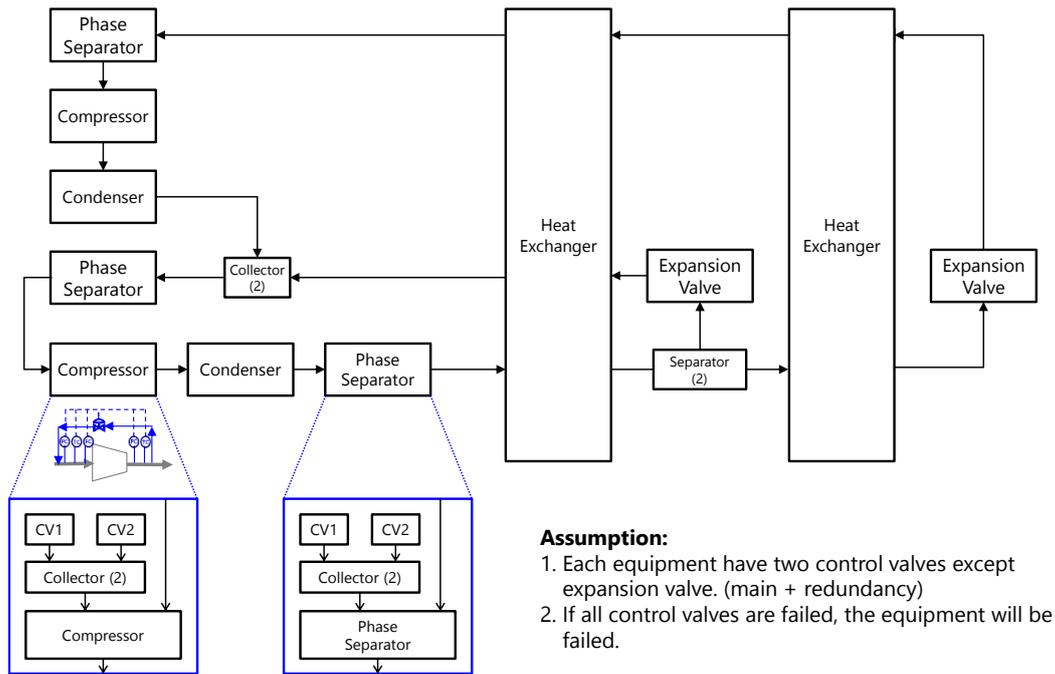


Figure 4-38 RBD of the DMR cycle: precooling cycle with a control valve

The equipment redundancy is also considered. Figure 4-39 shows that the concept of full redundancy is applied to the condenser and the expansion valve. The condenser and the expansion valve have one standby each, with 100% capacities. On the other hand, the compressor has one standby with 50% capacity, as shown in Figure 4-40. In the case of the compressor, its cost is higher than that of the condenser and the expansion valve. Thus, the cost can be reduced by using the standby with 50% capacity.

compressors have one standby each, with 50% capacities.

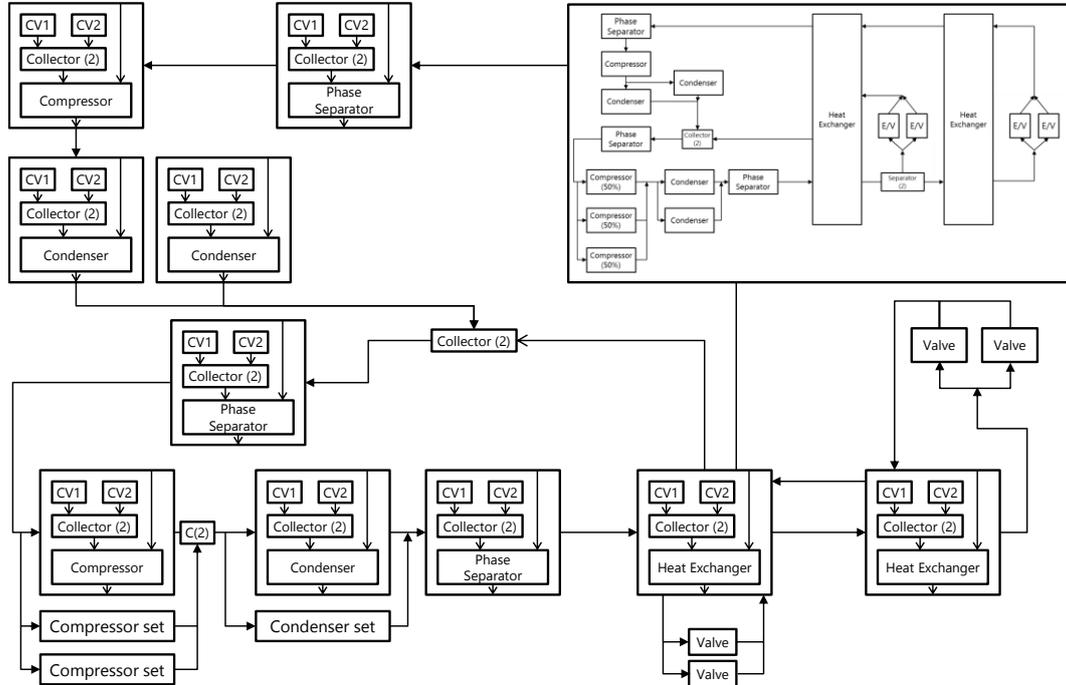


Figure 4-41 RBD of the DMR cycle: precooling cycle

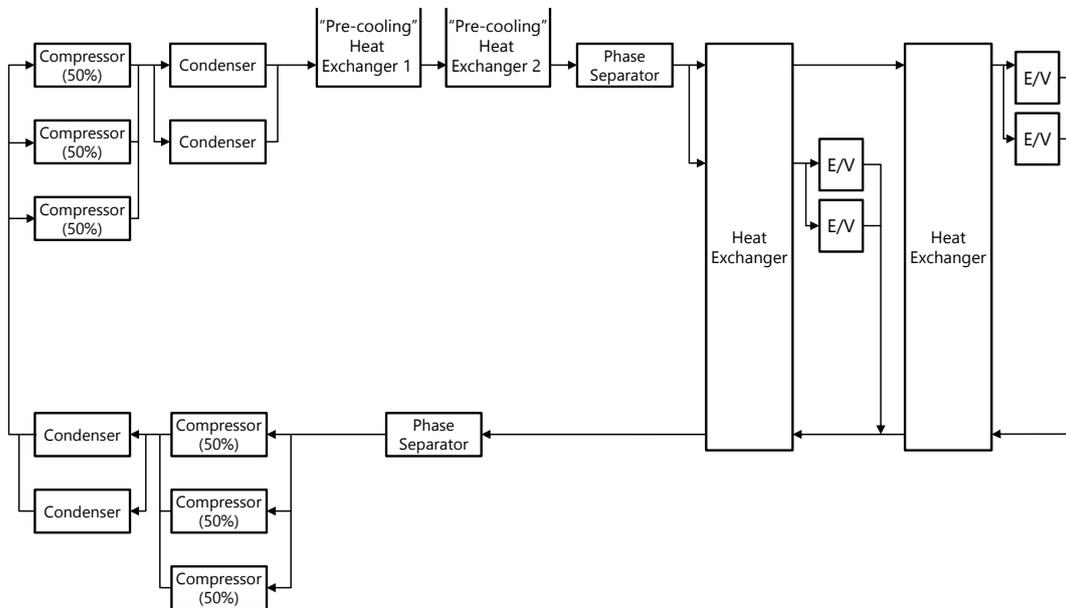


Figure 4-42 RBD of the DMR cycle: main cooling cycle

(3) Checking the system failure of the DMR cycle

After conversion to an RBD, Monte Carlo simulation is performed repeatedly. All the equipment check their own failure for every unit time, and the analyzer model checks its system failure by following the cycle shown in Figure 4-43. The probability of system failure is calculated based on the ratio of the number of system failures to the total number of simulations.

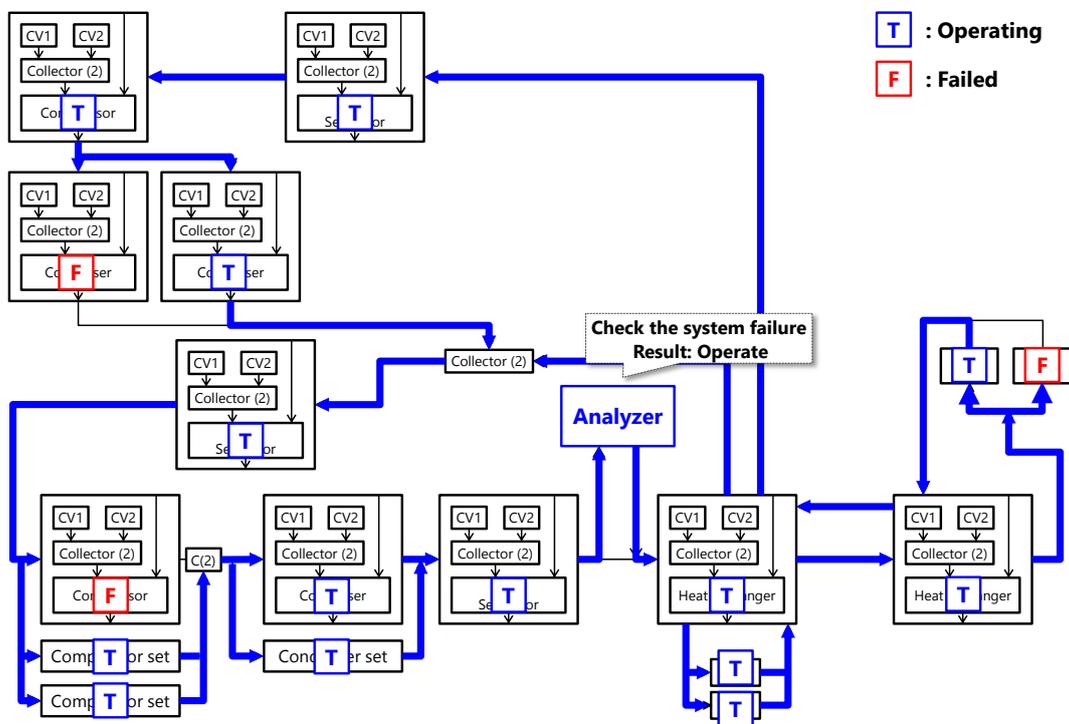


Figure 4-43 Checking the system failure of the precooling cycle of the DMR cycle

(4) Reliability analysis result of the DMR cycle

Table 4-4 shows the result of the reliability analysis for the DMR cycle. The operating time of the system is assumed to be 10,000 hours, and the simulation is performed 50,000

times. The probability of system failure is similar to the result of the FT. The execution time is two times longer than that of the FT with the Bayesian network.

Table 4-4 Comparison of reliability analysis results for the DMR cycle

Reliability analysis method	Probability of system failure	Execution time [s]
FTA and the Bayesian network	4.79%	301 [sec]
RBD based on DEVS	4.762%	702 [sec]

In Monte Carlo simulation, the repetition of the simulation affects the convergence of the results. Table 4-5 shows the probability convergence according to the number of calculations. Fifty-thousand-time calculation is reasonable considering the convergence and the execution time.

Table 4-5 Probability convergence according to the number of calculations

Number of calculations	Probability of system failure	Execution time [s]
1,000 times	5.8%	16
5,000 times	5.16%	75
10,000 times	4.941%	142
50,000 times	4.762%	702
100,000 times	4.779%	1,413

4.4.3. Reliability Analysis Results of Other Conventional Liquefaction Cycles

In this paper, reliability analysis of other conventional liquefaction cycles, such as the C3MR, cascade, N₂ expander, and Niche cycles, is performed using RBD based on the DEVS formalism. The results of the reliability analysis are described in the following sections.

(1) C3MR cycle

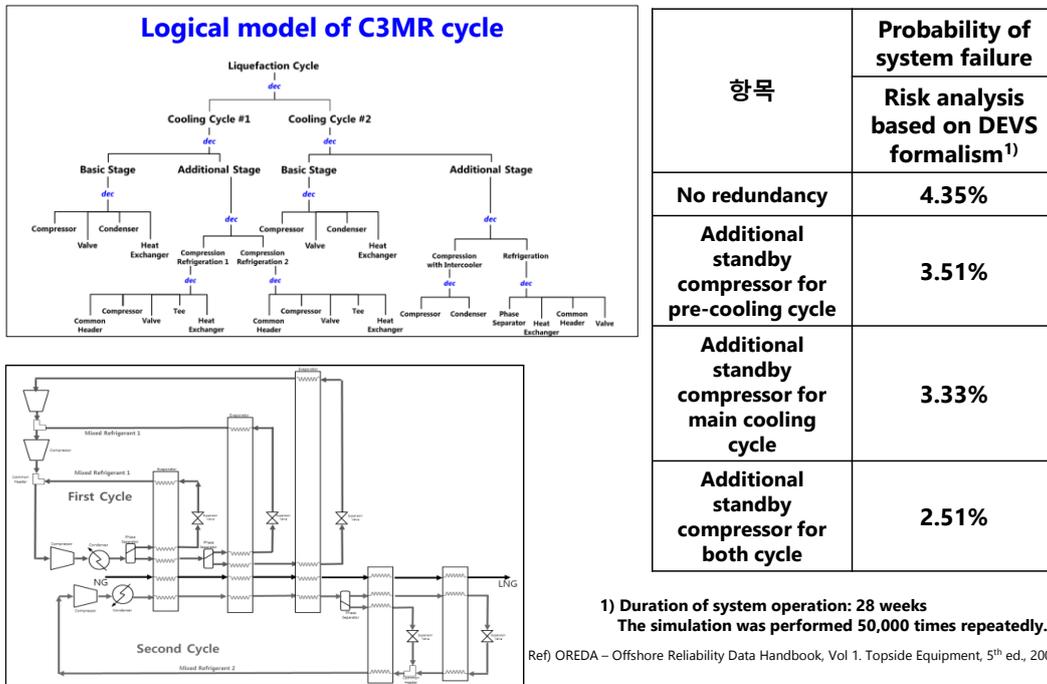


Figure 4-44 Reliability analysis of the C3MR cycle

(2) Cascade cycle

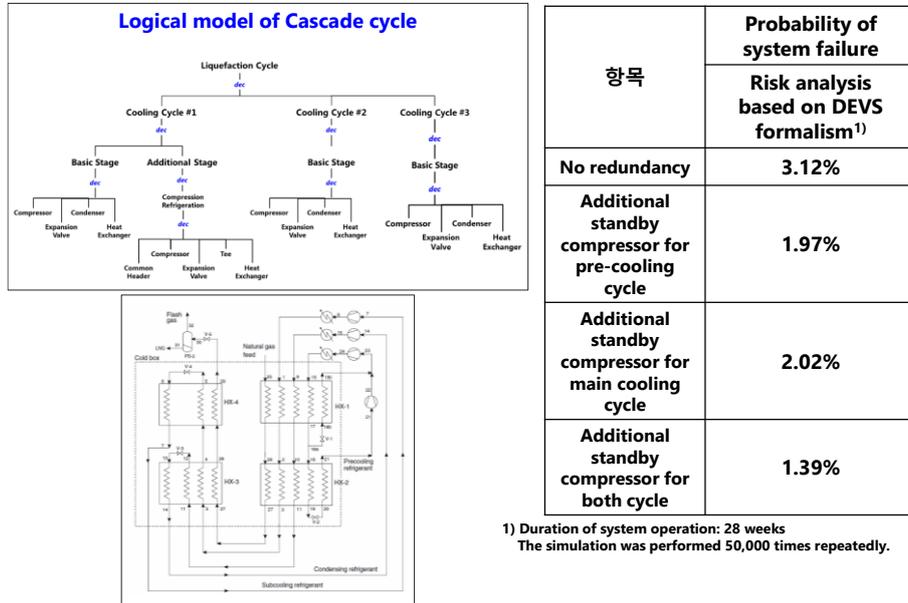


Figure 4-45 Reliability analysis of the cascade cycle

(3) Single N₂ expander cycle

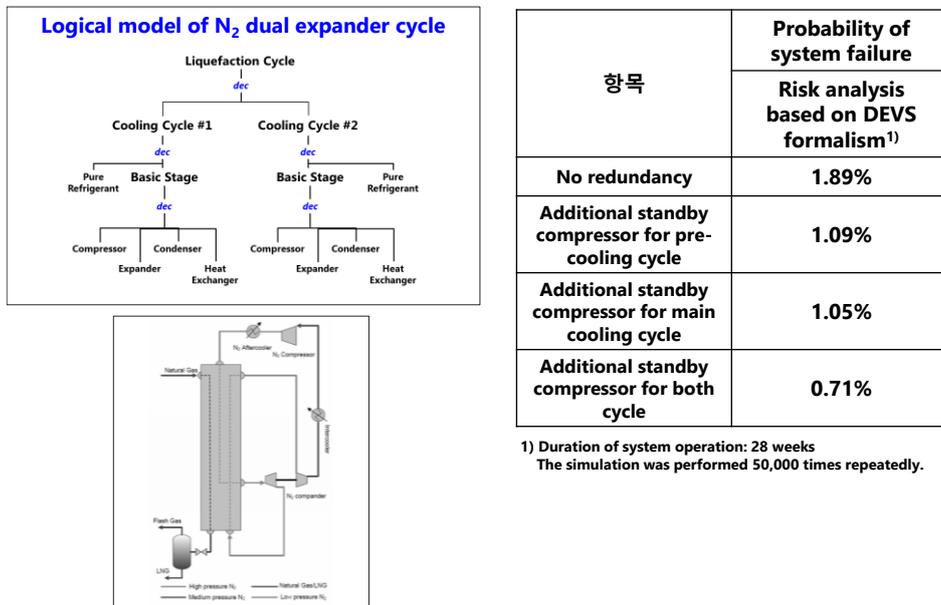
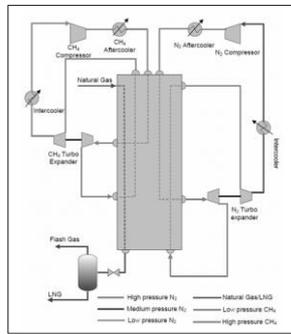
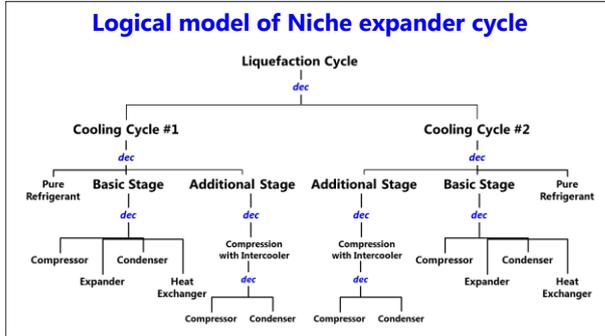


Figure 4-46 Reliability analysis of the single N₂ expander cycle

(4) Niche cycle



항목	Probability of system failure
	Risk analysis based on DEVS formalism ¹⁾
No redundancy	2.11%
Additional standby compressor for pre-cooling cycle	1.23%
Additional standby compressor for main cooling cycle	1.25%
Additional standby compressor for both cycle	0.81%

1) Duration of system operation: 28 weeks
The simulation was performed 50,000 times repeatedly.

Figure 4-47 Reliability analysis of the Niche cycle

5. Effect of Ship Motion in the Heat Exchanger as Determined Using the Lattice Boltzmann Method

The heat exchanger is the main device in the liquefaction cycle. The performance of the heat exchanger during the tilt and movements caused by the ship motion has been one subject of uncertainty for installing an LNG plant on floating plants. LNG FPSOs have to handle potentially extreme sea and weather conditions, so the performance of the heat exchanger should be considered in a point of the reliability and availability.

5.1. Modeling of Cryogenic Heat Exchangers

5.1.1. Cryogenic Heat Exchangers

For LNG purposes, the use of two main heat exchangers, the spiral wound heat exchanger (SWHE) and the plate fin heat exchanger (PFHE), is common. Overall, the SWHE is used for the mixed-refrigerant cycles while the PFHE is common in the expander and cascade cycles.

(1) Plate fin heat exchanger (PFHE)

The PFHE consists of aluminum fins. The refrigerator and natural gas moves in a crossed-flow configuration, offering optimal heat exchange. The design can be modified to apply a number of different streams in one unit. Multiple units can be arranged in cold

boxes that are totally insulated for extremely hot weather conditions. The small pipe works inside the cold boxes have straightforward designs, but the large bore pipe work outside the boxes may cause technical problems when connecting aluminum to stainless steel. The main advantages of the PFHE are its compactness, low equipment weight, and capability to treat many process streams in one unit. Its disadvantages, on the other hand, are its vulnerability to thermal shock, higher maintenance, and lower efficiency. A conceptual drawing of a PFHE is shown in Figure 5-1(a).

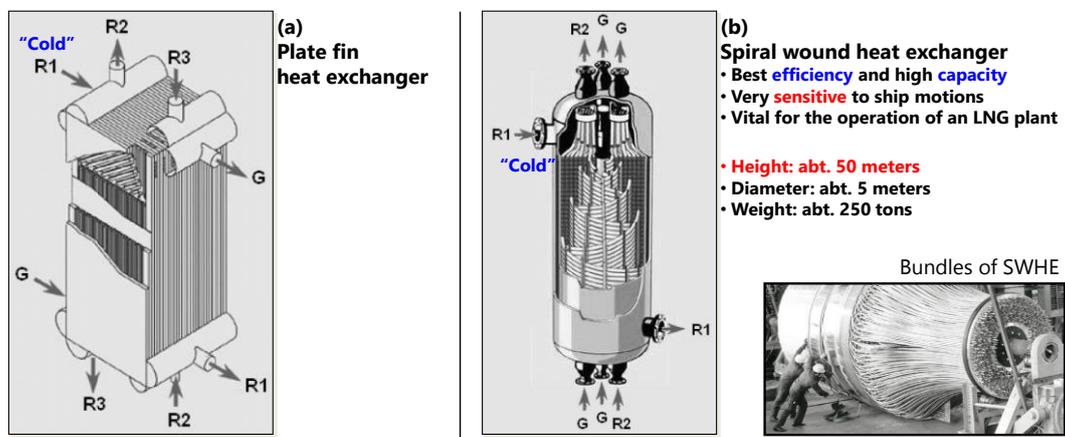


Figure 5-1 Cryogenic heat exchangers: plate fin heat exchanger and spiral wound heat exchanger

(2) Spiral wound heat exchanger (SWHE)

A conceptual drawing of the SWHE is shown in Figure 5-1(b). The SWHE is a vertically standing tube, as shown in Figure 5-2. Natural gas flows from the bottom to the top, and is dispersed in many helical bundles. The diameters of these bundles are typically around 10 mm. Liquid refrigerant flow is injected at the topside and diverts a liquid film over the bundles. During the cooling process, the liquid refrigerant vaporizes and exits the heat exchanger in gaseous form. Increasing the tube-side design pressure will result in higher production. Its disadvantages are its relatively high cost, size, and weight, and its limited

flexibility with respect to variable composition gas streams. Its advantages, on the other hand, are its efficiency, large cooling surface per area shell, tolerance of thermal shock (robustness), and reliability in large-capacity plants.

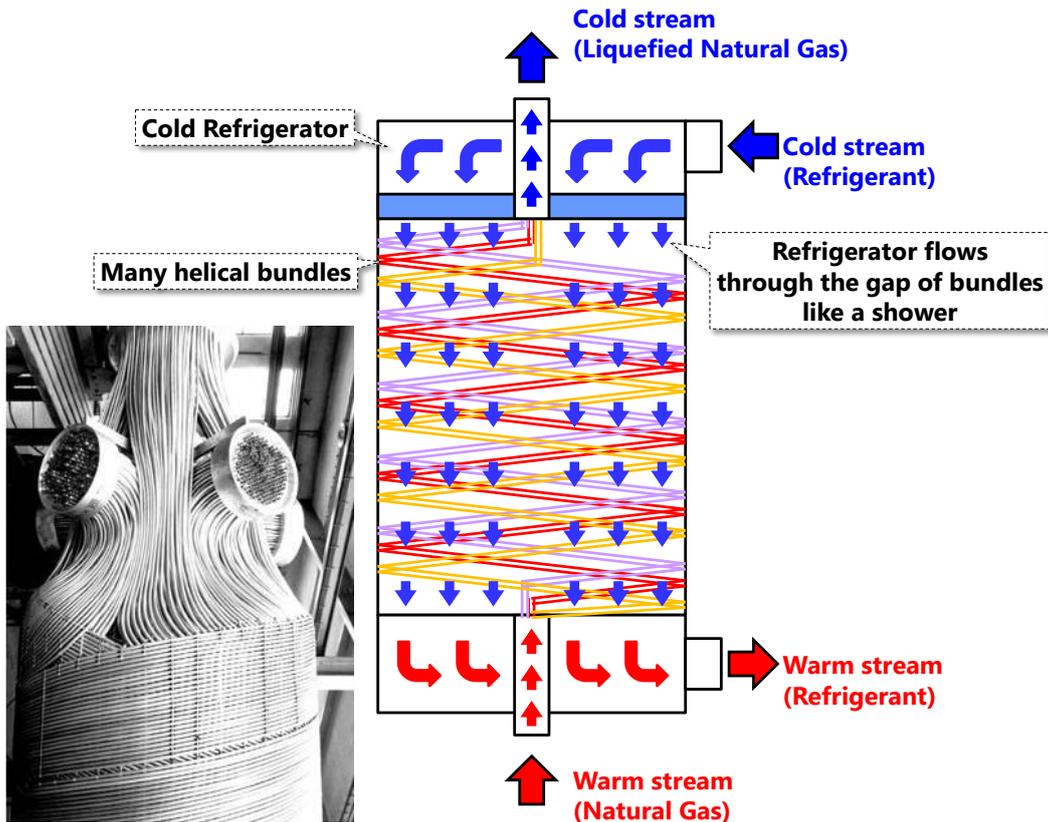


Figure 5-2 Concept of the spiral wound heat exchanger (SWHE)

5.1.2. Simplified Model of the Spiral Wound Heat Exchanger

As shown in Figure 5-2, the SWHE has a complex structure, making the effectiveness analysis of the heat exchanger more complex. Thus, a simplified model of the heat exchanger is introduced, as shown in Figure 5-3.

The simplified heat exchanger model is defined in a 2D plane. The bundles in the heat

exchanger are simplified into small rectangles and are distributed uniformly in the heat exchanger. The temperatures of the small rectangles are continuously initialized from bottom to top, and the cold refrigerator drops from top to bottom.

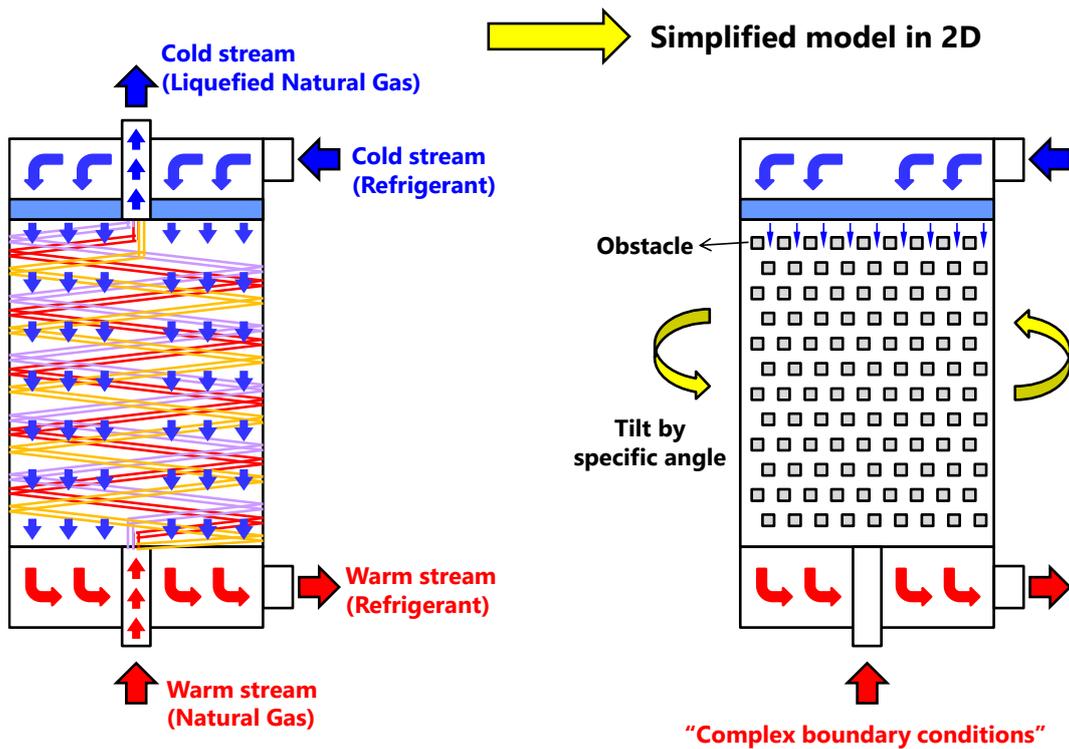


Figure 5-3 Simplified model of the heat exchanger

To analyze this simplified heat exchanger model, a numerical analysis method should be needed. The finite difference method (FDM) and finite element method (FEM) are conventionally used in fluid dynamics and solid mechanics. However, if the boundaries conditions of the model is more complex, these two method will have more complex equations to analyze the problem. In that case, Lattice Boltzmann Method (LBM) has an advantage to model the complex boundary conditions because the analysis model is represented as cellular form. Furthermore, it is easy to treat multi-phase and multi-component flows, and it can be naturally adapted to parallel processes computing.

Table 5-1 Comparison of advantages and disadvantages between numerical analysis methods

Computational Method	Advantage	Disadvantage
Finite Difference Method	<ul style="list-style-type: none"> - Preferred in fluid dynamics - Straightforward method in which any partial differential equation can be discretized 	<ul style="list-style-type: none"> - Not simple to adapt to boundaries of arbitrary shape Limited by computer resource to make fine grid
Finite Element Method	<ul style="list-style-type: none"> - Preferred in solid mechanics - Much simpler to implement moving boundaries than FDM 	<ul style="list-style-type: none"> - More complicated than FDM mathematically - Require considerable mathematical manipulation before solution can be found
Lattice Boltzmann Method	<ul style="list-style-type: none"> - Simple rules to update the state of the system - Simple rules for boundary condition including curved or moving boundary - Well suited for parallel computation 	<ul style="list-style-type: none"> - Restricted to very high viscosities or very small scales in time and space - Small wiggles on curved or moving boundary

The simplified heat exchanger model is simpler than the real model of the heat exchanger, but it also has complex boundary conditions. To analyze the temperature distribution of the heat exchanger with these complex boundary conditions, the lattice Boltzmann method (LBM) is used in this thesis.

5.2. Lattice Boltzmann Method

5.2.1. Concept of the LBM

The LBM is a new and promising method in computational fluid dynamics. There are two main approaches in simulating physical phenomena (heat, mass, and momentum), continuum and discrete as shown in Figure 5-4 (Mohamad, 2011). In continuum approach shown in Figure 5-4(a), ordinary or partial differential equations can be achieved by applying conservation of energy, mass, and momentum for an infinitesimal control volume. Since it is difficult to solve the governing differential equations for many reasons (nonlinearity, complex boundary conditions, complex geometry, etc.), therefore finite difference, finite volume, finite element, etc., schemes are used to convert the differential equations with a given boundary and initial conditions into a system of algebraic equations. This algebraic equations can be solved iteratively until convergence is insured. We can look at each step as each volume or node or element contains a collection of particles, and the scale is *macroscopic*.

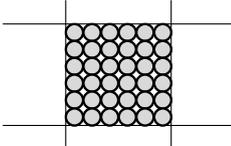
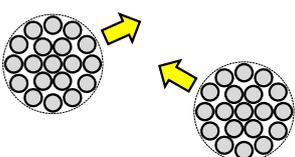
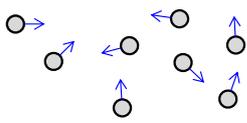
(a) Continuum	(b) Lattice Boltzmann Method	(c) Discrete Particle
<p>Control Volume</p> 	<p>Group of Particles</p> 	<p>Single Particle (Atom, Molecule)</p> 
<p>Macroscopic</p>	<p>Mesoscopic (Group by similar properties)</p>	<p>Microscopic</p>
<p>Ordinary/Partial Differential Equation</p>	<p>Explicit Equation of Distribution Function</p>	<p>Ordinary Differential Equation of Newton's 2nd Law</p>
<p>Difficult to solve (nonlinear, complex boundary & geometry)</p>	<p>Applicable to complex geometry, manageable computational resources</p>	<p>Need massive numbers of calculation</p>

Figure 5-4 Analysis techniques of physical phenomena

On the other extreme, the medium can be considered made of small particles (atom, molecule) and these particles collide with each other as shown in Figure 5-4(c). This scale is *microscopic*. Hence, we need to identify the inter-particle forces and solve ordinary differential equation of Newton's second law (momentum conservation). At each time step, we need to identify location and velocity of each particle, i.e., trajectory of the particles. Fundamentally, the microscopic approach is simple and can handle complex geometries without any difficulties. However, the main drawback of the microscopic approach for a large system is need of large computer resource.

As shown in Figure 5-4(b), the main idea of the LBM is to bridge the gap between microscale and macroscale by not considering each particle behavior alone but behavior of a collection of particles as a unit. The property of the collection of particles is represented by a *distribution function*. The distribution function acts as a representative for collection of particles. This scale is called *mesoscale*.

The LBM has advantages of both macroscopic and microscopic approaches, with manageable computer resources. It is easy to apply for complex domains, easy to treat multi-phase and multi-component flows. Furthermore, it can be naturally adapted to parallel processes computing.

5.2.2. Algorithms to Calculate Using the LBM

(1) Lattice gas automata: an origin of lattice Boltzmann method

The LBM evolved from the lattice gas automata (LGA) method. The field of LGA started in 1973 with the paper of Hardy, de Pazzis, and Pomeau (1973). These authors showed a kind of billiard game with collisions that conserved mass and momentum. The HPP model was improved into the LGA model with hexagonal symmetry, and the LGA model yields the incompressible Navier-Stokes equation with a macroscopic limit. A few years later, LBM arose as an offshoot of LGA. Its higher flexibility compared with LGA led to artificial microscopic models for several nonlinear partial differential equations, including the advection-diffusion and Navier-Stokes equations.

The purpose of LGA is to simulate the behavior and interaction of many single particles in a gas as simply as possible. LGAs can be seen as very simple molecular dynamic methods. Gas is modeled as a multitude of hard spheres moving along a regular grid, with a discrete set of possible velocities \vec{c}_i for each particle. The collision between particles is handled by a set of elastic collision rules that must conserve the system's quantities of mass m and momentum \vec{p} .

As mentioned earlier, the first LGA model was proposed in 1973 by Hardy, de Pazzis, and Pomeau (Hardy et al., 1973). It was named "HPP," after the initials of its three authors. HPP is a 2D lattice-gas cellular automata model over a square lattice. In the HPP model, the grid is 2D and square, so that each node in the grid has four neighbors. The particles can have four possible velocities, as shown in Figure 5-5(a). For each time step, each particle is moved forward one step in the direction of its velocity. When two or more particles meet in the same node after a time step, a collision occurs. To conserve mass and

momentum, the number of particles and the total velocity of all the particles in the node must be the same before and after the collision. When two particles collide head-on, they are thrown out at right angles to their original velocities, as shown in Figure 5-5(b). This conserves momentum as the sum of the velocities of the two particles is zero in both configurations.

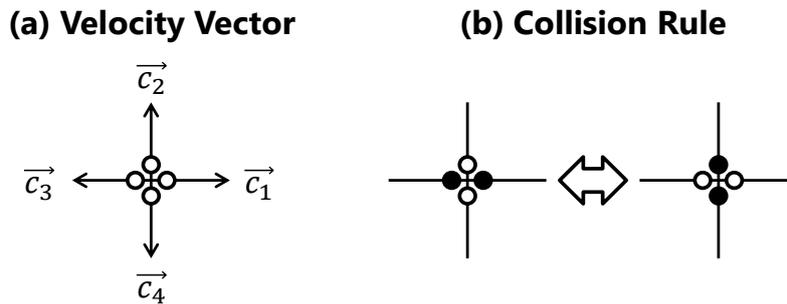
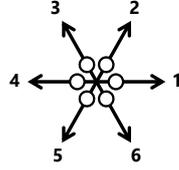


Figure 5-5 (a) Velocity vectors of the HPP model. (b) Head-on collision rule for the HPP model.

The HPP model is the simplest LGA, but it does not lead to the Navier-Stokes equation within some limits. This weakness alone is crippling to the point that the HPP model is not useful for fluid simulations (Wolf-Gladrow, 2005). For this reason, this model was abandoned in the late 1980s in favor of the FHP model, which manifests an isotropic Navier-Stokes behavior by changing the shape of the lattice.

In 1986, Frisch, Hasslacher, and Pomeau (1986) showed that an LGA model over a lattice with a larger symmetry group than that of the square lattice yields the incompressible Navier-Stokes equation within the macroscopic limit. This model that has hexagonal symmetry was named “FHP,” after the initials of its three authors. The FHP lattice is composed of triangles, as shown in Figure 5-6. It is invariant under rotations by $n \cdot 60^\circ$ about an axis, through a node, and is perpendicular to the lattice plane.

(a) Velocity Vector



(b) Example of Collision Rule

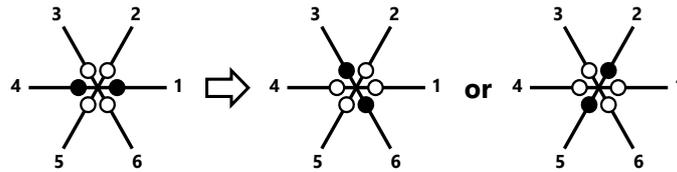


Figure 5-6 (a) Velocity vectors of the FHP model. (b) Sample head-on collision rule for the FHP model.

In the FHP model, at each node and each link to the nearest neighbor, a cell may be emptied or occupied by one particle at most. All the particles have the same mass and are indistinguishable. The evolution in time proceeds via alternating collisions and streaming (also called *propagation*). The collisions are strictly local — that is, only the particles of a single node are involved. With these essential properties of the FHP model, its general evolution equation becomes as follows:

$$n_i(\vec{x} + \vec{c}_i, t + 1) - n_i(\vec{x}, t) = \Omega_i(\vec{x}, t), \quad (5-1)$$

In the above equation, $n_i(\vec{x}, t)$ is the Boolean array for each i , wherein \vec{x} and t indicate the discrete points in space and time, and $\Omega_i(x, t)$ is the collision operator, which is dependent on $n_i(\vec{x}, t)$, and which can have the following three values: -1, 0, and 1 for each i . This equation expresses that a particle has moved away from direction \vec{c}_i due to a collision, stayed on its path, or moved towards direction \vec{c}_i due to a collision. Ha et al. (2011) adjusted these LGA to the acoustic problem.

(2) Lattice Boltzmann method

LBM was introduced in 1988 as a way of avoiding certain weaknesses of lattice gases while retaining their distinct advantages: parallelization and simplicity. Instead of handling single particles, LBM handles particle distributions and treats a collision in a manner different from its handling of LGA. With the FHP model, the Boolean occupation number n_i was replaced with the distribution function f_i so that f_i is a number between 0 and 1. Thus, the evolution equation for a lattice gas, equation (1), becomes the following lattice Boltzmann equation:

$$f_i(\vec{x} + \vec{c}_i, t + 1) - f_i(\vec{x}, t) = \Omega_i(\vec{x}, t). \quad (5-2)$$

This removed the need for the averaging used by the LGA to find some macroscopic quantities, such as velocity and momentum, along with its inherent statistical weakness. While the basic LBM is used to simulate the behavior of compressible and incompressible isothermal fluids, many models expand it in different ways, thus enabling it to simulate complex fluids, thermo-hydrodynamics, etc.

LBM is based on a regular triangle grid. Each node in the grid has several different variables associated with it [$f_i(\vec{x}, t)$]. These variables represent the density of the particles traveling in direction \vec{c}_i at node at \vec{x} and time t . In the most commonly used 2D lattice, there are six different velocity vectors \vec{c}_i .

The mass density, $\rho(\vec{x}, t)$, and the flow velocity, $u(\vec{x}, t)$, are defined by the sums of the distribution functions $f_i(\vec{x}, t)$, as follows:

$$\rho(\vec{x}, t) = \sum f_i(\vec{x}, t), \text{ and} \quad (5-3)$$

$$\rho(\vec{x}, t)u(\vec{x}, t) = \sum \vec{c}_i f_i(\vec{x}, t). \quad (5-4)$$

By using a one-dimensional rod, the algorithms of the LBM is described in detail as follows.

① Problem definition of one-dimensional rod

Figure 5-7 shows a one-dimensional rod with three nodes divided by uniform length (1m in this example). Two assumptions are given: particles can move to the left/right direction or stay current position; and particles move to the neighbor node during the unit time step.

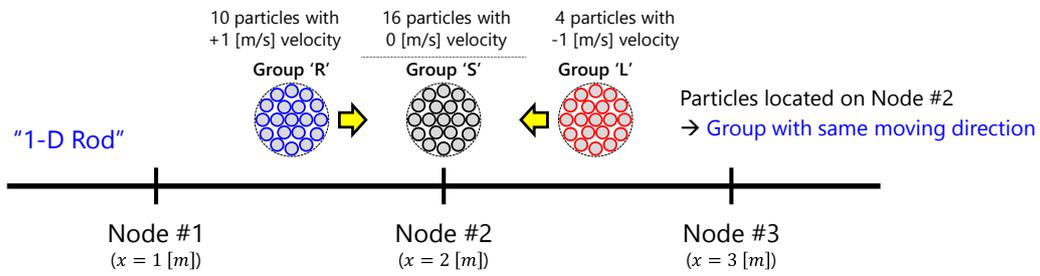


Figure 5-7 Concept of the lattice Boltzmann method: application to one-dimensional rod

The particles are in groups with same moving direction. For example, particles moving to the right direction on Node #2 is in *Group R*, particles staying on Node #2 is in *Group S*, and particles moving to the left direction on Node #2 is in *Group L*. Thus, Group R has 10 particles, Group S has 16 particles, and Group L has 4 particles as shown in Figure 5-7. For the simple calculation, this is assumed as distribution function f_i in the LBM, so the distribution functions for each direction at $t = 0$ on Node #2 is as follows:

$$f_0(2,0) = 10, f_R(2,0) = 16, \text{ and } f_L(2,0) = 4. \quad (5-5)$$

The notation of equation (D-6) is shown in Figure 5-8.

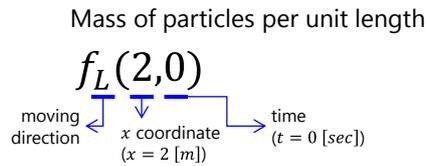


Figure 5-8 Simplified distribution function for an example of one-dimensional rod

② Streaming (propagation)

The process of streaming in the LBM means that particles move to the neighbor node during the unit time step. For example, if Group S and L are not exist on Node #2, Group R can move to Node #2 without collision after 1 second. This procedure can be expressed by

$$\begin{array}{ccc}
 \text{"10 particles"} & & \text{"10 particles"} \\
 f_R(2 + 1, 0 + 1) & = & f_R(2, 0) \\
 \text{"after moving"} & & \text{"before moving"}
 \end{array} \tag{5-6}$$

Equation (5-6) can be generally expressed as follows:

$$f_i(\mathbf{r} + \mathbf{c}_i \Delta t, t + \Delta t) = f_i(\mathbf{r}, t), \tag{5-7}$$

where i is a moving direction, \mathbf{r} is a position vector of particles, \mathbf{c}_i is a velocity vector of particles, and Δt is the unit time step.

③ Collision

On the other hand, if Group S and L are exist on Node #2, particles of Group R will collide with the particles of Group S and L during moving to Node #3 (see Figure 5-9(a)). Because of collision between particle groups, some particles in Group R stay or change their own direction to the left. Additionally some particles in Group S and L participate in

Group R because of the collision, and move to Node #3.

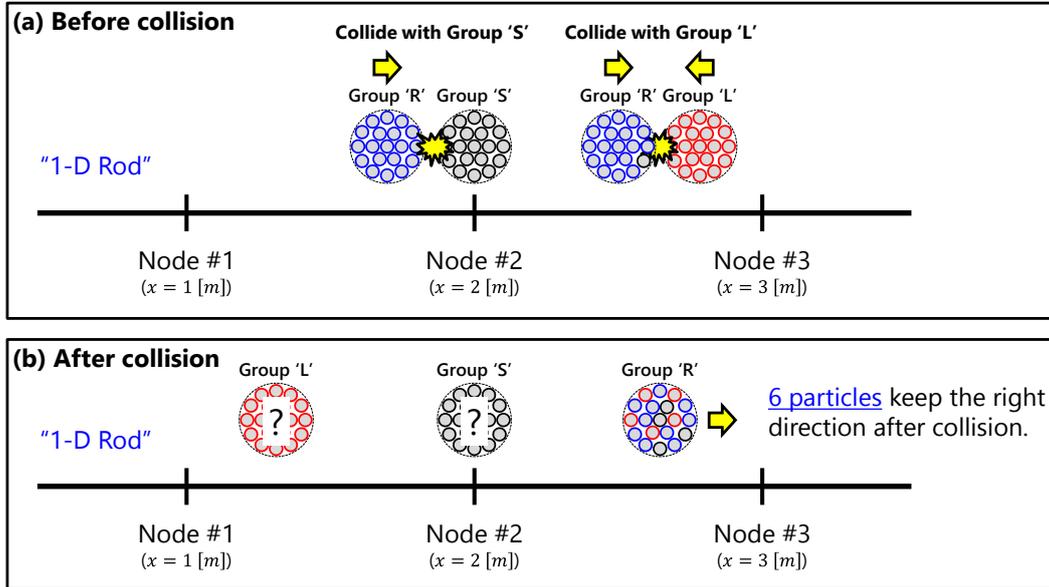


Figure 5-9 Collision process of the lattice Boltzmann method: (a) before collision, (c) particles of Group R after collision

The collision process of Group R can be expressed by

$$f_R(2 + 1, 0 + 1) = f_R(2, 0) + \Omega_R(2, 0) \quad (5-8)$$

"6 particles"
"10 particles"
"-4 particles"

"after moving"
"before moving"
"collision"

Equation (5-8) can be generally expressed as follows:

$$f_i(\mathbf{r} + \mathbf{c}_i \Delta t, t + \Delta t) = f_i(\mathbf{r}, t) + \Omega_i(\mathbf{r}, t), \quad (5-9)$$

where Ω_i is a collision operator representing the collision procedure of the particles.

④ Collision operator and equilibrium distribution function

From the equation (5-9), it is obvious that the determination of the collision operator is important to analyze specific physical phenomena. The key idea to describe the collision process is that all particles eventually comes to rest as time goes by. It is assumed that the distribution of particles at equilibrium state depends on the number of particles (actually density) ρ and average velocity of particles \mathbf{u} (see Figure 5-10(b) and (c)), and is so called *equilibrium distribution function* f_i^{eq} .

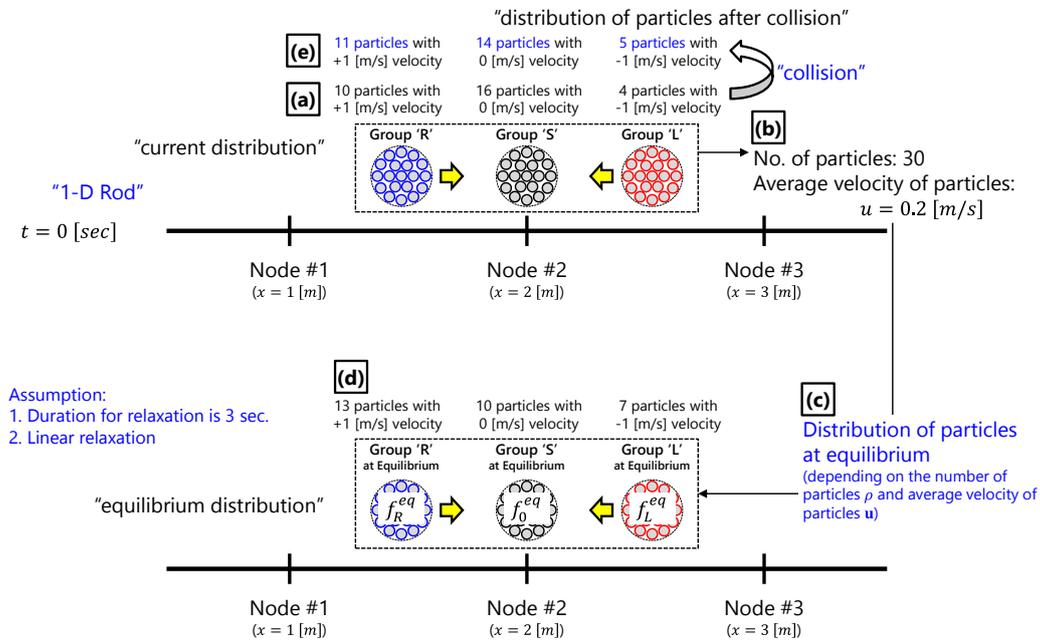


Figure 5-10 Equilibrium distribution function and collision operator of the lattice Boltzmann method

Before the collision, the particles are distributed in each group of Node #2 as shown in Figure 5-10(a). The particle distributions at equilibrium state can be calculated as shown in Figure 5-10(d). Another assumption is that the particles at current time comes to rest linearly according to the duration for relaxation and particle distributions as equilibrium

state. This can be expressed by

$$f_R(2+1,0+1) = f_R(2,0) + \underbrace{\Omega_R(2,0)}_{\text{"relaxation to equilibrium"}} \longrightarrow \Omega_R(2,0) = \frac{1}{3} \left[\underbrace{f_R^{eq}(2,0)}_{\text{equilibrium distribution}} - \underbrace{f_R(2,0)}_{\text{current distribution}} \right] \quad (5-10)$$

Total relaxation time ←
↓
↓

This is Bhatnagar-Gross-Krook (BGK) collision operator. Using the equation (5-17), the number of particles can be calculated as shown in Figure 5-10(e). The LB equation with BGK collision operator is generally expressed as follows:

$$f_i(\mathbf{r} + \mathbf{c}_i \Delta t, t + \Delta t) = f_i(\mathbf{r}, t) + \frac{\Delta t}{\tau} [f_i^{eq}(\mathbf{r}, t) - f_i(\mathbf{r}, t)], \quad (5-11)$$

where f_i^{eq} is particle distribution function for each direction i at equilibrium, and τ is a total relaxation time to equilibrium state.

(3) Bhatnagar-Gross-Krook (BGK) collision operator for the LBM

In the early 1990s, the simplified collision operator Ω_i for LBM was suggested by several authors. In 1992, Qian, d'Humieres, and Lallemand proposed the use of a simplified collision operator similar to the one that Bhatnagar, Gross, and Krook proposed for the Boltzmann equation in 1954 (Qian et al., 1992). What they considered the simplest and most efficient method was the method of relaxation. In computational fluid dynamics, it is known that the method of relaxation has a good stability property if the relaxation parameter ω is between 0 and 2. This method can be described simply as follows: If $N(t)$ is a quantity at time t and N_e is its predicted value in the equilibrium state, $N(t+1)$ is given by

$$N(t+1) = (1 - \omega)N(t) + \omega N_e. \quad (5-12)$$

If $0 < \omega < 1$, it is called *subrelaxation*, and if $1 < \omega < 2$, it is called *overrelaxation*. The scheme is linearly stable for $0 < \omega < 2$. Similar to this concept, the BGK operator used on the LBM is given by

$$\Omega_i(\vec{x}, t) = -\frac{1}{\tau} \left[f_i(\vec{x}, t) - f_i^{(0)}(\vec{x}, t) \right], \quad (5-13)$$

where τ is the time relaxation constant and $f_i^{(0)}$ is the equilibrium distribution of the particles. The operator itself represents the relaxation of the distribution function f_i towards the equilibrium value $f_i^{(0)}$.

As the collision operator must preserve both the mass and the momentum,

$$\sum_i \Omega_i = 0, \text{ and} \quad (5-14)$$

$$\sum_i \vec{c}_i \Omega_i = 0. \quad (5-15)$$

From equations (5-14) and (5-15), it is clear that the equilibrium distribution must preserve the mass and momentum, as follows:

$$\rho = \sum_i f_i^{(0)} = \sum_i f_i, \text{ and} \quad (5-16)$$

$$\rho \vec{u} = \sum_i \vec{c}_i f_i^{(0)} = \sum_i \vec{c}_i f_i. \quad (5-17)$$

If the BGK collision operator in equation (5-13) is inserted into the lattice Boltzmann equation (5-13), the lattice Boltzmann equation for the BGK operator,

$$f_i(\vec{x} + \vec{c}_i, t + 1) = \left(1 - \frac{1}{\tau} \right) f_i(\vec{x}, t) + \frac{1}{\tau} f_i^{(0)}(\vec{x}, t), \quad (5-18)$$

is derived, where τ is related to the kinematic shear viscosity through

$$v = c_s^2 \left(\tau - \frac{1}{2} \right), \quad (5-19)$$

where c_s is the speed of sound in the fluid, which is $1/\sqrt{3}$ in lattice units for the FHP lattice.

(4) Lattice arrangements

In LBM, the solution domain should be divided into lattices. At each lattice node, the distribution function resides at each direction to the neighbor lattices. The numbers of directions and linkages depend on the lattice arrangement.

In this paper, it was assumed that the behavior of oil slick movement and spread is based on a 2D model. The lattice with nine directions is quite commonly used for a 2D model. The velocity vectors \vec{c}_k towards the neighborhood are $c_0 = (0,0)$, $c_1 = (1,0)$, $c_2 = (0,1)$, $c_3 = (-1,0)$, $c_4 = (0,-1)$, $c_5 = (1,1)$, $c_6 = (-1,1)$, $c_7 = (-1,-1)$, and $c_8 = (1,-1)$. The weighting factors w_k for the corresponding distribution functions are $4/9, 1/9, 1/9, 1/9, 1/9, 1/36, 1/36, 1/36, 1/36$. In these lattice arrangements, speed c_s associated with the lattice size and dimension is $1/\sqrt{3}$ (Mohamad, 2011).

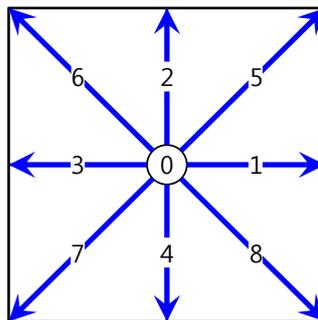


Figure 5-11 Lattice arrangements for a 2D problem with nine directions (D2Q9)

(5) Boundary conditions

There are two simple lattice Boltzmann boundaries: the periodic boundary and the hard wall. Periodic edges act as if they are connected to the opposite edge of the system. If the left and right edges of a system are periodic, a particle distribution that streams leftward at the left edge of the system will reappear at the right edge and will head towards the left. In a 2D system, the left-right boundaries and/or the top-bottom boundaries can be periodic. Figure 5-12 shows the neighboring nodes across the periodic boundaries.

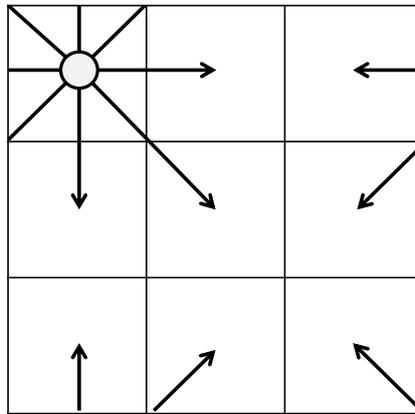


Figure 5-12 Particle streaming paths from a corner node in a periodic 2D system in LBM

The other boundary is the hard wall, which reflects particles back and guarantees a non-slip condition with zero velocity at the wall. As shown in Figure 5-13 and Figure 5-14, two variations of this boundary condition exist: the on-grid bounce-back boundary and the mid-grid bounce-back boundary. In these two boundaries, certain nodes in the grid are marked as walls and are thus not part of the fluid. The particle distributions in the wall nodes are not relaxed towards the equilibrium and do not act according to the lattice Boltzmann equation.

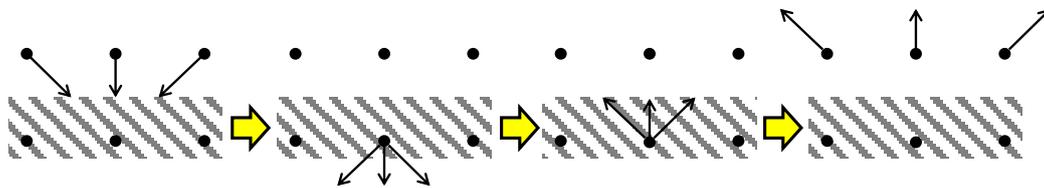


Figure 5-13 On-grid bounce-back boundary in LBM

In the on-grid bounce-back boundary, the particles that reach the wall node in one-time step are reflected back in the next. On the other hand, the particles that are set to stream into a wall node are instead reflected in the mid-grid bounce-back boundary. As the mid-grid boundary yields greater accuracy than the on-grid boundary (Chen & Doolen, 1998), mid-grid walls were used in the simulations.

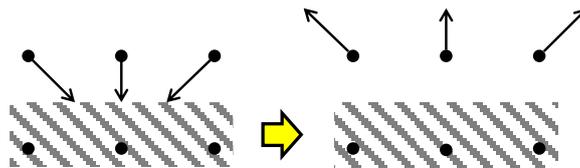


Figure 5-14 Mid-grid bounce-back boundary in LBM

5.3. Analysis of the Motion Effect on the Heat Exchanger Using LBM

The analysis result using LBM is used for the reliability analysis of the liquefaction cycles. The temperature distribution of the heat exchanger is analyzed using LBM. This distribution is used to calculate the effectiveness of the heat exchanger.

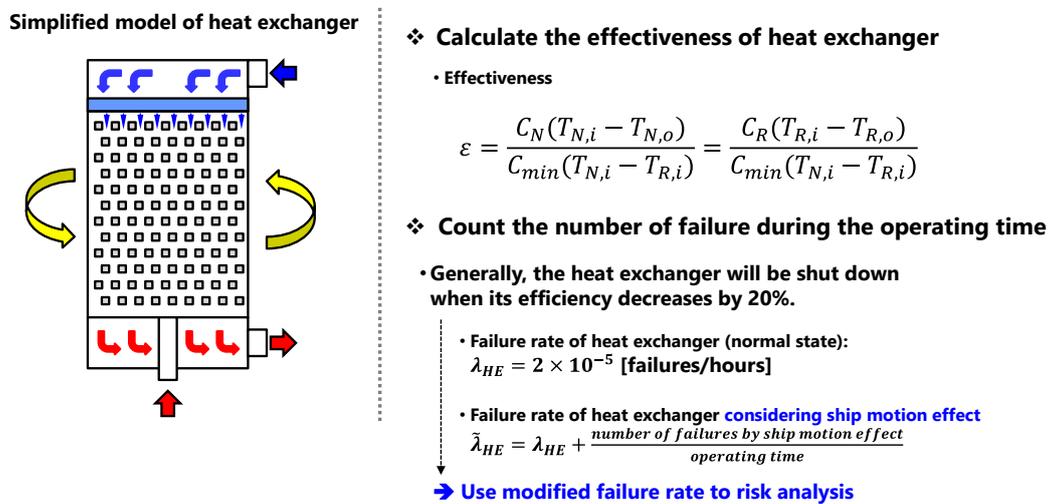


Figure 5-15 Analysis of the motion effect on the heat exchanger using LBM and its application to reliability analysis

In general, if the effectiveness of the heat exchanger decreases by 20%, the heat exchanger should be shut down to preserve the equipment and the whole liquefaction cycle. Thus, the number of failures can be counted by calculating the effectiveness of the heat exchanger as time goes by. This failure number can be applied to the failure rate of the heat exchanger, and the modified failure rate can be used for the reliability analysis. The analysis of the heat exchanger using the LBM is remained as a future work.

6. Conclusions

In this thesis, a general logical model for LNG FPSO liquefaction cycles was developed using the system entity structure (SES). The SES is an ontological framework for representing the elements of a system and their relationships in a hierarchical manner. By pruning the SES, various alternatives of the logical model for the liquefaction cycle can be generated automatically.

Process simulation is performed to determine the optimal operating conditions of the liquefaction cycles in terms of effectiveness. The sequential modular simulator is used for the process simulation because it has some advantages, like equipment modularization, easy debugging, and convenience for modifying liquefaction cycles. Each equipment for the sequential modular simulator is modeled based on the DEVS (discrete event system specification) formalism, and is connected by the logical model of liquefaction cycles. Process simulation is performed for the DMR cycle using the sequential modular simulator.

The reliability block diagram (RBD) method, which is a graphical representation of a system diagram connecting the subsystems of the components according to their reliability relationships, is used in this thesis. It can be easily generated from the generic logical model of the liquefaction cycle. A quantitative analysis of RBD can be conducted using Monte Carlo simulation, and the liquefaction system for reliability simulation is modeled based on the DEVS formalism. Reliability simulations are performed for the conventional liquefaction cycles, such as the C3MR, DMR, cascade, dual N₂ expander, and Niche (N₂-methane) cycles.

The influence of ship motion on the performance of the heat exchanger was considered.

The change in effectiveness due to the ship motion can be calculated using the lattice Boltzmann method (LBM). During the operating time, the failure of the heat exchanger can be counted considering this effectiveness, and the result can be applied to the reliability simulation.

Future work will focus on the process simulation of the conventional liquefaction cycles such as the C3MR, Cascade, N₂ expander, and Niche (N₂-methane) cycles. Through this study, the efficiency of the conventional liquefaction cycles will be compared. In addition, the change in effectiveness due to the tilt and movements caused by the ship motion will be calculated and its result will be applied to the reliability simulation of the liquefaction cycles.

References

- [1] Bang, K.W., 2006. Combined discrete event and discrete time simulation framework for shipbuilding process planning, Master Thesis, Seoul National University.
- [2] Barclay M., and Denton, N., 2005. "Selecting offshore LNG processes," LNG journal in Foster Wheeler Energy Ltd., UK, pp. 34-36.
- [3] Barclay, M., and Shukri, T., 2000. "Enhanced single mixed refrigerant process for stranded gas liquefaction," Proceedings of 79th Annual GPA Convention, Atlanta, USA.
- [4] Bliault, A., 2001. "Shell Floating LNG Plant, Technology Ready for Project Development," Offshore, 61 (5), p. 102.
- [5] Bobbio, A., Portinale, L., Minichino, M., and Ciancamerla, E., 2001. "Improving the analysis of dependable systems by mapping fault trees into Bayesian networks," Reliability Engineering & System Safety, 71(3), pp. 249-260.
- [6] Borgnakke, C. and Sonntag, R. E., 2009. Fundamentals of thermodynamics, 7th edition, John Wiley & Sons.
- [7] Boyd, M.A., 1991. Dynamic Fault Tree Models: Techniques for Analysis of Advanced Fault-Tolerant Computer Systems, Doctoral Thesis, Dept. of Computer Science, Duke Univ.
- [8] Burden, R.L. and Faires, J.D., 2010. Numerical Analysis, 9th edition, Brooks Cole.
- [9] Cengel, Y.A., 2007. Introduction to thermodynamics and heat transfer, 2nd edition, McGraw-Hill Science/Engineering/Math.

- [10] Cha, J.H., Lee, K.Y., Ham, S.H., Roh, M.I., Park, K.P., and Suh, H.W., 2009. "Discrete event/discrete time simulation of block erection by a floating crane based on multibody system dynamics," Proceedings of 19th International Offshore and Polar Engineering Conference (ISOPE), Osaka International Convention Center, Osaka, Japan, pp. 678-685.
- [11] Chang, K.P., Rausand, M., and Vatn, J., 2008. "Reliability assessment of reliquefaction systems on LNG carriers," Reliability Engineering & System Safety, 93(9), pp.1345-1353.
- [12] Cullinane, J., Yeh, N., and Grave, E., 2011. "Effects of Tower Motion on Packing Efficiency," Proceedings of Brasil Offshore Conference and Exhibition, Macaé, Brazil, pp. 1-14.
- [13] Dahmani, Y., and Hamri, M.E., 2011. "Event triggering estimation for Cell-DEVS: wildfire spread simulation case," Proceedings of 2011 Fifth UKSim European Symposium on Computer Modeling and Simulation (EMS), pp. 144-149.
- [14] Distefano, S., and Puliafito, A., 2009. "Dependability evaluation with dynamic reliability block diagrams and dynamic fault trees," IEEE Transactions on Dependable and Secure Computing, 6(1), pp.4-17.
- [15] Edgar, T.F., 2001. Optimization of chemical processes, 2nd edition, McGraw-Hill.
- [16] Gu, Y., and Ju, Y., 2008. "LNG-FPSO: Offshore LNG solution," Frontiers of Energy and Power Engineering in China, 2(3), pp. 249-255.
- [17] Ha, S., Cha, J.H., Roh, M.I., and Lee, K.Y., 2012. "Implementation of the submarine diving simulation in a distributed environment," International Journal of Naval Architecture and Ocean Engineering, 4(3), pp. 211-227.

- [18] Ha, S., Ku, N.K., Lee, K.Y., 2011. "Battle space model based on lattice gas automata for underwater warfare simulation", Proceedings of Asia Simulation Conference 2011, pp. 361-376.
- [19] Ha, S., Ku, N.K., Roh, M.I., and Lee, K.Y., 2012. "Cell-based evacuation simulation considering human behavior in a passenger ship," Ocean Engineering, 53, pp.138-152.
- [20] Hardy, J., Pomeau, J., and de Pazzis, O., 1973. "Time evolution of a two-dimensional classical lattice system", Physical Review Letters, 31(5), pp. 276-279.
- [21] Hetland, J., Kvamsdal, H.M., Haugen, G., Major, F., Karstad, V., and Tjellander, G., 2009. "Integrating a full carbon capture scheme onto a 450 MWe NGCC electric power generation hub for offshore operations: presenting the Sevan GTW concept," Applied Energy, 86(11), pp. 2298-2307.
- [22] Hwang, J.H., Roh, M.I., and Lee, K.Y., 2013. "Determination of the optimal operating conditions of the dual mixed refrigerant cycle for the LNG FPSO topside liquefaction process," Computers & Chemical Engineering, 49, pp. 25-36.
- [23] Jensen, J.B., 2008. Optimal operation of refrigeration cycles, Doctoral Thesis, Norwegian University of Science and Technology.
- [24] Kim, H., Park, C., Lee, J., and Kim, W., 2010. "Optimal degree of superheating of the liquefaction process for application of LNG FPSO", Proceeding of the SNAK (Society of Naval Architecture of Korea) Spring Conference, Korea, pp. 961-965 (in Korean).
- [25] Lee, G.C., Smith, R., and Zhu, X.X., 2002. "Optimal synthesis of mixed refrigerant systems for low temperature processes," Industrial & Engineering Chemistry Research, 41, pp. 5016-5028.

- [26] Lee, H., and Zeigler, B.P., 2010. "SES-based ontological process for high level information fusion," Proceedings of the 2010 Spring Simulation Multiconference, San Diego, CA, USA, pp. 129:1–129:10.
- [27] Lee, J.C., 2012. Optimal synthesis of liquefaction cycles for LNG FPSO considering operating conditions, Master Thesis, Seoul National University.
- [28] Lee, K.Y., Cho, S.H., and Roh, M.I., 2002. "An efficient global-local hybrid optimization method using design sensitivity analysis," International Journal of Vehicle Design, 28(4), pp. 300-317.
- [29] Lex, T., Ohlig, K., Fredheim, A.O., and Jenssen, C.B., 2007. "Ready for floating LNG - qualification of spiral wound heat exchangers," Proceedings of 15th International Conferences on Liquefied Natural Gas, Barcelona, Spain, pp. PO-27.1-PO-27.14.
- [30] Mohamad, A.A., 2011. Lattice Boltzmann method: fundamentals and engineering applications with computer codes, 1st edition, Springer.
- [31] Moran, M. J. and Shapiro, H. N., 2008. Fundamentals of engineering thermodynamics, 6th edition, Wiley.
- [32] Morbach, J., Yang, A., and Marquardt, W., 2007. "OntoCAPE-A large-scale ontology for chemical process engineering," Engineering Applications of Artificial Intelligence, 20(2), pp.147-161.
- [33] Nishida, N., Stephanopoulos, G., and Westerberg, A.W., 1981. "A review of process synthesis," AIChE Journal, 27(3), pp. 321-351.
- [34] Pekdemir, T., and White, G., 2007. "Marine Motion Effect: Is Your Processing Equipment Immune to Seasickness," Presented at Gas Processors Association (GPA) Annual Conference, Oslo, Norway.

- [35] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., 2007. Numerical recipes - The art of scientific computing, 3rd edition, Cambridge University Press.
- [36] Shin, S.K., and Seong, P.H., 2008. "Review of various dynamic modeling methods and development of an intuitive modeling method for dynamic systems," Journal of Korean Nuclear Society, 40(5), pp. 375-386.
- [37] Smith, J.M., Van Ness, H.C., and Abbott, M.M., 2005. Introduction to chemical engineering thermodynamics, 7th edition, McGrawHill.
- [38] Stamatelatos, M., 2002. Fault Tree Handbook with Aerospace Applications Version 1.1.
- [39] Venkatarathnam, G., 2008. Cryogenic mixed refrigerant processes, Springer, New York, USA.
- [40] Vesely, W.E., Goldberg, F.F., Roberts, N.H., and Haasl, D.F., 1981. Fault tree handbook, NUREG-0492, US Nuclear Regulatory Commission.
- [41] Wolf-Gladrow, D.A., 2005. Lattice-gas cellular automata and lattice Boltzmann models: an introduction, 1st edition, Springer.
- [42] Zeigler, B.P., 1990. Object-oriented simulation with hierarchical, modular models, Academic Press, Inc.
- [43] Zeigler, B.P., Praehofer, H., and Kim, T.G., 2000. Theory of modeling and simulation, Academic Press, New York, USA.

APPENDICES

A. Simulation Kernel based on Discrete Event System Specification (DEVS) and Discrete Time System Specification (DTSS) Formalism

A.1. DEVS and DTSS Formalism

A virtual object which acts as an actual object is called a *simulation model* (simply, *model*) and the process which makes such object is called *modelling*. A model consists of state variables, external and internal events, and state transition functions. As a set of instructions, a model needs some agent capable of actually executing the instructions and generating behaviour and the agent is called a *simulation engine*. To derive the process that status variables change by means of external and internal events, that is, to find the change of state variables according to the change of time is called *simulation*. A simulation that the state of a model changes by means of any events is called a *discrete event simulation*. The discrete event simulation processes the events, which change state variables of a model, in the order in which they occur. A simulation which calculates the state of a model every unit time is called a *discrete time simulation*. The discrete time simulation is being mostly used for analyzing dynamics or mechanics systems because it calculates the state of a model every unit time.

Ziegler et al. (2000) proposed formal structures, Discrete Event System Specification

(DEVS) and Discrete Time System Specification (DTSS), which can handle simulation models of a discrete event and discrete time. They are widely used as a standard formalism of modelling and simulation. This chapter presents the concept of modeling and simulation and the structure of simulation kernel based on DEVS and DTSS formalism.

A.1.1. DEVS Formalism

The DEVS formalism is a hierarchical and modular modeling approach, centered on the state concept. In its basic form, it does not take in account the system structure evolution; only the states can evolve/move. Each system is described by two points: functional (behavioral) and structural aspect. Likewise, DEVS formalism is composed of two types of models: *atomic models* and *coupled models*. The atomic model represents the basic behavior of the system whereas the coupled model, its internal structure.

(1) Atomic model

The *atomic models* are the basic components of the DEVS formalism. Their operation is close to the “state-machines”. Formally, an atomic model of the DEVS formalism is specified by 7 tuples:

$$M = \langle X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta \rangle. \quad (A-1)$$

These tuples are described in Table A-1.

Table A-1 Components of an atomic model based on DEVS formalism

Content		Description
X	Input	$X = \{(p, v) \mid p = IPorts, v \in X_p\}$ is the set of input event, where $IPorts$ represents the set of input ports and X_p represents the set of values for the input ports.

Y	Output	$Y = \{(p, v) \mid p = OPorts, v \in Y_p\}$ is the set of output event, where $OPorts$ represents the set of output ports and Y_p represents the set of values for the output ports.
S	State Variable	S is the set of sequential states.
δ_{ext}	External Transition Function	$\delta_{ext}: Q \times X \rightarrow S$ is the external state transition function, with $Q = \{(s, e) \mid s \in S, e \in [0, ta(s)]\}$ and e is the elapsed time since the last state transition. It defines the state changes due to external events.
δ_{int}	Internal Transition Function	$\delta_{int}: S \rightarrow S$ is the internal state transition function, and it models the states changes caused by internal events.
λ	Output Function	$\lambda: S \rightarrow Y$ is the output function, and maps the internal state onto the output set.
ta	Time Advance Function	$ta = S \rightarrow R^+ \cup \infty$ is the time advance function, and it describes the lifetime of the state.

At any given moment, a DEVS model is in a state $s \in S$. In the absence of external events, it remains in that state for a lifetime defined by $ta(s)$. When $ta(s)$ expires, the model outputs the value $\lambda(s)$ through a port $y \in \gamma$, and it then changes to a new state given by $\delta_{int}(s)$. A transition that occurs due to the consumption of time indicated by $ta(s)$ is called an internal transition. On the other hand, an external transition occurs due to the reception of an external event. In this case, the external transition function determines the new state, given by $\delta_{ext}(s, e, x)$, where s is the current state, e is the time elapsed since the last transition, and $x \in X$ is the external event that has been received.

The time advance function can take any real value between 0 and ∞ . A state for which $ta(s) = 0$ is called a transient state (which will trigger an instantaneous internal transition). In contrast, if $ta(s) = \infty$, then s is said to be a passive state, in which the system will remain perpetually unless an external event is received (can be used as a termination

condition).

(2) Coupled model

A *coupled model* is modular and presents a hierarchical structure, which allows the creation of complex models starting from atomic and/or coupled models. It is described as follows:

$$CM = \langle X, Y, \{M_d\}, EIC, EOC, IC, select \rangle. \quad (A-2)$$

The detailed description of these components are shown in Table A-2.

Table A-2 Components of a coupled model based on DEVS formalism

Content		Description
X	Input	$X = \{(p, v) \mid p = IPorts, v \in X_p\}$ is the set of input event, where $IPorts$ represents the set of input ports and X_p represents the set of values for the input ports.
Y	Output	$Y = \{(p, v) \mid p = OPorts, v \in Y_p\}$ is the set of output event, where $OPorts$ represents the set of output ports and Y_p represents the set of values for the output ports.
$\{M_d\}$	Set of component models	These components are either atomic or coupled model.
EIC	External Input Coupling	This represents the relationship between its own external inputs and inputs of child models.
EOC	External Output Coupling	This represents the relationship between its own external outputs and output of child models.
IC	Internal Coupling	This represents the relationship between child models.
$select$	Tiebreaker	This function set priorities of child models when the events fires simultaneously.

A.1.2. DTSS Formalism

The DTSS formalism is a model structure which continuously calculates the state of a model every unit time. An atomic model based on the DTSS formalism has similar structure to that of the DEVS formalism, and it is connected with the atomic model of the DEVS formalism. Formally, an atomic model of the DTSS formalism is specified as follows:

$$M = \langle X, Y, Q, f, \int f, C_{int}, \lambda \rangle. \quad (A-3)$$

The detailed description of these components are shown in Table A-3.

Table A-3 Components of a atomic model based on DTSS formalism

Content		Description
X	Input	$X = \{(p, v) \mid p = IPorts, v \in X_p\}$ is the set of input event, where $IPorts$ represents the set of input ports and X_p represents the set of values for the input ports.
Y	Output	$Y = \{(p, v) \mid p = OPorts, v \in Y_p\}$ is the set of output event, where $OPorts$ represents the set of output ports and Y_p represents the set of values for the output ports.
Q	State Variable	S is the set of sequential states.
f	Rate of Change	$f: Q \times X \rightarrow Q$, and it defines the state changes due to external events.
$\int f$	Integral Function	$\int f: Q \rightarrow Q$, and it changes its own state by intergration method every unit time according to given function.
C_{int}	State Event Function	It sends a trigger when specific state variables reach the given constraints.
λ	Output Function	$\lambda: S \rightarrow Y$ is the output function, and maps the internal state onto the output set.

A.2. Simulation based on the DEVS Formalism

Figure A-1 shows the concept of torpedo-firing simulation based on the DEVS formalism. The definition of simulation models and the simulation procedure is detailed as follows.

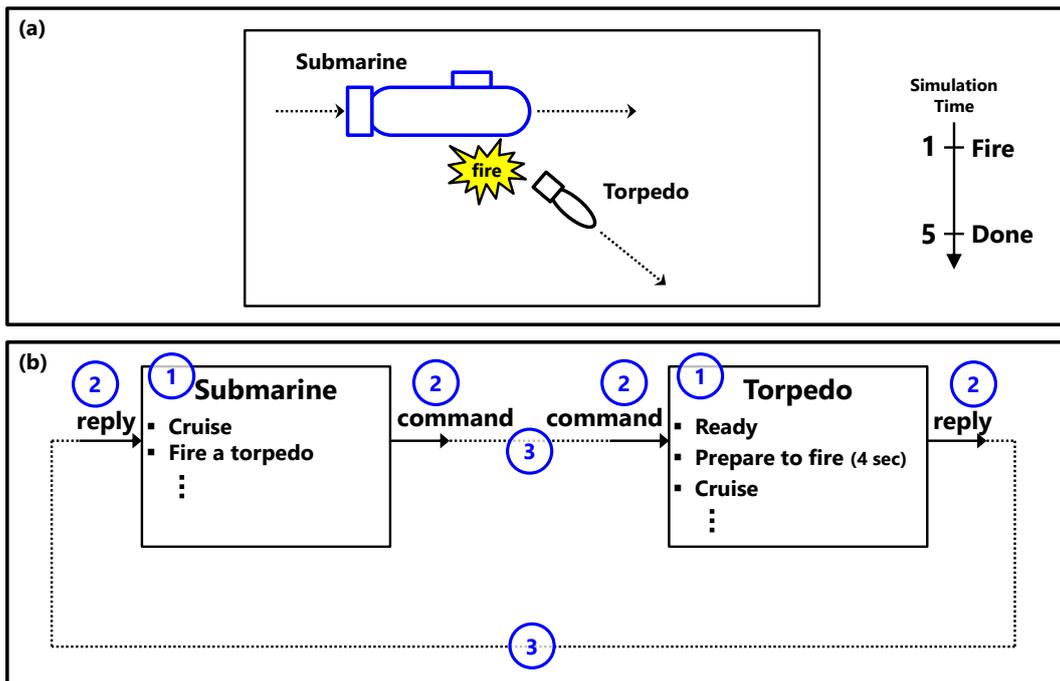


Figure A-1 (a) Concept of torpedo-firing simulation, (b) Configuration of model based on the DEVS formalism

A.2.1. Definition of Simulation Model

In this example, we want to configure a simulation which represents the procedure of firing torpedo. The procedure for making simulation models is as follows.

(1) Define the functions of the simulation model (Figure A-1 (b)-①)

As shown in Figure A-1 (b), the torpedo-firing simulation needs two simulation models: *submarine* model and *torpedo* model. These two model has the function for calculating their own position every unit time. The *submarine* model has the function to send torpedo-firing command to the *torpedo* model, and the *torpedo* model has the function for firing torpedo after receiving this command from the *submarine* model. In this example, the preparing time to fire a torpedo is assumed as 4 seconds.

(2) Input and output configuration of simulation models (Figure A-1 (b)-②)

In the simulation, each model has its own input and output to exchanger the messages with other models. The *submarine* model has two ports: output *command* for sending a torpedo-firing command and input *reply* for receiving a end-signal. The *torpedo* model also has two ports: input *command* for receiving a torpedo-firing command and output *reply* for announcing the completion of firing torpedo.

(3) Interconnection between simulation models (Figure A-1 (b)-③)

In the torpedo-firing simulation, the *submarine* model and the *torpedo* model exchange the messages with each other. The output *command* of the *submarine* model is connected with the input *command* of the *torpedo* model, and the output *reply* of the *torpedo* model is connected with the input *reply* of the *submarine* model.

A.2.2. Procedure of Simulation Based on DEVS and DTSS Formalism

After finishing the functional definition of simulation models, simulation models are defined based on the DEVS formalism. In this example, as shown in Figure A-2, the submarine and torpedo model is defined as an atomic model, which is composed of 7 components: state variables, inputs, outputs, external transition function, internal transition function, output function, and time advance function. Each atomic model is connected with a *simulator*, which manages the events of the connected atomic model and fires the event at suitable moment.

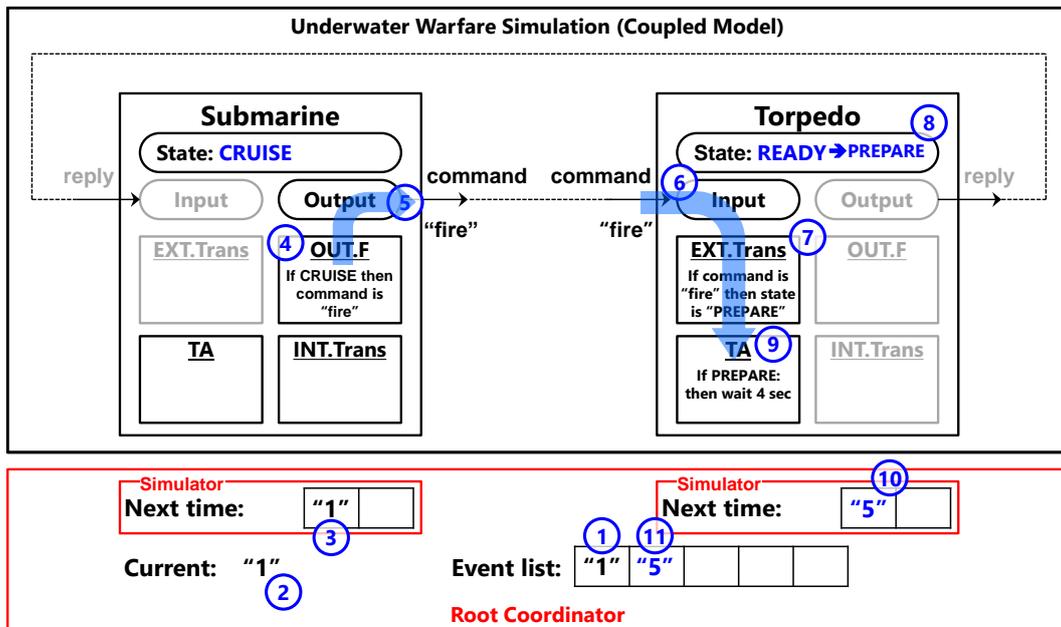


Figure A-2 Procedure of torpedo-firing simulation based on the DEVS formalism (1/3)

Two atomic models, submarine and torpedo, belong to the coupled model *Underwater Warfare Simulation*, and this coupled model is also connected with a *coordinator*, which manages the event list of the child models. If the coupled model represents whole system for simulation, the connected coordinator is a *root coordinator*, and it manages the time of

the simulation additionally.

The DEVS-based simulation is executed as following procedure.

- (a) Choose the nearest time in the event lists of the root coordinator, and set current time to it. (Figure A-2-①, ②)
- (b) Choose the model which has the same time in the simulator, so *Submarine* model was chosen. (Figure A-2-③)
- (c) Execute the output function of the model chosen (Figure A-2-④). Since current state of *Submarine* is *CRUISE*, *Submarine* will send the message *fire* through the output command according to its output function (Figure A-2-⑤).
- (d) After sending the message *fire*, this message is transfer to the input *command* of *Torpedo* according to the pre-defined interconnection between the models Figure A-2-⑥).
- (e) After receiving the message, the simulator of *Torpedo* executes the external transition function. This function confirms that the message is *fire*, and changes the state to *PREPARE* (Figure A-2-⑦, ⑧).
- (f) The simulator of *Torpedo* executes the time advance function, and set the next time of the simulator according to the lifetime of current state *PREPARE*. Thus, the next time of the simulator is changed to 5 seconds (Figure A-2-⑨, ⑩), and this time

also registers to the event lists of the root coordinator at the same time (Figure A-2-⑪).

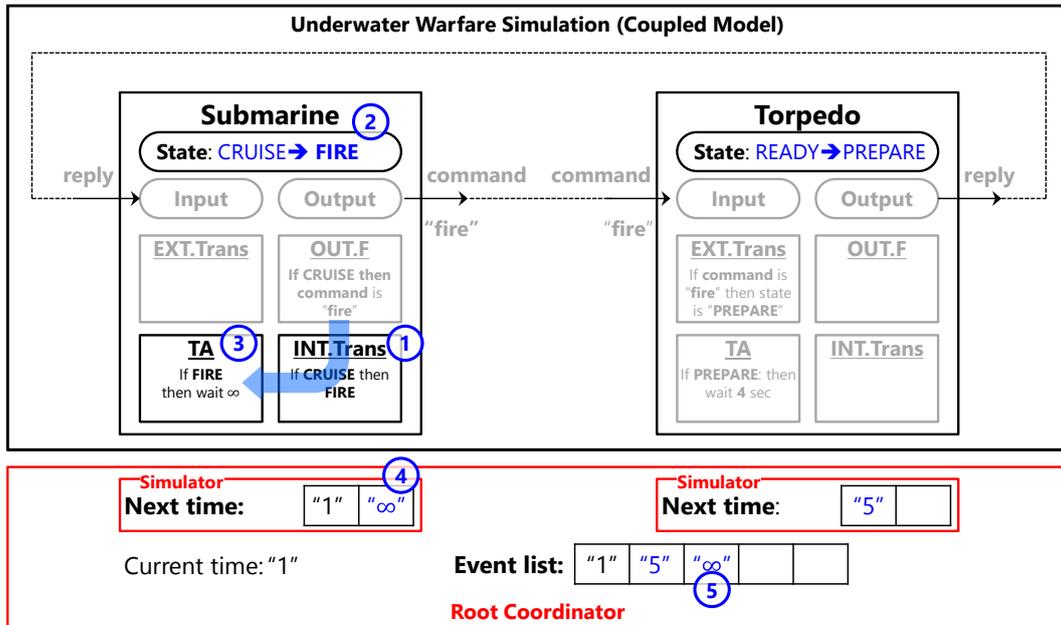


Figure A-3 Procedure of torpedo-firing simulation based on the DEVS formalism (2/3)

- (g) After finishing the execution of connected model *Torpedo*, return to *Submarine* and execute the internal transition function (Figure A-3-①). This function confirms current state, and changes current state to *FIRE* (Figure A-3-②).
- (h) The simulator of *Submarine* executes the time advance function (Figure A-3-③). Since current state is *FIRE*, this function registers the next time as infinity (∞) to wait until other external event will arrive (Figure A-3-④, ⑤).
- (i) After finishing the execution of *Submarine*, choose the time 5 in the event lists of

the root coordinator in order to execute next event (Figure A-4-①), and change current time to 5 (Figure A-4-②). After then, the following procedure is same as step (b).

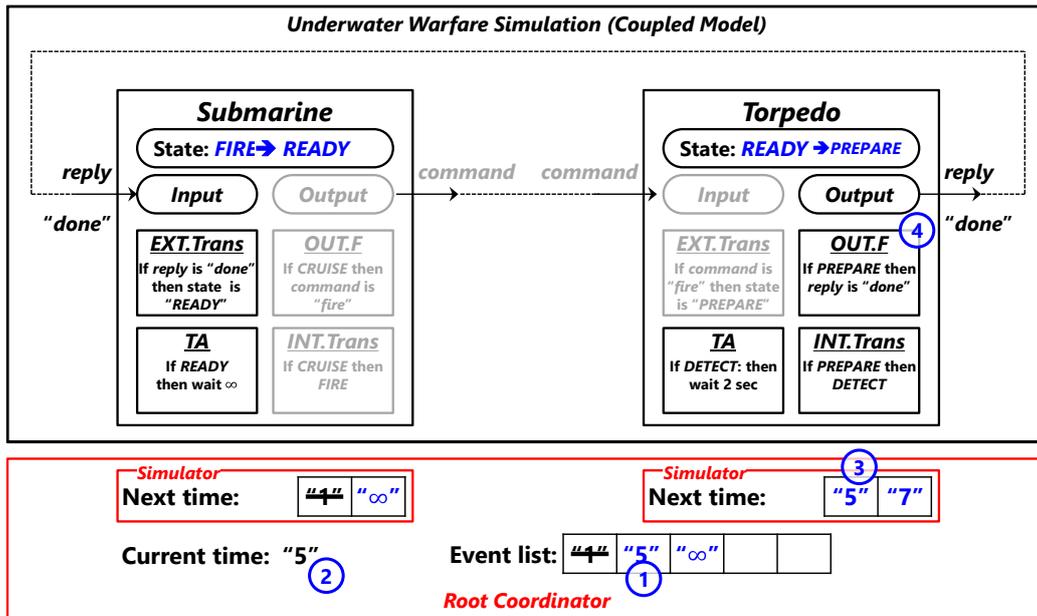


Figure A-4 Procedure of torpedo-firing simulation based on the DEVS formalism (3/3)

As shown in torpedo-firing simulation above, the procedure for simulation based on the DEVS formalism is based on two rules:

- Rule 1: execute the output function → send the message through the output → receive the message through the connected input → execute the external transition function according to the received message → execute the time advance function
- Rule 2: after finishing the execute of the output function (finishing Rule 1), execute the internal transition function → execute the time advance function.

Moreover, as shown in the example above, four functions are dependent on the state variables, inputs, and outputs. The change of the state variables affects these four functions,

the change of the inputs affects the external transition function, and the change of the outputs affects the output function.

A.3. Simulation based on the DEVS and DTSS Formalism

As shown in Figure A-1, the preparing duration for firing a torpedo is assumed as 4 seconds in the torpedo-firing simulation. However, if it is necessary to calculate the preparing duration more precisely, we should calculate the duration using precise equations. In this case, a discrete time simulation model is needed to calculate the equations every unit time.

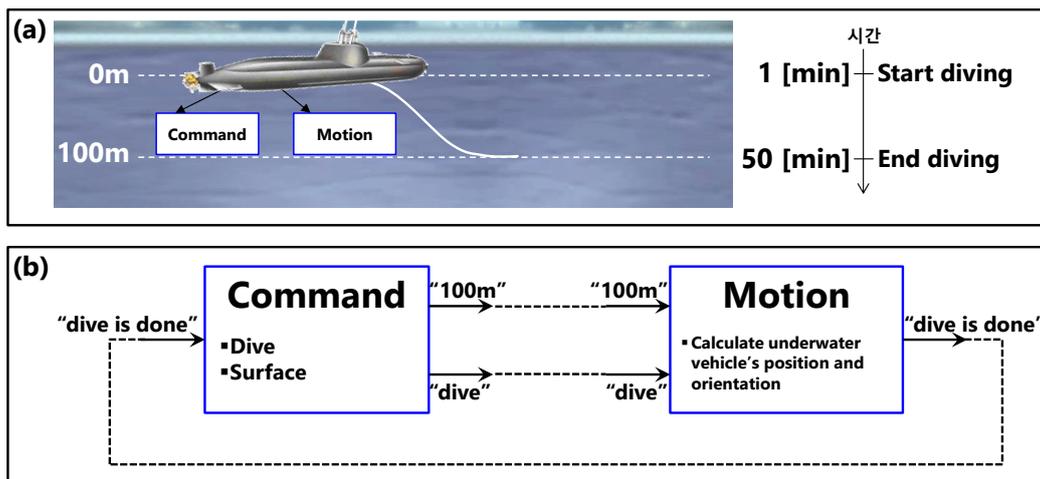


Figure A-5 (a) Concept of submarine diving simulation, (b) configuration of models for submarine diving simulation

Figure A-5 shows the configuration of models for submarine diving simulation based on the DEVS and DTSS formalism. The procedure of simulation based on the DTSS formalism is as follows.

(1) Activation of the discrete time model

- (a) Find the nearest time in the event list of the root coordinator, and set current time to it (Figure A-6-①, ②).
- (b) Find the simulator whose next time is same as current time (Figure A-6-③).

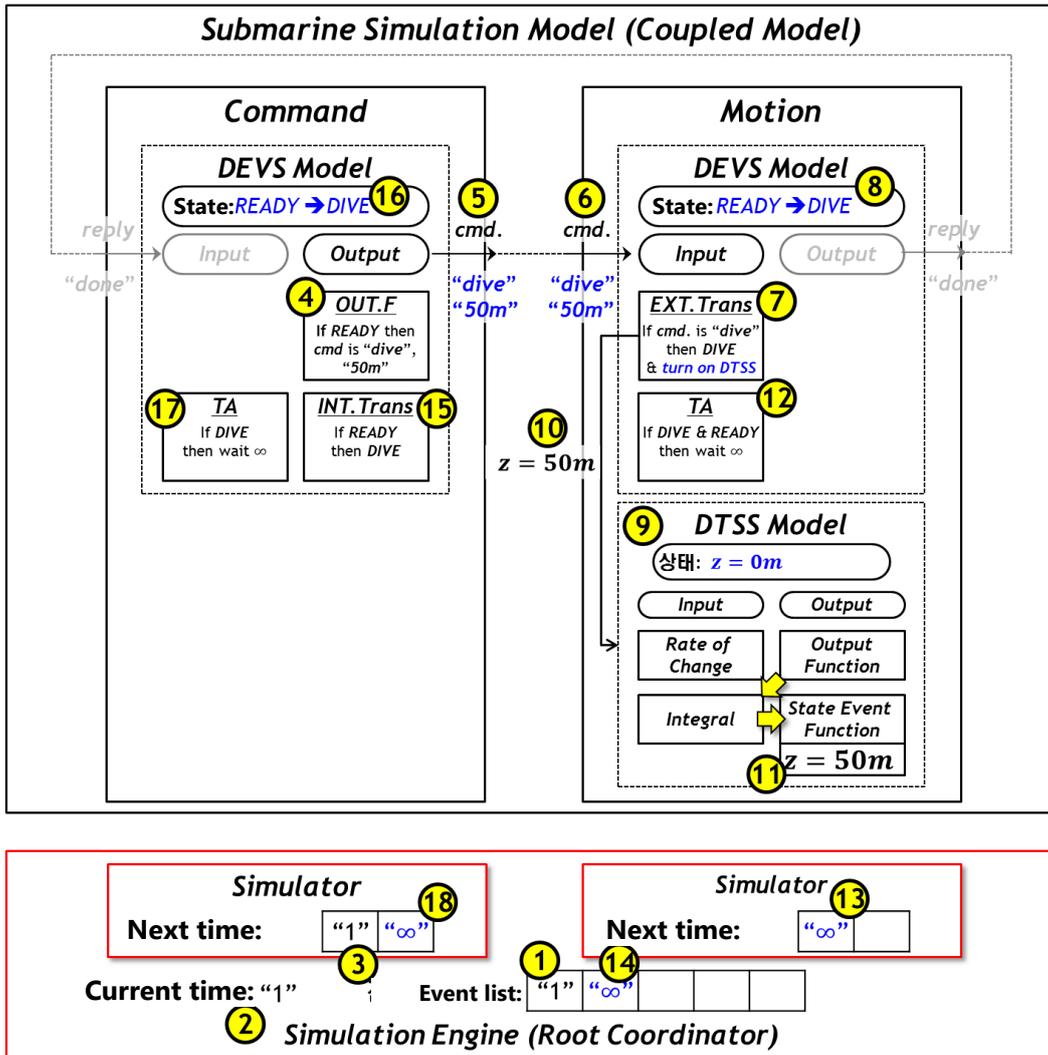


Figure A-6 Procedure of submarine diving simulation based on the DEVS and DTSS formalism

(1/3)

- (c) Execute the output function of *Command* model which is connected with the

simulator found on step (b) (Figure A-6-④). After executing the output function, *Command* model sends the values ‘*dive*’ and ‘*50m*’ through the output *cmd* (Figure A-6-⑤).

(d) The values ‘*dive*’ and ‘*50m*’ arrives through the input *cmd* of *Motion* model which is connected with the output *cmd* of *Command* model (Figure A-6-⑥).

(e) After receiving the values through the input *cmd*, the simulator of *Motion* model executes the external transition function (Figure A-6-⑦). According to the values received through the input *cmd*, the external transition function of *Motion* model changes the state of *Motion* model to *DIVE* (Figure A-6-⑧), and activate the DTSS model (Figure A-6-⑨). At the same time, the constraints of the DTSS model is set to the “*target depth = 50m*” (Figure A-6-⑩, ⑪).

(f) Execute the time advance function of *Motion* model (Figure A-6-⑫) and register the lifetime of the state *DIVE* (∞ in this example) to the next time of the simulator (Figure A-6-⑬, ⑭).

(g) After executing the time advance function of *Motion* model, execute the internal transition function of *Command* model (Figure A-6-⑮) and change the state of *Command* model to *DIVE* (Figure A-6-⑯).

- (h) Execute the time advance function of *Command* model (Figure A-6-⑰) and register the lifetime of the state *DIVE* (∞ in this example) to the next time of the simulator (Figure A-6-⑱).

(2) Procedure of discrete time simulation based on the DTSS formalism

As shown in Figure A-6, all times in the event list of the root coordinator is ∞ after finishing the execution of *Command* and *Motion* model. In case of the discrete time simulation based on the DEVS formalism, the simulation is finished if all times in the event list of the root coordinator is ∞ . However, if there are some activated DTSS models, the simulation will be continued by following rules.

- 1) If there is an activated DTSS model, the simulation time will increase by pre-defined unit time,
- 2) Execute the activated DTSS models. Execute the output function, integral function, and state event function of the DTSS model sequentially.
- 3) The state event function checks the pre-defined goal of the DTSS model. If the goal is satisfied, register current time to the next time of the connected simulator.
- 4) After finishing the execution of the activated DTSS models, execute the DEVS models by aforementioned rules in section A.2.

By following the rules from 1) to 4), the procedure of submarine diving simulation is as follows.

- (a) Since the DTSS model of *Motion* model is activated, the simulation time

increases by the pre-defined unit time (1 minute in this example) by the rule 1)
(Figure A-7-①).

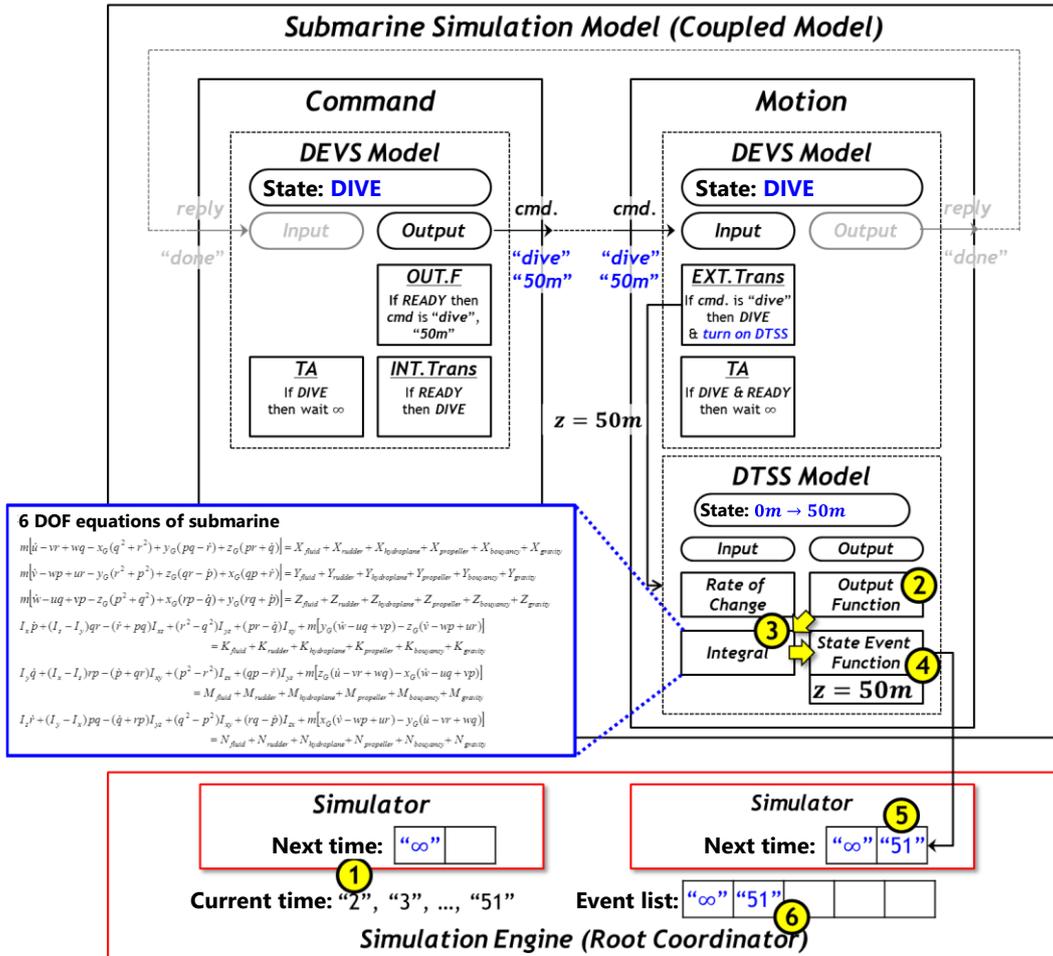


Figure A-7 Procedure of submarine diving simulation based on the DEVS and DTSS formalism

(2/3)

(b) By the rule 2), execute the DTSS model of *Motion* model. At first, execute the output function of the DTSS model, and send the values through the output (Figure A-7-②).

- (c) Change the state of the DTSS model by executing its integral function (Figure A-7-③). The integral function has 6 DOF equations of submarine, and calculate it every unit time.
- (d) The state event function of the DTSS model checks that the state variable satisfies the constraints (“target depth = 50m” in this example) (Figure A-7-④). If the constraints are not satisfied, execute the DEVS model by the rule 4). In this example, the next time of all simulator is set to ∞ , so there is no DEVS model to execute.
- (e) By the rules from 1) to 4), execute the DTSS model every unit time repeatedly.
- (f) When the simulation reaches 51 minutes, the state event function of the DTSS model confirms that the target depth reaches 50 meters. After then, the state event function registers current time to the next time of the connected simulator (Figure A-7-⑤). Current time is also registered to the event list of the root coordinator (Figure A-7-⑥).
- (g) If the execution of the DTSS model is finished, the DEVS models are executed by the execution rules.

(3) Deactivation of the discrete time simulation model

Since the state event function of the DTSS model registers current time to the next time of the connected simulator, the DEVS models are executed by the rules on DEVS formalism. The execution procedure is as follows.

- (a) Choose the simulator whose next time is same as current time. In this example, the simulator of *Motion* model is chosen (Figure A-8-②).

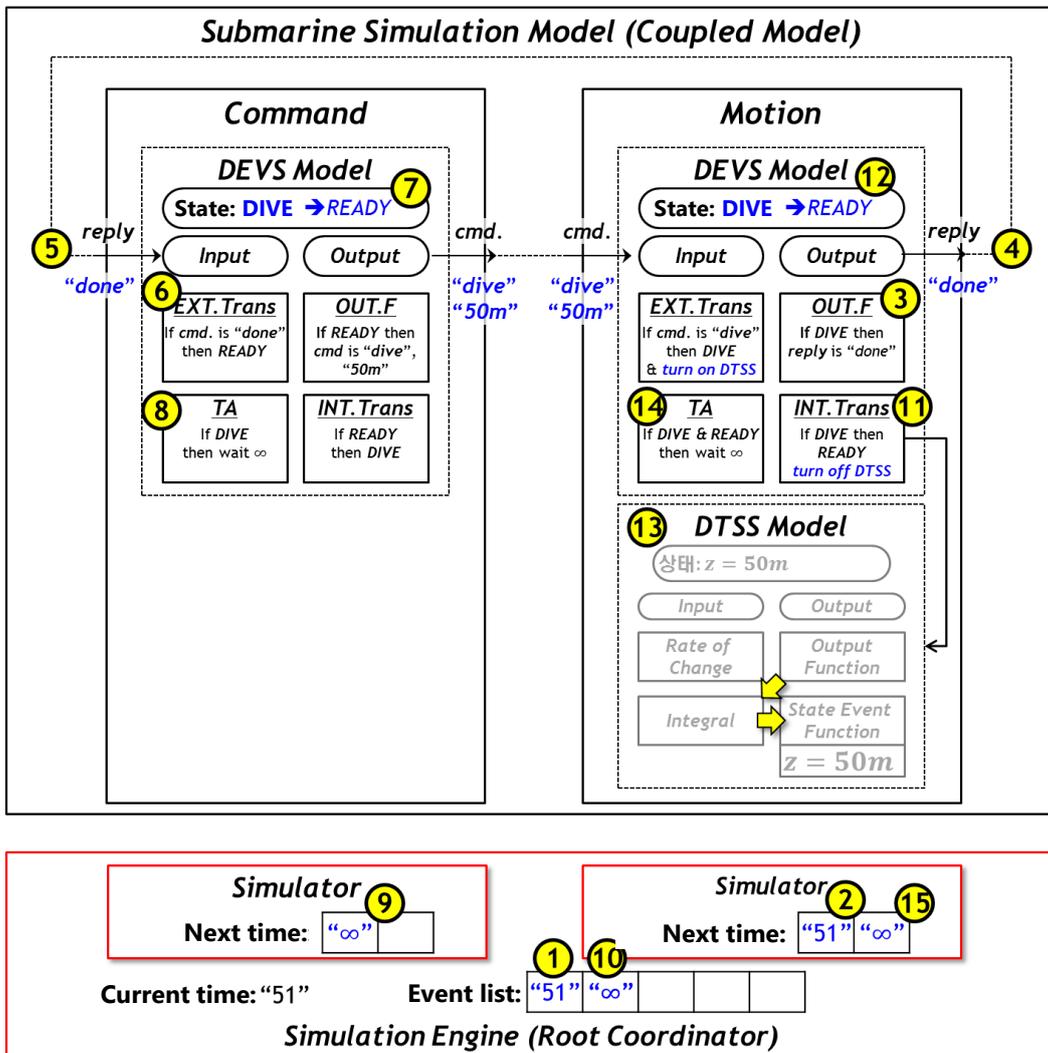


Figure A-8 Procedure of submarine diving simulation based on the DEVS and DTSS formalism (3/3)

- (b) The simulator of *Motion* model executes the output function of *Motion* model (Figure A-8-③). By executing the output function, the value *done* is sent through

the output *reply* of *Motion* model (Figure A-8-④).

- (c) After sending the value through the output *reply* of *Motion* model, the input *reply* of *Command* model receives the value *done* (Figure A-8-⑤).
- (d) After receiving the value through the input *reply* of *Command* model, the simulator of *Command* model executes the external transition function of *Command* model (Figure A-8-⑥). According to the value of the input *reply*, the external transition function changes the state of *Command* model to *READY* (Figure A-8-⑦).
- (e) The simulator of *Command* model executes the time advance function (Figure A-8-⑧). This function registers the lifetime of current state *READY* (∞ in this example) to the next time (Figure A-8-⑨, ⑩).
- (f) After finishing the execution of the time advanced function, the simulator of *Motion* model executes the internal transition function (Figure A-8-⑪). This function changes the state of *Motion* model to *READY* (Figure A-8-⑫), and deactivates the DTSS model of *Motion* model (Figure A-8-⑬).
- (g) The simulator of *Motion* model executes the time advance function of *Motion* model (Figure A-8-⑭). This function registers the lifetime of current state *READY* (∞ in this example) to the next time of the simulator (Figure A-8-⑮).

A.4. Structure of the Simulation Kernel

The simulation kernel developed in this thesis has three types of the class. One is a class which can define the variables such as state variables, inputs, and outputs. Another is a class which can define atomic or coupled models based on the DEVS formalism. The other is a class which can define simulators or a coordinators based on the DEVS formalism.

A.4.1. Variables Used on Simulation Kernel

The variables used on the simulation kernel based on the DEVS formalism are divided with two types: state variables and messages. The state variables are defined by inheriting *CState* class, and the messages for input and output are defined by inheriting *CValue* class. As shown in Figure A-9, *CState* and *CValue* classes are defined by inheriting *CObject* class in Microsoft Foundation Class (MFC) libraries for the convenience.

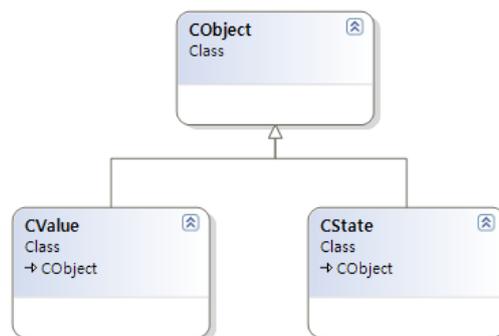


Figure A-9 Class for the definition of variables using on the simulation based on the DEVS and DTSS formalism

The messages are defined by inheriting *CValue* class. As shown in Figure A-10, there are four types of classes for the messages in the simulation kernel: *Integer* class for integer

values, *Float* class for real values, *String* class for string values, and *Message* class for custom messages.

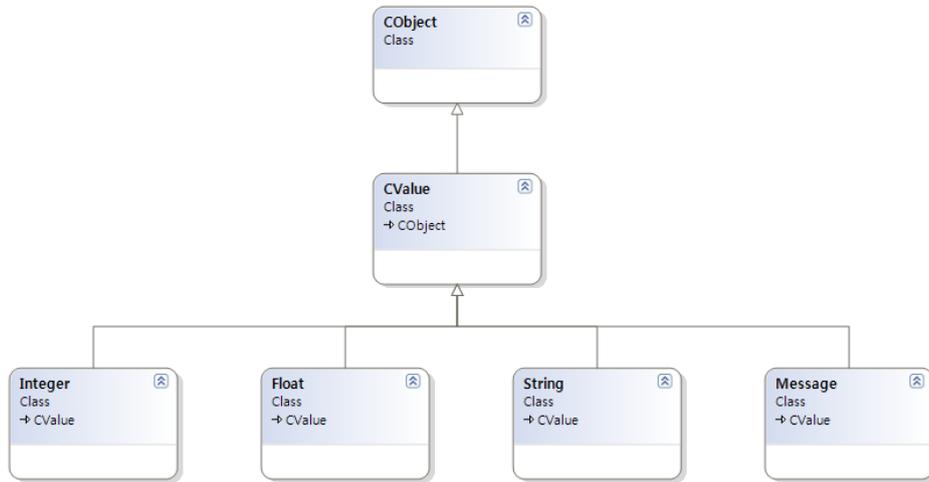


Figure A-10 Classes for messages in the simulation kernel: defined by inheriting ‘CValue’ class

Table A-4 Classes for the messages in the simulation kernel based on the DEVS formalism

Class	Description
Integer	Class for integer values
Float	Class for real values
String	Class for string values
Message	Class for custom messages

The classes for messages are used to define the state variables, inputs, and outputs. For example, the state variables, which have the integer data, have an instance by inheriting *Integer* class. *Message* class is used when amount of data is much more.

(1) Classes for inputs, outputs, and state variables

The state variables, inputs, and outputs are defined by inheriting *CState* class (Figure A-11). *CStateVariable* class is used to define the state variables of the simulation model.

CStatePort class is used to define the inputs and outputs of the simulation kernel. The classes for inputs and outputs are classified in four types: *CStateInputPort*, *CStateOutputPort*, *CStateContinuousInputPort*, and *CStateContinuousOutputPort* class.

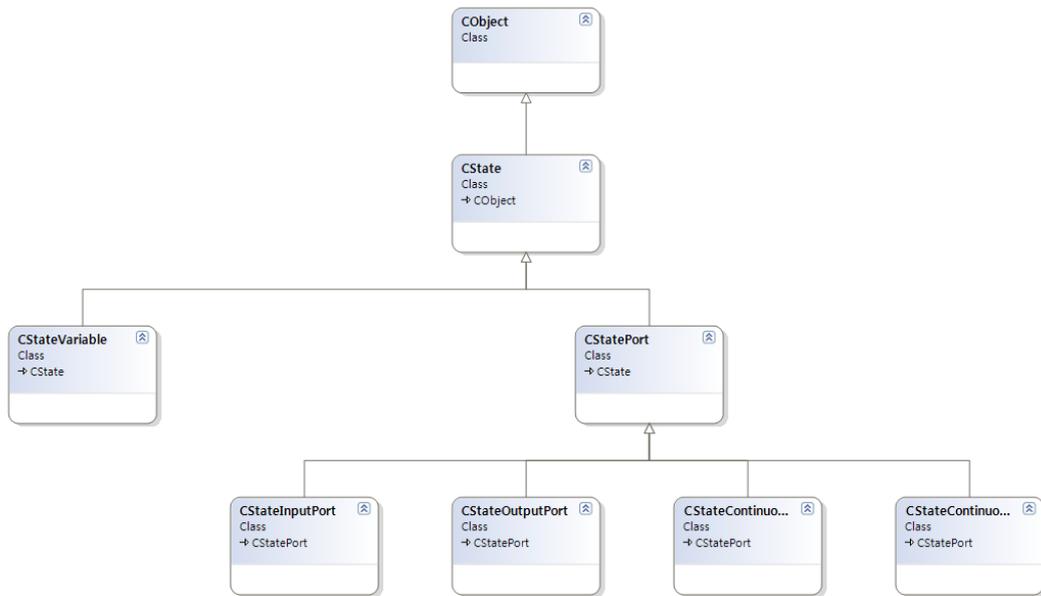


Figure A-11 Classes for inputs, outputs, and state variables: defined by inheriting ‘CState’ class

Table A-5 Classes for inputs, outputs, and state variables

Class	Description
CStateVariable	Class to define the state variables
CStateInputPort	Class to define the input of the DEVS model
CStateOutputPort	Class to define the output of the DEVS model
CStateContinuousInputPort	Class to define the input of the DTSS model
CStateContinuousOutputPort	Class to define the output of the DTSS model

A.4.2. Simulation Model

Figure A-12 shows the class diagram of the simulation models in the simulation kernel. The simulation kernel supplies *CAtomicModel* class to define an atomic model and *CCoupledModel* class to define a coupled model based on the DEVS formalism. In addition, the kernel supplies *CContinuousModel* class to define a DTSS model. The integral function of the DTSS model is defined by inheriting *CEquation* class. *CAtomicModel*, *CCoupledModel*, and *CContinuousModel* classes inherit *CModel* class. *CModel* and *CEquation* class inherit *CObject* class. *CModel* class contains common components of atomic and coupled models for inputs and outputs.

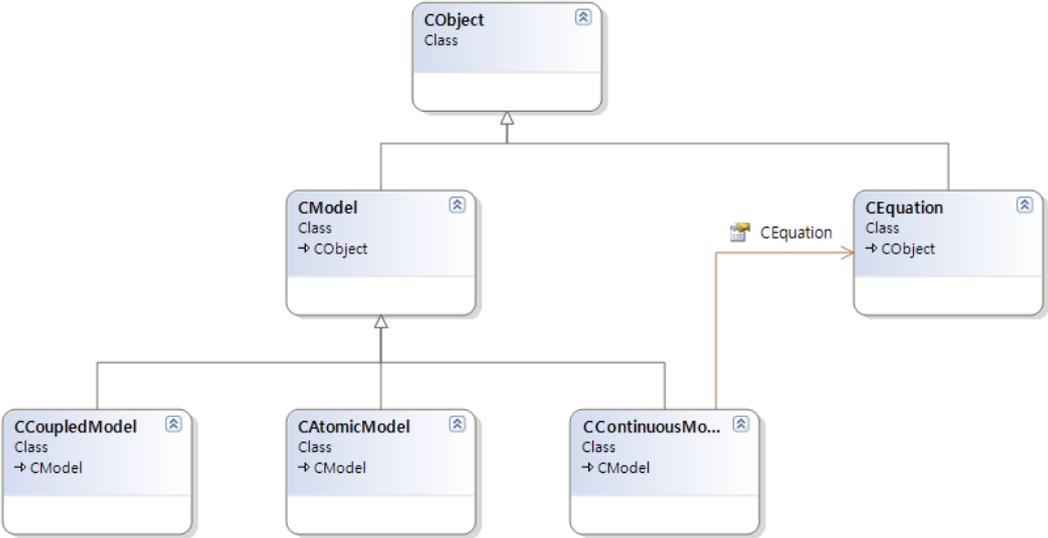


Figure A-12 Classes to define an atomic model and a coupled model

A.4.3. Simulation Engine

Figure A-13 shows the class diagram to define a simulator and a coordinator. All classes inherit *CObject* class in MFC libraries for the convenience. *CSimulationEngine* class is a class for executing a simulation. *CCoordinator* class is a class to define a coordinator which is connected with a coupled model based on the DEVS formalism. *CSimulator* class is a class to define a simulator which is connected with an atomic model based on the DEVS formalism. *CRootCoordinator* class is a class to define a root coordinator.

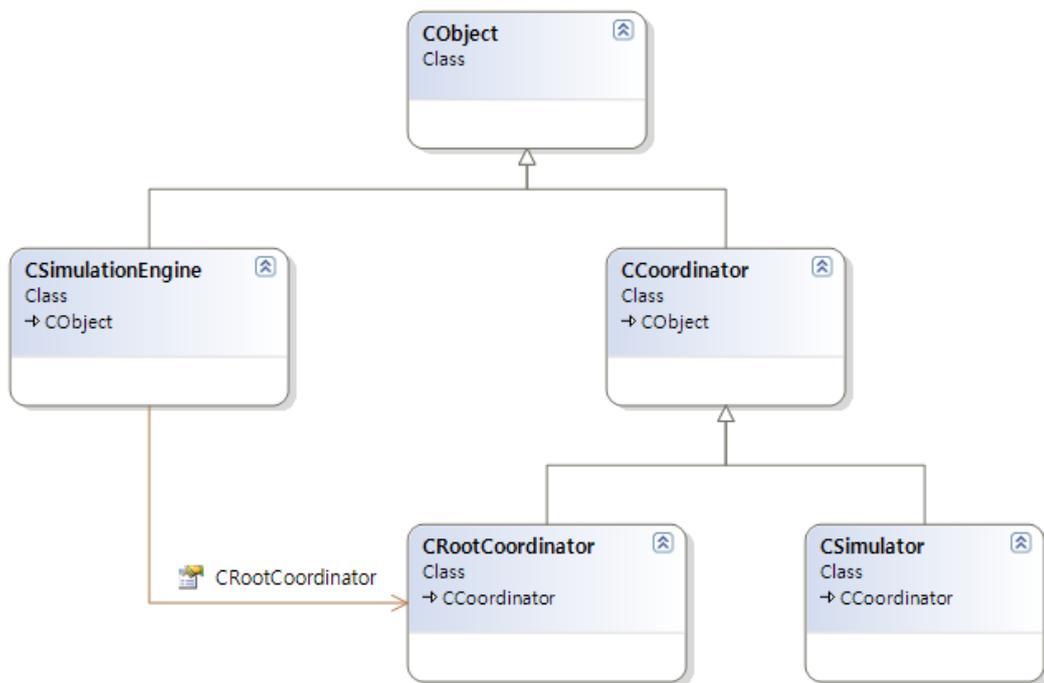


Figure A-13 Classes to define a simulator and a coordinator

B. Mathematical Model of Equipment

Based on Thermodynamics

The cycles have seven main equipment parts: the compressor, condenser, expansion valve, evaporator, phase separator, common header and tee. Many types of liquefaction cycles have been determined according to their respective syntheses. In this section, thermodynamic-based mathematical models of seven main equipment parts are derived to calculate the operating conditions according to Lee's studies (Lee, 2012).

B.1. Compressor

The compressor, usually driven by an electric motor, brings the refrigerant to a high pressure, which also raises its temperature. To increase the pressure, work is supplied to this device from an external source, through a rotating shaft. The state of the refrigerant is changed through the compression process, from a low-pressure saturated vapor to a high-pressure superheated vapor.

(1) Design variables

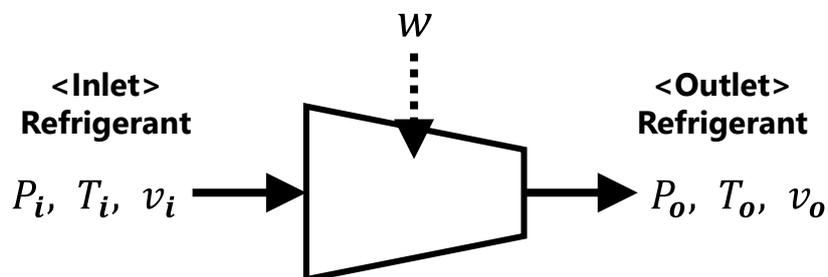


Figure B-1 Design variables of a compressor

As shown in Figure B-1, the operating conditions are defined as the design variables, including the pressure (\mathbf{P} , Pa), temperature (\mathbf{T} , K), specific volume (\mathbf{v} , m³/g) at the inlet and outlet of the compressor, and the specific work provided to the compressor (\mathbf{w} , J/g). The operating conditions of the inlet and outlet are classified by the subscript ‘ \mathbf{i} ’ and ‘ \mathbf{o} ’.

(2) Equality constraints

The first law of thermodynamics (energy conservation)

For an open system at a steady state, the rates of energy transfer by heat and work are constant with time, and the time rate of change for the energy contained within the control volume at any time is equal to zero, so the energy balance, or first law, for the open system is as follows (Cengel, 2007):

$$\sum_{out} \dot{m} \left(h + \frac{V^2}{2} + gz \right) - \sum_{in} \dot{m} \left(h + \frac{V^2}{2} + gz \right) = \dot{Q} + \dot{W}_S, \quad (\text{B-1})$$

where m is the mass flow rate of the refrigerant, h is the specific enthalpy of the refrigerant, V is the speed of the refrigerant in the stream, g is the gravitational acceleration, z is the height of the refrigerant above the ground, Q is the rate of heat transfer between the equipment and the surrounding environment, and W_S is the power provided from the surrounding environment to the equipment.

When the fluid experiences negligible changes in its kinetic and potential energy, the energy balance equation is reduced as follows:

$$\sum_{out} \dot{m} \cdot h - \sum_{in} \dot{m} \cdot h = \dot{Q} + \dot{W}_S. \quad (\text{B-2})$$

Moreover, there is only one flow stream entering and one flow stream leaving the

compressor.

$$\dot{m} \cdot h_o - \dot{m} \cdot h_i = \dot{Q} + \dot{W}_S \quad (\text{B-3})$$

As there is insufficient time to transfer much heat from the refrigerant, \dot{Q} becomes zero. The power provided to the compressor is represented as $\dot{W}_S = \dot{m} \cdot w$, where w is the specific work provided to the compressor. Substituting these and transposing $\dot{m} \cdot h_i$ to the righthand side of the equation, the equation is reformulated as follows:

$$\dot{m} \cdot h_i + \dot{m} \cdot w = \dot{m} \cdot h_o. \quad (\text{B-4})$$

This equation indicates that the energy of the refrigerant at the inlet of the compressor ($\dot{m} \cdot h_i$) was increased by the power provided to the compressor ($\dot{m} \cdot w$). The specific enthalpy (h) is the function of the pressure, specific volume, and temperature.

The second law of thermodynamics

For an open system at a steady state, the rates of energy transfer by heat and work are constant with time, and at a steady state, the time rate of change of the entropy contained within the control volume at any time is equal to zero. Thus, the entropy balance, or second law, for open systems is as follows (Moran & Shapiro, 2008):

$$\sum_{in} \dot{m} \cdot s - \sum_{out} \dot{m} \cdot s + \sum_j \frac{\dot{Q}_j}{T_j} + \dot{S}_{gen} = 0, \quad (\text{B-5})$$

where S is the specific entropy, Q_j is the heat transfer between the surrounding environments at temperature T_j , and S_{gen} is the entropy generated by the irreversible process. For the compressors, the heat transfer is usually negligible ($\dot{Q}_j = 0$) because the

compressors are typically well insulated and because there is insufficient time to transfer much heat from the refrigerant. Thus, the entropy balance for these single-stream steady-flow devices is reduced as follows:

$$\dot{m} \cdot s_i - \dot{m} \cdot s_o + \dot{S}_{gen} = 0. \quad (\text{B-6})$$

As the compression process is assumed to be adiabatic and reversible, an isentropic process can serve as an appropriate model for the process of the compressor (Cengel, 2008), and an isentropic process indicates that the quality of the energy (i.e., entropy) is not changed during the compression. Therefore, the entropy generated by the irreversible process becomes zero ($\dot{S}_{gen} = 0$).

$$\dot{m} \cdot s_i = \dot{m} \cdot s_o \quad (\text{B-7})$$

Equation of state

The equation of state is an equation representing the relationship between the pressure (P), specific volume (v), and temperature (T) of a substance. One form of energy possession in a system is intermolecular potential energy, an energy associated with the forces between molecules. At very low densities, the average distance between molecules is so large that there is intermolecular independence, a situation referred to as an *ideal gas*. Under this approximation, it has been experimentally observed that to a close approximation, a very-low-density gas behaves according to the ideal-gas equation of state, as follows:

$$Pv = RT, \quad (\text{B-8})$$

where R is the universal gas constant, whose value, for any gas, is

$$R = 8.314 \frac{N \cdot m}{mol \cdot K} = 8.314 \frac{J}{mol \cdot K}, \quad (B-9)$$

and where T is the absolute temperature in kelvins (K) (Borgnakke & Sonntag, 2009).

To improve the equation of state for liquids and vapors, the equation of state for an ideal gas is modified via experimentation and experience. As an example, consider the SRK equation, which is a type of cubic equation of state, as follows:

$$v = \frac{RT}{P} + b - \frac{a(T)}{P} \frac{v-b}{(v-\varepsilon b)(v-\sigma b)}, \quad (B-10)$$

where

$$b = \Omega \frac{RT_c}{P_c},$$

$\Omega = 0.08664$ for SRK equation,

T_c : critical temperature of the refrigerant ($T_{c,Ammonia} = 405.7$ K),

P_c : critical pressure of the refrigerant ($P_{c,Ammonia} = 112.80$ bar),

$$a(T) = \psi \frac{\alpha(T_r, \omega) R^2 T_c^2}{P_c},$$

$\Psi = 0.42748$ for SRK equation,

$$\alpha(T_r, \omega) = [1 + (0.480 + 1.574\omega - 0.176\omega^2)(1 - (T_r)^{0.5})]^2,$$

$T_r = \frac{T}{T_c}$, ω : acentric factor ($\omega_{Ammonia} = 0.253$), and

$\varepsilon = 0, \sigma = 1$ for SRK equation.

When the SRK equation of state is applied to the inlet and outlet of the compressor, the following equations are obtained:

$$v_i = \frac{RT_i}{P_i} + b - \frac{a(T_i)}{P_i} \frac{v_i - b}{(v_i - \varepsilon b)(v_i - \sigma b)}, \text{ and} \quad (\text{B-11})$$

$$v_o = \frac{RT_o}{P_o} + b - \frac{a(T_o)}{P_o} \frac{v_o - b}{(v_o - \varepsilon b)(v_o - \sigma b)}. \quad (\text{B-12})$$

Isentropic efficiency of a compressor

In the compression process, if the fluid in the compressor compresses reversibly and adiabatically, the process is isentropic, as shown in equation (B-7). This equation fixes the final state of the fluid and determines the specific enthalpy of the compressor's outlet. According to the energy balance, or the first law, for open systems and single-stream steady-flow devices, the shaft work for this special case is as follows (Smith, 2005):

$$h_i(P_i, v_i, T_i) + w_s = h_o(P_o, v_s, T_s), \quad (\text{B-13})$$

where w_s is the minimum shaft work required for the compression of a gas from a given initial state (T_i, v_i, P_i) to a given discharge pressure (P_o); and T_s and v_s are calculated by solving the following equations when the values of P_i, v_i, T_i , and P_o are given:

$$s_i(P_i, v_i, T_i) = s_o(P_o, v_s, T_s), \text{ and} \quad (\text{B-14})$$

$$v_s = \frac{RT_s}{P_o} + b - \frac{a(T_s)}{P_o} \frac{v_s - b}{(v_s - \varepsilon b)(v_s - \sigma b)}. \quad (\text{B-15})$$

In the actual liquefaction cycle, the compression process is irreversible. Considering the irreversibility of the compressor, the efficiency of a compressor is defined as the ratio of the work input required to raise the pressure of a refrigerant to a specified value in an

isentropic manner to the actual work input. It is called “isentropic efficiency of a compressor”:

$$\eta = \frac{\text{Isentropic compressor work}}{\text{Actual compressor work}} = \frac{w_s}{w_a}, \quad (\text{B-16})$$

where η is the isentropic efficiency of a compressor, and the value of η is usually given between 0.8 and 0.9 for a well-designed compressor.

The actual compressor work is calculated using equation (B-4).

(3) Inequality constraints

Condition of the inlet state for the compressor

If the liquid flows into the compressor, the compressor will be damaged (Kim et al., 2010). To avoid damage to the compressor, the pressure of the refrigerant at the inlet streams of the compressor is decreased to below the pressure of the dew point at the given temperature. The dew point is the point at which the last drops of liquid (dew) disappear (Smith et al., 2005). Therefore, if the pressure of the refrigerant is lower than that of the dew point at the given temperature, the state of the refrigerant is vapor. By applying this to compressor, the following inequality constraint can be obtained:

$$P_i \leq P_{Dew,i}. \quad (\text{B-17})$$

B.2. Condenser

The hot refrigerant passes through the condenser, which is an array of thin tubes that transfers heat from the refrigerant to the atmosphere. As it cools, the state of the refrigerant is changed from a high-pressure superheated vapor to a high-pressure saturated liquid.

(1) Design variables

As shown in Figure B-2, the operating conditions are defined as the design variables, including the pressure (P , Pa), temperature (T , K), specific volume (v , m³/g) at the inlet and outlet of the condenser, and the specific heat transfer from the refrigerant to the atmosphere (q_H , J/g).

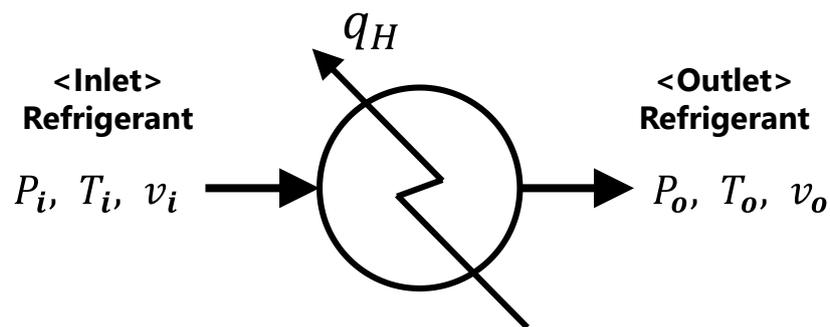


Figure B-2 Design variables of a condenser

(2) Equality constraints

The first law of thermodynamics (energy conservation)

For the condenser, the energy conservation for single-stream steady-flow devices represented in equation (B-3) is applied as follows:

$$\dot{m} \cdot h_o - \dot{m} \cdot h_i = \dot{Q} + \dot{W}_s. \quad (\text{B-18})$$

The condenser is a device where one moving fluid stream exchanges heat with the atmosphere without mixing. Thus, the condenser typically involves no work interaction ($\dot{W}_s = 0$). The heat transfer from the refrigerant to the atmosphere is represented as $\dot{Q} = -\dot{m} \cdot q_H$, where q_H is the specific heat transfer from the refrigerant to the atmosphere. Substituting these considerations, the equation is reformulated as follows:

$$\dot{m} \cdot h_o - \dot{m} \cdot h_i = -\dot{m} \cdot q_H. \quad (\text{B-19})$$

By transposing $\dot{m} \cdot h_o$ to the righthand side of the equation, the equation is represented as follows:

$$\dot{m} \cdot h_i = \dot{m} \cdot q_H + \dot{m} \cdot h_o. \quad (\text{B-20})$$

This equation indicates that the energy of the refrigerant at the inlet of the condenser ($\dot{m} \cdot h_i$) is equal to the summation of the heat transfer from the refrigerant to the atmosphere and the energy of the refrigerant at the outlet of the condenser ($\dot{m} \cdot h_o$).

Isobaric process

It is assumed that the pressure drop in the condenser is zero. Considering this condition, the equality constraint is formulated as follows:

$$P_i = P_o. \quad (\text{B-21})$$

Equation of state

As with the compressor, when the SRK equation of state is applied to the inlet and outlet of the condenser, the following equations are obtained:

$$v_i = \frac{RT_i}{P_i} + b - \frac{a(T_i)}{P_i} \frac{v_i - b}{(v_i - \varepsilon b)(v_i - \sigma b)}, \text{ and} \quad (\text{B-22})$$

$$v_o = \frac{RT_o}{P_o} + b - \frac{a(T_o)}{P_o} \frac{v_o - b}{(v_o - \varepsilon b)(v_o - \sigma b)}. \quad (\text{B-23})$$

Saturated pressure and temperature

As the refrigerant is in the saturated liquid phase at the outlet of the condenser, the pressure at the outlet is equal to the saturated pressure at the temperature of the outlet:

$$P_o = P_{sat}(T_o), \quad (\text{B-24})$$

where P_{sat} is the saturated pressure in bar at temperature T_o in kelvins.

The Antoine equation represents the relationship between the saturation pressure and the temperature:

$$P_{sat}(T_o) = 10^{A - \frac{B}{T_o + C - 273.15}}, \quad (\text{B-25})$$

where A , B , and C are the constant values, depending on the substance. For example, the value of A for ammonia is 4.48540, that of B is 926.1320, and that of C is 240.17.

As a result, the equality constraint is obtained as follows:

$$P_o = 10^{A - \frac{B}{T_o + C - 273.15}}. \quad (\text{B-26})$$

(3) Inequality constraints

Outlet temperature of the condenser

With an infinite transfer area for the condenser, the temperature at the condenser outlet is equal to the ambient temperature of 25°C. The increase in q_H is too small, however, to increase the transfer area of the condenser infinitely. Considering this practical aspect, the minimum value of the difference between the ambient temperature and the outlet temperature of the condenser is assumed as follows:

$$T_o > T_{amb} + \Delta T_{min}, \quad (\text{B-27})$$

where

T_{amb} : ambient temperature; and

ΔT_{min} : minimum value of the difference between the ambient temperature and the outlet temperature.

B.3. Expansion Valve

Expansion valves are flow-restricting devices that create turbulence in the refrigerant and cause a significant pressure drop in the fluid. When the refrigerant's pressure decreases, the refrigerant's temperature also tends to decrease. The state of the refrigerant is changed from a high-pressure saturated liquid to a low-pressure mixture of liquid and vapor.

(1) Design variables

As shown in Figure B-3, the operating conditions are defined as the design variables, including the pressure (P , Pa), temperature (T , K), and specific volume (v , m³/g) at the inlet and outlet of the expansion valve.

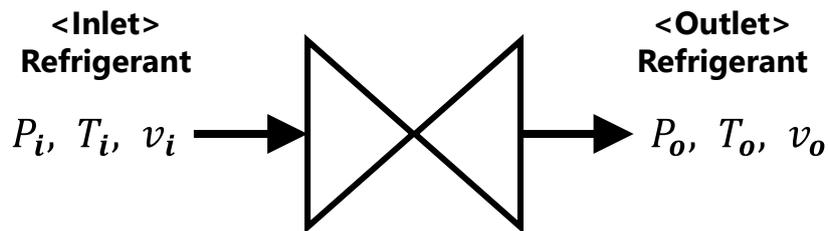


Figure B-3 Design variables of an expansion valve

(2) Equality constraints

The first law of thermodynamics (energy conservation)

For the expansion valve, the energy conservation for the single-stream steady-flow devices represented in equation (B-3) is applied as follows:

$$\dot{m} \cdot h_o - \dot{m} \cdot h_i = \dot{Q} + \dot{W}_S. \quad (\text{B-28})$$

As there is insufficient time to transfer much heat from the atmosphere to the refrigerant

in the expansion valve, the expansion process is assumed to be an adiabatic process ($\dot{Q} = 0$), and no work is done to decrease the pressure ($\dot{W}_s = 0$). Therefore, the energy values at the inlet and outlet “enthalpy” of the expansion valve are as follows:

$$\dot{m} \cdot h_o - \dot{m} \cdot h_i = 0, \quad (\text{B-29})$$

$$\dot{m} \cdot h_i = \dot{m} \cdot h_o, \text{ and} \quad (\text{B-30})$$

$$\dot{m} \cdot h_i(P_i, v_i, T_i) = \dot{m} \cdot h_o(P_o, v_o, T_o). \quad (\text{B-31})$$

At this point, the state at the outlet of the expansion valve is a mixture of liquid and vapor. To calculate the specific enthalpy for the mixture of liquid and vapor, the vapor fraction was introduced. The vapor fraction (v_f) is the ratio of the mass of the vapor to the total mass of the mixture of saturated liquid and saturated vapor. Considering the vapor fraction, equation (B-30) is reformulated as follows:

$$\dot{m} h_i(P_i, v_i, T_i) = \dot{m} [(1 - v_f) \cdot h_{o,l}(P_o, v_{o,l}, T_o) + v_f \cdot h_{o,v}(P_o, v_{o,v}, T_o)]. \quad (\text{B-32})$$

Saturated pressure and temperature

As the refrigerant is in the liquid and vapor phase at the outlet of the expansion valve, the pressure at the outlet is equal to the saturated pressure at the temperature of the outlet. As with the condenser, the following equality constraint is formulated:

$$P_o = 10^{A - \frac{B}{T_o + C - 273.15}}. \quad (\text{B-33})$$

Equation of state

As with the compressor, applying the SRK equation of state to the inlet and outlet of the expansion valve yields the following equations:

$$v_i = \frac{RT_i}{P_i} + b - \frac{a(T_i)}{P_i} \frac{v_i - b}{(v_i - \varepsilon b)(v_i - \sigma b)}, \text{ and} \quad (\text{B-34})$$

$$v_o = \frac{RT_o}{P_o} + b - \frac{a(T_o)}{P_o} \frac{v_o - b}{(v_o - \varepsilon b)(v_o - \sigma b)}. \quad (\text{B-35})$$

At this point, the state at the outlet of the expansion valve is a mixture of liquid and vapor. To calculate the specific volume for the mixture of liquid and vapor, the vapor fraction is also introduced, as with the specific enthalpy.

Considering the vapor fraction, equation (B-35) is reformulated as follows:

$$v_o = (1 - v_f) \cdot v_{o,l} + v_f \cdot v_{o,v}, \quad (\text{B-36})$$

Where

$$v_{o,l} = \frac{RT_o}{P_o} + b - \frac{a(T_o)}{P_o} \frac{v_{o,l} - b}{(v_{o,l} - \varepsilon b)(v_{o,l} - \sigma b)}, \text{ and} \quad (\text{B-37})$$

$$v_{o,v} = \frac{RT_o}{P_o} + b - \frac{a(T_o)}{P_o} \frac{v_{o,v} - b}{(v_{o,v} - \varepsilon b)(v_{o,v} - \sigma b)}. \quad (\text{B-38})$$

The three roots for the SRK equation of state indicate the specific volume at the given temperature and pressure. It is considered that the smallest root is for the liquid phase and the largest root is for the vapor phase. The third root has no physical meaning.

B.4. Heat Exchanger

Heat exchangers are devices where several moving fluid streams exchange heat without mixing. Thus, heat exchangers typically involve no work interaction. Moreover, heat exchangers are used for heat transfer between the two fluids within the device, and the outer shell is usually well insulated to prevent any heat loss to the surrounding medium.

(1) Design variables

As shown in Figure B-4, the operating conditions are defined as the design variables, including the pressure (P , Pa), temperature (T , K), specific volume (v , m³/g), and mass flow rate (\dot{m} , mol/s) at the inlet and outlet of the heat exchanger.

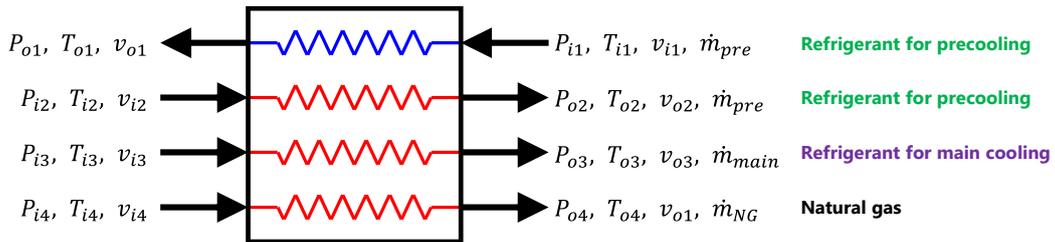


Figure B-4 Design variables of a heat exchanger

(2) Equality constraints

The first law of thermodynamics (energy conservation)

For the heat exchanger, the energy conservation for multi-stream steady-flow devices, represented in equation (B-2) is applied as follows:

$$\left[\dot{m}_{pre} \cdot h_{o1} + \dot{m}_{pre} \cdot h_{o2} + \dot{m}_{main} \cdot h_{o3} + \dot{m}_{NG} \cdot h_{o4} \right] - \left[\dot{m}_{pre} \cdot h_{i1} + \dot{m}_{pre} \cdot h_{i2} + \dot{m}_{main} \cdot h_{i3} + \dot{m}_{NG} \cdot h_{i4} \right] = \dot{Q} + \dot{W}_s \quad (\text{B-39})$$

As heat exchangers are devices where several moving fluid streams exchange heat without mixing, they typically involve no work interaction ($\dot{W}_s = 0$). Moreover, heat exchangers are used for heat transfer between the two fluids within the device, and the outer shell is usually well insulated to prevent any heat loss to the surrounding medium. When the entire heat exchanger is selected as the control volume, \dot{Q} becomes zero because the boundary for this case lies just beneath the insulation, and because little or no heat crosses the boundary (Cengel, 2008). Substituting these considerations, the equation is reformulated as follows:

$$\begin{aligned} \dot{m}_{pre} \cdot h_{o1} + \dot{m}_{pre} \cdot h_{o2} + \dot{m}_{main} \cdot h_{o3} + \dot{m}_{NG} \cdot h_{o4} = \dot{m}_{pre} \cdot h_{i1} + \\ \dot{m}_{pre} \cdot h_{i2} + \dot{m}_{main} \cdot h_{i3} + \dot{m}_{NG} \cdot h_{i4}. \end{aligned} \quad (B-40)$$

This equation indicates that the total energy of the outlet streams of the heat exchanger is equal to the total energy of the inlet streams of the heat exchanger.

Isobaric process

It is assumed that the pressure drop in the heat exchanger is zero. Considering this condition, the equality constraints are formulated as follows:

$$P_{i1} = P_{o1}, P_{i2} = P_{o2}, P_{i3} = P_{o3}, \text{ and } P_{i4} = P_{o4}. \quad (B-41)$$

Conservation condition of the output temperature

For the heat exchanger, it is assumed that the temperatures of the outlet streams are the same, for the efficiency of the heat exchanger. Therefore, it is a type of economic constraint while the above constraints are thermodynamic constraints. By applying this to the heat exchangers, the following equality constraints can be obtained:

$$T_{o2} = T_{o3} = T_{o4}. \quad (\text{B-42})$$

Equation of state

As with the other equipment, when the SRK equation of state is applied to the inlet and outlet of the heat exchanger, the following equations are obtained:

$$v_{in} = \frac{RT_{in}}{P_{in}} + b - \frac{a(T_{in})}{P_{in}} \frac{v_{in}-b}{(v_{in}-\varepsilon b)(v_{in}-\sigma b)}, \text{ and} \quad (\text{B-43})$$

$$v_{on} = \frac{RT_{on}}{P_{on}} + b - \frac{a(T_{on})}{P_{on}} \frac{v_{on}-b}{(v_{on}-\varepsilon b)(v_{on}-\sigma b)}. \quad (\text{B-44})$$

where $n = 1, 2, 3, 4$.

(3) Inequality constraints

Minimum temperature difference in the heat exchanger

The hot stream in the heat exchanger transfers heat to the cold stream. The minimum temperature difference in the heat exchanger is the minimum temperature difference between the hot stream and the cold stream. A small minimum temperature difference value means that much energy is recovered, but this requires a large heat exchanger. On the other hand, a larger minimum temperature difference value requires less area for the heat exchanger, but the outlet temperature of the hot stream is higher, and less energy is recovered (Jensen, 2008). The minimum temperature difference is assumed to be 3.0 [K] (Venkatarathnam, 2008). Applying this to the heat exchanger, where the stream 1 is the cold stream and the others form the hot stream, will give rise to the following constraints:

$$T_{i2} - T_{o1} \geq 3.0, \text{ and } T_{o2} - T_{i1} \geq 3.0 \quad (\text{B-45})$$

B.5. Tee

The tee is the equipment used to separate an inlet stream of a refrigerant into two outlet streams. It is assumed that there is no pressure drop in the refrigerant through this equipment, and that there is no heat transfer between the refrigerant and the surroundings.

(1) Design variables

As shown in Figure B-5, the operating conditions are defined as the design variables, including the pressure (P , Pa), temperature (T , K), specific volume (v , m³/g) at the inlet and outlet of the tee, and flow rate ratio between inlet and outlet 1 (c).

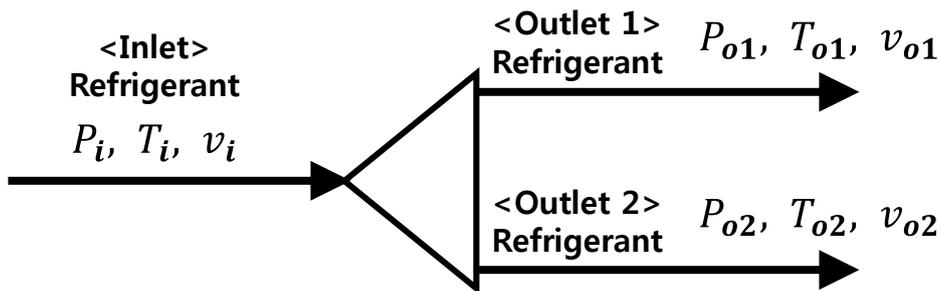


Figure B-5 Design variables of a tee

(2) Equality constraints

The first law of thermodynamics (energy conservation)

For the tee, the energy conservation, represented in equation (B-2), is applied as follows:

$$[c \cdot h_{o1} + (1 - c) \cdot h_{o2}] - h_i = \dot{Q} + \dot{W}_S, \quad (\text{B-46})$$

where c is the flow rate ratio between inlet and outlet 1.

It is assumed that there is no heat transfer between the refrigerant and the surroundings, and that the tee is usually well insulated ($\dot{Q} = 0$). Moreover, the tee usually does not involve any kind of work ($\dot{W}_s = 0$). Substituting these considerations, equation (B-46) is reformulated as follows:

$$[c \cdot h_{o1} + (1 - c) \cdot h_{o2}] = h_i. \quad (\text{B-47})$$

This equation indicates that the total energy of the tee's outlet streams is equal to the total energy of the tee's inlet streams.

Isobaric process

It is assumed that the pressure drop in the tee is zero. Considering this condition, the equality constraints are formulated as follows:

$$P_i = P_{o1} = P_{o2}. \quad (\text{B-48})$$

Isothermal condition

It is assumed that there is no heat transfer in the tees. Thus, the temperatures of the outlet streams are the same. Applying this to the tee will give rise to the following equality constraint:

$$T_{o1} = T_{o2}. \quad (\text{B-49})$$

Equation of state

As with the other equipment, applying the SRK equation of state to the inlet and outlet of the tee will yield the following equations:

$$v_i = \frac{RT_i}{P_i} + b - \frac{a(T_i)}{P_i} \frac{v_i - b}{(v_i - \varepsilon b)(v_i - \sigma b)}, \quad (\text{B-50})$$

$$v_{o1} = \frac{RT_{o1}}{P_{o1}} + b - \frac{a(T_{o1})}{P_{o1}} \frac{v_{o1} - b}{(v_{o1} - \varepsilon b)(v_{o1} - \sigma b)}, \text{ and} \quad (\text{B-51})$$

$$v_{o2} = \frac{RT_{o2}}{P_{o2}} + b - \frac{a(T_{o2})}{P_{o2}} \frac{v_{o2} - b}{(v_{o2} - \varepsilon b)(v_{o2} - \sigma b)}. \quad (\text{B-52})$$

B.6. Phase Separator

A phase separator is used to separate a vapor-liquid mixture fluid into vapor and liquid. As the boiling point of the heavy hydrocarbon is higher than that of the light hydrocarbon, the heavy hydrocarbon is liquefied more easily than the light hydrocarbon. Thus, one component of the liquid at the outlet stream of the phase separator is usually heavy hydrocarbon. Outlet stream 1, as shown in Figure B-6, is the refrigerant in the vapor phase of inlet stream, and outlet stream 2 is the refrigerant in the liquid phase of inlet stream. It is assumed that there is no pressure drop in the refrigerant through the phase separator, and that there is no heat transfer between the refrigerant and its surroundings.

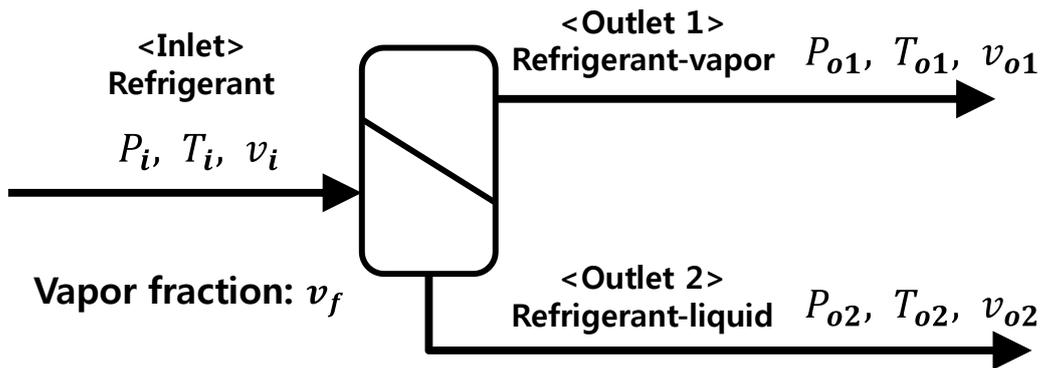


Figure B-6 Design variables of a phase separator

(1) Design variables

As shown in Figure B-6, the operating conditions are defined as the design variables, including the pressure (P , Pa), temperature (T , K), specific volume (v , m^3/g) at the inlet and outlet of the phase separator, and vapor fraction at the inlet stream (v_f).

(2) Equality constraints

The first law of thermodynamics (energy conservation)

For the phase separator, the energy conservation, represented in equation (B-2), is applied as follows:

$$[v_f \cdot h_{o1} + (1 - v_f) \cdot h_{o2}] - h_i = \dot{Q} + \dot{W}_s. \quad (\text{B-53})$$

It is assumed that there is no heat transfer between the refrigerant and its surroundings as the phase separator is usually well insulated ($\dot{Q} = 0$). Moreover, the phase separator usually does not involve any kind of work ($\dot{W}_s = 0$). Substituting these considerations, equation (4-104) is reformulated as follows:

$$h_i = [v_f \cdot h_{o1} + (1 - v_f) \cdot h_{o2}]. \quad (\text{B-54})$$

This equation indicates that the total energy of the phase separator's outlet streams is equal to the total energy of the phase separator's inlet streams, as with the tee.

Isobaric process

It is assumed that the pressure drop in the phase separator is zero. Considering this condition, the equality constraints are formulated as follows:

$$P_i = P_{o1} = P_{o2}. \quad (\text{B-55})$$

Isothermal condition

It is assumed that there is no heat transfer in the phase separator. Thus, the temperatures

of the outlet streams are the same. Applying this to the phase separator will give rise to the following equality constraints:

$$T_i = T_{o1} = T_{o2}. \quad (\text{B-56})$$

Equation of state

As with the other equipment, applying the SRK equation of state to the inlet and outlet of the phase separator will give rise to the following equations:

$$v_i = \frac{RT_i}{P_i} + b - \frac{a(T_i)}{P_i} \frac{v_i - b}{(v_i - \varepsilon b)(v_i - \sigma b)}, \quad (\text{B-57})$$

$$v_{o1} = \frac{RT_{o1}}{P_{o1}} + b - \frac{a(T_{o1})}{P_{o1}} \frac{v_{o1} - b}{(v_{o1} - \varepsilon b)(v_{o1} - \sigma b)}, \text{ and} \quad (\text{B-58})$$

$$v_{o2} = \frac{RT_{o2}}{P_{o2}} + b - \frac{a(T_{o2})}{P_{o2}} \frac{v_{o2} - b}{(v_{o2} - \varepsilon b)(v_{o2} - \sigma b)}. \quad (\text{B-59})$$

B.7. Common Header

A common header is used to combine the two refrigerant streams separated by the tee or the phase separator, as shown in Figure B-7. For this equipment, there is no heat transfer between the refrigerant and its surroundings.

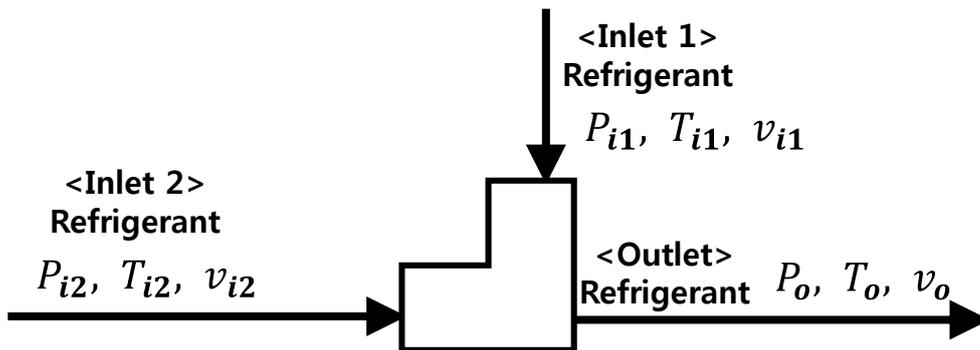


Figure B-7 Common header

(1) Design variables

As shown in Figure B-7, the operating conditions are defined as the design variables, including the pressure (P , Pa), temperature (T , K), and specific volume (v , m^3/g) at the inlet and outlet of the common header.

(2) Equality constraints

The first law of thermodynamics (energy conservation)

For the common header, the energy conservation represented in equation (4-2) is applied as follows:

$$h_o - [c \cdot h_{i1} + (1 - c) \cdot h_{i2}] = \dot{Q} + \dot{W}_s. \quad (\text{B-60})$$

It is assumed that there is no heat transfer between the refrigerant and its surroundings as the common header is usually well insulated ($\dot{Q} = 0$). Moreover, the common header usually does not involve any kind of work ($\dot{W}_s = 0$). Substituting these considerations, equation (4-123) is reformulated as follows:

$$[c \cdot h_{i1} + (1 - c) \cdot h_{i2}] = h_o. \quad (\text{B-61})$$

This equation indicates that the total energy of the common header's outlet stream is equal to the total energy of the common header's inlet streams.

Isobaric process

To prevent a backflow in the common headers, it is assumed that the pressures of the inlet streams are the same. Considering this condition, the equality constraints are formulated as follows:

$$P_{i1} = P_{i2} = P_o. \quad (\text{B-62})$$

Equation of state

As with the other equipment, applying the SRK equation of state to the inlet and outlet of the common header will give rise to the following equations:

$$v_{i1} = \frac{RT_{i1}}{P_{i1}} + b - \frac{a(T_{i1})}{P_{i1}} \frac{v_{i1}-b}{(v_{i1}-\varepsilon b)(v_{i1}-\sigma b)}, \quad (\text{B-63})$$

$$v_{i2} = \frac{RT_{i2}}{P_{i2}} + b - \frac{a(T_{i2})}{P_{i2}} \frac{v_{i2}-b}{(v_{i2}-\varepsilon b)(v_{i2}-\sigma b)}, \text{ and} \quad (\text{B-64})$$

$$v_o = \frac{RT_o}{P_o} + b - \frac{a(T_o)}{P_o} \frac{v_o-b}{(v_o-\varepsilon b)(v_o-\sigma b)}. \quad (\text{B-65})$$

B.8. Calculation of the Specific Enthalpy for a Pure Substance

The specific enthalpy (h) is the summation of the specific internal energy of the flow (u) and the flow work (Pv), as follows:

$$h = u + Pv. \quad (\text{B-66})$$

The flow work is the energy needed to push a fluid into or out of a control volume, and it is equal to Pv . The enthalpy is useful for analyzing the energy conservation or energy balance, which is the first law of thermodynamics, of flow processes, as applied to the compressor, condenser, and expansion valve and evaporator, for the calculation of heat and work.

Many tables of thermodynamic properties do not give values for internal energy. To allow the calculation of enthalpy from the pressure, specific volume, and temperature, the following equation is derived using the definition ($h = u + Pv$) and the equation of state, and via experimentation.

$$h = h^{IG} + h^R, \quad (\text{B-67})$$

where

h^{IG} : ideal gas value of the specific enthalpy [J/mol], and

h^R : residual specific enthalpy [J/mol].

The residual specific enthalpy is the correction of the ideal gas state values to the real

gas values. The ideal gas value of the specific enthalpy is a function of the temperature, and it is calculated by the following equation, derived from the experiment:

$$h^{IG} = h^{IG}(T) = (a + bT + cT^2 + dT^3 + eT^4 + fT^5)M + h_{basis} \quad (\text{B-68})$$

where

a , b , c , d , e , and f : characteristic constants of the particular substance. If the substance is ammonia (e.g., $a = -1.8514$; $b = 1.9937$; $c = -5.3266 \times 10^{-4}$; $d = 2.0615 \times 10^{-6}$; $e = -1.3386 \times 10^{-9}$; and $f = 3.0533 \times 10^{-13}$);

T : temperature [K];

M : molar mass (g/mol) (e.g., the molar mass of ammonia is 17.031 [g/mol]); and

h_{basis} : specific enthalpy at the reference state [J/mol] (e.g., the specific enthalpy at the reference state of ammonia is -55757.7).

The residual specific enthalpy is a function of the pressure, specific volume, and temperature, and it is calculated using the following equation (Sandler, 1998):

$$h^R = h^R(P, v, T) = RT(Z - 1) + \frac{T\left(\frac{\partial a}{\partial T}\right) - a(T)}{b} \frac{1}{(\sigma - \varepsilon)} \ln \left[\frac{Z + \sigma \cdot \beta}{Z + \varepsilon \cdot \beta} \right], \quad (\text{B-69})$$

where

$$a(T) = \psi \frac{\alpha(T_r, \omega) R^2 T_c^2}{P_c};$$

$$T_r = \frac{T}{T_c};$$

$$Z = \frac{v}{v-b} - \frac{v}{R \cdot T} \cdot \frac{a(T)}{(v+\varepsilon \cdot b)(v+\sigma \cdot b)};$$

$$\beta = \frac{bP}{RT}; \text{ and}$$

$$b = \Omega \frac{RT_c}{P_c}.$$

The values of ω , critical pressure (P_c), and temperature (T_c) depend on the substance.

The values of parameters a , σ , ε , Ω , and Ψ depend on the type of cubic equation of state. For example, the values of the parameters for the SRK equation of state are given in Table B-1.

Table B-1 Values of the parameters for the SRK equation

Parameter	Values
A	$\alpha(T_r; \omega) = [1 + (0.480 + 1.574\omega - 0.176\omega^2)(1 - (T_r)^{0.5})]^2$
σ	1
ε	0
Ω	0.08664
Ψ	0.42748

To calculate $\frac{da}{dT}$, the central difference approximation is used, as follows:

$$\left(\frac{da}{dT}\right) = \frac{a(T+e)-a(T-e)}{2 \cdot e}, \quad (e = 10^{-6}). \quad (\text{B-70})$$

B.9. Calculation of the Specific Entropy for a Pure Substance

While the first law of thermodynamics is concerned with the quantity of energy and the transformation of energy from one form to another, with no regard to its quality, the second law is concerned with the quality of energy. Entropy is a property associated with the second law of thermodynamics, and represents the quality of energy. The second law of thermodynamics indicates that the actual processes involve a decreasing quality of energy, an increase of entropy.

Entropy can be viewed as a measure of molecular disorder, or molecular randomness. As a system becomes more disordered, the positions of the molecules become less predictable, and the entropy increases. Based on this viewpoint, vapor has higher entropy than the liquid from which it is formed. As particles acquire greater freedom to move about when they are vaporized, they are distributed throughout the entire container rather than being restricted to a small volume. Moreover, increasing the temperature of a substance increases its entropy. Raising the temperature increases the kinetic energy of the molecules, and this increases their freedom of motion. In a solid, the molecules vibrate with greater amplitude at higher temperatures. In a liquid or vapor, they move about more rapidly.

To allow the calculation of entropy from the pressure, specific volume, and temperature, the following equation is derived using the definition ($ds = dq/T$) and the equation of state, and via experimentation:

$$s = s^{IG} + s^R, \quad (\text{B-71})$$

where

s^{IG} : ideal gas value of the specific entropy [J/(mol·K)]; and

s^R : residual specific entropy [J/(mol·K)].

The residual specific entropy is the correction of the ideal gas state values to the real gas values, as with the residual specific enthalpy. The ideal gas value of the specific entropy is a function of the temperature, and it is calculated using the following equation, derived from experimentation:

$$s^{IG} = \left(g + b \cdot \ln(T) + 2 \cdot c \cdot T + \frac{3}{2} \cdot d \cdot T^2 + \frac{4}{3} \cdot e \cdot T^3 + \frac{5}{4} \cdot f \cdot T^4 \right) \cdot M, \quad (\text{B-72})$$

where

b , c , d , e , and f : characteristic constants of the particular substance (the values are the same as those used in the ideal gas value of the specific enthalpy);

g : entropy coefficient (i.e., entropy of ideal gas at $T=0$ K)=1.0 [the value of the entropy coefficient, g , is set at 1.0 (vs. 0.0), so that any division involving this coefficient will not return a divide by zero error].

Residual specific entropy is a function of the pressure, specific volume, and temperature, and it is calculated using the following equation (Sandler, 1998):

$$s^R = s^R(P, v, T) = R \ln(Z - \beta) - R \ln\left(\frac{P}{P_0}\right) + \frac{\left(\frac{\partial a}{\partial T}\right)}{b} \frac{1}{(\sigma - \varepsilon)} \ln \left[\frac{Z + \sigma \cdot \beta}{Z + \varepsilon \cdot \beta} \right], \quad (\text{B-73})$$

where

$$a(T) = \psi \frac{\alpha(T_r, \omega) R^2 T_c^2}{P_c};$$

$$T_r = \frac{T}{T_c};$$

$$Z = \frac{v}{v-b} - \frac{v}{R \cdot T} \cdot \frac{a(T)}{(v+\varepsilon \cdot b)(v+\sigma \cdot b)};$$

$$\beta = \frac{bP}{RT}; \text{ and}$$

$$b = \Omega \frac{RT_c}{P_c}.$$

The values of ω , critical pressure (P_c), and temperature (T_c) depend on the substance. The values of the parameters a , σ , ε , Ω , and Ψ depend on the type of cubic equation of state. For example, the values of the parameters for the SRK equation of state are given in Table B-1.

C. Methods of Solving Nonlinear Equations

(1) Bisection method

Once an interval containing a root is known, several classical procedures are available to refine it. These proceed with varying degrees of speed and certainty towards the answer. Unfortunately, the methods that are guaranteed to converge plod along slowly while those that rush to a solution can also, without warning, dash rapidly to infinity if measures are not taken to avoid such behavior.

The bisection method is a method that cannot fail. It should thus not be ignored as a method for serious problems. The idea is simple. Over a certain interval, the function is known to pass through zero because it changes its sign. Evaluate the function at the interval's midpoint and examine its sign. Use the midpoint to replace whichever limit has the same sign. After each iteration, the bounds containing the root decrease by a factor of two. If after n iterations the root is known to be within an interval of size ε_n , then after the next iteration it will be bracketed within an interval of size

$$\varepsilon_{n+1} = \varepsilon_n/2 \tag{C-1}$$

neither more nor less. Thus, the number of iterations required to achieve a given tolerance is known in advance in the following solution:

$$n = \log_2 \frac{\varepsilon_0}{\varepsilon}, \tag{C-2}$$

where ε_0 is the size of the initially bracketed interval and ε is the desired ending tolerance.

Bisection must succeed. If the interval happens to contain more than one root, bisection will find one of them. If the interval contains no roots and merely straddles a singularity, it will converge on the singularity (Press et al., 2007).

(2) Newton-Raphson method

Perhaps the most celebrated of all one-dimensional root-finding routines is Newton's method, also called *Newton-Raphson method*. This method requires the evaluation of both the function $f(x)$ and the derivative $f'(x)$, at an arbitrary point x . The Newton-Raphson formula consists of geometrically extending the tangent line at a current point x_i until it crosses zero, and then setting the next guess x_{i+1} to the abscissa of the zero crossing. Algebraically, the method was derived from the familiar Taylor series expansion of a function in the neighborhood of a point:

$$f(x + \delta) \approx f(x) + f'(x)\delta + \frac{f''(x)}{2}\delta^2 + \dots \quad (\text{C-3})$$

For small-enough values of δ for well-behaved functions, the terms beyond linear are unimportant; hence, $f(x + \delta) = 0$ implies

$$\delta = -\frac{f(x)}{f'(x)}. \quad (\text{C-4})$$

Due to this method's rate of convergence, it is a powerful method. Within a small distance ε of x , the function and its derivative are approximately

$$f(x + \varepsilon) = f(x) + \varepsilon f'(x) + \varepsilon^2 \frac{f''(x)}{2} + \dots, \quad (\text{C-5})$$

$$f'(x + \varepsilon) = f'(x) + \varepsilon f''(x) + \dots$$

By the Newton-Raphson formula,

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}, \quad (\text{C-6})$$

so that

$$\varepsilon_{i+1} = \varepsilon_i - \frac{f(x_i)}{f'(x_i)}. \quad (\text{C-7})$$

When a trial solution x_i differs from the true root by ε_i , equation (C-6) can be used to express $f(x_i)$, $f'(x_i)$ in equation (C-7) in terms of ε_i and the derivatives of the root itself. The result is a recurrence relation for the deviations of the trial solutions:

$$\varepsilon_{i+1} = -\varepsilon_i^2 \frac{f''(x_i)}{f'(x_i)}. \quad (\text{C-8})$$

Equation (C-8) indicates that the Newton-Raphson formula converges quadratically compared with equation (C-2). This very strong convergence property makes the Newton-Raphson method the method of choice for any function whose derivative can be evaluated efficiently and whose derivative is continuous and nonzero in the neighborhood of a root.

Far from a root, however, where the higher-order terms in the series are important, the Newton-Raphson formula can provide grossly inaccurate and meaningless corrections. For instance, the initial guess for the root may be so far from the true root as to let the search interval include a local maximum or minimum of the function. This can have significantly negative consequences. If iteration places a trial guess near such a local extremum so that the first derivative nearly vanishes, the Newton-Raphson method sends its solution off to limbo, with a very small hope of recovery (Press et al., 2007).

D. Comparison of Calculation Results for One-Dimensional Heat Equation Using the Lattice Boltzmann Method and Finite Difference Method

To know the difference between the LBM and the finite difference method (FDM), these two methods are applied to a simple example as shown in Figure D-1. The problem statement of Figure D-1 is as follows. The FDM and LBM was applied to solve this problem. The detailed descriptions are in following sections.

- Solve for the temperature distribution of a long, thin rod with a length of 10 cm and the following values: $k' = 0.49 \text{ cal}/(\text{s} \cdot \text{cm} \cdot ^\circ\text{C})$. At $t = 0$, the temperature of the rod is zero and the boundary conditions are fixed for all times at $T(0) = 100^\circ\text{C}$ and $T(10) = 50^\circ\text{C}$. Note that the rod is aluminum with $C = 0.2174 \text{ cal}/(\text{g} \cdot ^\circ\text{C})$ and $\rho = 2.7 \text{ g}/\text{cm}^3$. Therefore, $\alpha = \frac{0.49}{2.7 \cdot 0.2174} = 0.835 \text{ cm}^2/\text{s}$.

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}, \quad T = T(x, t)$$

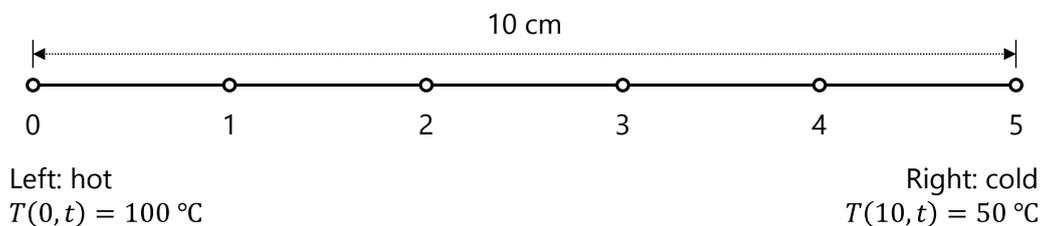


Figure D-1 One-dimensional heat equation for 1-D rod

D.1. Solving the Equation Using the Explicit Method of the FDM

The heat-conduction equation requires approximations for the second derivative in space and the first derivative in time. The former is represented in the same fashion as for the Laplace equation by a centered finite-divided difference:

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^k - 2T_i^k + T_{i-1}^k}{\Delta x^2}, \quad (\text{D-1})$$

where k is the number of iteration step and it means that $t = k\Delta t$, and i means an index of discretized spatial point. A forward finite-divided difference is used to approximate the time derivative

$$\frac{\partial T}{\partial t} = \frac{T_i^{k+1} - T_i^k}{\Delta t}. \quad (\text{D-2})$$

Subscribing these two equation into one-dimensional heat equation in the problem yields

$$\frac{T_i^{k+1} - T_i^k}{\Delta t} = \alpha \frac{T_{i+1}^k - 2T_i^k + T_{i-1}^k}{\Delta x^2}, \quad (\text{D-3})$$

which can be solved for

$$T_i^{k+1} = T_i^k + \lambda(T_{i+1}^k - 2T_i^k + T_{i-1}^k), \quad \lambda = \alpha \frac{\Delta t}{\Delta x^2}. \quad (\text{D-4})$$

This equation can be written for all the interior nodes on the rod. It then provides an explicit means to compute values at each node for a future time based on the present values at the node and its neighbors.

Figure D-2 shows the calculation result of this problem using the explicit method of the FDM with the following conditions: $\Delta x = 2 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$.

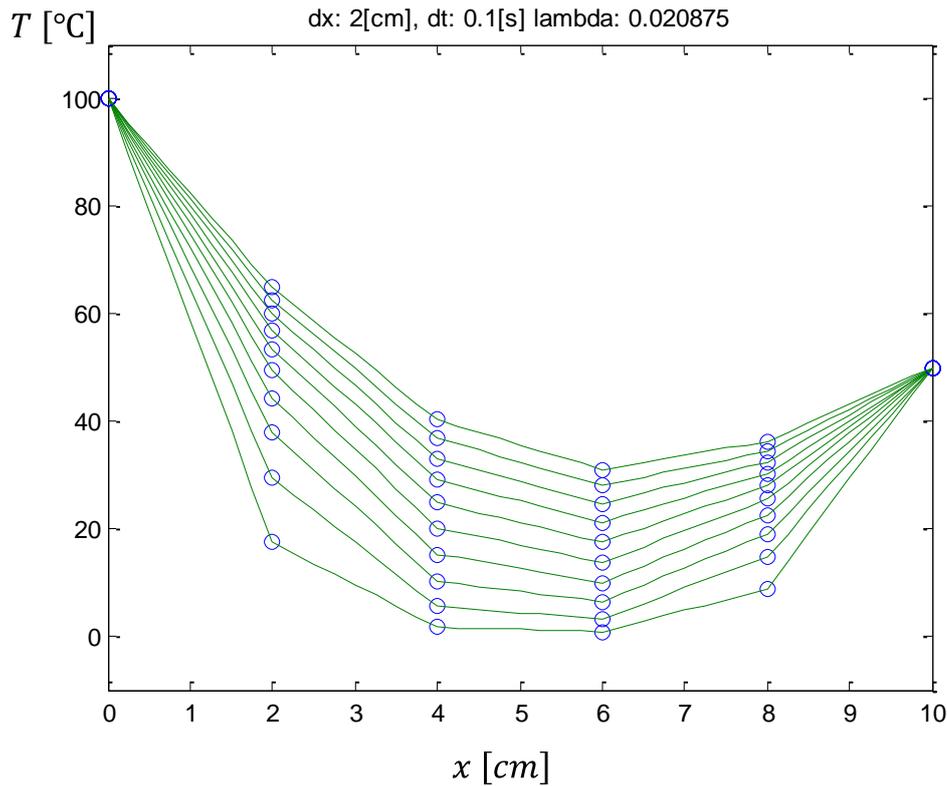


Figure D-2 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 2 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

Figure D-3, Figure D-4, and Figure D-5 show the change of the calculation results by increasing the spatial step size Δx . Stability means that errors at any stage of the computation are not amplified but are attenuated as the computation progresses. It can be shown (Carnahan et al., 1969) that the explicit method is both convergent and stable if $\lambda \leq \frac{1}{2}$ or $\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{k}$. Figure D-5 is an example of instability caused by violating the stability

condition $\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{k}$.

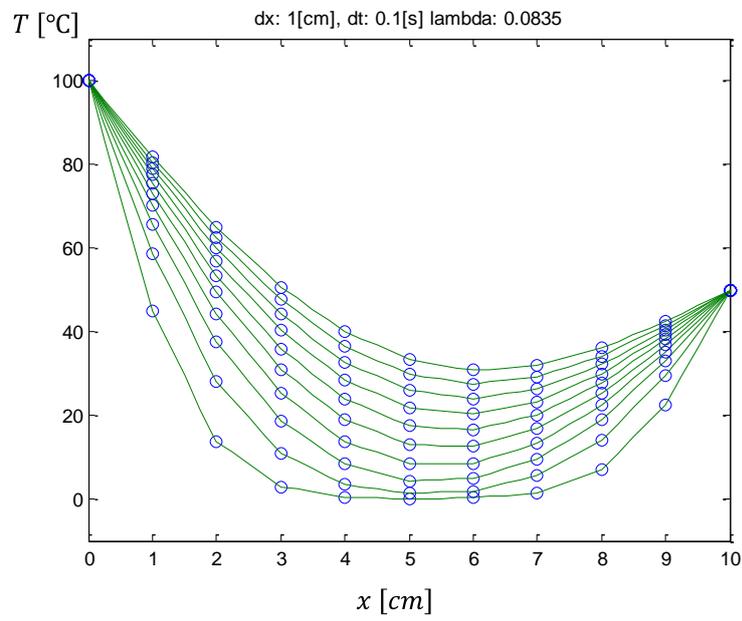


Figure D-3 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 1$ [cm], $\Delta t = 0.1$ [s], and $\alpha = 0.835$ [cm²/s])

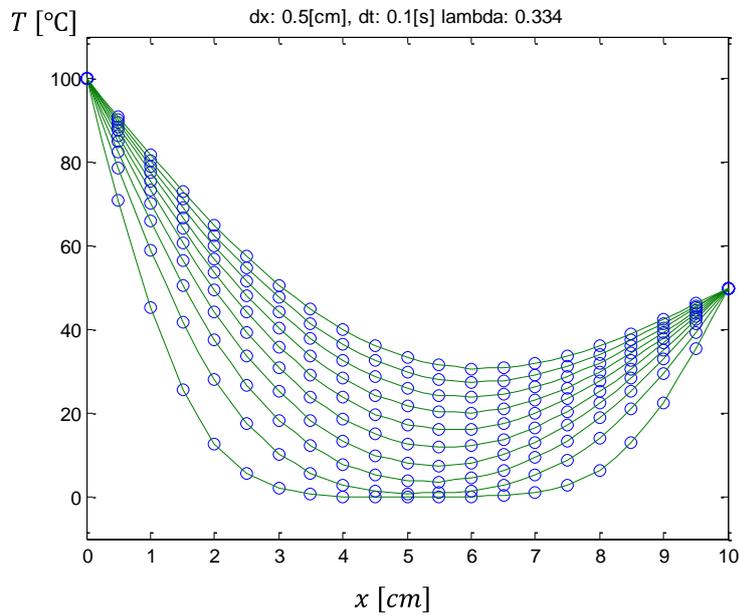


Figure D-4 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 0.5 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

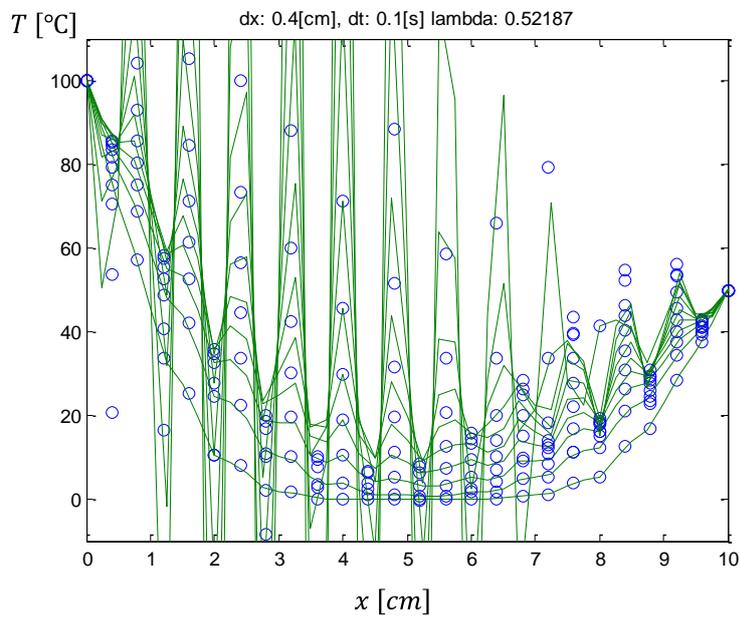


Figure D-5 Calculation result of one-dimensional heat equation using the explicit method of the FDM ($\Delta x = 0.4 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

D.2. Solving the Equation Using the Implicit Method of the FDM

As noted previously, explicit finite-difference formulations have problems related to stability. Implicit methods overcome both these difficulties at the expense of somewhat more complicated algorithms.

The fundamental difference between explicit and implicit approximations is depicted in Figure D-6. For the explicit form, we approximate the spatial derivative at time level k . In implicit methods, the spatial derivative is approximated at an advanced time level $k + 1$.

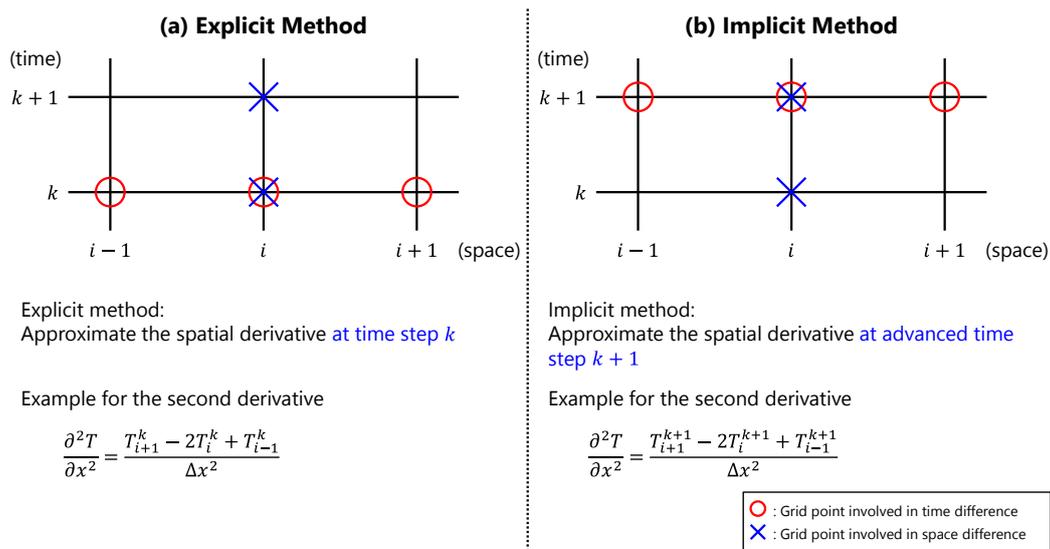


Figure D-6 Computational molecules demonstrating the fundamental differences between (a) explicit and (b) implicit methods

Figure D-7 shows the procedure of applying the implicit method to one-dimensional heat equation. The last equation in Figure D-7 applies to all but the first and the last interior nodes, which must be modified to reflect the boundary conditions.

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}, \quad T = T(x, t)$$

Left Hand Side

$$\frac{\partial T}{\partial t} = \frac{T_i^{k+1} - T_i^k}{\Delta t}$$

k : k^{th} step ($t = k\Delta t$)
 i : i^{th} point

Right Hand Side

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1}}{\Delta x^2}$$

$$\frac{T_i^{k+1} - T_i^k}{\Delta t} = \alpha \frac{T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1}}{\Delta x^2}$$

$$T_i^{k+1} = T_i^k + \alpha \frac{\Delta t}{\Delta x^2} (T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1})$$

$$T_i^{k+1} = T_i^k + \lambda (T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1}), \quad \lambda = \alpha \frac{\Delta t}{\Delta x^2}$$

$$-\lambda T_{i-1}^{k+1} + (1 + 2\lambda)T_i^{k+1} - \lambda T_{i+1}^{k+1} = T_i^k, \quad \lambda = \alpha \frac{\Delta t}{\Delta x^2}$$

Figure D-7 Applying an implicit method of the FDM to one-dimensional heat equation

Using the implicit method, the difference equation for all nodes except the first ($i = 1$) and last ($i = m$) interior nodes is as follows:

$$-\lambda T_{i-1}^{k+1} + (1 + 2\lambda)T_i^{k+1} - \lambda T_{i+1}^{k+1} = T_i^k, \quad \lambda = \alpha \frac{\Delta t}{\Delta x^2}. \quad (\text{D-5})$$

The boundary condition at the left end of the rod ($i = 0$) is given by

$$T_0^{k+1} = f_0(t^{k+1}), \quad (\text{D-6})$$

where $f_0(t^{k+1})$ is a function describing how the left boundary temperature changes with time. Thus, subscribing equation (D-6) to equation (D-5), the difference equation for the first interior node ($i = 1$):

$$(1 + 2\lambda)T_1^{k+1} - \lambda T_2^{k+1} = T_1^k + \lambda f_0(t^{k+1}). \quad (\text{D-7})$$

The boundary condition at the right end of the rod ($i = m + 1$) is given by

$$T_{m+1}^{k+1} = f_{m+1}(t^{k+1}), \quad (\text{D-8})$$

where $f_{m+1}(t^{k+1})$ is a function describing how the right boundary temperature changes with time. Thus, subscribing equation (D-8) to equation (D-5), the difference equation for the last interior node ($i = m + 1$):

$$-\lambda T_{m-1}^{k+1} + (1 + 2\lambda)T_m^{k+1} = T_m^k + \lambda f_{m+1}(t^{k+1}). \quad (\text{D-9})$$

When the equations (D-5), (D-7), and (D-9) are written for all the interior nodes, the resulting set of m linear algebraic equations has m unknowns. In addition, the method has the added bonus that the system is tridiagonal.

$$\begin{bmatrix} 1 + 2\lambda & -\lambda & 0 & 0 & \cdots & 0 \\ -\lambda & 1 + 2\lambda & -\lambda & 0 & \cdots & 0 \\ 0 & -\lambda & 1 + 2\lambda & -\lambda & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & -\lambda & 1 + 2\lambda & -\lambda \\ 0 & \cdots & 0 & 0 & -\lambda & 1 + 2\lambda \end{bmatrix} \begin{bmatrix} T_1^{k+1} \\ T_2^{k+1} \\ T_3^{k+1} \\ \vdots \\ T_{m-1}^{k+1} \\ T_m^{k+1} \end{bmatrix} = \begin{bmatrix} T_1^k \\ T_2^k \\ T_3^k \\ \vdots \\ T_{m-1}^k \\ T_m^k \end{bmatrix}. \quad (\text{D-10})$$

Figure D-8 shows the calculation result of this problem using the implicit method of the FDM with the following conditions: $\Delta x = 2$ [cm], $\Delta t = 0.1$ [s], and $\alpha = 0.835$ [cm²/s]. Figure D-9, Figure D-10, and Figure D-11 show the change of the calculation results by increasing the spatial step size Δx . Even if the stability condition $\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{k}$ is violated, Figure D-11 shows the stable solution.

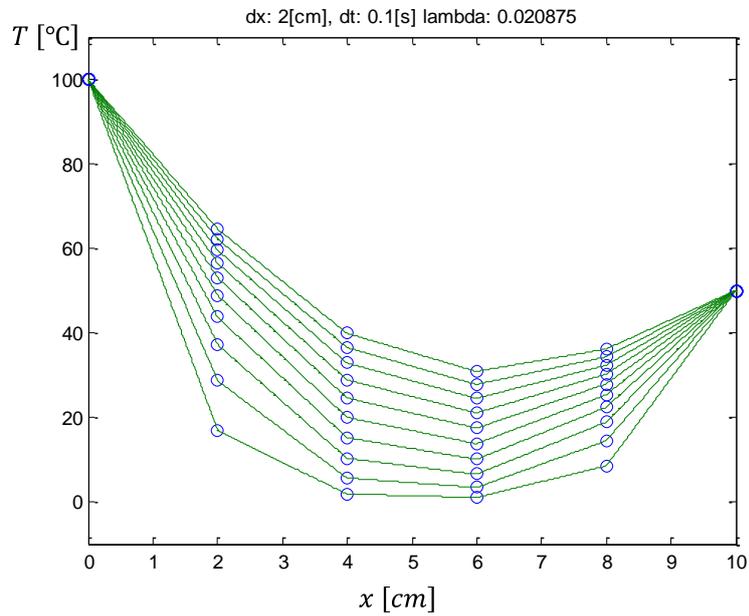


Figure D-8 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 2 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

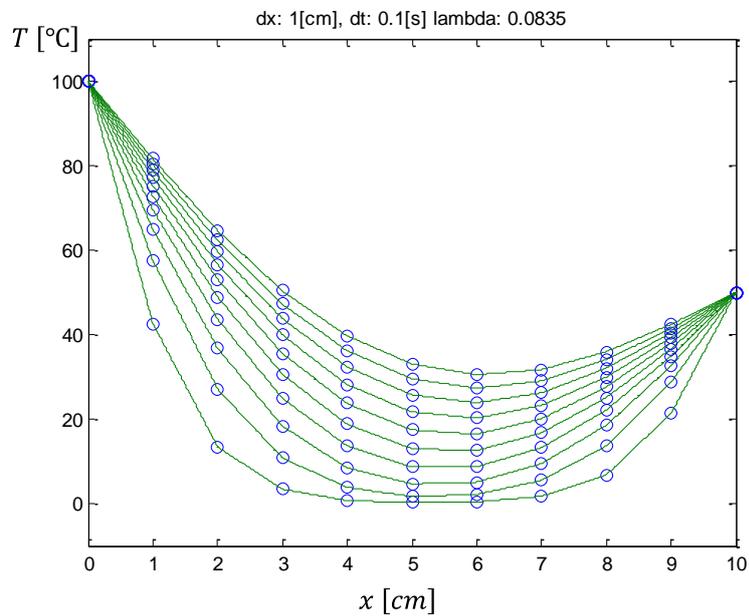


Figure D-9 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 1 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

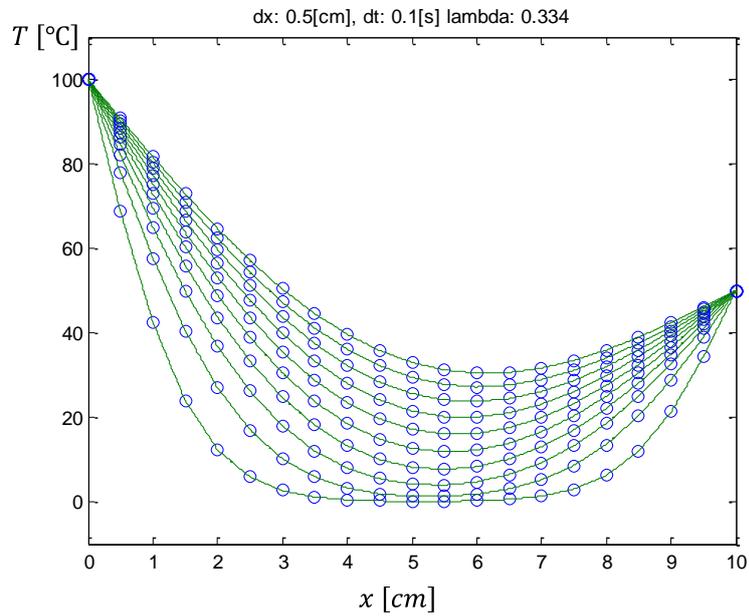


Figure D-10 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 0.5$ [cm], $\Delta t = 0.1$ [s], and $\alpha = 0.835$ [cm^2/s])

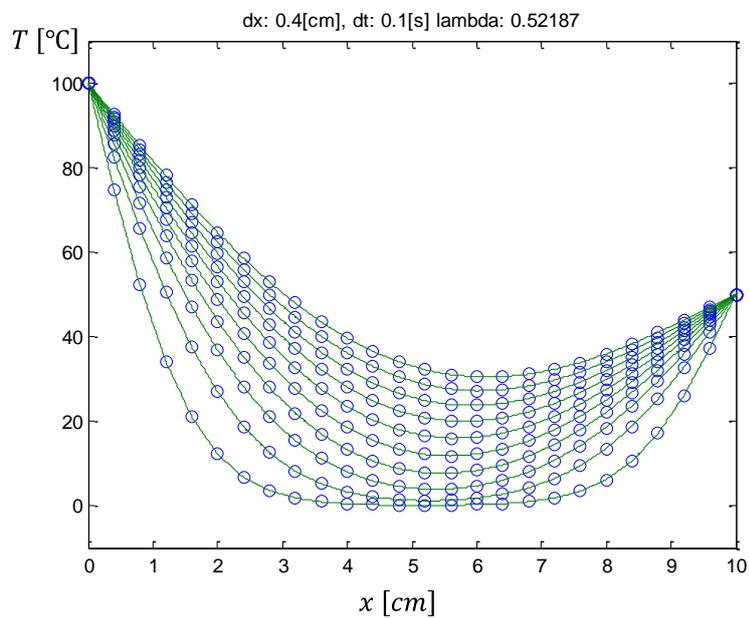


Figure D-11 Calculation result of one-dimensional heat equation using the implicit method of the FDM ($\Delta x = 0.4$ [cm], $\Delta t = 0.1$ [s], and $\alpha = 0.835$ [cm^2/s])

D.3. Solving the Equation Using Crank-Nicolson Method of the FDM

Whereas the implicit method described is stable and convergent, it has the defect that the temporal difference approximation is first-order accurate, whereas the spatial difference approximation is second-order accurate. Although the simple implicit method is unconditionally stable, there is an accuracy limit to the use of large time steps. Consequently, it is not that much more efficient than the explicit approaches for most time variable problems.

The Crank-Nicolson method provides an alternative implicit scheme that is second-order accurate in both space and time. To provide this accuracy, difference approximations are developed at the midpoint of the time increment (Figure D-12).

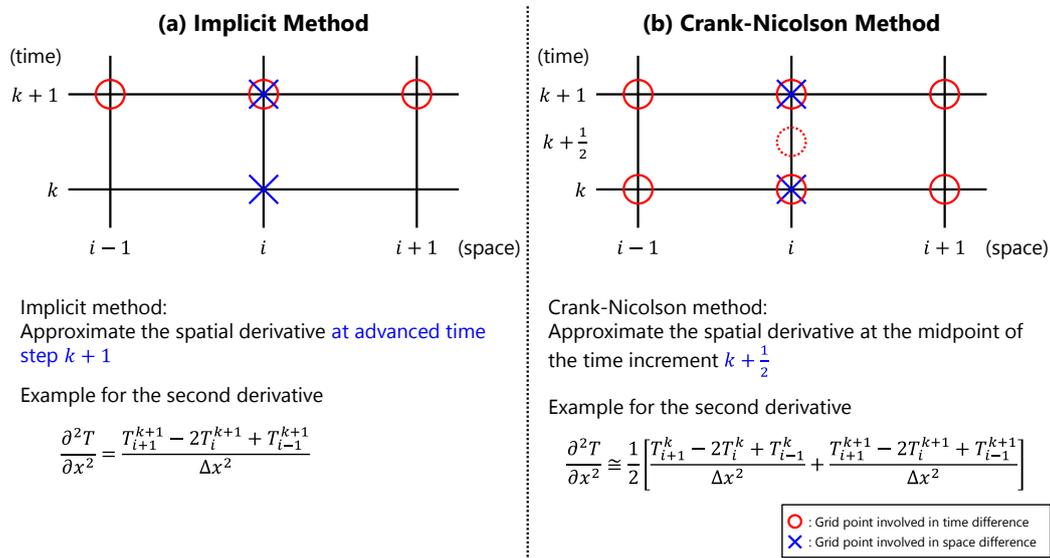


Figure D-12 Computational molecules demonstrating the fundamental differences between (a) simple implicit and (b) Crank-Nicolson methods

Figure D-13 shows the procedure of applying the implicit method to one-dimensional heat equation. The last equation in Figure D-13 applies to all but the first and the last interior nodes, which must be modified to reflect the boundary conditions.

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}, \quad T = T(x, t)$$

Left Hand Side	Right Hand Side
$\frac{\partial T}{\partial t} = \frac{T_i^{k+1} - T_i^k}{\Delta t}$	$\frac{\partial^2 T}{\partial x^2} \cong \frac{1}{2} \left[\frac{T_{i+1}^k - 2T_i^k + T_{i-1}^k}{\Delta x^2} + \frac{T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1}}{\Delta x^2} \right]$
$k: k^{\text{th}} \text{ step } (t = k\Delta t)$ $i: i^{\text{th}} \text{ point}$	
$\frac{T_i^{k+1} - T_i^k}{\Delta t} = \alpha \cdot \frac{1}{2} \left[\frac{T_{i+1}^k - 2T_i^k + T_{i-1}^k}{\Delta x^2} + \frac{T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1}}{\Delta x^2} \right]$	
$T_i^{k+1} = T_i^k + \frac{\alpha \Delta t}{2 \Delta x^2} \left[\frac{T_{i+1}^k - 2T_i^k + T_{i-1}^k}{\Delta x^2} + \frac{T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1}}{\Delta x^2} \right]$	
$T_i^{k+1} = T_i^k + \frac{\lambda}{2} \left[\frac{T_{i+1}^k - 2T_i^k + T_{i-1}^k}{\Delta x^2} + \frac{T_{i+1}^{k+1} - 2T_i^{k+1} + T_{i-1}^{k+1}}{\Delta x^2} \right], \quad \lambda = \alpha \frac{\Delta t}{\Delta x^2}$	
$-\lambda T_{i-1}^{k+1} + 2(1 + \lambda)T_i^{k+1} - \lambda T_{i+1}^{k+1} = \lambda T_{i-1}^k + 2(1 - \lambda)T_i^k + \lambda T_{i+1}^k, \quad \lambda = \alpha \frac{\Delta t}{\Delta x^2}$	

Figure D-13 Applying the Crank-Nicolson method of the FDM to one-dimensional heat equation

Using the Crank-Nicolson method, the difference equation for all nodes except the first($i = 1$) and last($i = m$) interior nodes is as follows:

$$-\lambda T_{i-1}^{k+1} + 2(1 + \lambda)T_i^{k+1} - \lambda T_{i+1}^{k+1} = \lambda T_{i-1}^k + 2(1 - \lambda)T_i^k + \lambda T_{i+1}^k, \quad (\text{D-11})$$

where $\lambda = \alpha \frac{\Delta t}{\Delta x^2}$. The boundary condition at the left end of the rod ($i = 0$) is given by

$$T_0^{k+1} = f_0(t^{k+1}), \quad (\text{D-12})$$

where $f_0(t^{k+1})$ is a function describing how the left boundary temperature changes with time. Thus, subscribing equation (D-12) to equation (D-11), the difference equation for the first interior node ($i = 1$):

$$2(1 + \lambda)T_1^{k+1} - \lambda T_2^{k+1} = \lambda f_0(t^k) + 2(1 - \lambda)T_1^k + \lambda T_2^k + \lambda f_0(t^{k+1}). \quad (\text{D-13})$$

The boundary condition at the right end of the rod ($i = m + 1$) is given by

$$T_{m+1}^{k+1} = f_{m+1}(t^{k+1}), \quad (\text{D-14})$$

where $f_{m+1}(t^{k+1})$ is a function describing how the right boundary temperature changes with time. Thus, subscribing equation (D-14) to equation (D-11), the difference equation for the last interior node ($i = m + 1$):

$$-\lambda T_{m-1}^{k+1} + 2(1 + \lambda)T_m^{k+1} = \lambda T_{m-1}^k + 2(1 - \lambda)T_m^k + \lambda f_{m+1}(t^k) + \lambda f_{m+1}(t^{k+1}). \quad (\text{D-15})$$

When the equations (D-5), (D-7), and (D-9) are written for all the interior nodes, the resulting set of m linear algebraic equations has m unknowns. In addition, the method has the added bonus that the system is tridiagonal.

$$\begin{bmatrix} 2(1 + \lambda) & -\lambda & 0 & 0 & \cdots & 0 \\ -\lambda & 2(1 + \lambda) & -\lambda & 0 & \cdots & 0 \\ 0 & -\lambda & 2(1 + \lambda) & -\lambda & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & -\lambda & 2(1 + \lambda) & -\lambda \\ 0 & \cdots & 0 & 0 & -\lambda & 2(1 + \lambda) \end{bmatrix} \begin{bmatrix} T_1^{k+1} \\ T_2^{k+1} \\ T_3^{k+1} \\ \vdots \\ T_{m-1}^{k+1} \\ T_m^{k+1} \end{bmatrix} = \begin{bmatrix} \lambda f_0(t^k) + 2(1 - \lambda)T_1^k + \lambda T_2^k + \lambda f_0(t^{k+1}) \\ \lambda T_1^k + 2(1 - \lambda)T_2^k + \lambda T_3^k \\ \lambda T_2^k + 2(1 - \lambda)T_3^k + \lambda T_4^k \\ \vdots \\ \lambda T_{m-2}^k + 2(1 - \lambda)T_{m-1}^k + \lambda T_m^k \\ \lambda T_{m-1}^k + 2(1 - \lambda)T_m^k + \lambda f_{m+1}(t^k) + \lambda f_{m+1}(t^{k+1}) \end{bmatrix}. \quad (\text{D-16})$$

Figure D-14 shows the calculation result of this problem using the Crank-Nicolson

method of the FDM with the following conditions: $\Delta x = 2 \text{ [cm]}$, $\Delta t = 0.1 \text{ [s]}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$. Figure D-15, Figure D-16, and Figure D-17 show the change of the calculation results by increasing the spatial step size Δx . Even if the stability condition $\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{k}$ is violated, Figure D-17 shows the stable solution.

The Crank-Nicolson method is often used for solving linear parabolic PDEs in one spatial dimension. Its advantages become even more pronounced for more complicated applications such as those involving unequally spaced meshes. Such nonuniform spacing is often advantageous where we have foreknowledge that the solution varies rapidly in local portions of the system.

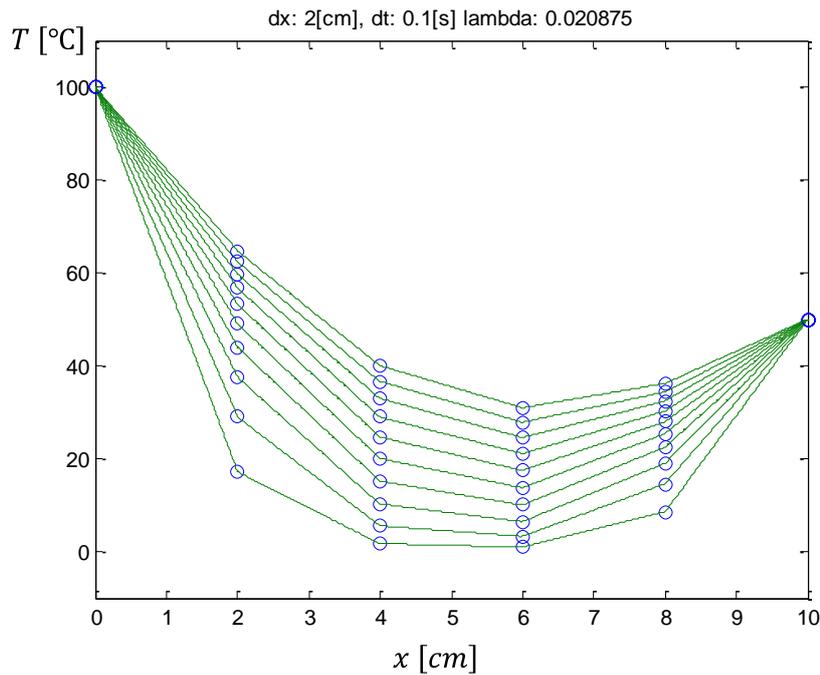


Figure D-14 Calculation result of one-dimensional heat equation using the Crank-Nicolson method of the FDM ($\Delta x = 2 \text{ [cm]}$, $\Delta t = 0.1 \text{ [s]}$, and $\alpha = 0.835 \text{ [cm}^2/\text{s]}$)

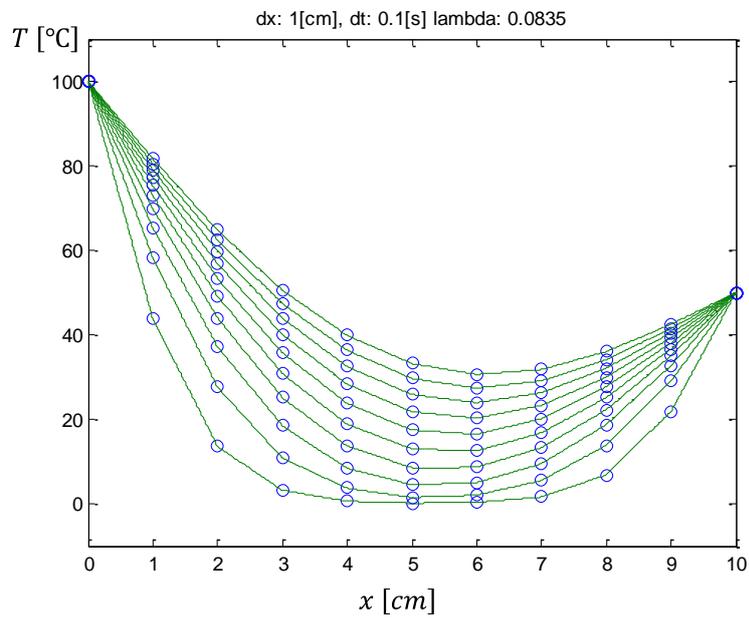


Figure D-15 Calculation result of one-dimensional heat equation using the Crank-Nicolson method of the FDM ($\Delta x = 1 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

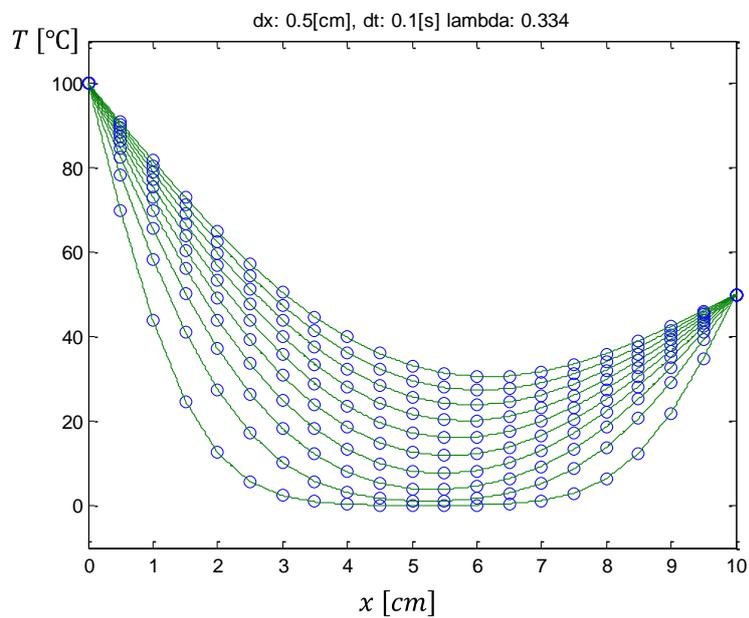


Figure D-16 Calculation result of one-dimensional heat equation using the Crank-Nicolson method of the FDM ($\Delta x = 0.5 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

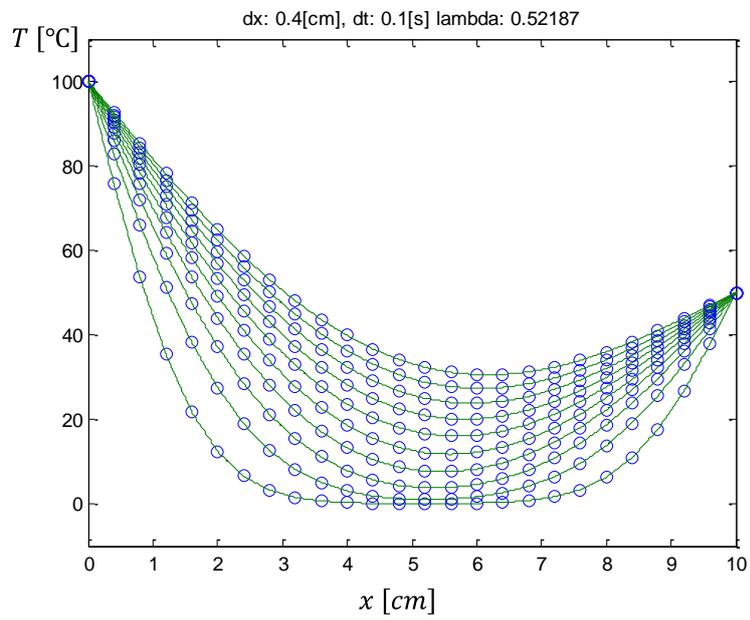


Figure D-17 Calculation result of one-dimensional heat equation using the Crank-Nicolson method of the FDM ($\Delta x = 0.4$ [cm], $\Delta t = 0.1$ [s], and $\alpha = 0.835$ [cm²/s])

D.4. Solving the Equation Using LBM

In this example, the LB equation for one-dimensional problem is given by

$$f_k(x + c_k \Delta t, t + \Delta t) = f_k(x, t)[1 - \omega] + \omega f_k^{eq}(x, t), \quad (\text{D-17})$$

which is derived from the kinetic equation for the distribution function by applying the BGK approximation. In equation (D-17) ω is called a relaxation time, and given by

$$\alpha = \frac{\Delta x^2}{\Delta t D} \left(\frac{1}{\omega} - \frac{1}{2} \right). \quad (\text{D-18})$$

The conditions of the problem is assumed as $\alpha = 0.835$, $\Delta x = 2.0$, $\Delta t = 0.1$, and $D = 1$, so by using the (D-18), we can calculate that $\omega = 1.9198$.

The temperature of the rod $T(x, t)$ is related with the distribution function $f_k(x, t)$, as,

$$T(x, t) = \sum_{k=1}^2 f_k(x, t). \quad (\text{D-19})$$

The equilibrium distribution function $f_k^{eq}(x, t)$ can be chosen as

$$f_k^{eq}(x, t) = w_k T(x, t), \quad (\text{D-20})$$

where w_k stands for the weighting factor in the direction k . The weighting factor should satisfy the following criterion,

$$\sum w_k = 1. \quad (\text{D-21})$$

The weight factor for one-dimensional problem is given by $w_k = 0.5$; for $k=1$ and 2 .

Figure D-18 shows schematic diagram for three nodes with necessary linkages, central

and neighboring nodes, where the velocity vectors are $c_1 = 1$ and $c_2 = -1$.

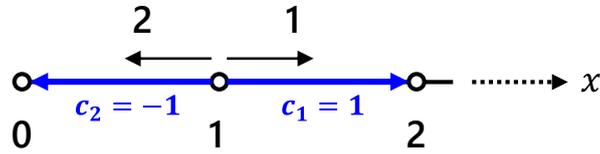


Figure D-18 Lattices for one-dimensional heat equation using the LBM

From the equation (D-17), the LB equation for the direction 1 (right) in Figure D-18 is derived as

$$f_1(x + 1, t + 1) = f_1(x, t)[1 - \omega] + \omega f_1^{eq}(x, t), \quad (D-22)$$

where the equilibrium function to the direction 1 is

$$f_1^{eq}(x, t) = 0.5T(x, t) = 0.5[f_1(x, t) + f_2(x, t)]. \quad (D-23)$$

In the same manner, from the equation (D-17), the LB equation for the direction 2 (left) in Figure D-18 is derived as

$$f_2(x - 1, t + 1) = f_2(x, t)[1 - \omega] + \omega f_2^{eq}(x, t), \quad (D-24)$$

where the equilibrium function to the direction 2 is

$$f_2^{eq}(x, t) = 0.5T(x, t) = 0.5[f_1(x, t) + f_2(x, t)]. \quad (D-25)$$

Using the equations (D-23) and (D-25), the procedure of the calculation using the LBM at step $k = 1$ is shown in Figure D-19. As you can see, the boundary conditions are needed at the first and last nodes.

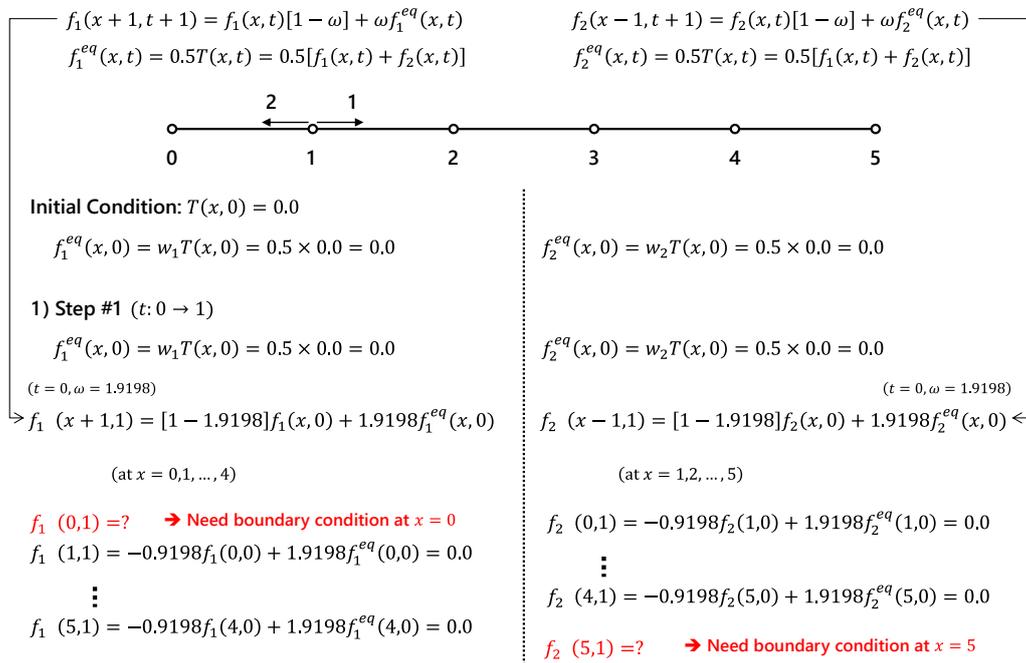


Figure D-19 Procedure of the calculation using the LBM for one-dimensional heat equation at step $k = 1$

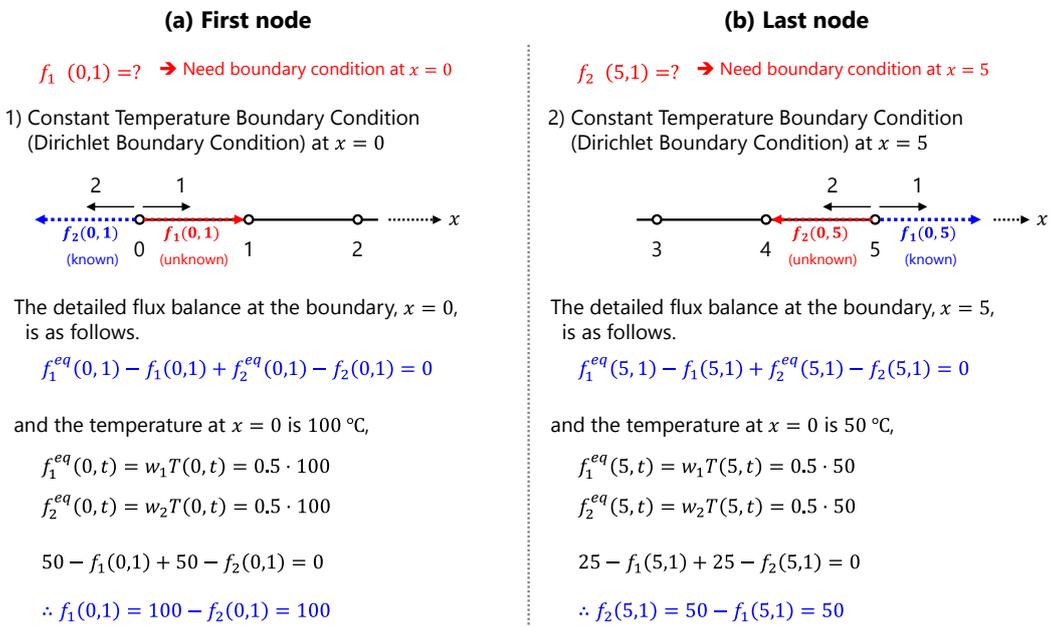


Figure D-20 Consideration of the boundary conditions using the LBM for one-dimensional heat equation at step $k = 1$

As shown in Figure D-20, the boundary conditions are considered at step $k = 1$ as follows:

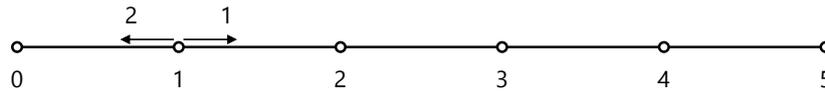
$$f_1(0,1) = 100 - f_2(0,1) = 100, \quad (\text{D-26})$$

$$f_2(5,1) = 50 - f_1(5,1) = 50, \quad (\text{D-27})$$

From the equations (D-23), (D-25), (D-26), and (D-27), the LB equations are given as shown in Figure D-21. The lattice distributions by calculating these LB equations are also shown in this figure.

$$f_1(x+1, t+1) = f_1(x, t)[1 - \omega] + \omega f_1^{eq}(x, t) \quad f_2(x-1, t+1) = f_2(x, t)[1 - \omega] + \omega f_2^{eq}(x, t)$$

$$f_1^{eq}(x, t) = 0.5T(x, t) = 0.5[f_1(x, t) + f_2(x, t)] \quad f_2^{eq}(x, t) = 0.5T(x, t) = 0.5[f_1(x, t) + f_2(x, t)]$$



$$f_1(x, 1) = \begin{cases} 0.0, & (\text{except } x = 0) \\ 100.0, & (x = 0) \end{cases} \quad f_2(x, 1) = \begin{cases} 0.0, & (\text{except } x = 5) \\ 50.0, & (x = 5) \end{cases}$$

$$T(x, t) = \sum f_k(x, t) \longrightarrow \therefore T(x, 1) = \begin{cases} 0.0, & (\text{except } x = 0, 5) \\ 100.0, & (x = 0) \\ 50.0, & (x = 5) \end{cases}$$

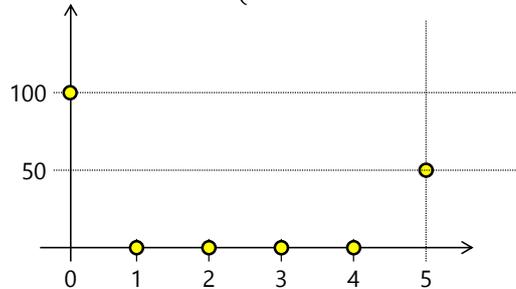
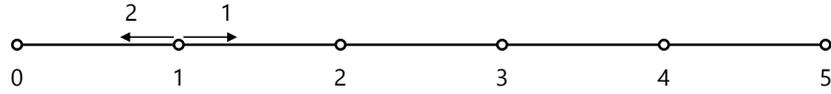


Figure D-21 LB equations and lattice distributions for one-dimensional heat equation at step $k = 1$

In the same manner, the LB equations at step $k = 2$ are also derived as shown in Figure D-22. The lattice distributions at step $k = 2$ by calculating these LB equations are also shown in Figure D-22.

$$f_1(x+1, t+1) = f_1(x, t)[1 - \omega] + \omega f_1^{eq}(x, t) \quad f_2(x-1, t+1) = f_2(x, t)[1 - \omega] + \omega f_2^{eq}(x, t)$$

$$f_1^{eq}(x, t) = 0.5T(x, t) = 0.5[f_1(x, t) + f_2(x, t)] \quad f_2^{eq}(x, t) = 0.5T(x, t) = 0.5[f_1(x, t) + f_2(x, t)]$$



$$f_1(0, 2) = 100, f_1(1, 2) = 4.01$$

$$f_1(x, 2) = 0, (x = 2, \dots, 5)$$

$$f_2(x, 2) = 0, (x = 0, \dots, 3)$$

$$f_2(4, 2) = 2.00, f_2(5, 2) = 50$$

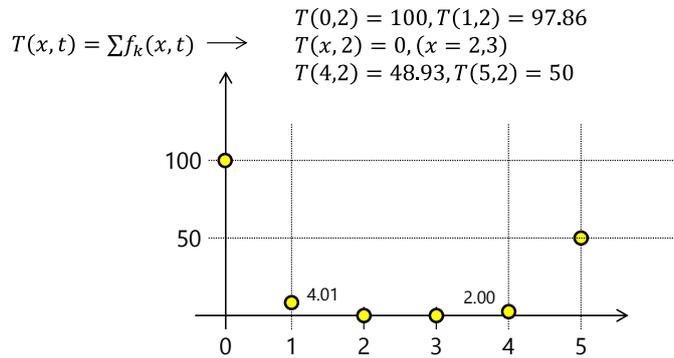


Figure D-22 LB equations and lattice distributions for one-dimensional heat equation at step $k = 2$

Figure D-23 shows the calculation result of this problem using the LBM with the following conditions: $\Delta x = 2 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$. Figure D-24, Figure D-25, and Figure D-26 show the change of the calculation results by increasing the spatial step size Δx , and those are still stable even if the spatial step size increases.

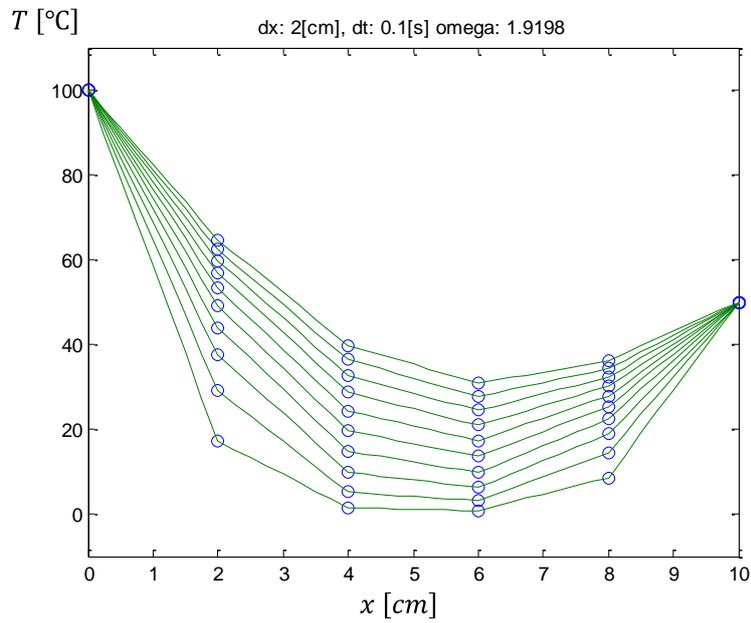


Figure D-23 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 2$ [cm], $\Delta t = 0.1$ [s], and $\alpha = 0.835$ [cm²/s])

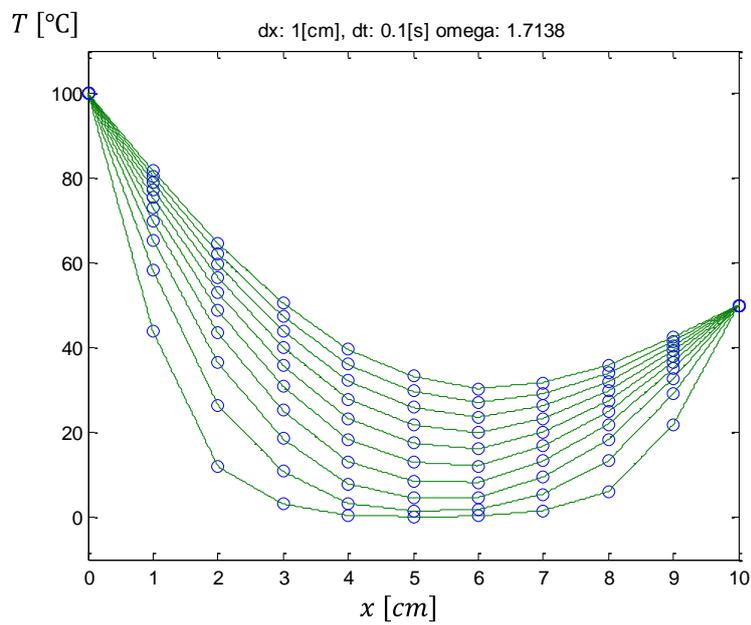


Figure D-24 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 1$ [cm], $\Delta t = 0.1$ [s], and $\alpha = 0.835$ [cm²/s])

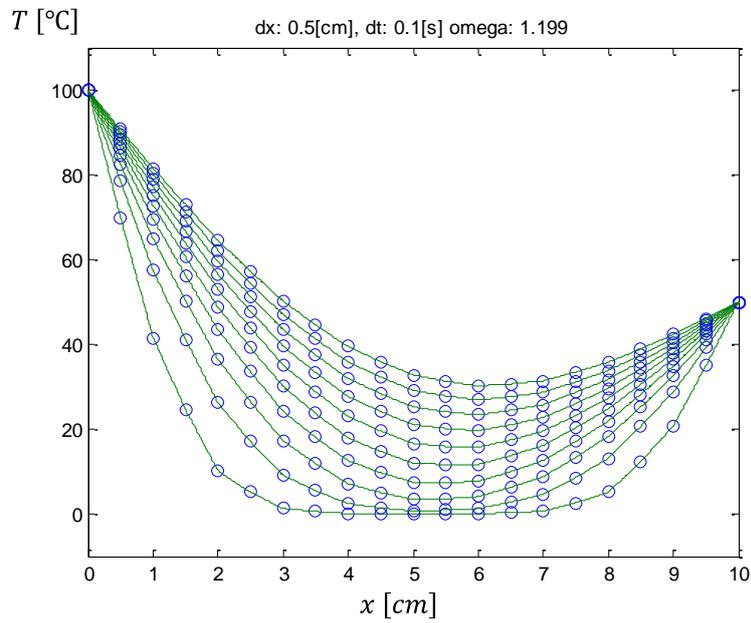


Figure D-25 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 0.5 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

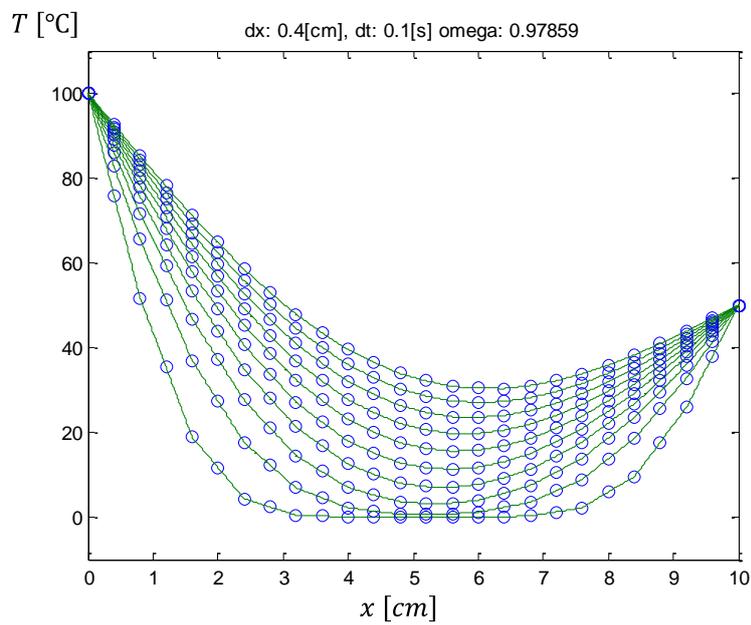


Figure D-26 Calculation result of one-dimensional heat equation using the LBM ($\Delta x = 0.4 [cm]$, $\Delta t = 0.1 [s]$, and $\alpha = 0.835 [cm^2/s]$)

D.5. Comparison of the Calculation Results Using the FDM and the LBM

Table D-1 shows the comparison of the calculation result for one-dimensional heat equation with the conditions $\Delta x = 2 \text{ cm}$ and $\alpha = 0.835 \text{ cm}^2/\text{s}$. This table compares the temperature of the rod at $t = 10 \text{ [s]}$ and $x = 2 \text{ [cm]}$. Note that the analytic solution is $T(2,10) = 64.8018 \text{ }^\circ\text{C}$. As shown in Table D-1, only the explicit method of the FDM is not converged when the unit time step increases. The Crank-Nicolson method shows the most exact solution among the methods, followed by the LBM.

Table D-1 Comparison of the calculation result for one-dimensional heat equation: temperature at
 $t = 10 \text{ [s]}$ at $x = 2 \text{ [cm]}$

Δt	Finite Difference Method				Lattice Boltzmann Method	
	λ	Temperature ($^\circ\text{C}$)			ω	Temperature ($^\circ\text{C}$)
		Explicit	Implicit	Crank-Nicolson		
0.1	0.0209	64.8513 (0.0764%)	64.6087 (-0.2987%)	64.6083 (-0.1101%)	1.9198	64.6856 (-0.1793%)
0.2	0.0418	64.9720 (0.2626%)	64.4859 (-0.4875%)	64.7316 (-0.1084%)	1.8459	64.6429 (-0.2453%)
0.5	0.1044	65.3302 (0.8154%)	64.1158 (-1.0586%)	64.7394 (-0.0963%)	1.6546	64.5238 (-0.4291%)
1.0	0.2087	65.9147 (1.7174%)	63.4909 (-2.0229%)	64.7671 (-0.0536%)	1.4109	64.3444 (-0.7058%)
2.0	0.4175	67.1209 (3.5787%)	62.2246 (-3.9770%)	64.8743 (0.1119%)	1.0899	61.1616 (-5.6174%)
5.0	1.0438	-9.1328 (-114.0934%)	58.4926 (-9.7362%)	64.7925 (-0.0144%)	0.6478	67.6113 (4.3355%)
10.0	2.0875	208.75 (-222.1360%)	53.0061 (-18.2027%)	79.7713 (23.1004%)	0.3865	0 (-100.0%)

국문 초록

선체 운동을 고려한 LNG FPSO 액화 사이클의 온톨로지 모델링 및 DEVS 형식론 기반 공정 및 신뢰도 시뮬레이션

LNG FPSO(Liquefied Natural Gas-Floating, Production, Storage and Offloading)에서 액화 공정 시스템(Liquefaction Process System)은 천연 가스를 액화시키는 Topside Process System 의 핵심 공정 중 하나이다. LNG FPSO 의 액화 공정은 Topside Process System 건조 비용의 70%, 그리고 전체 건조 비용의 30%에서 40%를 차지한다. 따라서, 액화 공정의 최적 설계는 LNG FPSO 의 설계 및 생산 비용과 직접적인 연관이 있다. 액화 공정 시스템의 사이클을 구성하는 주요 장비에는 Compressor, Condenser, Expansion Valve, Heat Exchanger, Phase Separator, Common Header, Tee 가 있으며, 이들의 조합에 따라 다양한 액화 사이클의 설계 대안이 도출된다. 본 논문에서는 LNG FPSO 의 설계 제약 조건에 따라 최적의 액화 사이클을 도출하기 위해, 다양한 액화 사이클의 대안을 자동으로 생성할 수 있는 온톨로지 모델링 방법을 제안하고, 이를 이용하여 다양한 액화 사이클에 대한 공정 및 신뢰도 분석 시뮬레이션을 수행하였다.

다양한 액화 사이클의 설계 대안을 도출하기 위해 온톨로지 모델링 방법에 기반한 액화 사이클의 논리 모델을 제안하였다. 액화 사이클의 논리 모델이란 액화 사이클을 구성하는 장비와 장비들 간의 연결 및 종속 관계를 표현한 것을

말한다. 액화 사이클을 구성하기 위한 장비들의 조합 규칙은 C3MR 사이클, DMR 사이클 등과 같이 기존에 육상에서 사용되고 있는 액화 사이클을 분석하여 추출하였다. 액화 사이클의 논리 모델은 개체들 간의 연결 및 종속 관계를 트리 구조 형태로 잘 표현할 수 있는 온톨로지 모델링 방법 중 하나인 System Entity Structure (SES)를 이용하여 표현하였다. SES 로 표현한 논리 모델을 기반으로 가지치기(Pruning) 과정을 거치면 다양한 액화 사이클의 논리 모델들을 자동으로 생성할 수 있으며, 이를 공정 시뮬레이션과 신뢰도 분석 시뮬레이션에 이용할 수 있다.

최적의 액화 사이클을 찾기 위해서는 주어진 설계 제약 조건에 따라 여러 가지 설계 대안을 평가해야 한다. 설계 제약 조건 중 하나인 액화 사이클의 효율을 극대화 하기 위해서는 공정 시뮬레이션을 통해 액화 사이클의 최적 운전 조건을 결정해야 한다. 따라서 공정 시뮬레이션을 위해, 운전 조건을 계산하기 위한 장비의 물리 모델들을 열역학에 기반하여 정의하였다. 그리고 이들을 SES 로부터 자동 생성된 액화 사이클의 논리 모델과 결합하면 다양한 설계 대안에 대한 공정 시뮬레이션을 수행할 수 있으며, 이로부터 최적의 효율을 가지는 액화 사이클을 도출할 수 있다. 장비의 모듈화와 액화 사이클의 변경을 용이하게 하기 위해 Sequential Modular Simulator 방법을 도입하였으며, 운전 조건을 계산하기 위한 장비의 물리 모델은 이산 사건 시스템 (Discrete Event System Specification; DEVS) 형식론을 기반으로 정의하였다.

LNG FPSO 의 또 다른 설계 제약 조건으로는 액화 사이클의 신뢰도가 있다. 전통적으로 시스템의 신뢰도를 분석하기 위해 Fault Tree Analysis (FTA) 방법을 이용해 왔다. 그러나 FTA 방법은 해당 분야의 전문가가 시스템을 분석하고 Fault Tree 를 설계해야 하기에 시간과 비용이 소요된다. 따라서 본 논문에서는

시스템의 구성도(Configuration)를 그대로 사용하는 Reliability Block Diagram (RBD) 방법을 이용하였다. 액화 사이클의 논리 모델은 시스템의 구성도에 해당하는 장비들 간의 연결 및 종속 관계를 가지고 있기에, RBD 로 손쉽게 전환할 수 있다. 신뢰도를 분석하기 위해 시간 흐름에 따른 각 장비의 작동 여부를 판단할 수 있는 물리 모델을 DEVS 형식론에 기반하여 정의하였으며, 액화 사이클의 고장 확률을 계산하기 위해 몬테칼로 시뮬레이션을 이용하였다. 공정 및 신뢰도 분석 시뮬레이션은 전통적으로 많이 사용되는 액화 사이클인 C3MR 사이클, DMR 사이클, N₂ Expander 사이클, Niche 사이클, Cascade 사이클에 적용하였다.

LNG FPSO 가 설치될 해상 환경을 고려하기 위해 선체의 운동이 액화 사이클의 주요 장비 중 하나인 열 교환기(Heat Exchanger)에 어떤 영향을 미치는지에 대해 격자 볼츠만 방법을 도입하여 분석하였다. 그리고 분석 결과를 바탕으로 선체 운동에 의한 열 교환기의 고장 횟수를 계산하고 이를 신뢰도 분석 시뮬레이션에 반영하여, 선체 운동에 의해 액화 사이클의 신뢰도가 어떻게 변하는지에 대한 분석을 수행하였다.

Keywords: LNG FPSO, 액화 사이클, 온톨로지 모델링, 공정 및 신뢰도 시뮬레이션, DEVS 형식론

Student number: 2003-21035

후기

10 년이다.

2003 년, 대학원 석사과정에 입학한 후 10 년이라는 세월이 지났다. 겨우 서른 초반의 나이에 지나온 세월을 돌이켜 본다는 것이 웃길 수 있지만, 나에게는 지난 10 년의 세월이 짧지만은 않았던 것 같다.

지금도 그렇지만 대학 때의 나는 딱히 성실한 사람이 아니었다. 대학 때의 나는 게임에 미쳐 살았고 학점은 그럭저럭 중간을 유지하는 수준이었다. 게임 길드를 가입하고 당시 페인 양성 게임인 디아블로 II 와 당시 유행하던 오락실의 게임들을 섭렵했다. 시간이 흘러 4 학년이 되자 진로에 대해서 고민하고 대학원 석사 과정에 입학할 것을 4 학년 1 학기 때 결정하였다. 당시에는 아무것도 모르던 22 살의 나이에 선박기본설계 수업을 듣고 단순히 컴퓨터가 좋아서 (컴퓨터도 결국은 게임 때문에 좋아한 것 같다) 지금의 연구실을 결정하였다. 지금 다시 생각해도 웃기지만, 석사 진학 결정은 4 학년 1 학기 때 했지만 인턴 생활은 여름 방학이 끝난 직후인 4 학년 2 학기부터 시작했다. 당시에 ASDAL 연구실은 힘들기로 유명했기에 인턴 생활을 하기 전 마지막을 불태워 보자는 강력한 의지에서 대학의 마지막 여름 방학을 거의 매일 오락실에서 보냈다고 해도 과언이 아니다. 하루에 10 시간 동안 오락실에 있었던 때도 있었다.

어찌됐든 2002 년 9 월, 나의 연구실 인턴 생활이 시작되었다. 4 학년 졸업 즈음의 CAD/CAM 학회의 발표를 시작으로 나의 대학원 생활은 본격적으로 시작되었다. 최근에 논문을 마무리 하면서 2004 년 석박사 통합 과정을 결정하게 되었는지 곰곰이 생각해 보았지만 왜 박사 과정을 지원했는지 명확한 이유가 기억나지 않는다. 당시 시작한 대우조선해양과의 로봇 프로젝트가 많은 영향을 주었고, 당시 박사 과정으로 있었던 조두연 박사, 노명일 박사에게 대한 동경, 그리고 교수님과의 면담이 영향을 주지 않았나 싶다.

대학원 생활을 돌이켜 보면 교수님의 배려 덕분에 진행된 과제도 꽤나 된다. 로봇 프로젝트를 시작으로 각종 수중운동체 관련 과제, 그리고 승객 탈출 시뮬레이션 과제 등

여러 가지 프로젝트를 연구실 구성원들과 함께 했다. 사실 나는 로봇 프로젝트를 시작할 때 이 과제의 끝이 내 박사 과정의 끝이 될 줄 알았다. ‘로봇’이라는 주제에 대한 두려움이 있었지만, 6년이라는 과제 연구 기간이 예상 기간과 얼추 비슷하였기 때문이다. 하지만 내가 부족한 탓인지 교수님과의 잦은 트러블로 결과는 그렇지 않았다.

10 년이면 강산도 변한다는데, 돌이켜 보면 나도 많은 부분이 바뀌었다. 내 옆에는 나를 바라 보고 있는 아내와 든든한 후원을 해 주시는 처갓집 식구들이 생겼고. 20 대 초반의 훌쩍한 몸매는 어느 새 배가 나온 아저씨 몸매가 되었고, 문득 거울을 보니 팔자 주름도 조금 더 깊게 패인 것 같다. 가치관도 많이 달라지고 사람을 대하는 자세, 일하는 자세도 나름대로 많이 바뀐 것 같다. 무엇보다 대학 때와 같이 게임에 미쳐 있지는 않다 (컴퓨터에 미쳐 있기는 하지만...).

10 년의 연구실 생활 동안 정말 여러 가지 일들이 있었지만 그 중에서 가장 기억에 남는 것은 로봇 프로젝트의 당등산 시연회 때의 사건이다. 왼쪽의 사진에서는 다행히 열심히 용접을 하고 있는 RRX3 이지만, 당시에는 시연 바로 직전까지 레이저 센서 뚜껑이 안 열리는 것을 비롯해서 여러 가지 문제가 있었다. 어찌됐건 시연 준비는 무사히(?) 마무리 되었고, 당시 대우조선해양 부사장님을 비롯하여 여러 사람이 참석한 가운데 시연이 시작되었다.



시연 초반의 과정이 어제 일처럼 아직도 생생하다. 시연은 로봇이 옆으로 이동하고 다시 앞으로 이동한 후 용접하는 것으로 계획되어 있었다. 시연이 시작되고 교수님께서 부사장님 옆에서 열심히 로봇의 이동 과정을 설명을 하신다. 로봇은 옆으로 잘 이동하고 있다. 그런데 갑자기 로봇이 멈췄다... 약 3 초 정도 로봇이 멈춰 있고, 교수님과 남국이가 나를 쳐다 보고 있다...

지금와서 이야기 하지만 로봇이 멈춘 것은 나의 실수 였다. 당시 고질적인 문제였던 레이저 센서 뚜껑이 열리지 않아서 뚜껑을 열어둔 채로 시연을 하기로 계획하였는데, 뚜껑을 열지 않고 시연을 시작한 탓에 부랴부랴 로봇이 옆으로 이동하는 중에 레이저 센서 뚜껑을 여는 명령을 키보드로 입력하였다. 근데, 입력하고 나서 생각해 보니 이동

도중의 키보드 명령은 로봇의 동작을 일시 정지 상태로 만들게 된다는 것을 바로 기억하게 되었다. 로봇은 3 초 멈추어 있었지만 나는 키보드로 명령을 입력한 직후부터 어떻게 할지 미친듯이 생각을 하기 시작했다. 다행히 로봇 제어 프로그램의 버그를 기억해 내었지만, 2 개의 명령어 중 하나를 짚어야만 하는 상황... 14 번이나 15 번이냐의 선택의 기로에 놓여 있다가 교수님과 남국이의 다가온 시선을 느끼고 약간의 확신만을 가진 채 에라 모르겠다라는 심정으로 하나를 선택했는데, 다행히 로봇이 움직이고 시연이 계속 진행 되었다. 시연이 끝나고 누군가가 와서 나에게 물었다. 로봇을 동작 시키려면 계속 키보드로 뭔가를 입력해야 하나고... 허허..

대학원 생활의 전반기에 진행한 로봇 프로젝트는 나에게 많은 추억을 남겼다. 로봇 프로젝트 현장 적용을 위해 남국, 민재와 함께 몇 개월 간 대우조선해양 현장에서 이리 저리 돌아다니던 일도 생생하다. 현장 테스트 바로 직전에 나는 4 주 훈련을 위해 훈련소를 다녀왔는데, 훈련소에서 걸린 감기가 나올 당시에 많이 심해졌다. 그러나 4 주 훈련을 마치고 나오자마자, 교수님께서 전화를 하시고 현장 테스트를 위해 거제도로 내려가야겠다는 말씀을 전하셨다. 감기 몸살이 너무 심해서 하루 이틀 쉰 후 거제도로 내려갈 계획을 잡았는데, 하필이면 또 집 보일러가 고장나서 아픈 몸에 벌벌 떨면서 집에서 잤던 기억이 난다.

대학원 생활의 일들이 주마등 같이 지나간다. 같이 일하던 동훈이형에게 반항(?)한 일, 니킥 사건, UVRC 발표 준비를 하다가 교수님께 꿀밤을 맞은 일도...

대학원 생활이 개인적으로 힘들어진 시기는 2007 년 여름의 일본 여행 전후이다. 당시 나는 주어진 일을 미루고 회피하고자만 했었다. 연구실에서 여러 가지 일이 있었으나 이리 저리 피하기만 하다가 결국 일본 여행을 빌미로 교수님께서 부탁하신 국제 학회 발표 자료 준비도 회피하였다. 주환이형은 여행을 떠나기 직전 나를 말렸지만, 나는 그냥 떠났고, 결국 교수님께 좋지 않은 강력한 인상을 남겼다.

돌이켜보니 대학 때 가지 못한 국외 여행을 대학원 들어와서는 꽤나 많이 다닌 것 같다. 첫 여행이었던 스페인 여행은 약 10 여년 전 일이지만 아직도 기억에 남는다. 외국인과 살라살라 대화를 잘하는 정한이의 영어 실력에 놀랐고, 발품 팔아 걸어 다니던 마드리드 시내의 지도는 아직도 기억에 생생하다. 특히 기억에 남는 건 마지막 날인 금요일 밤, 클럽 데이의 흥대와 비슷한 느낌의 거리와 거리를 활보하는 젊은 늘씬한

서양 미녀들이 나에게 충격을 주었다. 문제는 맥주 석 잔 먹고 뺏어서 후반부는 기억이 잘 안다는 것이... 이외에도 연구실 구성원들과 함께한 중국, 일본 여행, 그리고 학회 발표를 위해 간 이탈리아, 미국, 그리스 등등.. 참으로 많이도 다녔다. 덕분에 간도 좀 커졌는지 국제 학회에 나가서 되지도 않는 영어로 발표하는 스킬은 늘었다.

지난 인생에 대한 잡설은 이제 그만해야겠다. 곰곰히 생각해 보면 여러 일들이 있었겠지만 논문 마감 시간이 다가오기에... 박사 학위는 하나의 마침표라고 생각하고 또 다른 목표를 위해 준비나 성실히 해야겠다.

논문에 도움을 주신 여러 분들께 이 자리를 빌어 감사의 말씀을 드립니다.

먼저 논문 지도에 물심 양면으로 지원해 주신 이규열 교수님께 감사의 말씀을 드립니다. 느릿느릿한 자세로 늘상 교수님의 마음에 들지 않는 행동으로 교수님의 화를 돋구는 제자였기에 더욱 더 죄송한 마음과 함께 감사의 마음을 전합니다. 교수님께서서는 항상 저를 보시면서 조금 더 빠르게, 그리고 냉철하게 행동할 필요가 있다라고 강조하시지만 지난 10 여년의 연구실 생활 동안 제대로 교수님의 기대에 부응하지 못한 것 같습니다. 교수님께서 지도해 주신 학문적인 가치관과 자세는 제 인생에 커다란 길을 제시해 주셨으며, 앞으로도 조금 더 멋진 제자가 될 수 있도록 노력하겠습니다.

그리고 논문을 지도해 주신 심사 위원 교수님들께 감사의 말씀을 드립니다. 매 심사마다 날카로운 지적을 해 주시고, 심사 이외에도 논문의 내용에 대해 직접 연락하셔서 조언을 주신 양영순 교수님께 감사 드립니다. 지난 3 여년간 수중운동체특화센터 과제를 함께 하면서 학위 논문의 핵심 주제인 DEVS, SES 에 대해 많은 도움을 주신 항공대학교 지승도 교수님께서 감사의 말씀을 드립니다. 또한, 논문 심사에 참석하셔서 논문의 핵심 내용을 부각시킬 수 있도록 많은 조언을 주신 기계항공공학부 김찬중 교수님께도 감사 드립니다. 그리고 학부 때부터 대학원까지 항상 많은 조언과 가르침을 주셨던 서울대학교 조선해양공학과 교수님들께도 진심으로 감사를 드립니다.

항상 동고동락하면서 연구실 생활을 한 ASDAL 졸업생들께도 진심으로 감사 드립니다.

남국아, 네가 정말 고생했다. 대학 동기라서 싫은 소리 한번 못 하고 네 할일도 많은 텐데 나를 도와주라, 연구실 일 하라 정말 많이 힘들었지? 민재와 함께 세 명이서

지내온 시간이 근 14 여년, 형제보다도 더 오랜 시간 함께하면서 결혼식 사회를 포함해서 도움 받은 일이 너무나 많기에 글로는 다 표현할 수가 없구나. 네 열정은 항상 보면서 감탄하기에 곧 좋은 기회가 있을 것을 확신하고 친구로써 응원할게.

주환이형, 형이 언젠가 남국이와 함께 저를 불러 놓고 이야기를 하셨죠~? 졸업하려면 집에 가지 말라고... 그 땐 싫다라고 이야기 했는데, 결국 형의 예언대로 졸업 전 1 년은 연구실에서 지냈네요. 재운이형과 함께 세 명이서 워크래프트를 하던 때가 엇그제 같은데 벌써 10 년이 흘렀네요. 대학원 생활 동안 지속적으로 신경 써 주시고 조언해 주시던 것 정말 감사 드립니다.

광필이형, 정말 감사 드립니다. 형이 연구실에 계시는 동안 학문적인 것뿐만 아니라 그 외의 것들을 정말 많이 배울 수 있었네요. 형의 철두철미한 자세는 그 때도 그렇고 아직도 감탄을 하면서 바라 보고 있습니다. 단 20% 만이라도 형의 자세를 따라갈 수만 있으면 좋으련만.. 다시 한번 연구실에 계신 동안 대학원 생활뿐만 아니라 여러 가지 측면에서 많은 조언 주신 점, 정말 감사 드립니다.

명일이형, 항상 형의 깨끗한 자세를 보면서 많은 것을 배웁니다. 국외 논문 투고를 위해 많은 도움을 주신 것도 정말 감사 드립니다. 그리고 두연이형, 졸업이 늦어질 때 진심 어린 조언으로 후배의 마음을 다독여 주신 점, 진심으로 감사 드립니다.

같이 학위를 준비하면서 함께 고생한 지현이형, 모르는 것이 있을 때 마다 물어보면 척척 답변을 해 주셔서 정말 많은 도움이 되었습니다. 준채야, 네 논문이 없었으면 내 논문도 없었다라고 해도 과언이 아니다. 네가 정리해 놓은 여러 자료들이 내 학위 논문에 정말 도움이 많이 되었구나. 성차별 없이 언제나 깨끗하게 연구실 생활을 하면서 도움을 준 아라야, 정말 고맙다.

재운이형, 형의 명확한 태도는 언제 봐도 일품이에요! 로봇 프로젝트 초창기 함께 하면서 형에게 많은 것을 배울 수 있었어요. 그리고 연구실에서 나가시는 순간까지 저를 걱정해 주신 점 진심으로 감사 드립니다.

나에게 평생 인연을 소개시켜 준 재승이, 승객 탈출 시뮬레이션을 하면서 함께 고생했던 윤옥이, 같이 로봇 과제 하면서 동고동락 했던 (권)정한이와 정우, 연구실에서뿐만 아니라 회사 가서도 많은 도움을 준 승호에게도 고맙다는 말을 전한다. 그리고 회사원의 표본을 나에게 처음으로 일깨워 주신 최우영 차장님, 나를 변호하다

날벼락을 맞은 성준이, 냉철하면서도 나름(?) 여린 (김)정한이, 매사에 성실한 성진이, 삼성중공업에서 열심히 설계를 하고 계실 선빈이 형까지, 연구실 생활 초창기 멤버들이 함께 일하고, 게임하고, 술 마시던 시절이 정말 그립습니다. 그리고 간간히 빌 때마다 좋은 이야기를 많이 해 주신 황준규 차장님께도 감사의 말씀을 드립니다. 연구실 행정 업무를 하느라 고생하는 지연이, 그리고 인턴 생활 하면서 나에게 잔소리를 꽤나(?) 들었던 재복이, 성혁이, 성이에게 고맙다는 말을 전합니다.

민재야, 네가 3 월이면 부산으로 내려가는 데, 복도를 지나면서 자주 보던 네 모습을 그리워 지겠구나. 같이 활동하는 piracy 영역(?)이 있으니 온라인으로는 자주 연락하겠지? 아무쪼록 네 부인도 순산하길 기원하고 하는 일 잘 되길 기원한다.

비록 같은 연구실은 아니지만 함께 동고동락했던 IT 연구실의 구성원들에게도 감사의 말을 전합니다. 수중운동체특화센터 과제 2 단계 말년에 함께 고생했던 명조, 한 때 같은 연구실 멤버였던 보람이와 현진이, 함께 회식을 하고 술잔을 기울였던 정호형, 승철이형, 현호에게도 고맙다는 말을 전합니다.

그리고 지난 10 여년의 연구실 생활 동안 함께 프로젝트를 진행한 분들께도 감사의 말씀을 드립니다. 수중운동체특화센터 과제를 진행하면서 많은 지원과 조언을 주신 이심용 박사님, 나영인 박사님, 그리고 차기중어뢰 과제를 수행하면서 여러 인맥을 연결해 주신 홍우영 교수님께도 감사 드립니다. 함께 수중운동체특화센터 일을 마무리 지은 항공대의 은복이와 석훈이 커플, 그리고 이화여대 박혜진 연구원에게도 고맙다는 말을 전합니다. 로봇 프로젝트를 진행하면서 도움을 주셨던 대우조선해양 연구소의 김수호 이사님, 강성원 부장님, 윤호중 당시 과장님을 비롯하여 지원해 주신 많은 분들께도 진심으로 감사 드립니다. 그리고 현재 진행하고 있는 SBP 과제의 마스터이신 황호진 박사님, UVRC 과제에 이어 이 과제에서도 잠깐(?) 함께 했던 효광이형을 비롯하여 함께 과제에 도움을 주시는 많은 분들께 감사의 말씀을 드립니다.

마지막으로 정말 저를 아끼고 끝까지 믿고 지원해 준 가족들에게 감사의 말씀을 드립니다.

나의 아내, 숙희야... 부족한 남편을 만나서 여러 모로 힘들었지? 네가 작년 중순

병원을 다닐 때, 그 원인 중의 하나가 나왔기에 남몰래 몇 번 눈물을 훔쳤던 기억이 나는구나. 결혼 후 4 년이라는 시간이 지났지만 함께 있었던 시간은 턱없이 적기에 정말 미안한 마음뿐이네. 우리 둘을 보는 사람들이 신혼 같다고들 이야기 하지만, 그런 이야기를 들을 때마다, 사실 신혼 같은 신혼 생활을 함께하지 못하였기에 늘 미안한 마음뿐이었어. 늘 연구실 생활을 이유로 늦고, 약속을 안 지키는 남편이 마냥 밉고 그리웠을테니... 비록 박사 학위가 끝은 아니지만, 하나의 마침표를 찍고 다음으로 나갈 수 있기에 계획한 목표를 이루고 좋은 남편이 될 수 있도록 항상 노력할께.

아버지, 어머니... 계속 늦어지는 졸업에도 믿음을 잃지 않고 아들을 지원해 주셔서 정말 감사드립니다. 다른 자식들은 취직하고 첫 월급을 타는 동안, 아들은 33 년 인생동안 늘 받기만 하고 제대로 해 드린 것이 없네요. 박사 학위가 끝은 아니지만, 좋은 진로를 택하여 어머니, 아버지께서 자랑스러워 하실 만한 아들이 되도록 노력하겠습니다. 항상 덕망 있는 인품으로 존경 받으시는 아버지, 그리고 누구보다 활기찬 생활을 하고 계시는 어머니, 정말 존경합니다. 그리고 아련아, 오빠가 결혼하기 전, 오빠를 서포트 하느라 정말 고생 많았지? 요즘 회사일에 너무 치어 몸과 마음이 너무 힘들어 보여서 오빠는 걱정이구나. 회사 생활이 좀 더 나아져서 생활의 활력을 되찾았으면 좋겠구나.

물심양면으로 많은 지원을 해 주신 장인어른, 장모님! 정말 감사합니다. 늦어지는 졸업에도 아무 말 없이 기다려 주신 장인어른, 그리고 지난 1 년간 집에도 잘 안 들어가면서 연구실 생활을 할 동안 항상 시집 간 딸을 챙겨 주신 장모님께 진심으로 감사 드립니다. 이제는 한 가정의 아내가 된 민희, 처갓집의 공주 주희, 그리고 미국에서 열공 중인 동호까지, 저희 집과는 또 다른 분위기의 화기애애함과 에너지를 주어 항상 저를 웃게 만들어 주는 처갓집 식구들에게 감사의 말을 전합니다.

논문 제본을 인쇄소에 맡기기 전인 2013 년 2 월 18 일 새벽 1 시 54 분, 한결 같이 머무르던 서울대학교 34 동 308 호 연구실에서 박사 학위 논문의 마지막인 이 글을 남기며 박사 학위의 마침표를 찍는다.