T2R2 東京科学大学 リサーチリポジトリ
Science Tokyo Research Repository

論文 / 著書情報
Article / Book Information

題目(和文)	持続可能なプロセス情報管理を目指したエージェント指向運転・設計システムに関する研究
Title(English)	
著者(和文)	木村直樹
Author(English)	Naoki KIMURA
出典(和文)	学位:博士（工学）, 学位授与機関:東京工業大学, 報告番号:甲第6423号, 授与年月日:2006年3月26日, 学位の種別:課程博士, 審査員:黒田千秋
Citation(English)	Degree:Doctor of Engineering, Conferring organization: Tokyo Institute of Technology, Report number:甲第6423号, Conferred date:2006/3/26, Degree Type:Course doctor, Examiner:
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Powered by T2R2 (Science Tokyo Research Repository)

2005 年度　博士 (工学) 論文

持続可能なプロセス情報管理を目指した
エージェント指向運転•設計システムに関する研究

東京工業大学 大学院 理工学研究科 化学工学専攻

木村 直樹

目　　次

第 1 章　緒論

1.1 本研究の概要と既往の研究 1

1.1.1 マルチエージェント 1

1.1.2 製造業へのマルチエージェントシステムの適用 4

1.1.3 マイクロ化学プロセス 5

1.1.4 知識共有のためのモデル化 6

1.1.5 エージェントの協調型意思決定機構 8

1.2 本研究の目的と論文の構成 14

第 2 章　マルチエージェントを用いた運転操作シミュレーション

2.1 シミュレーションフレームワーク 15

2.1.1 装置エージェント 15

2.1.2 製品エージェント 16

2.1.3 組立エージェント 16

2.1.4 プロセスマネージャー 17

2.1.5 レシピ 17

2.1.6 通信ネットワーク 20

2.1.7 契約ネットプロトコルによる装置の獲得 25

2.1.8 階層化意思決定法の導入 27

2.2 マイクロ化学プロセス運転シミュレーション 43

2.2.1 シミュレーション対象および方法 43

2.2.2 AHP の評価基準 43

2.2.3 シミュレーション結果および考察 49

2.3 本章の結論 70

第 3 章　意思決定機構の高度化

3.1 AHP への確信度の導入 71

3.1.1 シミュレーション対象および方法 73

3.1.2 シミュレーション結果および考察 74

3.2 比較優位指標の導入 83
3.2.1 比較優位指標......83
3.2.2 シミュレーション対象および方法......85
3.2.3 シミュレーション結果および考察......85
3.3 本章の結論 94

第 4 章　データベースからのシミュレーションモデルの動的な取得
4.1 知識データベース 95
4.1.1 知識データベースの構造......96
4.1.2 知識の XML 化......97
4.2 デバイスのクラス化 107
4.2.1 デバイスの機能に基づくクラス化......107
4.2.2 デバイスの形状に基づくクラス化......108
4.3 シミュレーションモデルのクラス化 113
4.4 シミュレーションモデルの動的取得シミュレーション 116
4.4.1 シミュレーションモデル取得のための通信シーケンス......116
4.4.2 データベースエージェントの導入......119
4.4.3 シミュレーション対象および方法......119
4.4.4 シミュレーション結果および考察......119
4.5 本章の結論 125

第 5 章　運転を考慮した設計システム
5.1 シミュレーションモデル評価機構の導入 126
5.1.1 シミュレーションモデル評価機構......126
5.1.2 シミュレーションモデルの評価基準......128
5.2 シミュレーション 136
5.2.1 シミュレーション対象および方法......136
5.2.2 シミュレーション結果および考察......137
5.3 本章の結論 153

第 6 章　結論 154

Appendix A　UML **158**

1. UML クラス図 158

2. UML シーケンス図 159

3. UML コラボレーション図 159

Appendix B　比較優位説 **162**

Appendix C　管型反応器モデル **164**

1. 表型モデル 164

2. 関数型モデル 164

参考文献 **166**

Nomenclature **171**

謝辞 **174**

図　一　覧

Fig.1.1　キューブ型マイクロ化学プロセス 11
Fig.1.2　機能的知識・オントロジー階層および機能概念オントロジー 12
Fig.1.3　汎用機能分解木および達成方式知識の階層 13
Fig.2.1　装置エージェントの構造の概略 31
Fig.2.2　製品エージェントが装置エージェント間を動き回る様子 32
Fig.2.3　S88.01 における抽象度に基づくレシピの分類 33
Fig.2.4　S88.01 におけるレシピの階層構造 34
Fig.2.5　本研究で用いたレシピ構造と S88.01 に基づくレシピ構造の比較 35
Fig.2.6　XML 形式によるレシピの記述例 36
Fig.2.7　Black board 方式の通信の概略 37
Fig.2.8　hosts ファイルを用いたネットワークの概略 38
Fig.2.9　DNS を用いたネットワークの概略 39
Fig.2.10　ルーティングサーバを用いたネットワークの概略 40
Fig.2.11　契約ネットプロトコルに基づく通信シーケンス 41
Fig.2.12　AHP の構造 42
Fig.2.13　本研究で用いた AHP の構造 55
Fig.2.14　バスタブ曲線 56
Fig.2.15　m = 0.5, 1, 2 における信頼度 , 確率密度関数 , 故障率の経時変化 ... 57
Fig.2.16　ある装置における信頼度の経時変化 58
Fig.2.17　少品種生産シミュレーションにおける運転結果の一例 59
Fig.2.18　少品種生産シミュレーションにおける製品評価値の推移 60
Fig.2.19　少品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の推移 61
Fig.2.20　多品種生産シミュレーションにおける製品評価値の経時変化 62
Fig.2.21　多品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の経時変化 63
Fig.2.22　装置入れ替えシミュレーションにおける運転結果の一例 64
Fig.2.23　装置入れ替えシミュレーションにおける製品評価値の経時変化 65
Fig.2.24　運転評価を用いない AHP の構造 66
Fig.2.25　運転評価を用いた場合と用いない場合の
シミュレーションにおける運転結果の比較 67

Fig.2.26 運転評価を用いないシミュレーションにおける
製品評価値の経時変化 68
Fig.2.27 運転評価を用いないシミュレーションにおける
評価基準項目の重みおよび整合度の経時変化 69
Fig.3.1 少品種生産シミュレーションにおける
交渉条件の違いによる製品評価値の比較 79
Fig.3.2 確信度を用いた少品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の推移 80
Fig.3.3 多品種生産シミュレーションにおける
交渉条件の違いによる製品評価値の比較 81
Fig.3.4 確信度を用いた多品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の経時変化 82
Fig.3.5 複数の募集メッセージに対する入札の様子 91
Fig.3.6 装置エージェントの入札方法の違いによる運転結果の比較 92
Fig.3.7 募集順序の違いによる運転結果の比較 93
Fig.4.1 *MDCOs* 知識データベースの構造の概略 100
Fig.4.2 知識データの XML 構造 (`research` エレメント) 101
Fig.4.3 知識データの XML 構造 (`header` エレメント) 102
Fig.4.4 知識データの XML 構造 (`research-method` エレメント) 103
Fig.4.5 知識データの XML 構造 (`model` エレメント) 104
Fig.4.6 `model` ノードの XML 表現の一例 105
Fig.4.7 表形式に成形された知識データの一例 106
Fig.4.8 マイクロ化学デバイスの機能に基づくクラス図 110
Fig.4.9 化学装置便覧による化学装置の分類の一部 111
Fig.4.10 マイクロ化学デバイスの装置の幾何学的形状に基づくクラス図 112
Fig.4.11 シミュレーションモデルのクラス図 115
Fig.4.12 クラスファイル取得シーケンス 121
Fig.4.13 異なるファイルサイズおよびネットワーク環境における
クラスファイルロード時間の比較 122
Fig.4.14 データベースエージェントを用いた
シミュレーションモデル取得シーケンス 123
Fig.4.15 表型モデルを用いた種々の操作条件における転化率の推算結果 124

Fig.5.1 シミュレーションモデルの評価機構を導入した
シミュレーションモデル取得シーケンス........................132
Fig.5.2 シミュレーションモデル選択におけるコラボレーション図................133
Fig.5.3 シミュレーションモデル評価のための AHP の構造..........................134
Fig.5.4 計算精度評価の概略..135
Fig.5.5 Clariant Process における反応経路..146
Fig.5.6 モデル選択シミュレーション I における転化率の推算結果..............147
Fig.5.7 種々の運転操作条件における計算精度評価値の変化.......................148
Fig.5.8 表型モデルと関数型モデル B における計算精度評価値の比較........149
Fig.5.9 種々の運転操作条件における
分布関数を用いた計算精度評価値の変化....................150
Fig.5.10 モデル選択シミュレーション II における転化率の推算結果...........151
Fig.5.11 装置形状の検討におけるシミュレーション結果の一例...................152
Fig.A-1 UML クラス図..160
Fig.A-2 UML シーケンス図...160
Fig.A-3 UML コラボレーション図.. 161
Fig.C-1 管型反応器モデルの概略..164

表　一　覧

Table 2.1　AHP の一対比較値の評価尺度 30
Table 2.2　AHP の一対比較値の一例 30
Table 2.3　AHP の重みの一例 50
Table 3.1　組立エージェントが持つ一対比較値 78
Table 3.2　組立エージェントが持つ確信度の値 78
Table 3.3　製品エージェントが持つ一対比較値 78
Table 3.4　製品エージェントが持つ確信度の値 78
Table 3.5　各装置エージェントの各製品についての処理速度 86
Table 3.6　各装置エージェントの各製品についての処理時間 86
Table 3.7　2 装置—2 製品における運転結果の組み合わせ 87
Table 3.8　各装置エージェントの各製品についての処理速度 88
Table 3.9　各装置エージェントの各製品についての処理時間 88
Table 3.10　各装置エージェントの各製品についての比較優位指標 88
Table 5.1　アルゴリズムの一般的な評価尺度 131
Table 5.2　シミュレーションモデル 136
Table 5.3　異なるシミュレーション目的における
シミュレーションモデル評価の一例 140
Table 5.4　種々の装置形状が設定された装置エージェント 142
Table 5.5　各装置エージェントの性能推算値の一例 145
Table 5.6　製品エージェントによる入札の評価値の一例 145
Table B-1　2 国 2 財モデルにおける労働投入量 162
Table B-2　2 国 2 財モデルにおける自給自足による生産量 163
Table B-3　2 国 2 財モデルにおける比較生産費 163
Table B-4　2 国 2 財モデルにおける分業による生産量 163

第 1 章
緒　　　論

本研究では、化学プロセスの運転•設計システムを対象としてマルチエージェントを用いた効率的で持続可能な情報管理を目指して研究を行なった。具体的にはマイクロ化学プロセスを対象とし、プロセス運転における評価機構について検討し、マルチエージェント環境における協調的な動作の発現を試みた。また、知識データベースシステムを構築し、データベースとマルチエージェントシステムを接続し、動的にシミュレーションモデルを取得することにより柔軟なシミュレーションシステムの構築を試みた。また、プロセスの運転を考慮した設計を行なうためにシミュレーションモデルの評価機構を検討した。

本章では、研究の概要と関連する既往の研究について 1.1 節で述べ、本研究の目的および論文構成を 1.2 節で述べる。

1.1 本研究の概要と既往の研究

1.1.1 マルチエージェント

エージェントとは、環境を ‘センサー’ で知覚し、環境に対して ‘エフェクター’ を通して動作するものである。エージェントがどのように動作するかは、センサーによって知覚した情報から動作へのマッピングによって決定される。従って、あら

かじめ組み込まれた‘知覚—動作対応表’を用いてマッピングを実現すれば、単純なエージェントを実現することが出来る。しかしこのようなエージェントは“正しい”又は“合理的な”動作をとることは出来るが、知的であるとは言えない。知的エージェントが満たす条件として“自律性”が挙げられる。自律性とはエージェントの「行動が自分自身の経験に基づいてなされる [1]」ことであるが、このような厳密な自律性のみでは、エージェントが‘正しい’動作を選択することは出来ない。従って、知的なエージェントは、環境に合わせてあらかじめ組み込まれた知識とエージェント自身の経験の両者に基づいて、合理的な動作を選択する必要がある。

このような合理的な動作を行なうエージェントがある一方で、コンピュータ上に人工の社会を構築し、社会における主体としてエージェントを多数 (マルチ) 用いて、エージェント間の相互作用により社会現象を説明しようという試みがなされてきた。そのような社会科学的なマルチエージェントシミュレーションにおいては、エージェントの内部モデルは KISS(Keep It Simple, Stupid) 原則と呼ばれるモデル構築方法が採られることが多い。これは「単純で視野の狭いミクロの主体から、何らかの秩序あるマクロ的なパターンを生み出す [2]」ことを目的としているためである。一方で、“エージェント”の本来の意味である「代理人」という立場から見ると、依頼主の要求をかなえるためにエージェントが自律的に行動する必要がある。すなわち、依頼主からは行動方針 (戦略) が与えられるが、個別かつ具体的な方法 (戦術) が指示されるわけではなく、エージェントが主体的に戦術を選択する自由が与えられている。‘代理人’としてのエージェントは、情報収集のためのモバイルエージェント [3] や、オークションの入札代行エージェント [4] が提案されている。

エージェントが主体的に戦術を選択し、‘正しい’行動を行なうために、知覚から動作へのマッピング、すなわち意思決定機構が重要となる。単純なマッピング

の例として先にも挙げた‘知覚—動作対応表’を用いた場合、項目数が多くなると、それらの項目を評価•選択する時間的•メモリ空間的な限界が生じたり、自律性を持たせにくいといった欠点がある。そこで、マルチエージェントシステムにおけるより高度な意思決定機構として、種々の方法が検討されている。一般的には、エージェントに効用関数を持たせ、効用の値を最大化するように行動を選択する方法が用いられる。また、エージェントに報酬を与えることにより、効用関数を学習させる‘強化学習’を用いたエージェント型の動的スケジューラ [5] や‘GA(遺伝的アルゴリズム)’を用いたスケジューラ [6] などが提案されている。また、‘AHP(階層化意思決定法)’[7] を用いた旅行先決定支援システム [8] やバッチプロセスの運転管理 [9–12] や、強化学習エージェントに AHP を導入し、より適切な行動が選択されるようにエージェントの行動を代替案とみなして重み付けを行なう方法 [13] も提案されている。

本研究では、装置エージェントの評価•選択やシミュレーションモデルの評価•選択に AHP を用いて意思決定を行なうこととする。AHP を用いた研究や解説書の多くでは、代替案の評価値を相対評価 (Relative Measurement) 法を用いて決めることが多い。しかし、代替案の相対評価は、代替案が追加された場合に一対比較のやり直しや、評価順位の逆転が起こったり、代替案数の増加に伴って一対比較が困難になったり、整合性が悪くなる [14] ことが指摘されている。そこで、本研究では、絶対評価 (Absolute Measurement) によって代替案の評価値を決定することとした。装置エージェントの評価•選択についての詳細は第 2 章で、シミュレーションモデルの評価•選択についての詳細は第 5 章で述べることとする。

1.1.2 製造業へのマルチエージェントシステムの適用

このようなマルチエージェントシステムを製造業へ適用する試みがなされている。特にマニュファクチャリングプロセスにおいては種々の試みがなされ、実際の加工装置に装置エージェントを搭載し、生産物 (製品パレット) に製品エージェントを対応させた自律的な加工装置制御システム [15–17] が試作されている。また、加工装置を効率的に割り当てるために、‘Scheduling Agent’ および ‘Optimasation Agent’ により、生産コスト、生産時間、機械の信頼性、ボトルネックの 4 つの評価項目を用いた運転計画の最適化 [18–19] や ‘Task Agent’ とすべての装置を一括して管理する‘Resource Manager’との交渉により動的にスケジューリングを行ない、コンフリクト解消のために再交渉を行なうシステム [20–22] などが提案されている。

また、特定の企業や工場における生産•運転計画の効率化•最適化のみならず、製品の原料から消費者までのサプライチェーン全体に渡って情報を共有しあい、モノやお金の流れを速くすることで効率的な生産や経営を目指したサプライチェーンマネージメントが研究されている。しかし、一般に化学工業におけるサプライチェーンは長大で厳密なプラント情報の取り扱いが要求される。そのため、他の製造業のサプライチェーンマネージメントに用いられた方法論をそのまま化学工業に適用することは難しい [23]。石油精製プロセスにおいて、原油価格や原油の品質、市場における製品価格や要求仕様を考慮したエージェントベースのサプライチェーンシステム [24] や化学物質に関する情報の精密な交換のために STEP(Standard for the Exchange of Product Model Data) を用いてエージェント同士の連携を容易にし、サプライチェーン上の各段階 (企業や工場) に相当するエージェントが、サプライチェーン全体の状況を把握したり流通性を高めるシステム [25] が提案されている。化学工業において種々の製品を生産する際に、化学物質の

情報を扱うことが難しいという問題点が挙げられる。産地により原料の組成に違いがあったり、製品のカットポイントなどの条件により製品の品質や量に違いが出来る [24] ため、それらの情報を精密に取り扱う必要がある。また、多品種を生産するためには装置同士を接続する配管の問題などが生じる。そこで、ロボット反応器と知的な作業ステーションを用いた自律分散型のパイプレスバッチプラント [9–12] や小型モジュール化した化学装置に分散型の制御機能を持たせ、それらのモジュールを組合わせる方法 [26] が提案されている。

本研究では、種々の化学製品を生産する化学プロセスを対象とし、種々の装置や製品が混在するプロセスにおける柔軟かつ効率的な運転操作を検討する。その詳細については第 2 章で述べることとする。また、種々の製品や装置における運転操作を考慮しつつ、プロセス設計を行なうシステムを検討する。その詳細は第 5 章で述べることとする。

1.1.3 マイクロ化学プロセス

環境や化学物質に対する社会的関心の高まりを受けて、精密かつクリーンに生産することが化学工業に求められている。微細加工技術や微量分析技術の発達と相まって化学プロセスの微小化が可能となった。マイクロ化学デバイスでは熱効率が高く、温度制御を敏速かつ精密に行なうことが出来るため、反応収率 [27] や反応選択率 [28–30] 向上の効果や触媒量減量 [31] を図ることが出来る。また、トルエンの直接フッ化 [32] やニトロベンゼンからアニリンへの還元反応 [33] などの発熱量が多く、暴走しやすい反応にも用いられ、発熱量の多い反応系でも十分に熱交換が可能な反応器 [34] なども研究されている。また、微小化により副反応が進行しない程度に電極電位を抑えることが出来るため電気化学反応 [35–37]

にも有用視されている。

マイクロ化学プロセスを実現するためにマイクロ化学デバイス同士を接続する方法として、積層型のマイクロミキサーモジュールにキャピラリ接続用のレセプターを組み込んだり [38]、嚙み合わせ型の接合方法 [39] が提案されている。また、モジュール化したマイクロ化学デバイスも数多く提案され、六角形積層型モジュールの FAMOS(Fraunhofer Alliance Modular Microreaction Systems) や立方体もしくは直方体モジュールを組合わせて反応プロセスを組み立てる 'Modular Miroreaction System' (Ehrfeld Mikrotechnik BTS 社) などが提案されている。

本研究では、Ehrfeld 社の 'Modular Miroreaction System' を参考にして、キューブ状のマイクロ化学プロセスを対象とする。Fig.1.1 にイメージ図を示す。マイクロリアクターやマイクロミキサーなどの単位操作ごとにモジュール化したマイクロ化学デバイスを台座上に組み合わせることにより化学プラントを構築することとした。

1.1.4 知識共有のためのモデル化

種々の分野の研究•開発において、知識の共有および再利用が注目されている。知識を共有するための試みの一つとして、設計作業における知識操作を記録することにより、設計文書へ自動的に設計記述を書き込んだり [40]、設計者にコメント入力を促すことによって、設計知識の獲得を行なうシステム [41] が提案されている。また、化学プロセス設計者とエージェントシステムとの対話により、プロセスフローシートを生成する設計支援システムでは、設計知識をオブジェクト化することにより知識のモジュール化と再利用性を高め、代替案の生成や設計案の決定における簡便性が高められた [42]。

シミュレーションモデルを入れ替えてシミュレーションを行なう方法も提案され

ている。定性モデルを扱うメタモデル機構の変更を伴わずに設計対象モデラの組み込みが可能なプラガブル・メタモデル機構 [43] や、シミュレーションモデルをリレーショナルデータベースに格納し、'SimSQL' 言語を用いてシミュレーションモデルを取り出して、シミュレーションに利用するシステム [44] が提案されている。しかし、'SimSQL' 言語で取り出せるモデルは最大で 3 つの整数型変数、3 つの小数型変数、2 つの文字列型のパラメータからなるモデルであった。このようなパラメータだけのモデルではなく、種々のデータを交換するために様々な方法が検討されている。物質を扱うモデルを 'substance'、'phase system'、'phase system mathematical model' の 3 つに分類してクラス化を行ない、化学プロセスのシミュレーションレベルからサプライチェーンマネージメントのレベルまでデータ交換が可能な一貫したモデル [45] や金属材料のデータを XML ベースの 'MatML' を用いて記述し、データベースに蓄積することで、web ブラウザや種々のアプリケーションで利用すること [46] が提案されている。

知識の共有や再利用のバックボーンとして、オントロジー [47] が盛んに研究されている。知識処理分野におけるオントロジーとは、「知識システムを構築する際に用いられる基本概念の体系的記述 [48]」のことであり、種々の概念を表現するために 5 層 (対象固有層、視点依存知識層、一般知識層、一般概念層、基盤概念層) からなる '機能的知識•オントロジー階層 (Fig.1.2(a))' が提案されている。対象の機能モデルを表現するために、機能概念オントロジー (Fig.1.2(b)) や機能分解木 (Fig.1.3(a)) や機能と物理原理をつなぐ知識として、達成方式知識 (Fig.1.3(b)) を構築するなど対象物の機能の分解および汎化を行なうことによって、知識の基本概念を記述する方法 [49–50] を提案している。また、「人工物のモデル化に関してデバイス中心の見方とプロセス中心の見方の 2 つの見方が存在 [51]」するとして

いる。人工物の工学的モデリングにおいては、デバイスオントロジーが盛んに用いられてきたと述べているが、ここで言うデバイスとは「入力を処理して利用者が要求するものに変換して出力する」ものであり、プロセスとは「自然現象もしくは自然法則」を表しているため、本研究で対象とするマイクロ化学プロセスにおける 'デバイス' や 'プロセス' とは概念の階層が異なっている。デバイスオントロジーは、デバイスの果たす役割、すなわち対象物に対する機能の視点でとらえたものである。

マイクロ化学プロセスにおけるデバイスの機能は化学工学における単位操作の概念によって記述することが出来る。しかし、前項で述べたようにモジュール化•規格化されたマイクロ化学プロセスが数多く提案されているため、それらのマイクロ化学デバイスにおける運転を考慮した設計を行なうためには、デバイスの機能性のみではなく、形状も考慮する必要がある。そこで、本研究ではデバイスの機能と形状の 2 つの視点からクラス化を検討することとした。詳細は第 4 章で記述する。

また、マイクロ化学プロセス技術の知識共有のために知識データベースを構築することとした。知識データは知識の再利用性を高めるため XML で記述し、データベースから動的にシミュレーションモデルを取得する方法の詳細を第 4 章で述べる。

1.1.5 エージェントの協調型意思決定機構

マルチエージェント環境における問題解決の際に、エージェントの振る舞いが利己的であるか、協調的であるかという点が注目される。協調的なエージェントの振る舞いは、様々な研究がなされており、オークションサイトごとに設置したビッ

ダーエージェント間の情報共有により協調的に入札を行なったり [4]、‘パーソナルレポジトリ’ に蓄積された情報を管理するエージェント同士の情報交換により、検索対象と検索の幅を広げる効果が得られる協調情報検索 [52] などが行なわれている。また、具体的な問題解決の手段として、パーツ設計と金型製作のプロセスに、マルチエージェントを用いた協調的な設計機構を導入することにより、コンフリクトの検出と解消を行なうシステム [53] や化学プラントの配管問題において ‘plot layout agent’、‘site layout agent’、‘maintenance/safety agent’ の 3 者間で設計案を共有することにより、パレート最適な設計を行なうシステム [54] が提案されている。

一方で、社会科学の分野においては、限られた資源に対して、エージェントが協調的もしくは利己的な振る舞いを行なうかを調べ、個々のエージェントの意思決定が環境全体に及ぼす影響が調べられている [55–56]。マルチエージェント環境では「個々が利益を追求すればするほど、実際に得られる利益が減少するため、個々は自己の利益を追求せずに放棄することが求められる [57]」という ‘社会的ジレンマ’ と呼ばれる状態に陥る。そこでは、権威者や上位からの制御により社会的ジレンマを解決する研究が行なわれている。同一のトピックを扱う情報検索エージェント同士がエージェントコミュニティを形成し、他のエージェントからの検索要求に多く答えられるエージェントほど自分自身の検索効率が向上するという「give and take」効果に基づく協調 [58] は、協調を行なうほど結果として自己の利益になるという動機付けを用いた例である。また、強化学習エージェントが環境の状況を識別し、状況に応じて自己の利益を放棄し得るエージェント [57] も提案されている。これらの方法はエージェント相互の情報交換により、自己もしくは環境全体に対して利得の大きい行動を選択するものである。

一般には「マルチエージェント環境において協調行動を引き出すためには、情

報の交換やネゴシエーションなどのコミュニケーションが重要 [59]」であると言われている。これらのコミュニケーションなくして、協調行動の発現はあり得ないのであろうかという疑問に対し、Sen らは他のエージェントの状態を知らなくとも、エージェントが協調的な行動をとることが出来る [60] ことを指摘している。また、それによって、コミュニケーション遅れや他のエージェントから受け取った情報の信頼性の問題、上位のエージェントなどの影響を排除できると述べた。なお、これらのコミュニケーション無しでの協調行動を発現させるためには、エージェント間での「目的」と「報酬関数」を共有することが条件となる。本研究では、エージェント間の利己的•自発的な行動による協調効果として、プロセス全体の生産効率の向上を目指す。「コミュニケーションを介さないでも協調行動を生み出すことができれば、適用分野が広くなる [59]」ことは言うまでもない。罰や将来の利益を見越した譲歩に基づいて行動した結果の協調効果ではなく、エージェントが利己的な行動をとった結果として協調効果を発現させるために、比較優位指標の導入を検討することとした。その詳細は第 3 章で述べる。

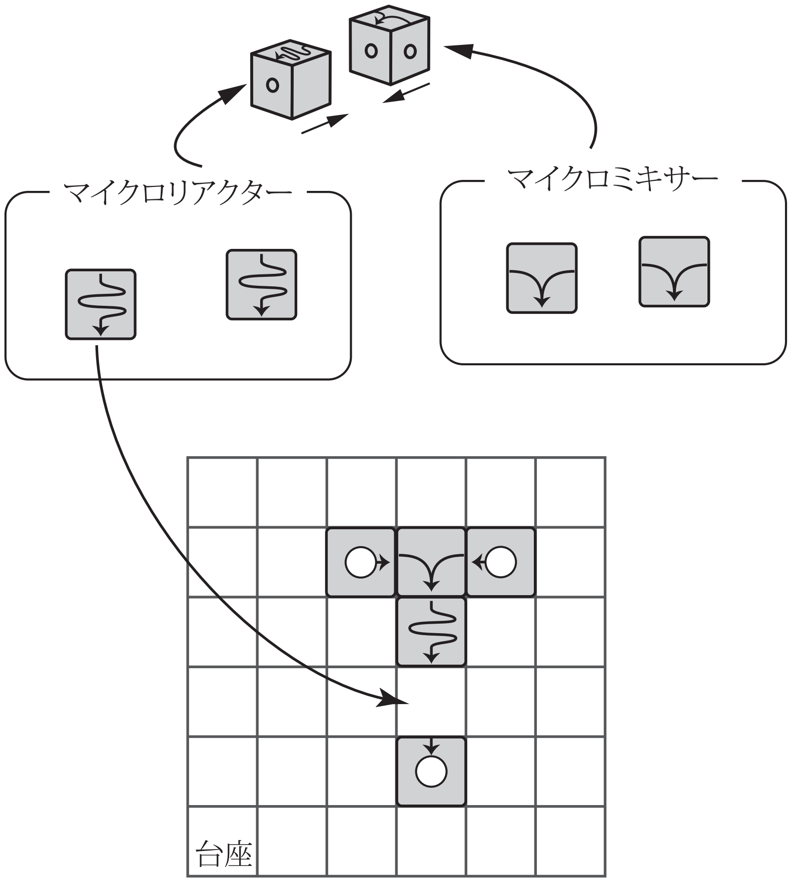
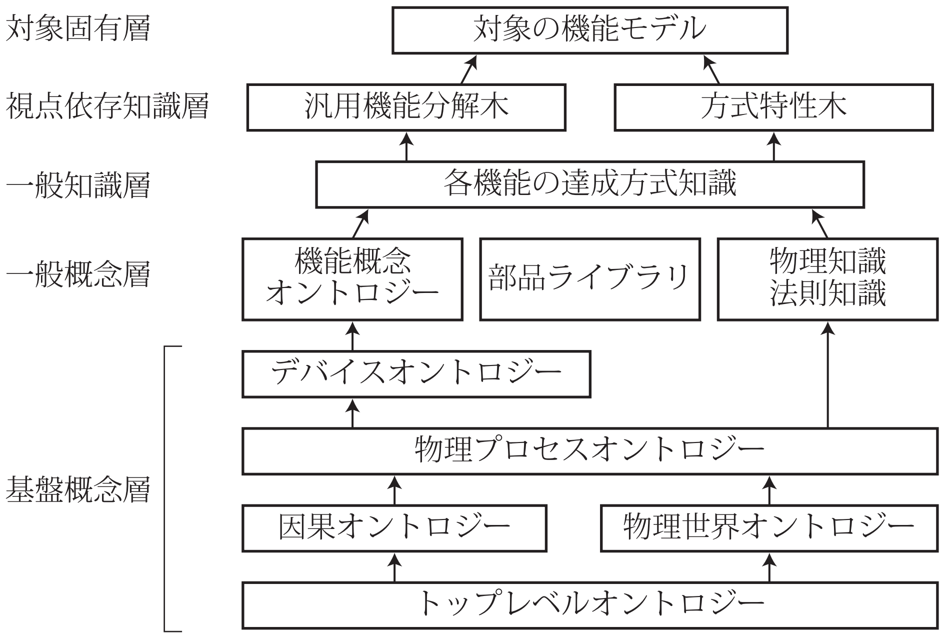
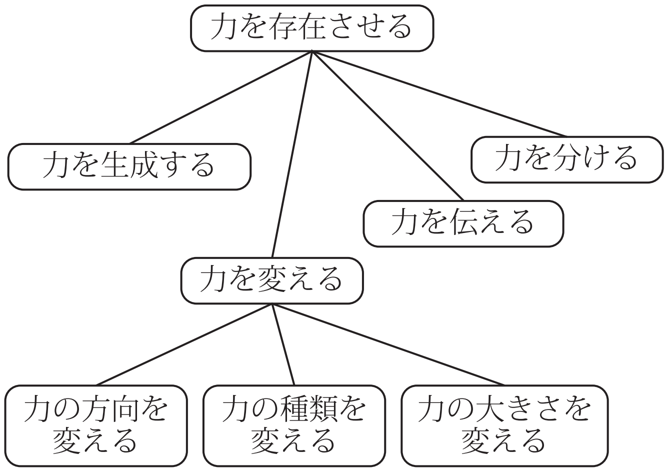


Fig.1.1　キューブ型マイクロ化学プロセス

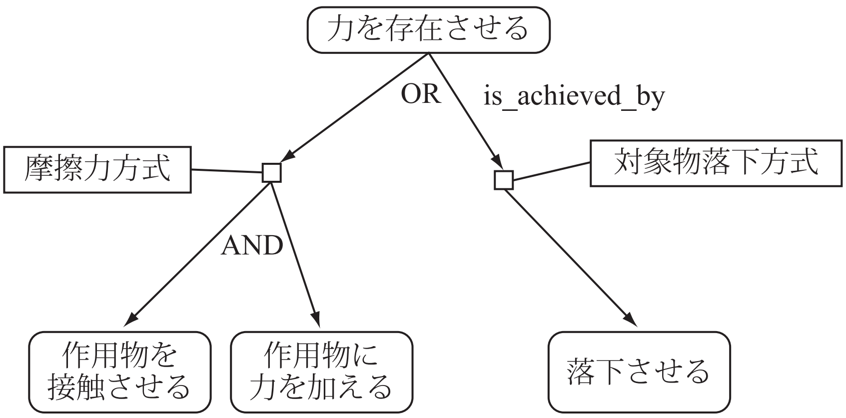


(a) 機能的知識・オントロジー階層

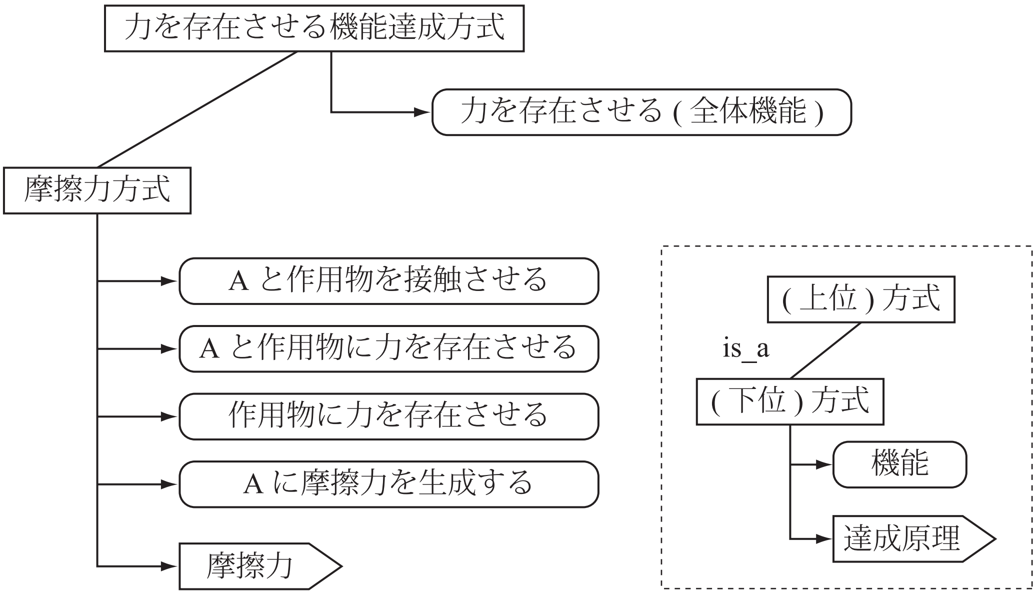


(b) 機能概念オントロジー

Fig.1.2　機能的知識・オントロジー階層および機能概念オントロジー



(a) 汎用機能分解木



(b) 達成方式知識の階層

Fig.1.3　汎用機能分解木および達成方式知識の階層

1.2 本研究の目的と論文の構成

マイクロ化学プロセスにおける持続可能なプロセス情報管理を目指し、マルチエージェントを用いた運転•設計システムの構築を目的として、マルチエージェント環境におけるプロセス情報やシミュレーションモデルの管理、評価および意思決定法について検討した。本論文は、6 章より構成される。

第 1 章「緒論」では、既往の研究および本研究の目的について述べる。

第 2 章「マルチエージェントを用いた運転操作シミュレーション」では、マイクロ化学プロセスにおけるマルチエージェントを用いた運転管理システムについて述べる。

第 3 章「意思決定機構の高度化」では、2 章で構築したマルチエージェントを用いた運転操作シミュレーションシステムの意思決定機構に確信度や比較優位指標を導入し、交渉や協調における高度な意思決定機構を提案する。

第 4 章「データベースからのシミュレーションモデルの動的な取得」では、マイクロ化学プロセスのマルチエージェントを用いたプロセス設計システムに関して、マイクロデバイスの性能やプロセス挙動を推算するためのシミュレーションモデルを共有したり、検索、管理するためのデータベースシステムを構築し、エージェントがデータベースから動的にシミュレーションモデルを取得する方法を提案する。

第 5 章「運転を考慮した設計システム」では、4 章で述べたエージェントの動的なシミュレーションモデル取得機構に、シミュレーションモデルの評価機構を導入し、プロセスの運転操作を考慮しながらマイクロ化学プロセスのプロセス設計を行なうシステムを提案する。

第 6 章「結論」では得られた成果をまとめ、本研究の結論を述べる。

第 2 章
マルチエージェントを用いた運転操作シミュレーション

プロセスを構成する装置や製品などの要素数や種類の増加に伴って、大規模かつ複雑化するマイクロ化学プロセスにおいて、飛躍的に増大するプロセスの情報を柔軟かつ効率的に管理する必要がある。そこで、本章では、マルチエージェントを用いた運転操作を検討する。

2.1 シミュレーションフレームワーク

2.1.1 装置エージェント

装置モジュールごとに実装されるエージェントを "装置エージェント (Equipment agent)" と呼ぶ。装置の作業履歴や仕事の処理を管理するエージェントである。各装置モジュールには、マイクロチップが埋め込まれ、そのマイクロチップ上に装置エージェントがインストールされることを想定している。装置エージェントは、情報交換のための通信インターフェースと、作業機構である装置の制御系とのインターフェース、およびエージェントの核となる意思決定機構からなる (Fig.2.1)。また、作業スケジュールおよび作業履歴を保持している。履歴から判断して、自己の装置の消耗が激しい場合には使用を中止することが出来る。本研究においては、2.2.2 項で述べる信頼度を用いて自己の消耗具合を判断することとした。

装置エージェントは、次項 (2.1.2) で述べる製品エージェントから仕事の募集を受け、装置の機能や作業スケジュールを考慮しながら入札を行なう。2.1.7 項に述べる契約ネットプロトコルに従って仕事の契約が成立すると、製品エージェントから担当する作業範囲のレシピ (2.1.5 項で述べる) を受け取る。作業開始から終了までの間、装置エージェントは随時、製品エージェントやプロセスマネージャー (2.1.4 項で述べる) に対して、作業の進捗状況の報告を行なう。

2.1.2 製品エージェント

レシピを保持して製品の作業工程の進捗状況を管理するエージェントを “製品エージェント (Product agent)” と呼ぶ。プロセスマネージャーにより、製品の注文ごとに生成される。製品の生産量や処理手順を記述したレシピを保持して、装置エージェント間を動き回るモバイルエージェント (Fig.2.2) である。レシピに従って装置を獲得し、獲得した装置エージェントに移動してレシピのコピーを渡し、処理を依頼する。同様の手順で必要なすべての装置を獲得する。生産終了後は、プロセスマネージャーに移動して生産終了を報告する。

2.1.3 組立エージェント

装置モジュールの組み立て、分解、洗浄・検査を管理するエージェントを “組立エージェント (Assembly agent)” と呼ぶ。

装置モジュールの入札メッセージは、一旦、組立エージェントに渡され、その装置を組み込んだ場合に要する組み立て時間やコストなどの情報を入札メッセージに付加して、製品エージェントに転送する。製品エージェントと装置エージェントが契約すると、装置エージェントから組み立て依頼が来るのでそれに従って、複数

の装置モジュールを接合し組み立てる。生産が終わると、組み立てた装置モジュールを分解し、洗浄・検査をして、再利用するための保管スペースに移動させる。

2.1.4 プロセスマネージャー

人間のオペレーターとシステムとのインターフェースとなるエージェントを "プロセスマネージャー (Process Manager)" と呼ぶ。製品の注文を受けて製品エージェントを生成したり、装置エージェントや製品エージェントから送られてくる作業の進捗状況をリアルタイムでモニタリングし、それらの情報を保存しておくことが可能である。

2.1.5 レシピ

レシピ (Recipe) は、製品の処理手順を記述したものである。ISA-S88.01[61] は、レシピを General Recipe, Site Recipe, Master Recipe, Control Recipe に分類した。これらの 4 分類を ER 図で記述したものを Fig.2.3 に示す。General Recipe はある企業における製品固有の処理手順を記述したものである。Site Recipe は、ある企業において複数の生産拠点を持つ場合に、General Recipe に対して特定の生産拠点 (site) 向けに具体的な環境要因 (e.g. 気温や湿度) や原料物質の地域依存性 (e.g. 水質) などを加味したものである。Master Recipe は、特定のプロセスセル向けの要因を加味したものである。プロセスセル (Process cell) とは複数の単位操作装置からなり一つまたはそれ以上の製品を製造することができる装置群であり、ある生産拠点内には一つ以上のプロセスセルが存在する。規模の小さい企業では、ある製品の生産拠点を一つしか持たない場合があるが、その場合には Site Recipe が省略され、General Recipe にプロセスセルを構成する装置群の特性

や制約などを加味して Master Recipe を生成する。Control Recipe はプロセスセル内で実際に使用する装置情報やバッチ ID、生産量などを記載した具体的なレシピである。これらの 4 分類はレシピの抽象度に基づく分類である。一方、レシピは構造的にも分類される。Procedure, Unit Procedure, Operation, Phase に分けられ、これらが階層的な構造を持ってレシピを構成する。レシピの階層的構造を Fig.2.4 に示す。Procedure は最上位の要素で 1 つの製品の処理工程全体に対応し、1 つ以上の Unit Procedure とその処理順序の情報を持つ。Unit Procedure は単位操作に相当し 1 つ以上の Operation とその処理順序の情報を持つ。Operation は 1 つ以上の Phase とその処理順序の情報からなり、単位操作における大まかな処理手順である。Phase は処理手順の最小の単位であり、具体的な制御系の操作などを表す。

本研究で用いたレシピ構造の一例を Fig.2.5 (a) に示し、S88.01 で定義されたレシピの構造を比較のために Fig.2.5 (b) に示す。概ね S88.01 に準拠した構造を持たせたことがわかる。本研究におけるレシピは、1 つの作業モジュールで処理できる作業範囲に対応したサブレシピに分割される (S88.01 における Unit Procedure に相当する)。レシピのヘッダー部分には、製品名、ロット番号、生産量が記述され、複数のサブレシピとそれらの処理順序の情報を持つ。サブレシピには、そのサブレシピを処理可能な装置のグループ名、処理時間および装置への負荷を計算するためのパラメータが記述されている。

既往の研究でも、種々のレシピ記述方法が提案されている。Control Recipe の記述において黒田らはプログラミング言語に似た構造をもつテキスト形式で記述した [10–13]。和田らは実験室で従来から使用されていたテキスト形式による記述方法にあわせて加工機械制御システムを導入した [15–17]。一般にテキスト形式によ

る記述は、人間の可読性を重視しており、コンピュータで処理する場合には手間と時間がかかる。一方、コンピュータでの処理が簡便かつ高速に行なうことが出来るバイナリ型式の場合には、人間は適当な支援ツール (アプリケーション) がなければ内容を理解することが非常に困難である。そこで、本研究では人間のオペレータとコンピュータ双方の可読性と処理のしやすさを持った XML(eXtensible Markup Language) 型式で記述することとした。XML 型式で記述したレシピの一例を Fig.2.6 に示す。この図は Fig.2.5 (a) と同じレシピを示している。XML では、< >で囲まれた要素 (e.g. <recipe>) に属性 (attribute) を持たせることが出来る (e.g. name, quantity)。また要素の下に要素を持たせ、階層的なデータを記述することが出来る。さらに、要素名や属性名はそのデータの内容にあわせて拡張することが可能である。XML は Internet や種々のアプリケーション間のデータ交換用のデータ記述言語としてデファクトスタンダートとなりつつある。SVG などの 2D 画像表示や 3D などの画像処理系、XHTML や XSLT などのページ記述言語や手続記述言語、報道情報のための NewsML など 1998 年に W3C(World Wide Web Consortium) により XML がリリースされて以降、様々な分野へ応用されている。化学工業においても S88.02 で定義されたデータ構造に準拠して、レシピ情報やバッチスケジュール / 履歴情報の交換のために、種々の方法が検討されてきた。Exchange Table を用いてデータ交換を行なったり、テキスト形式や ISO 10303(STEP/Express) 型式、SGML(Standard Generalized Markup Language) を用いた方法等が検討され、1995 年頃から RDBMS(Relational DataBase Management Systems) を用いた方法が主流となりつつあった。しかし、RDBMS はデータ保存には適した方法であったが、異なるシステム間でデータ交換を行なうためにサードパーティ製の変換モジュールが必要となりコストがかかるなどの問

題点があった。そこで、2002 年 4 月に Master Recipe、バッチスケジュール、バッチ履歴を交換するために WBF(World Batch Forum) において、BatchML(Batch Markup Language) バージョン 1 がリリースされ、 2003 年 9 月にバージョン 2[62] がリリースされた。本研究で用いたレシピの XML 記述方法は BatchML に準拠しているわけではないが、BatchML 型式に変換することによりデータ交換を行なうことが可能である。

2.1.6 通信ネットワーク

(1) エージェント間コミュニケーション

　マルチエージェントシステムにおいて、エージェント間のメッセージング方法として種々の方法が検討されている。同一のコンピュータ上に複数のエージェントが存在する場合、パイプ (又はパイプライン) を用いてプロセス間通信を行なう方法も用いられるが、エージェントが複数のコンピュータ上に物理的に分散している場合はこの方法を用いることが出来ない。

　ブラックボード (黒板) 方式は、特定のファイルやファイルサーバに “掲示板” を用意し、エージェントは他のエージェント宛のメッセージを掲示板に書き込むことでメッセージの授受を行なう方法であり、ファイルやファイルサーバにすべてのエージェントからアクセス出来れば、エージェントが物理的に分散している場合でも可能である。前述のパイプを用いたプロセス間通信のうち、共有メモリ領域を用いた “名前付きパイプ” はブラックボード方式の 1 種とみなすことが出来る。Fig.2.7 にブラックボード方式による通信の様子を示す。この方法ではエージェントは一定時間間隔でブラックボードを閲覧し、自分宛のメッセージの有無をチェックしなければならないため、エージェント数の増加に伴ってファイ

ル又はファイルサーバへのアクセスが集中してしまう問題がある。

ブロードキャスト方式は TCP(Transmission Control Protocol) 又は UDP(User Datagram Protocol) を用いたコネクションレス型通信方式である。コネクションレス型とは、相手 (受信側) が受信待機状態にあるものとみなして一方的に送信を行なう方法である。特定のネットワーク内の全てのノード (コンピュータなど) に同じメッセージが配布されるため、そのネットワーク空間にエージェントを接続すればメッセージの送受信が可能になるという利点がある。しかし、ブロードキャスト方式では特定のポート宛にメッセージが配信されるため、エージェントは所定のポートを監視 (受信待機) する必要がある。しかし、同一のコンピュータ上の複数のエージェントが同一のポートを監視することは出来ない (コンフリクトする) ため、2 つ以上のエージェントを 1 台のコンピュータ (ノード) 上で起動することが出来ないという欠点がある。実際に多数の装置モジュールを用意しそれぞれにマイクロチップを搭載した場合にはブロードキャスト方式が有力だと思われるが、本研究では 2 〜 3 台のコンピュータ上で 4 種類以上のエージェントを起動するためブロードキャスト方式は不向きである。

(2) 名前解決

これに対して www などのサーバ—クライアント型の通信やピアトゥピア (Peer to Peer) 型の通信もマルチエージェントシステムで用いられている。しかし、これらの通信方式の場合、“名前解決” と呼ばれる問題がつきまとう。小規模なネットワークの場合には、各マシンにホスト名と IP アドレスの対応表 (一般的には `hosts` ファイルと呼ばれる) を置き名前解決を行う場合がある。hosts ファイルを用いたネットワークの様子を Fig.2.8 に示す。しかし、ネットワーク上のノード

数が増えたり、対応表が頻繁に変更される場合には、各マシン上の対応表を同期させたり更新•管理が煩雑になってしまう。

そこで、現在、インターネットの世界では FQDN(Fully Qualified Domain Name) と呼ばれるホスト名 (e.g. www.chemeng.titech.ac.jp) と IP アドレス (e.g. 131.112.125.60) を対応させるために、DNS(Domain Name System) を用いて名前解決を行なっている。IPアドレスは無意味な数字の羅列であり覚えにくい、また、マシンの交換や物理的な場所の移動など様々な理由で IP アドレスが変更される場合があるが、FQDN と呼ばれる分かりやすいホスト名を設定し、DNS を使用して名前解決 (FQDN から IP アドレスに変換する) を図ることによって、クライアントユーザにマシンの物理的な所在や IP アドレスを意識させることなく、インターネットへの容易なアクセスを可能にしている。Fig.2.9 に DNS サーバを用いた通信の概要を示す。ドメインと呼ばれる実際の組織 (国や企業、部門など) と関連づけられたグループごとに DNS サーバが存在する。それらはルートネームサーバ (世界に 13 台ほど設置されているトップレベルの DNS サーバ) を頂点として階層的に構成されている。クライアントマシンから ‘ドメイン B’ 上の ‘ホスト A’ にアクセスしたい場合、クライアントマシンが所属するネットワークの DNS サーバに名前解決要求を出す。名前解決要求を受けた DNS サーバは、階層的な DNS サーバのネットワーク内を走査して ‘ドメイン B’ を管理する DNS サーバ検索し、‘ホスト A’ の IP アドレスを問い合わせ、クライアントマシンに返す。IP アドレスを取得したクライアントマシンは直接、‘ホスト A’ にアクセスすることが出来る。このようなシステムを用いると、クライアントマシンでは、所属ドメインの DNS サーバの情報を設定するだけでインターネットに接続されたすべてのホストの所在 (IP アドレス) を知ることが出来る。また、所属

ドメインを管理する DNS サーバの情報を書き換えるだけで、容易に新規のホスト追加や削除が可能である。

インターネットの世界には種々のプロトコルが混在している。例えば web 閲覧に用いられる HTTP(Hyper Text Transfer Protocol) やファイル転送に用いられる FTP(File Transfer Protocol)、電子メールのための SMTP(Simple Mail Transfer Protocol) • POP(Post Office Protocol) などがあるが、これらのアプリケーションプロトコルは Well known services と呼ばれそれぞれ特定のポート (e.g. HTTP:80, FTP:21, SMTP:25, POP:110) で実行されることが前提となっているため、DNS による名前解決ではホスト名—IP アドレス間のみの対応となっている。

(3) ルーティングサーバ

以上のような問題点を踏まえて、本研究ではエージェント間の情報交換のために独自のネットワークシステムを構築した。同一マシン上で複数のエージェントを起動することと名前解決を円滑に行うためにルーティングサーバ (Routing Server) を用いた通信ネットワークを構築した。Fig.2.10 にルーティングサーバの概要を示す。エージェントは起動時にルーティングサーバにアクセスし、自己の情報 (名前及び所属グループ名) を登録する。ルーティングサーバは、エージェントから他のエージェント宛のメッセージを渡されると、メッセージ上の宛先に基づいて当該エージェントにメッセージを配信する。これにより、各エージェントはルーティングサーバの IP アドレスおよびポート番号のみを設定すれば良く、他のエージェントの IP アドレスやポート番号を問い合わせずとも、すべてのエージェントと通信を行うことが出来る。また、ルーティングサーバがメッセージの振り分け (ルーティング) をするため、特定のエージェントのみを宛先に指定 (ユ

ニキャスト) したり、グループ名を宛先に指定 (グループ通信またはマルチキャスト) したり、全エージェントを宛先に指定 (ブロードキャスト) することも可能である。

ルーティングサーバを用いた通信ネットワークでは、エージェントとルーティングサーバ間の接続 (論理的な通信路) を保持したままの通信 (エンド—エンドセッション型) を行うこととした。‘セッション (session)’ とは 1 つのクライアントと 1 つのサーバ間での論理的な通信路を指したり、2 者間での一連の通信 (クライアントユーザによる一連の操作) を指す用語である。前述の HTTP や FTP などのアプリケーションプロトコルは、1 組のリクエスト (クライアント→サーバ) とレスポンス (サーバ→クライアント) からなるサーバ—クライアント型 (またはサーバアクセス型) の通信であるため、セッション ID と呼ばれるパラメータを使用して一連の通信であることを識別している。これに対し、本研究ではエージェントが正常に機能している間、エージェントとルーティングサーバ間のセッションは常に保持される。エージェント間でピアトゥピア通信を行なう場合、全てのエージェントが発信者 (クライアント) にも受信者 (サーバ) にもなる可能性がある。そのため全てのエージェントに受信待機用のサーバソケット (特定のポートを常に監視する機能) を持たせなければならないためプログラム容量が増加する。さらに複数のエージェントと同時に接続を保持する可能性があるため、n のエージェント間で相互の接続を行なうと最大 $n(n-1)/2$ 本の接続が必要となり、エージェント数の増加に伴ってネットワークの負荷が高くなる。しかし、本研究で用いたエンド—エンドセッション型通信では、各エージェントからルーティングサーバへ向けて接続を張るため、各エージェントに受信待機用のサーバソケットは不要である。さらに n のエージェントとルーティングサーバ間では

n 本の接続で良いためネットワークへの負荷も軽減することが出来る。なお、このようなシステムでは、エージェント数の増加に伴って、ルーティングサーバに負荷が集中することが予想されるが、ルーティングサーバを複数設置することが出来、スイッチングハブを用いたスター型ネットワークのように、各々のルーティングサーバに複数のエージェントを接続して、それらのルーティングサーバ同士を接続することによって、通信の負荷を分散させることが出来る。このようなルーティングサーバを用いたネットワークは、新しい装置の追加や故障による装置の入れ替えが頻繁に起こる環境においても、柔軟かつ効率よく通信を管理することが出来ると考えられる。

2.1.7 契約ネットプロトコルによる装置の獲得

装置の獲得には、契約ネットプロトコル (Contract-net Protocol) を用いた。その様子を UML シーケンス図で Fig.2.11 に示す。

(1) 製品エージェントは、サブレシピで指定されている機能を持つ装置エージェントに "募集" メッセージを送付する。募集メッセージには、仕事を開始する予定時刻が記載され、サブレシピのコピーが添付される。

(2) 募集メッセージを受け取った装置エージェントは、自己の処理速度や精度などの静的な情報とその仕事を請け負った場合の終了予定時刻や安全性などの動的な情報を "入札" メッセージに添付して、組立エージェントに送付する。但し、既に他の仕事の契約が成立しており、募集された指定時刻に仕事を開始できない場合には、自己のスケジュールの都合がよい開始時刻を代替案として提示する。また、募集された仕事を請け負った場合に、安全性が確保できないと判断される場合、すなわち安全性の評価値が一定の値を下回る場合には、入

札を行わないこともある。

(3) 装置エージェントからの “入札” メッセージを受け取った組立エージェントは、その装置を組立てた場合にかかる時間やコストなどの情報を追加して、正式な “入札” メッセージとして募集元の製品エージェントに送付する。

(4) 製品エージェントは、あらかじめ定めた入札締切時刻になると、受け取った複数の “入札” メッセージを評価し、意思決定を行う。

(5) 選ばれた装置エージェントに対して、“契約” メッセージを送付する。

(6) “契約” メッセージを受け取った装置エージェントは、自己のスケジュールを再確認した上で、“承諾” メッセージを返信する。この時点で契約が正式に成立する。但し、それより先に他の製品エージェントと契約が成立してしまった場合には、“拒否” メッセージを返信する。

(7) “承諾” メッセージを受け取った製品エージェントは、その装置エージェントに自分自身が移動する。一方、“拒否” メッセージを受け取った場合には、(4) で得られた “入札” の中から、次に評価の良い装置エージェントに “契約” メッセージを送付する。

(8) 移動が完了すると、製品エージェントは、装置エージェントにレシピのコピーを渡して作業を依頼する。

(9) 同時に次の工程の “募集” メッセージを送付して、同様な手順で装置を確保する。

(10) すべての工程の装置エージェントが確保されると、組立エージェントにより組み立てられて、生産を開始する。

2.1.8 階層化意思決定法の導入

意思決定プロセスは、まず決定すべき問題を明確にし、その問題を解決するための解決策や計画案などの代替案を生成する。次に代替案が複数ある場合には、それらを評価し選択を行なうことである。

本研究の意思決定プロセスでは、"装置エージェントの選択"という問題に対して、契約ネットプロトコルに基づいた入札により代替案を生成し、これらの代替案の評価•選択を行なう。意思決定法は様々なものが提案されているが、本研究では、階層化意思決定法 (AHP:Analytic Hierarchy Process) を用いた。AHP は、複数の評価基準を用いて多目的に評価を行なうことが出来る。また、人間の持つ主観的な評価を数値として評価することが出来る。一般的に、多目的評価における問題点として、複数の評価基準間の重みを一意に設定することが困難なことが挙げられるが AHP では、評価基準同士の重要度を主観的に比較することで重みを決定できるという利点がある。

AHP では、まず、意思決定問題を "問題 (goal)"、"評価基準"、"代替案" の 3 種類の要素で捉える。Fig.2.12 に一般的な AHP の階層構造を示す。最上位の "問題レベル" に問題 (goal) を置く。この階層は単一の要素からなる。それより下には複数の評価基準を持つ "評価基準レベル" を置く。評価基準レベルは、問題の性質によりさらに細分化した評価基準を持たせることが出来、1 つまたは複数の階層で構成される。問題を解決するための複数の代替案を最下層の "代替案レベル" に置く。

次に、各階層における要素間の重み付けを行なうために、ある階層における 2 つの要素の比較 (一対比較) を行なう。一対比較とは、2 つの評価基準についてどちらがより重要かということを主観的に判断することである。判断の尺度は、

Table 2.1 に示す 1, 3, 5, 7, 9 の 5 段階とその補間的な値およびそれらの逆数の計 17 段階で行なう。評価基準 i と評価基準 j について、2 つの評価基準の重要度が同じくらいの場合には、一対比較値 a_{ij} は 1 をとる。評価基準 j に比べて評価基準 i の方が絶対的に重要な場合には一対比較値 a_{ij} は 9 をとる。逆に評価基準 i に比べて評価基準 j の方が絶対的に重要な場合には、a_{ij} は逆数である 1/9 をとり、a_{ji} は 9 をとる。一対比較を同じ階層内のすべての組み合わせについて行なう。従って同一の階層内に n 個の要素 (評価基準) がある場合には、$n (n - 1) / 2$ 回の一対比較を行なうことになる。このようにして得られた一対比較値 a_{ij} の組を Table 2.2 に示す。ここで、対角成分 $a_{ii} = 1$ とし、$a_{ji} = 1 / a_{ij}$ の関係が成り立つようにする。

最後に、得られた一対比較値 a_{ij} の組をもとにして、各評価基準の重要度を計算する。評価基準 i の重要度を w_i、評価基準 j の重要度を w_j とすると、

$$a_{ij} = \frac{w_i}{w_j} \tag{2.1}$$

が成り立つ。すなわち、一対比較行列 $\boldsymbol{A}$ は、

$$\boldsymbol{A} = \{ a_{ij} \} = \{ \frac{w_i}{w_j} \} \tag{2.2}$$

となる。これより、

$$\boldsymbol{A} \boldsymbol{w} = n \boldsymbol{w} \tag{2.3}$$

が成り立つ固有値 n と固有ベクトル $\boldsymbol{w}$ が求められる。固有ベクトル $\boldsymbol{w}$ は、

$$\boldsymbol{w} = \{ w_i \} \tag{2.4}$$

である。すなわち、固有ベクトル $\boldsymbol{w}$ の各要素は各評価基準の重み w_i となる。

一対比較は、主観的な判断により、重みを算出する方法である。従って、n 個の評価基準に対して $n (n - 1) / 2$ 回の一対比較を行なった結果、それらの一対比較値および重みの整合性がとれない場合が生じる。そこで、一対比較値の整合性を評価するために、整合度 (Consistency Index) C.I. を次のように求める。行

列 $\boldsymbol{A}$ には、n 個の固有値があり、その和は n となる。最大固有値を λ_{max} とすると、(λ_{max} - n) は λ_{max} 以外の固有値の大きさを示す指標と見ることができる。(n - 1) 個の固有値でこの指標を持つので、その平均をとると、

$$\mathrm{C.I.} = \frac{\lambda_{max} - n}{n - 1} \tag{2.5}$$

となる。行列 $\boldsymbol{A}$ が完全な整合度を持つ場合には、C.I. = 0 であり、値が大きくなるほど不整合度が高いことを意味する。通常、C.I. 値が 0.1 以下 (場合によっては、0.15 以下) を許容範囲とし、整合度が許容範囲外の場合には、一対比較をやり直さなければならない。Table 2.2 に示した一対比較行列では、C.I. 値は 0.064 となり許容範囲内である。

Table 2.1　AHP の一対比較値の評価尺度

一対比較値 a_{ij}	定　義
1	項目 i と項目 j が**同じぐらい重要** (equal importance)
3	項目 i の方が項目 j に比べて、**少し重要** (weak importance)
5	項目 i の方が項目 j に比べて、**重要** (strong importance)
7	項目 i の方が項目 j に比べて、**かなり重要** (very strong importance)
9	項目 i の方が項目 j に比べて、**絶対的に重要** (absolute importance)
2, 4, 6, 8	上記の評価の補間的に用いる
上の数字の逆数	項目 j に比べて項目 i の方が重要な場合に用いる

Table 2.2　AHP の一対比較値の一例

a_{ij}		j			
		1	2	3	4
i	1	**1**	7	2	6
	2	1/7	**1**	1/5	3
	3	1/2	5	**1**	6
	4	1/6	1/3	1/6	**1**

C.I. = 0.064

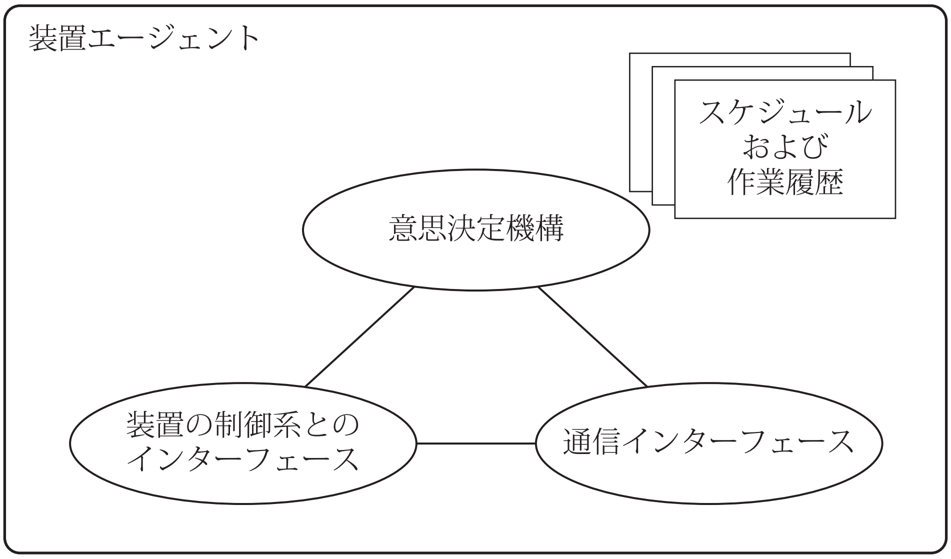


Fig.2.1　装置エージェントの構造の概略

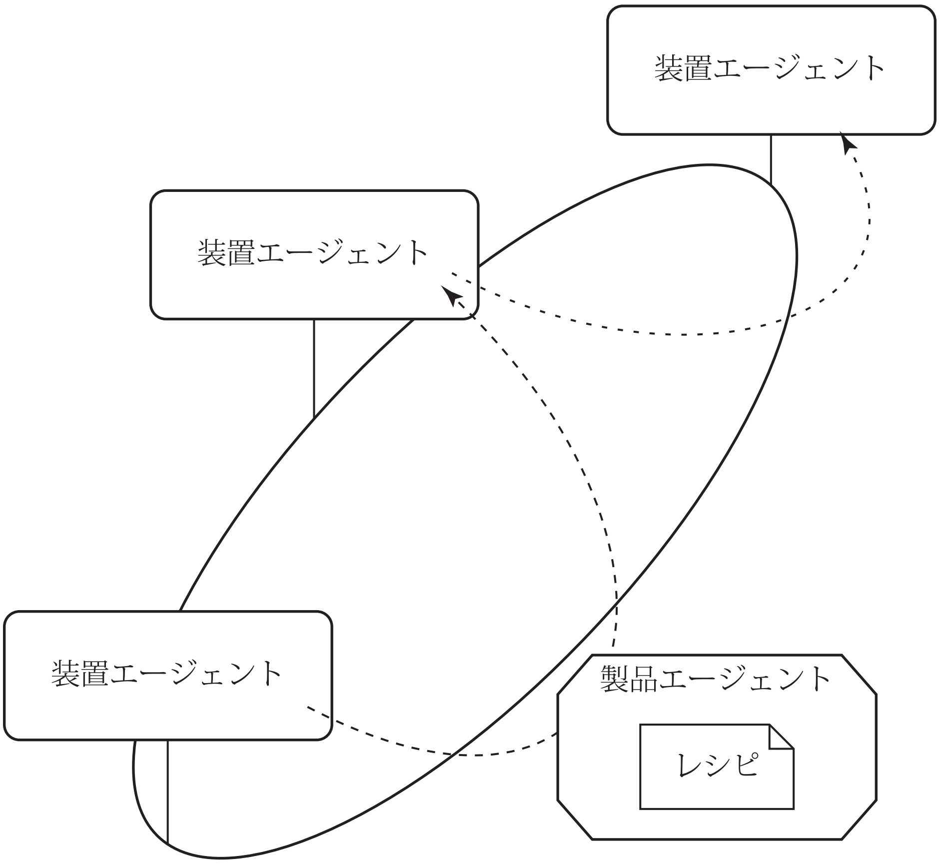


Fig.2.2　製品エージェントが装置エージェント間を動き回る様子

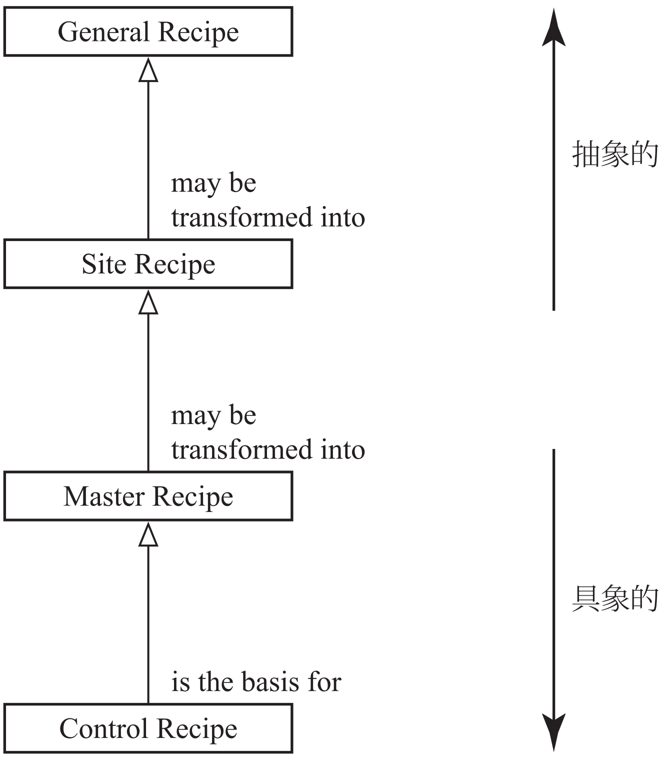


Fig.2.3　S88.01 における抽象度に基づくレシピの分類

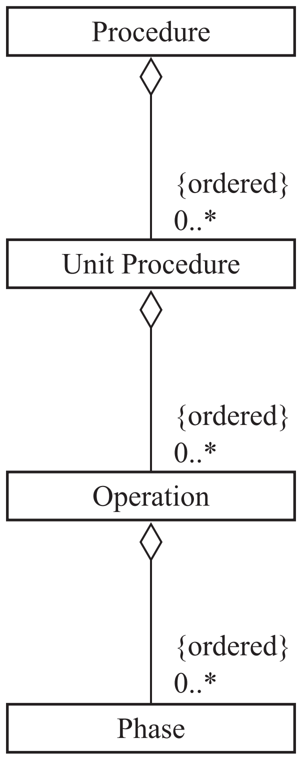
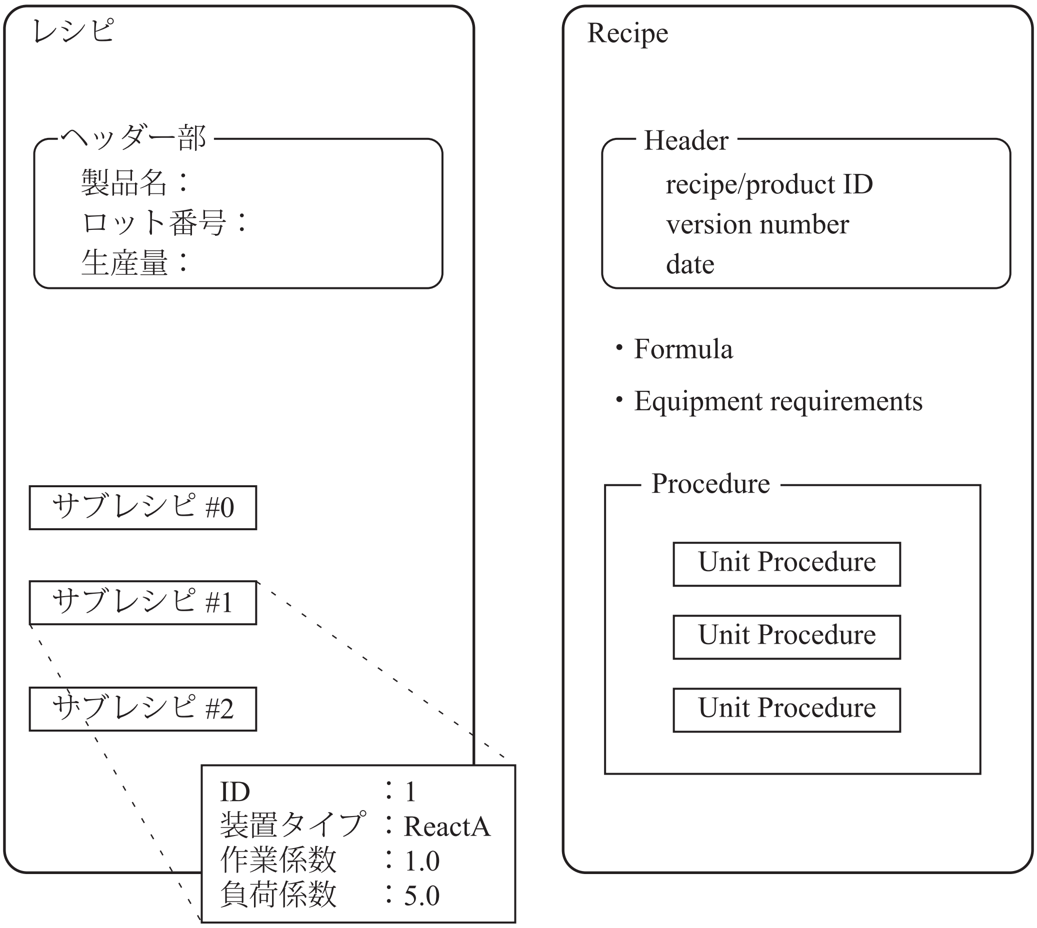


Fig.2.4　S88.01 におけるレシピの階層構造



(a) 本研究で用いたレシピ構造の一例 (b) S88.01 に基づくレシピ構造の一例

Fig.2.5　本研究で用いたレシピ構造と S88.01 に基づくレシピ構造の比較

```
<?xml version="1.0"?>        ←XML 宣言

<recipe name="ProductA" lotID="2" quantity="3800">
        製品名          ロット番号      生産量

   <subRecipeList first="0" last="2">

      <subRecipe id="0"
                 coef="1.0"
                 loadCoef="2.5"
                 equipType="MixA"/>

      <subRecipe id="1"
                 coef="1.0"
                 loadCoef="5.0"
                 equipType="ReactA"/>

      <subRecipe id="2"          ←サブレシピ ID
                 coef="1.5"        ←仕事係数
                 loadCoef="4.0"      ←負荷係数
                 equipType="SepA"/>    ←装置タイプの指定

   </subRecipeList>

   <connectionList>

      <connection from="0" to="1"/>          サブレシピの処理順序

      <connection from="1" to="2"/>

   </connectionList>

</recipe>
```

Fig.2.6　XML 形式によるレシピの記述例

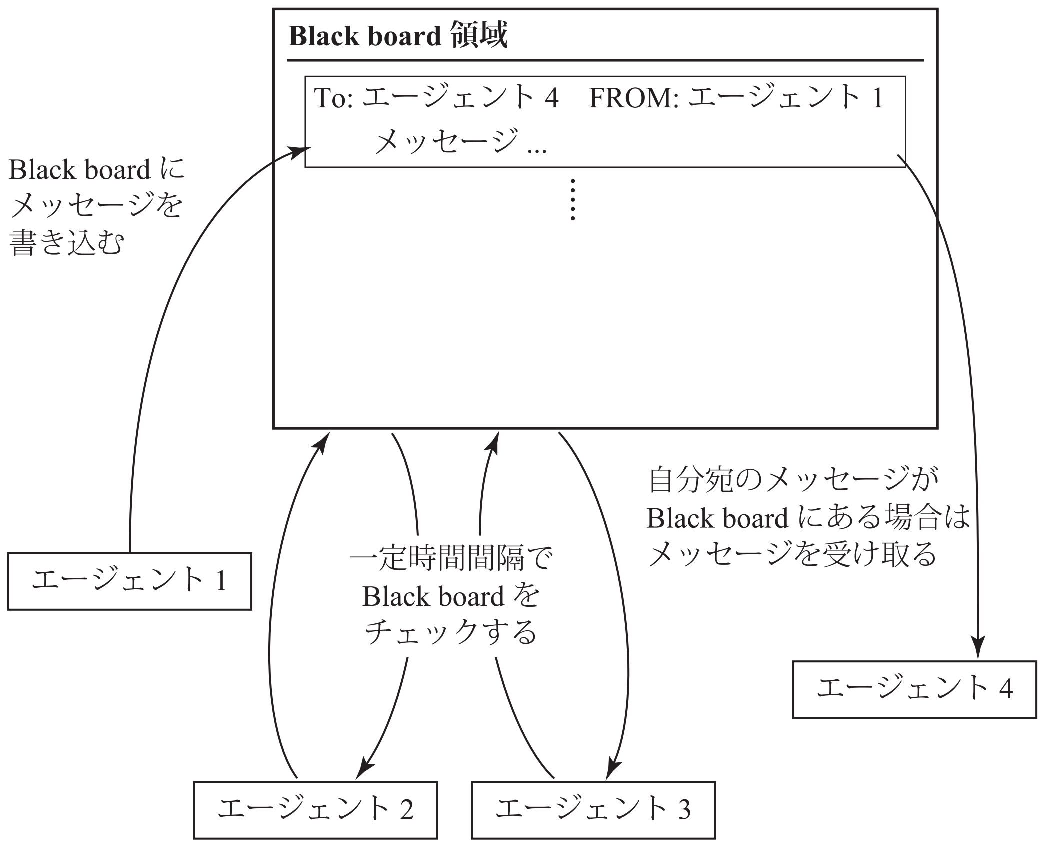


Fig.2.7　Black board 方式の通信の概略

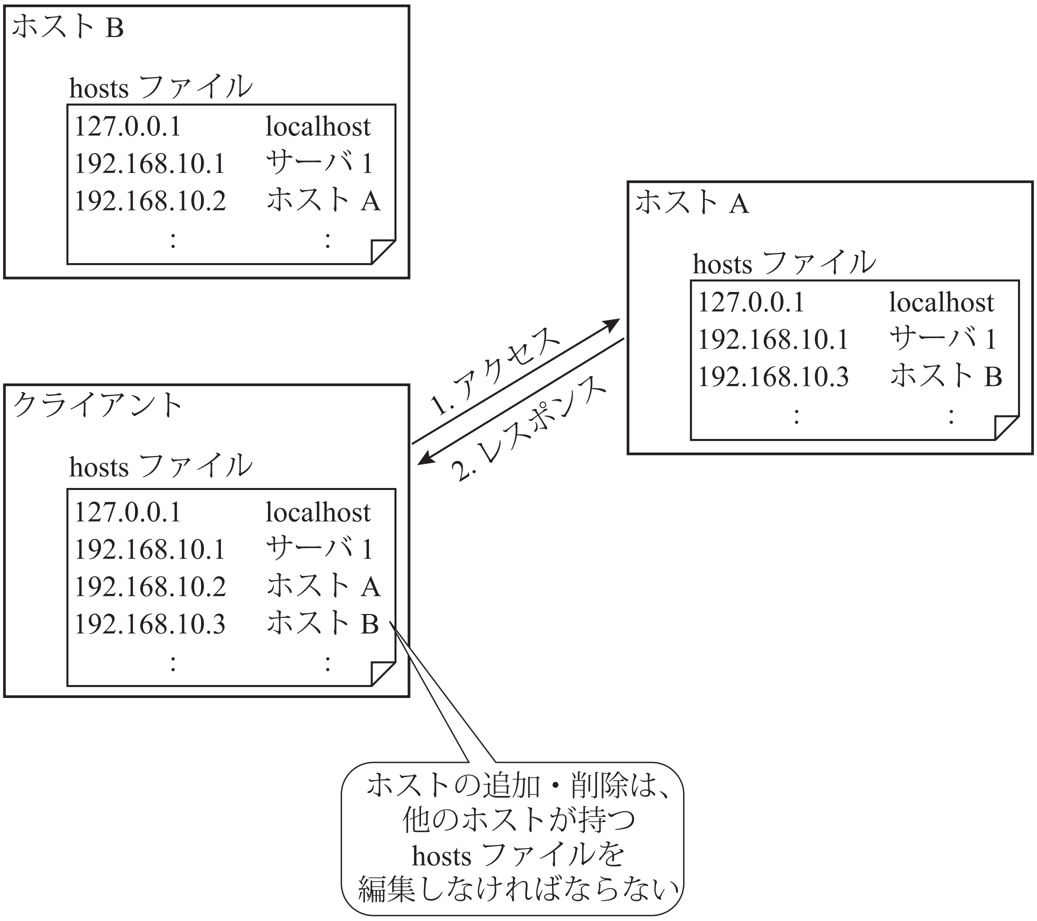


Fig.2.8　hosts ファイルを用いたネットワークの概略

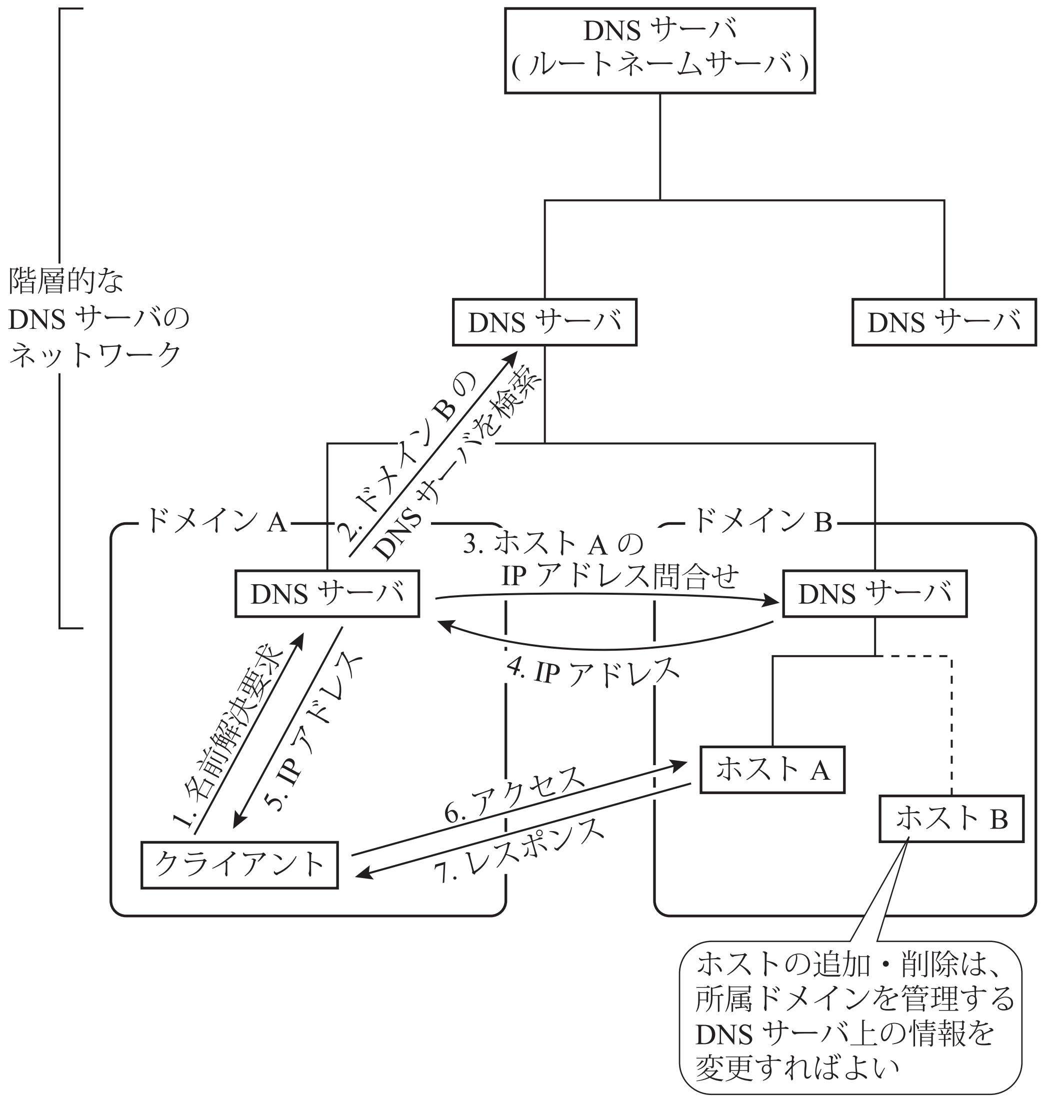


Fig.2.9　DNS を用いたネットワークの概略

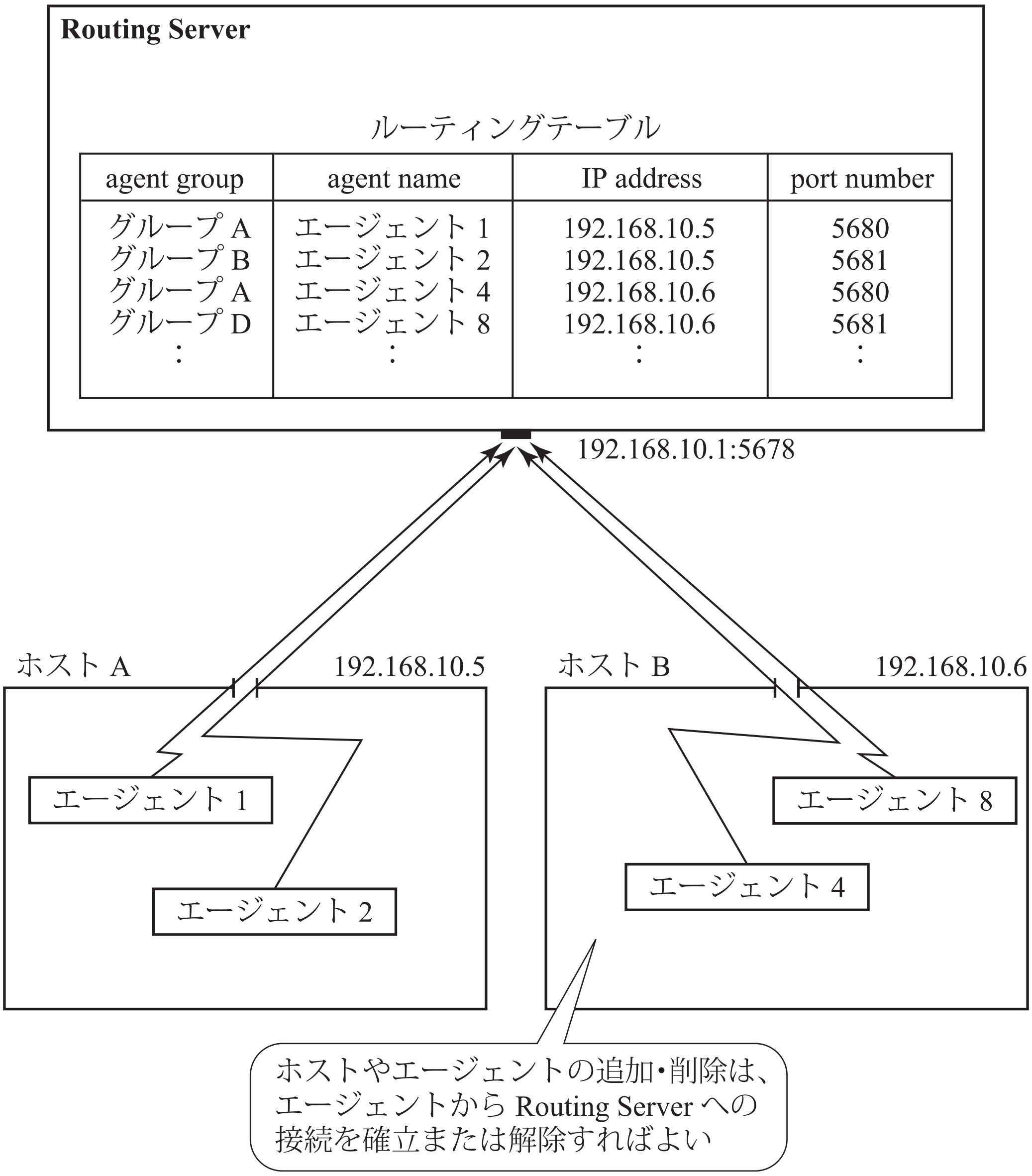


Fig.2.10　ルーティングサーバを用いたネットワークの概略

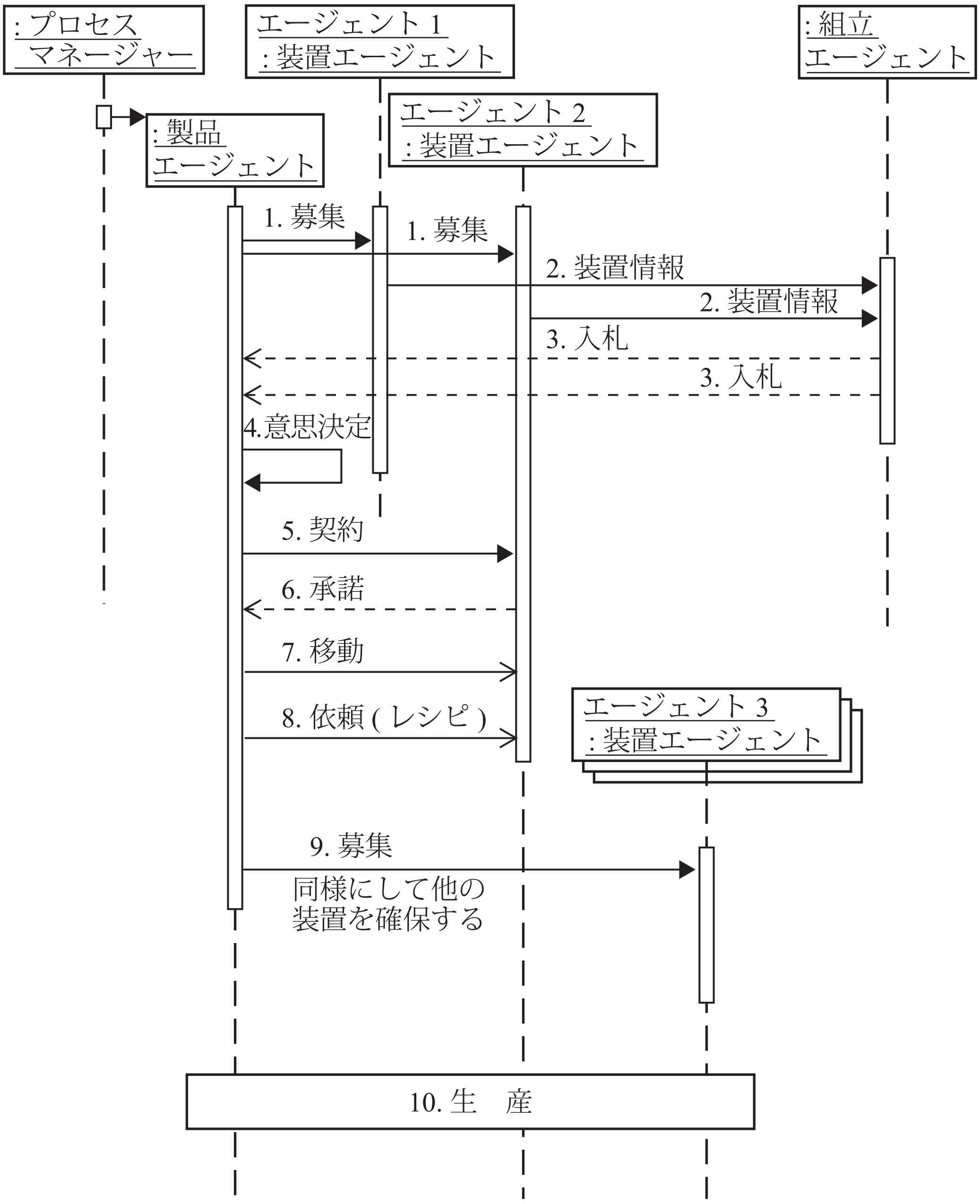


Fig.2.11　契約ネットプロトコルに基づく通信シーケンス

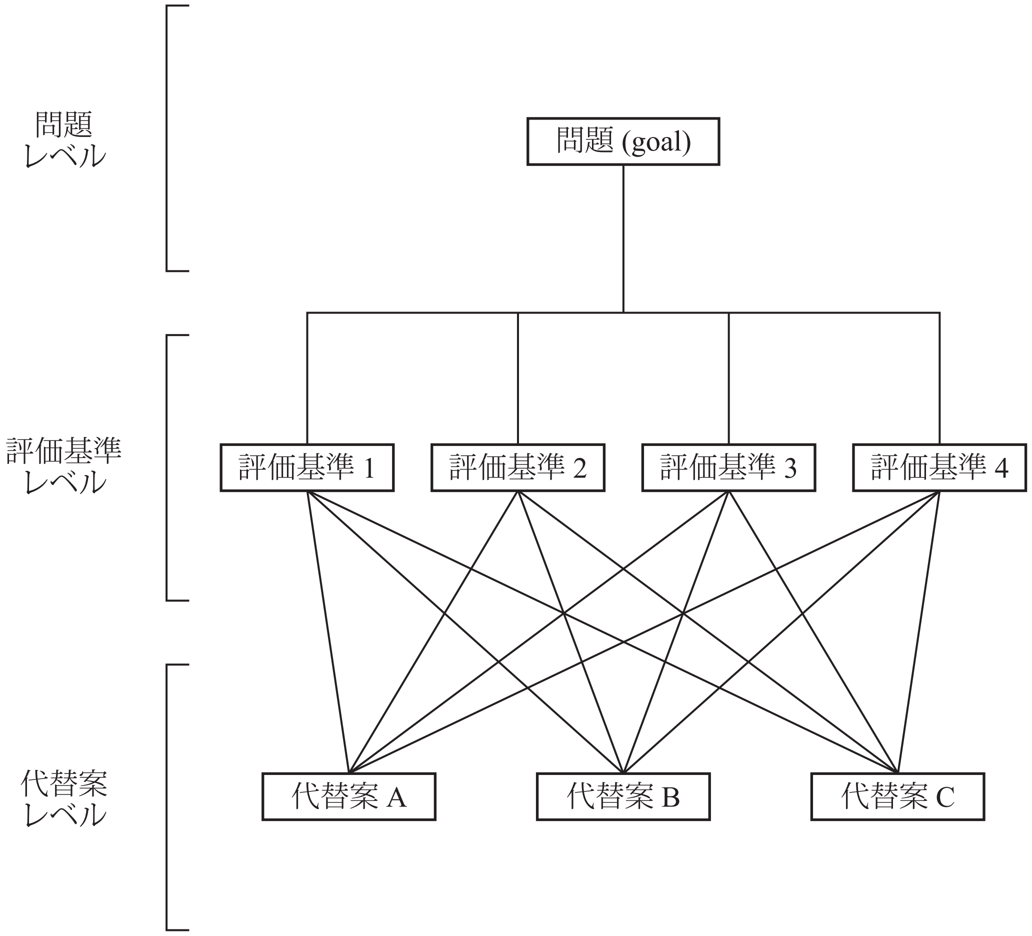


Fig.2.12　AHP の構造

2.2 マイクロ化学プロセス運転シミュレーション

2.2.1 シミュレーション対象および方法

対象とする化学プロセスは、混合→反応→分離の 3 つの単位操作からなる。製品の品種により、それぞれの単位操作における使用可能な装置タイプが異なる。本研究では、最大 7 品種のレシピを作成した。また、装置は各単位操作において最大 6 種類の装置タイプを設定し、同一の装置タイプでも処理速度等の性能が異なる装置を設定した。

本研究で用いたエージェントはすべて Java 言語を用いてプログラムし、2 ～ 3 台のマルチタスクコンピュータを用いてシミュレーションを行なった。

2.2.2 AHP の評価基準

本実験で用いた AHP の構造を Fig.2.13 に示す。評価基準は、機能性評価、動的評価、経済的評価、運転評価の 4 項目とする。装置エージェント i の総合評価値 $eval_{total,\,i}$ は

$$eval_{total,\,i} = eval_{f,\,i}\, w_f + eval_{d,\,i}\, w_d + eval_{e,\,i}\, w_e + eval_{o,\,i}\, w_o \qquad (2.6)$$

と表すことが出来る。ここで w_f, w_d, w_e, w_o は各評価基準 (機能性評価、動的評価、経済的評価、運転評価) の重みであり、$eval_{f,\,i}, eval_{d,\,i}, eval_{e,\,i}, eval_{o,\,i}$ は装置エージェント i に対する各評価基準の評価値である。各項目をそれぞれ次に述べるように細分化して、評価を行なう。

(1) 機能性 (functional) 評価

機能性評価は、装置固有の静的な情報であり、処理速度 (procSpeed) と処理精度 (procAcc) の 2 項目で評価する。

装置の処理速度 (procSpeed) は以下の式を用いて評価する。

$$eval_{procSpeed,\, i} = \frac{S_i}{S_{max}} \tag{2.7}$$

対象とする製品に対する装置エージェント i の処理速度を S_i とし、入札した装置エージェントの中で最速なものの処理速度を S_{max} とする。上式で規格化することにより、$eval_{procSpeed}$ は 0.0 ～ 1.0 となり、高速なものほど評価値が大きくなる。

処理精度 (procAcc) は、0.0 ～ 1.0 の範囲であらかじめ各装置エージェントが持っている値であり、装置エージェント i の処理精度の評価値を $eval_{procAcc,\, i}$ とする。値が大きいほど処理精度が高いことを意味している。

(2) 動的 (dynamic) 評価

動的評価は、プロセスの負荷により変動する時間的評価であり、即応性 (promptness) と組立時間 (assembleTime) の 2 項目で評価する。

即応性 (promptness) は、装置エージェントがその注文をいつから処理出来るかを示す指標であり、次式を用いて評価する。

$$eval_{prompt,\, i} = 1 - \frac{t_{prompt,\, i}}{t_{prompt,\, latest}} \tag{2.8}$$

募集メッセージに記載された処理開始指定時刻と装置エージェント i が作業を開始可能な時刻の差を $t_{prompt,i}$ とする。入札した装置エージェントの中で、一番遅い処理開始可能時刻との差を $t_{prompt,\, latest}$ とする。すなわち $max(t_{prompt,\, i}) = t_{prompt,\, latest}$ であり、評価値 $eval_{prompt}$ は 0.0 ～ 1.0 の値を取る。入札した装置エージェントの中で処理開始可能時刻が一番遅いエージェントの評価値は 0.0 となり、処理開始指定時刻と同時に処理開始出来るエージェントの評価値は 1.0 となる。ただし、$t_{prompt,\, latest} = 0.0$ すなわち、すべての装置エージェントが処理開始指定時刻と同時に処理開始可能な場合には、すべての装置エージェントの評価

値が 1.0 である。

組立時間 (assembleTime) は、組立エージェントが算出した組立•分解にかかる時間である。次式を用いて評価する。

$$eval_{asmblTime,\ i} = 1 - \frac{t_{asmblTime,\ i}}{t_{asmblTime,\ latest}} \qquad (2.9)$$

装置エージェント i の組立•分解にかかる時間を $t_{asmbleTime,\ i}$ とする。入札した装置エージェントの中で、組立•分解に最も時間を要する装置エージェントの組立•分解時間を $t_{asmbleTime,\ latest}$ とする。すなわち $max(\ t_{asmble,\ i}\ ) = t_{asmble,\ latest}$ であり、評価値 $eval_{asmbleTime}$ は 0.0 ～ 1.0 の値を取る。短時間で組立•分解が行なわれるものの評価値は大きくなり、入札した装置エージェントのうち最も組立•分解時間を要するものの評価値が 0.0 となる。

(3) 経済的 (economic) 評価

経済性評価は、コストの評価である。組立•分解コストおよび追加コストの 2 項目で評価する。

組立•分解コスト (assembleCost) は、組立エージェントが算出した組立•分解にかかるコストである。次式を用いて評価する。

$$eval_{asmblCost,\ i} = 1 - \frac{Cost_{asmbl,\ i}}{Cost_{asmbl,\ max}} \qquad (2.10)$$

装置エージェント i の組立•分解にかかる費用を $Cost_{asmbl,\ i}$ とする。入札した装置エージェントの中で、組立•分解に最もコストがかかる装置エージェントの組立•分解コストを $Cost_{asmbl,\ max}$ とする。すなわち $max(\ Cost_{asmble,\ i}\ ) = Cost_{asmbl,\ max}$ であり、評価値 $eval_{asmblCost}$ は 0.0 ～ 1.0 の値を取る。入札した装置エージェントの中で、最も組立•分解コストがかかるものの評価値は 0.0 となり、コストが低いものほど評価値は大きくなる。

追加コスト (additionalCost) は、装置と装置の接続に特別なアタッチメントを使用した場合などに発生するコストである。次式を用いて評価する。

$$eval_{addCost,\,i} = 1 - \frac{Cost_{add,\,i}}{Cost_{add,\,max}} \tag{2.11}$$

装置エージェント i の追加コストを $Cost_{add,i}$ とする。入札した装置エージェントの中で、最も追加コストがかかる装置エージェントの追加コストを $Cost_{add,\,max}$ とする。すなわち $max(\,Cost_{add,\,i}\,) = Cost_{add,\,max}$ であり、評価値 $eval_{addCost}$ は 0.0~1.0 の値を取る。入札した装置エージェントの中で最も追加コストがかかるものの評価値は 0.0 となり、コストが低いものほど評価値は大きくなる。但し、$max(\,Cost_{add,\,i}\,) = 0$ すなわち、すべての装置エージェントに追加コストがかからない場合には、すべての装置エージェントの評価値 $eval_{addCost} = 1.0$ である。

(4) 運転 (operational) 評価

運転評価は、安全性に関する評価項目である。

すべての機器や装置には有限な寿命があり、装置内部や環境から受ける様々なストレスによって故障が発生する。一般に機器の故障確率は確率分布と見なすことが出来、指数分布やガンマ分布、ワイブル (weibull) 分布、正規分布、対数正規分布などが用いられる [63]。不信頼度 $F(\,t\,)$ は、装置の供用開始 (時刻 $t = 0$) から時間 t 経過した時に装置が有している機能を失っている確率であり、累積故障分布関数や累積故障率とも呼ばれる。信頼度 $R(\,t\,)$ は不信頼度とは逆に時間 t 経過してもなお装置の機能を果たしている確率である。これより、

$$R(\,t\,) = 1 - F(\,t\,) \tag{2.12}$$

が成り立つ。また、確率密度関数 $f(\,t\,)$ は、

$$f(t) = \frac{\mathrm{d}F(t)}{\mathrm{d}t} = -\frac{\mathrm{d}R(t)}{\mathrm{d}t} \tag{2.13}$$

で定義される。故障率 $\lambda(t)$ は、単位時間内に故障する割合であり、故障率関数や瞬間故障率とも呼ばれる。数学的には、時刻 t から $t+\Delta t$ の間に故障した装置の割合 $f(t)\Delta t$ と、その時正常に機能していた装置の割合 $R(t)$ との比の極限 ($\Delta t \to 0$) をとったものであり、

$$\lambda(t) = \frac{f(t)}{R(t)} \tag{2.14}$$

で表される。一般に機器のリリース直後に故障が頻発する「初期故障期間」があり、その後、故障率が低く安定した期間「偶発故障期間」を経て、機器の摩耗や劣化により故障率が上昇する「摩耗故障期間」を経る。Fig.2.14 に縦軸に故障率 $\lambda(t)$、横軸に時間をとった U 字型の曲線を示す。故障率の低い期間すなわち偶発故障期間が長く続くことから、バスタブ曲線と呼ばれる。

本研究では、半導体集積回路等において寿命分布としてよく用いられるワイブル分布を用いて、装置エージェントの信頼性を評価した。ワイブル分布の信頼度 $R(t)$、確率密度関数 $f(t)$、故障率 $\lambda(t)$ は次式で表される。

$$\begin{aligned} R(t) &= \exp\left[-\left(\frac{t-\gamma}{\eta}\right)^{m}\right] \quad (t \geq \gamma > 0,\ m > 0,\ \eta > 0) \\ f(t) &= \frac{m}{\eta}\left(\frac{t-\gamma}{\eta}\right)^{m-1} \exp\left[-\left(\frac{t-\gamma}{\eta}\right)^{m}\right] \\ \lambda(t) &= \frac{m}{\eta}\left(\frac{t-\gamma}{\eta}\right)^{m-1} \end{aligned} \tag{2.15}$$

ここで、m は形状パラメータと呼ばれ、装置の故障メカニズムによって変化する値である。η は、尺度パラメータ又は特性寿命と呼ばれ、装置の特性を表す。γ は位置パラメータと呼ばれ、装置の供用を開始してからある時間 γ 経過までは、故障が発生しない場合に用いるが、通常はそのようなことはないので多くは $\gamma = 0$ として扱う。m の値が 1 より小さい場合、故障率 $\lambda(t)$ は時間 t の経過

とともに減少する。これは、バスタブ曲線における初期故障期間に相当する。$m = 1$ の場合、時間に関係なく故障率 $\lambda(t)$ は一定 (偶発故障期間に相当) であり、式 (2.15) は指数分布 ($f(t) = \lambda e^{-\lambda t}$) に簡略化出来る。m が 1 より大きい場合、時間 t の経過に伴って故障率 $\lambda(t)$ は増加する (摩耗故障期間に相当)。Fig.2.15 に $m = 0.5, 1, 2$ の場合の信頼度 $R(t)$、確率密度関数 $f(t)$、故障率 $\lambda(t)$ をそれぞれ示す。

装置へのストレスの大きさは、作業の有無や処理内容により異なることが予想される。従って、本研究では、レシピによって装置への負荷が異なるものとして、サブレシピごとに負荷の大きさの尺度 α を定義し、式 (2.15) を次のように変形して、レシピの処理内容を信頼度に反映させることとした。

$$
\begin{aligned}
R(t) &= \exp\left[-\left(\frac{t-\gamma}{\eta/\alpha}\right)^{m}\right] \quad (t \geq \gamma > 0, m > 0, \eta > 0) \\
f(t) &= \frac{m}{\eta/\alpha}\left(\frac{t-\gamma}{\eta/\alpha}\right)^{m-1} \exp\left[-\left(\frac{t-\gamma}{\eta/\alpha}\right)^{m}\right] \\
\lambda(t) &= \frac{m}{\eta/\alpha}\left(\frac{t-\gamma}{\eta/\alpha}\right)^{m-1}
\end{aligned}
\tag{2.16}
$$

装置がアイドル状態 (作業をしていない状態) の場合には $\alpha = 1.0$ である。サブレシピには $\alpha = 1.0 \sim 5.0$ の値を設定した。α の値が大きいほど、装置に対するストレスが大きく、η/α すなわち見かけ上の特性寿命の値が小さくなる。しかし、装置は種々の作業を行なうため装置へのストレスの度合い、すなわち α の値は常に変化する。装置の供用開始 ($t = 0$) からある時間 (t_1) までは、装置がアイドル状態にあるとする。すると時刻 t_1 における信頼度は $\alpha = 1.0, \gamma = 0$ を代入して

$$
R(t_1) = \exp\left[-\left(\frac{t_1}{\eta}\right)^{m}\right] \quad (t_1 > 0) \tag{2.17}
$$

と表される。次に時刻 t_1 から t_2 までの間、負荷 $\alpha_2(> 1.0)$ の作業を行なう。この時、$t_1 \sim t_2$ の作業における信頼度は

$$R'(t_2) = \exp\left[-\left(\frac{t_2 - t_1}{\eta/\alpha_2}\right)^{m} \right] \quad (t_2 > t_1 > 0) \tag{2.18}$$

と表されるが、この作業開始時点 ($t = t_1$) での信頼度は $R(t_1)$ であるので、作業終了時点 ($t = t_2$) における累積の信頼度は、

$$R(t_2) = R(t_1) \times R'(t_2) \quad (t_2 > t_1 > 0) \tag{2.19}$$

と表すことが出来る。

同様にすれば、作業 n が終了した時点 ($t = t_n$) での信頼度は、

$$R(t_n) = R(t_1) \times R'(t_2) \times .... \times R'(t_n) \quad (t_n >....> t_2 > t_1 > 0) \tag{2.20}$$

と表される。Fig.2.16 にある装置における信頼度の経時変化を示す。装置のアイドリング時に比べ、作業時は信頼度の減少の仕方が大きいことがわかる。このようにして得られた装置エージェント i の信頼度の値を安全性の評価値 $eval_{safety,\,i}$ とする。募集メッセージを受け取った装置エージェントは、募集された仕事が完了する時刻 ($t = t_{end}$) までにその装置が故障しない確率、すなわち信頼度 $R(t_{end})$ を安全性の評価値として入札を行なう。信頼度の値は $R(t) = 0.0$ ～ 1.0 であり、安全性の高い装置エージェントほど値が大きい。

2.2.3 シミュレーション結果および考察

(1) 少品種生産シミュレーション

単一の品種のみを生産するシミュレーションを行った。装置エージェントを 12 エージェント使用し、同一品種を 10 回受注するがそれぞれのロットの生産量は異なっている。組立エージェントおよびプロセスマネージャーは各1エージェントである。製品の受注時刻および生産量はランダムであり、受注時刻になるとプロセスマネージャーに製品の注文が渡されるため、各エージェントはいつど

のような仕事が入ってくるのかを事前に知ることは出来ない。

Fig.2.17 に結果の一例をガントチャート風に示した。横軸は時刻、縦軸は各装置エージェントを示す。実線で囲まれた部分は、装置エージェントが仕事を請負い実行したことを示し、上段には製品名およびロット番号、下段にはサブレシピ番号を示す。本シミュレーションでは連続式の生産とした。従って、生産開始前にすべての装置を確保し、開始予定時刻になると確保されたそれらの複数の装置が 1 つの製品を同時に処理していることが図からもわかる。2.1.7 項で述べた契約ネットプロトコルに従って製品エージェントが装置エージェントの獲得を行う。獲得した装置エージェントの評価は、すなわちその装置により処理された製品の評価とみなすことができる。製品エージェントが各工程において獲得した装置エージェントの評価値を平均して製品の評価値とした。

Fig.2.18 に評価値をプロットしたものを示す。横軸は製品のロット番号を示し受注順である。縦軸は製品の評価値を示す。全体的にロット番号が大きくなるにつれて、評価値が低下しているが、これは運転評価によるものと考えられる。装置が仕事を処理するごとに安全性の評価は低下していくため、製品の評価値にも大きく影響を与えていると考えられる。Table 2.3 に組立エージェントおよび

Table 2.3　AHP の重みの一例

	組立エージェント	製品エージェント
w_f	0.076	0.456
w_d	0.430	0.138
w_e	0.110	0.052
w_o	0.384	0.354
C.I.	0.022	0.031

装置エージェントが初期値として持つ各評価基準項目の重み (w_f, w_d, w_e, w_o) と整合度 C.I. を示す。また、Fig.2.19 に装置選択問題の意思決定に用いた AHP の重みの変化を示す。横軸にロット番号、縦軸に C.I. 値又は機能性評価 (w_f)、動的評価 (w_d)、経済的評価 (w_e)、運転評価 (w_o) を示す。C.I. 値は常に 0.1 以下であり整合性がとれた評価をしていることが分かる。組立エージェントおよび製品エージェントが、双方ともに経済的評価を重視していないため、経済的評価 (w_e) がすぐに収束していることがわかる。

(2) 多品種生産シミュレーション

次に複数の品種の製品を生産するシミュレーションを行った。7 品種、それぞれ生産量の異なる注文を計 35 回受注する。装置エージェントは 54 エージェント、組立エージェントおよびプロセスマネージャーは各 1 エージェントである。

Fig.2.20 に製品の評価値の推移を示す。横軸は製品の受注時刻、縦軸は製品の評価値を示す。評価値が時間とともに低下しているのは前述した安全性評価の影響である。評価値が乱高下するのは、製品エージェントは受注した製品の品種ごとに異なる AHP の重みを持っているため、製品の品種によってプロセスの向き / 不向きが現れるためと考えられる。

Fig.2.21 に装置選択問題の意思決定に用いた AHP の重みの変化を示す。横軸に時刻、縦軸に C.I. 値又は機能性評価 (w_f)、動的評価 (w_d)、経済的評価 (w_e)、運転評価 (w_o) を示す。C.I. 値は常に 0.1 以下であり整合性がとれた評価をしていることが分かる。w_f ~ w_o のすべての値が時間とともに大きく乱高下していることがわかる。

(3) 装置の入れ替えシミュレーション

装置エージェントを入れ替えるシミュレーションを行った。製品数等の条件は(2) で述べた多品種生産シミュレーションと同様に 7 品種 35 回の受注を行なった。時刻0では、装置エージェントを44エージェント用意し、時刻60で10エージェントを新規に追加し、時刻 80 で 16 エージェントを使用中止とした。装置エージェントの追加はルーティングサーバに接続するだけで、装置エージェントはそれ以降の募集に参加することが出来る。また、使用中止の際にはルーティングサーバとの接続を解除するだけでよい。Fig.2.22 にシミュレーション結果の一部をガントチャート風に示す。装置グループ "SepF" において、はじめに 2 つの装置エージェント ("SepF1" および "SepF2") で運転を行い、時刻 60 で 1 エージェント ("SepF3") を追加した。追加後、しばらくして新しい装置が使用されはじめていることがわかる。また、時刻 80 ではじめから運用していた装置エージェント "SepF1" および "SepF2" の使用を中止した。

Fig.2.23 に製品ごとの評価値の推移を示す。横軸は製品の受注時刻、縦軸は製品の評価値を示す。比較のために装置を入れ替えない場合 ((2) 多品種生産シミュレーション) の結果もプロットした。全体の結果としてはそれほど大きな差がないことが分かる。ただし、時刻 10 〜 40 の付近で装置の入れ替えを行なった方が評価値が低くなっていることがわかるが、これは、時刻 60 以前は同じ機種の装置エージェント数が少ないため、負荷が集中して安全性評価が小さくなったり、処理能力に対して受注量が多いため動的評価 (即応性) が小さくなっているためと考えられる。また、時刻 65 付近や時刻 90 付近では、入れ替えを行なった方が評価値が高くなっているが、これは、新しい装置では摩耗が進んでいないため運転評価が高いことによるものと考えられる。但し、これらの違

いは非常に微小なものであり、マルチエージェントを用いることで、生産結果に大きな影響を与えることなく、円滑に装置を入れ替えることが出来たことを示している。

(4) 運転評価を用いないシミュレーション

前項までのシミュレーションにおいて、製品の評価値が時間とともに低下していく結果が得られた。これは、前述したように時間の経過とともに装置が摩耗し安全性が低下するからである。そこで、Fig.2.24 に示すように運転評価を用いない AHP を用いてシミュレーションを行った。製品は 7 品種、それぞれ生産量の異なる注文を計 35 回受注する。装置エージェントは 54 エージェント、組立エージェントおよびプロセスマネージャーは各 1 エージェントである。

Fig.2.25 に結果の一例をガントチャート風に示す。横軸は時間、縦軸は装置エージェントを示す。Fig.2.25 (a) は運転評価を用いた運転結果である。Fig.2.25 (b) は運転評価を用いないシミュレーション結果である。両図とも同じ装置グループ、同じ仕事の部分を抜粋しているが、製品 “ProductC_4” は、(a) では装置エージェント “SepC1” によって処理されているが、(b) では装置エージェント “SepC2” によって処理されていることがわかる。これは、運転評価を考慮しない (b) では、装置の機能性評価や動的 (即応性) 評価の占める割合が大きいためと考えられる。稼働率の高い装置は、摩耗が進行し運転評価が低くなるため、運転評価を考慮する (a) では稼働率の低い装置エージェント “SepC1” が受注したと考えられる。

Fig.2.26 に製品ごとの評価値の推移を示す。横軸は製品の受注時刻、縦軸は製品の評価値を示す。比較のために運転評価を考慮する場合と、考慮しな

い場合のシミュレーション結果をプロットした。運転評価を入れない場合、全体的に評価値が低下し、激しく乱高下していることがわかる。これは、運転評価がないために相対的にそれ以外の 3 項目 (機能性評価、動的評価、経済的評価) の比率が高まり、製品の品種により装置やプロセスの向き / 不向きが出やすいこれら 3 項目の変動の影響が強く出たためと思われる。

Fig.2.27 に装置選択問題の意思決定に用いた AHP の重みの変化を示す。横軸に時刻、縦軸に C.I. 値又は機能性評価 (w_f)、動的評価 (w_d)、経済的評価 (w_e)、運転評価 (w_o) を示す。C.I. 値は常に 0.1 以下であり整合性がとれている。運転評価を考慮した場合の AHP の重みの変化は Fig.2.21 である。運転評価を考慮しない場合には、重みの変動幅が大きく評価が不安定なことがわかる。以上の結果より、運転評価を考慮して意思決定を行った方が、安定した AHP の重みを持ち、比較的安定した良い運転結果が得られることがわかった。

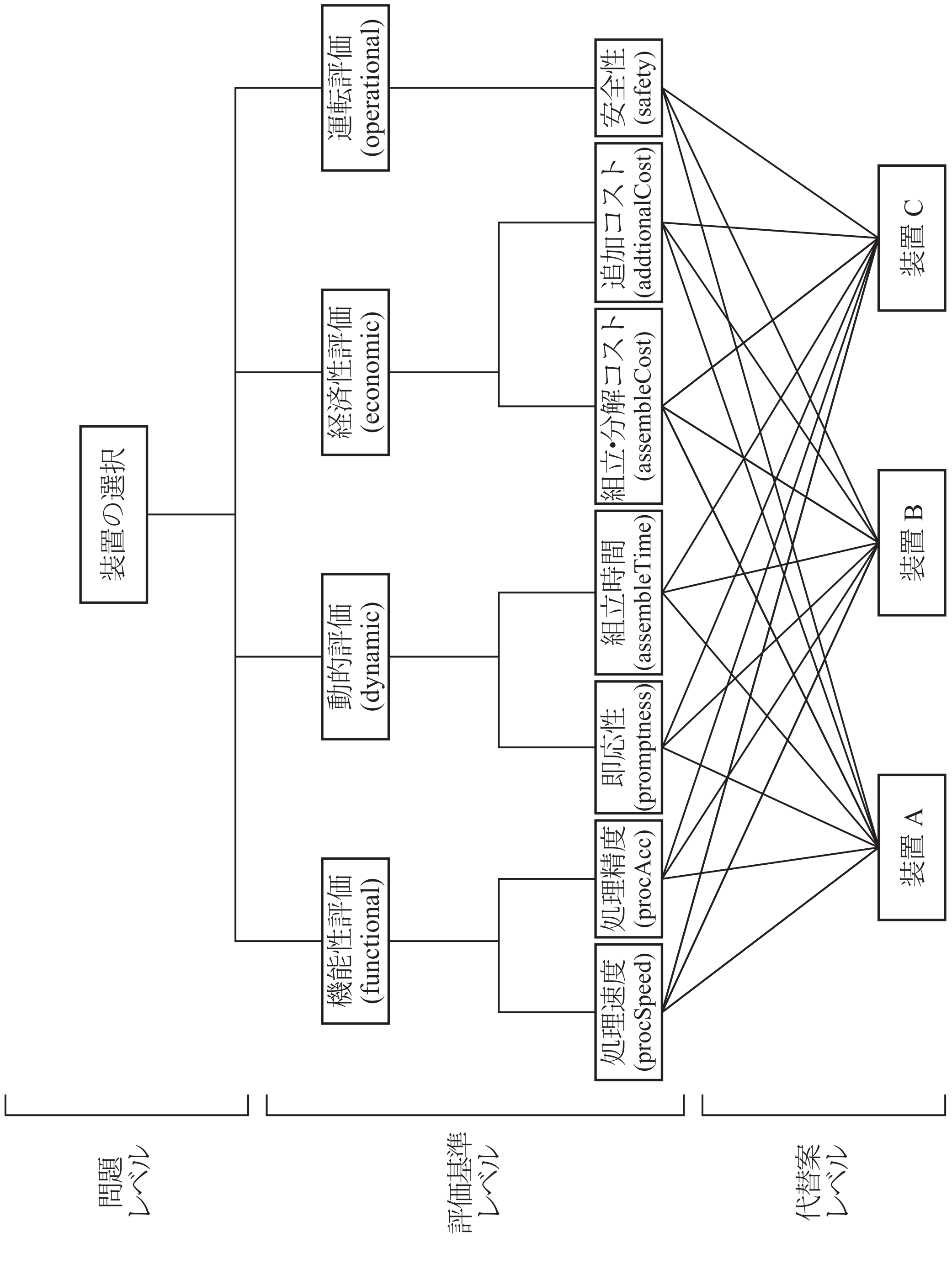


Fig.2.13　本研究で用いた AHP の構造

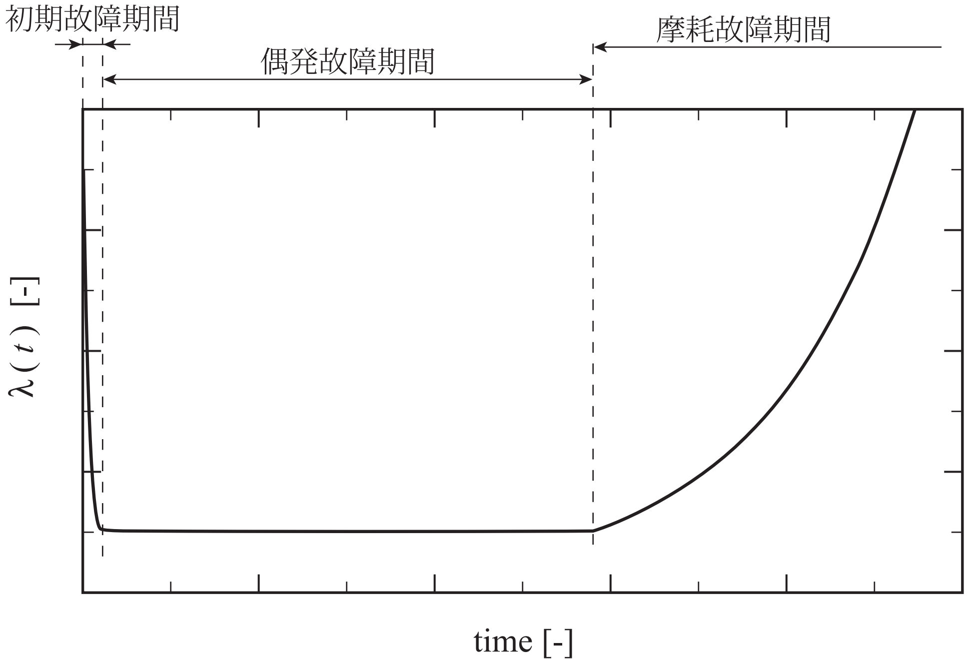


Fig.2.14　バスタブ曲線

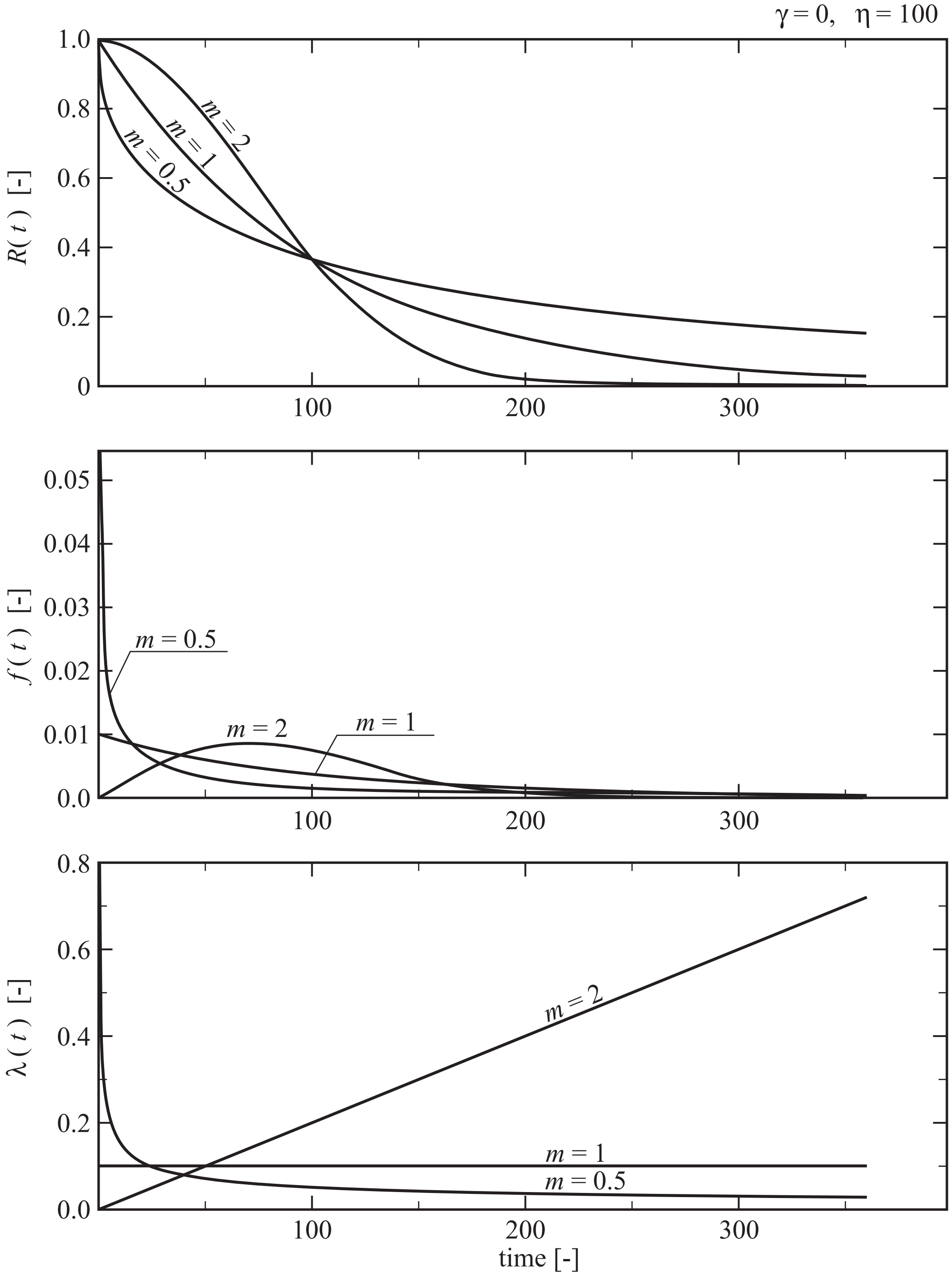


Fig.2.15　m = 0.5, 1, 2 における信頼度 , 確率密度関数 , 故障率の経時変化

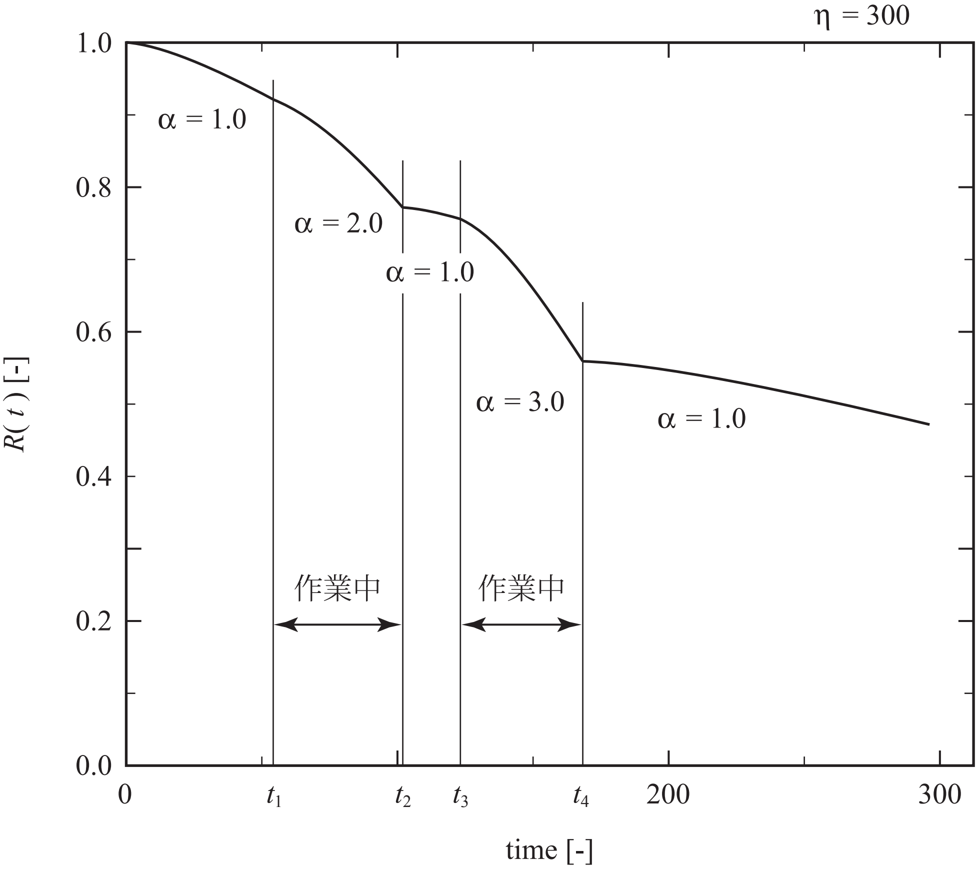


Fig.2.16　ある装置における信頼度の経時変化

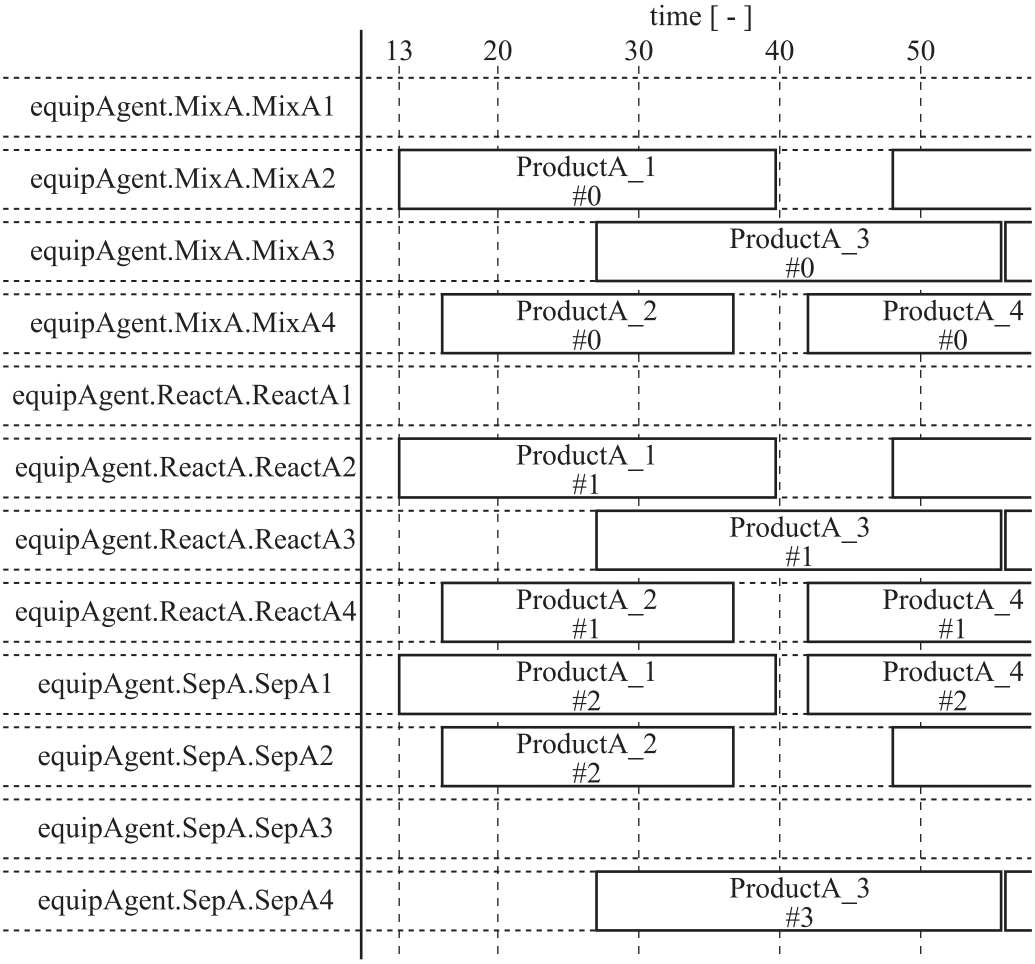


Fig.2.17　少品種生産シミュレーションにおける運転結果の一例

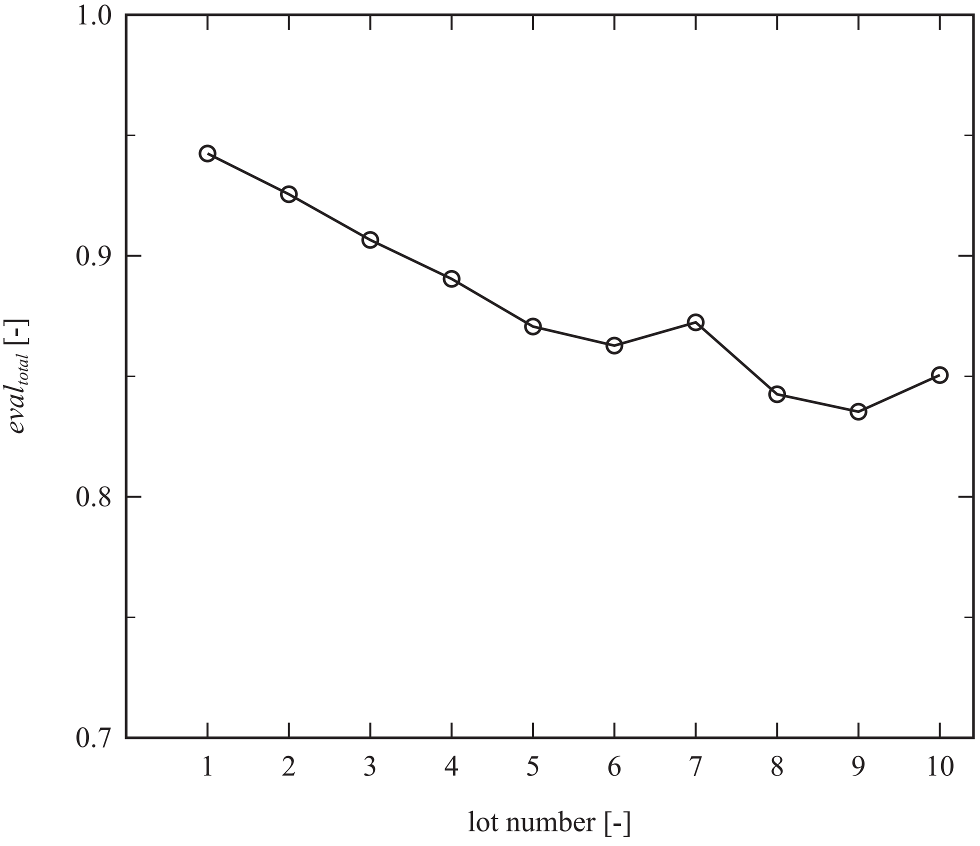


Fig.2.18　少品種生産シミュレーションにおける製品評価値の推移

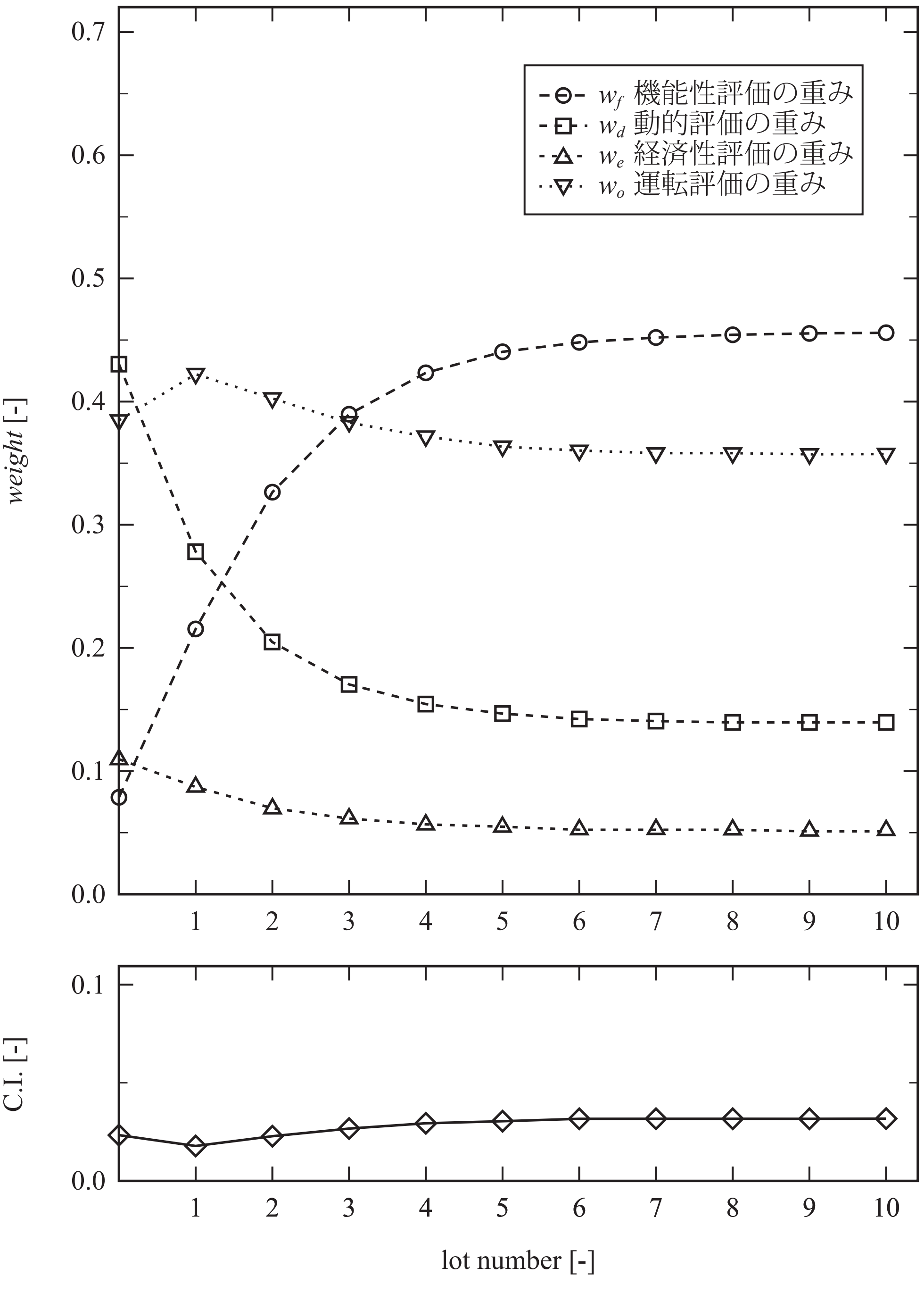


Fig.2.19　少品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の推移

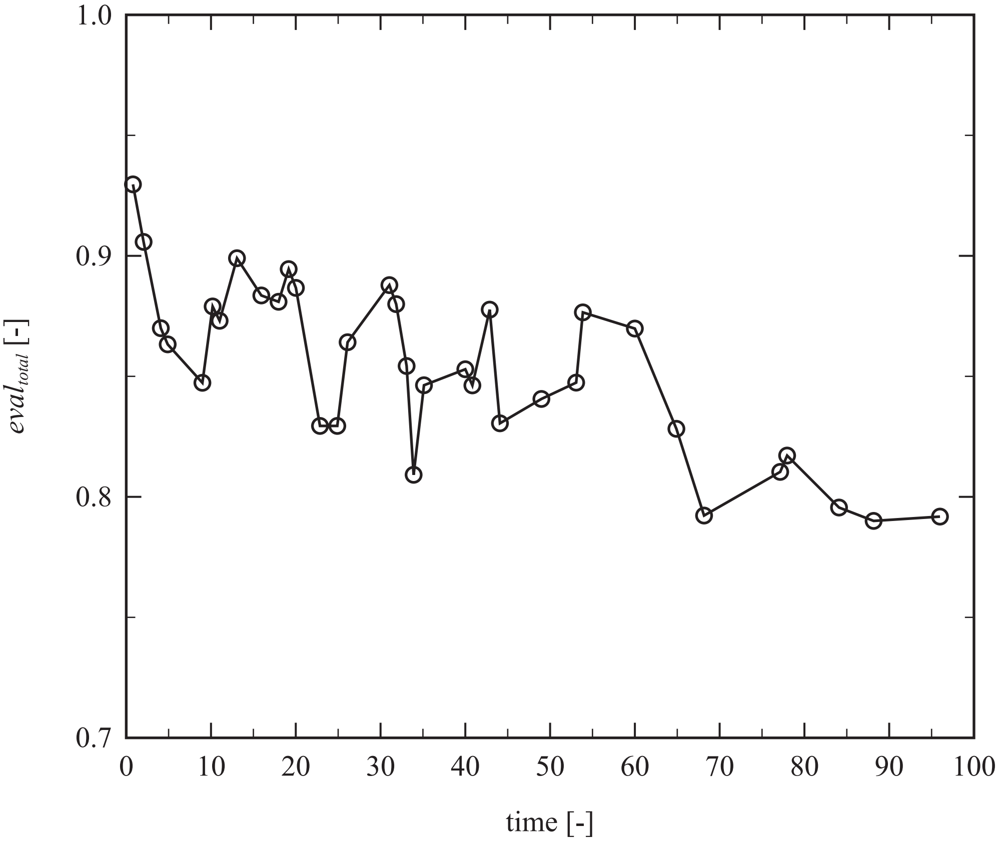


Fig.2.20　多品種生産シミュレーションにおける製品評価値の経時変化

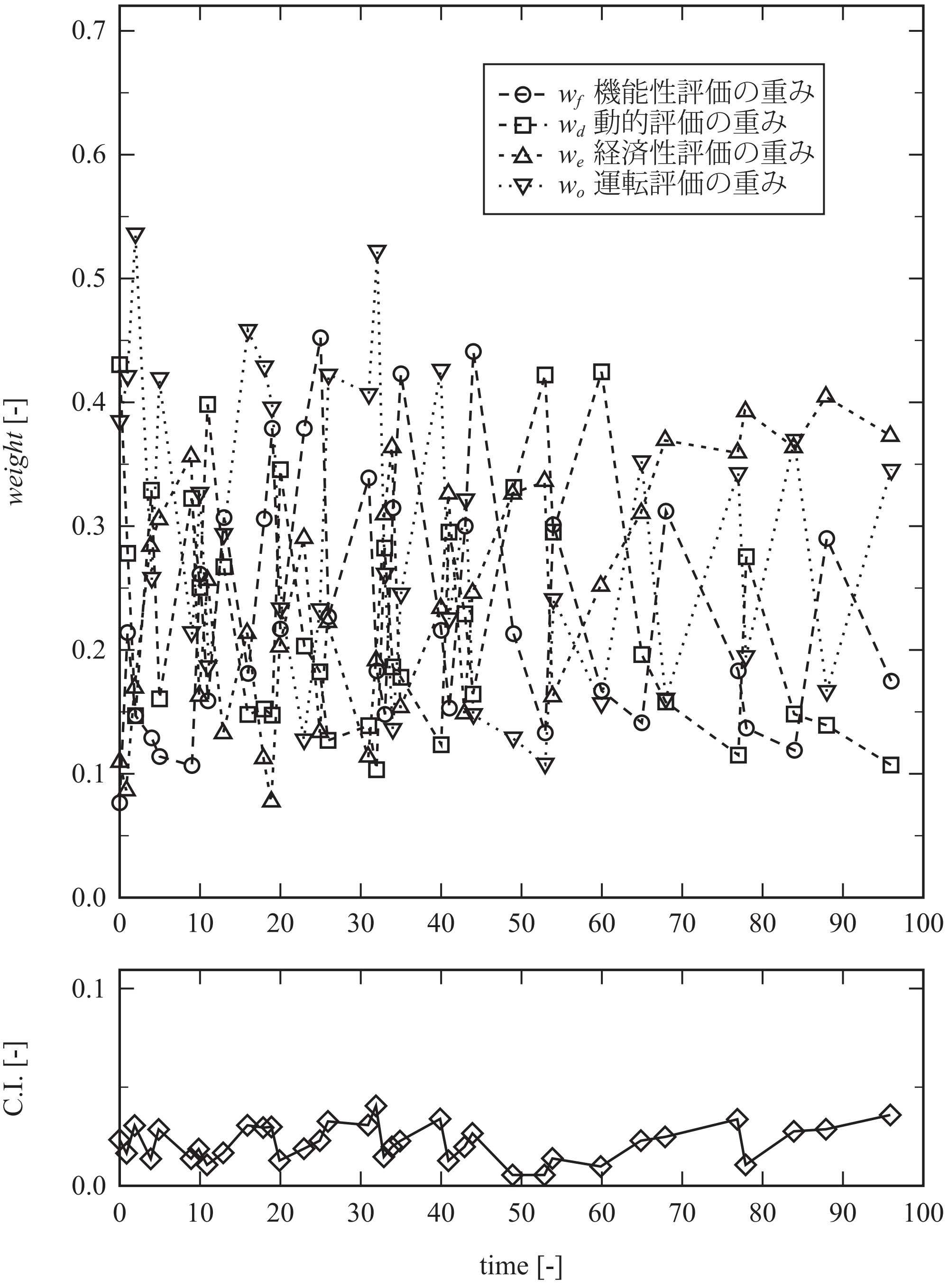
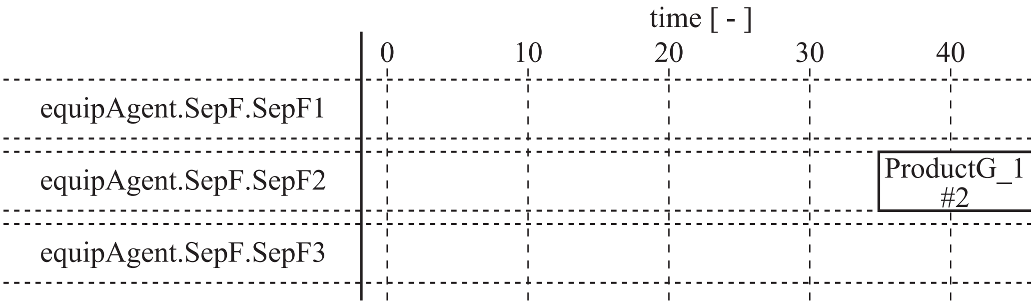
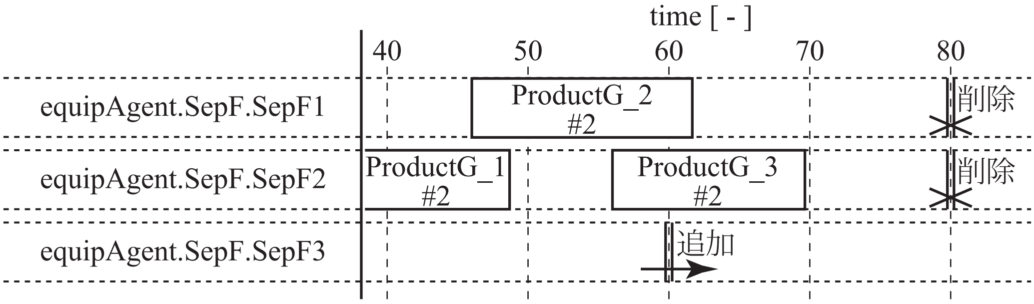


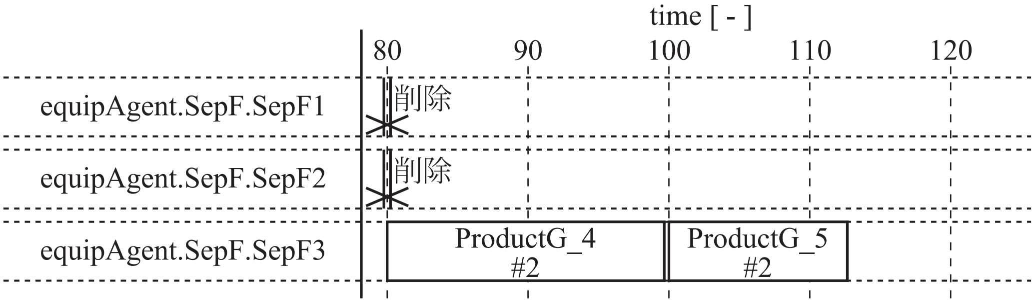
Fig.2.21　多品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の経時変化



(a) 時刻 0 から 40 までの運転結果



(b) 時刻 40 から 80 までの運転結果



(c) 時刻 80 から 120 までの運転結果

Fig.2.22　装置入れ替えシミュレーションにおける運転結果の一例

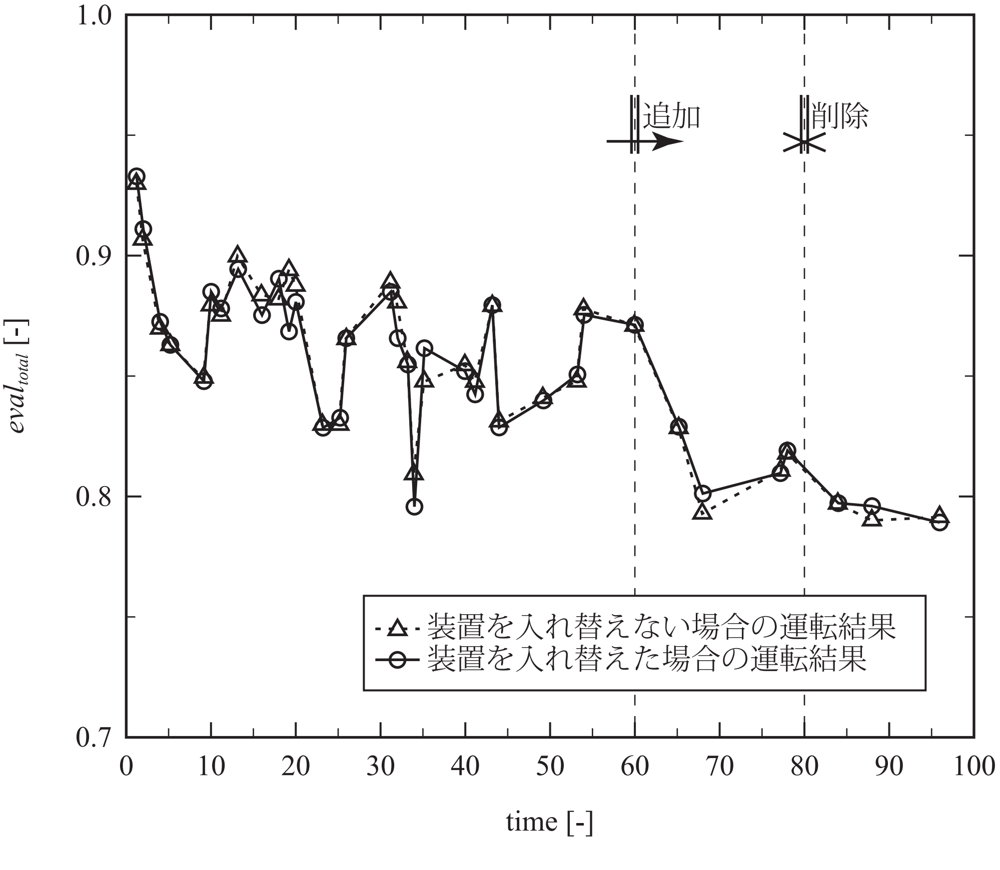


Fig.2.23　装置入れ替えシミュレーションにおける製品評価値の経時変化

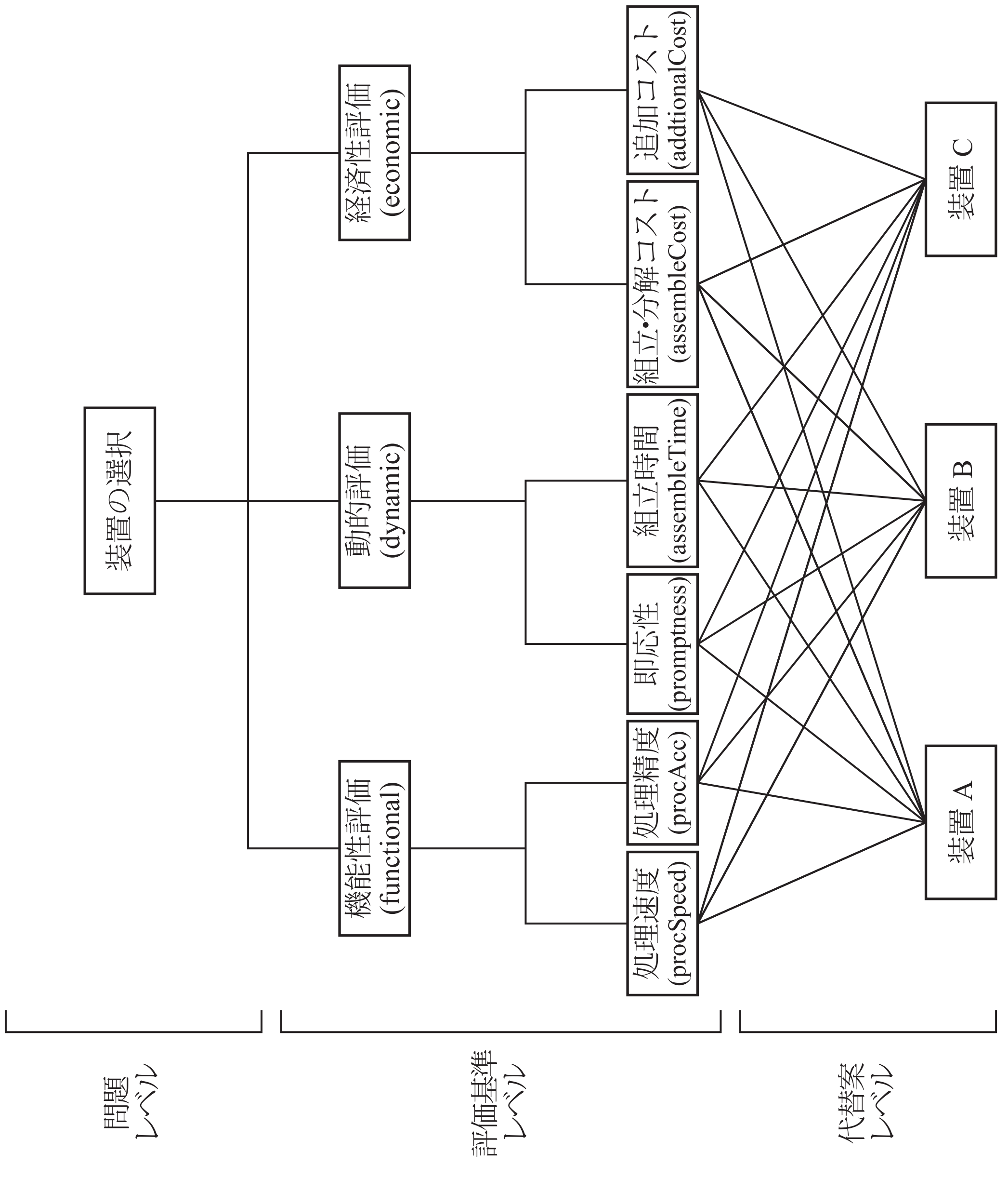
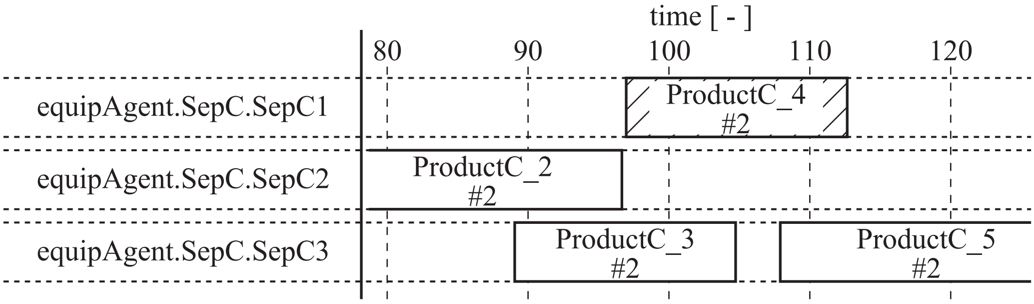
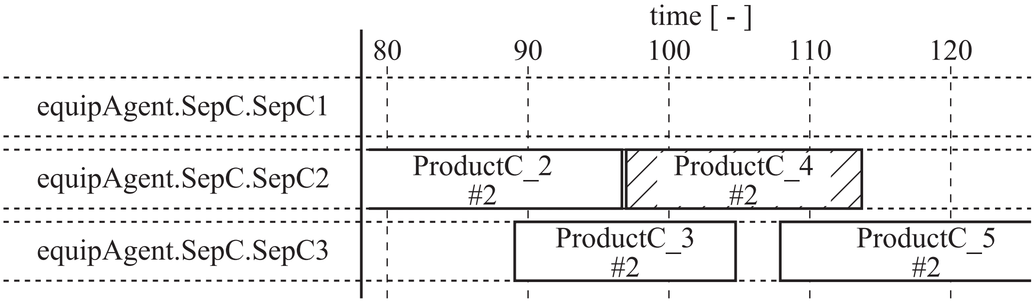


Fig.2.24 運転評価を用いない AHP の構造



(a) 運転評価を用いた運転結果の一例



(b) 運転評価を用いない運転結果一例

Fig.2.25　運転評価を用いた場合と用いない場合のシミュレーションにおける運転結果の比較

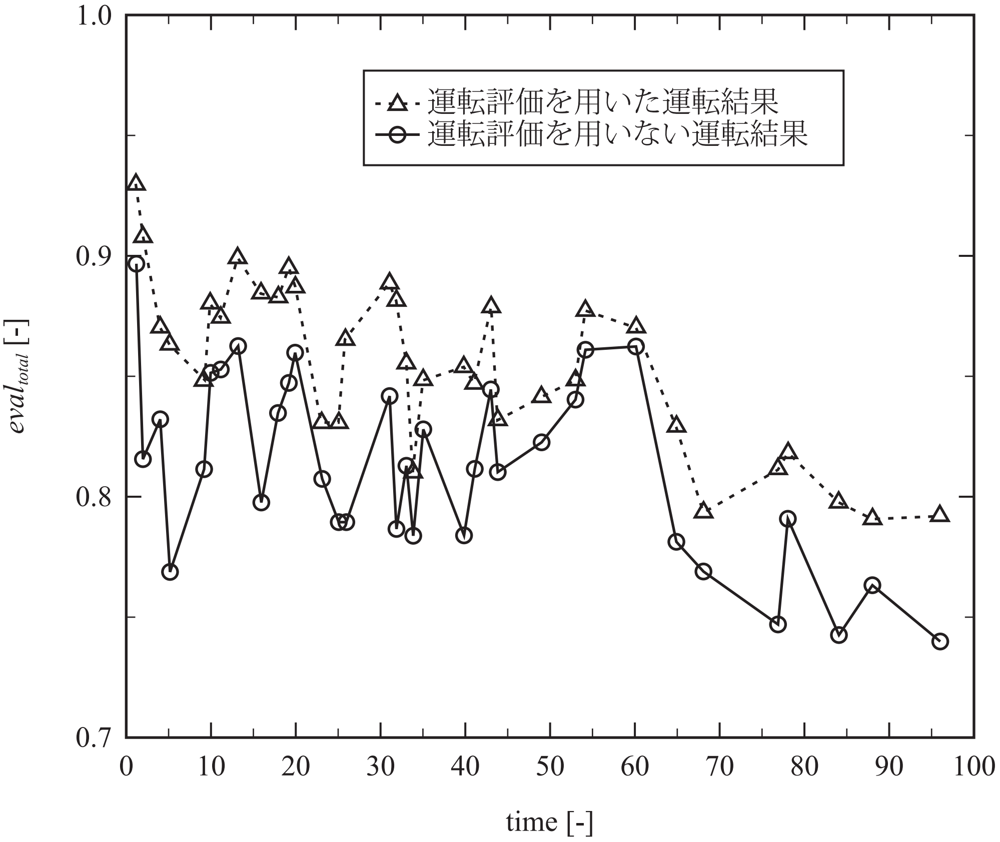


Fig.2.26　運転評価を用いないシミュレーションにおける
製品評価値の経時変化

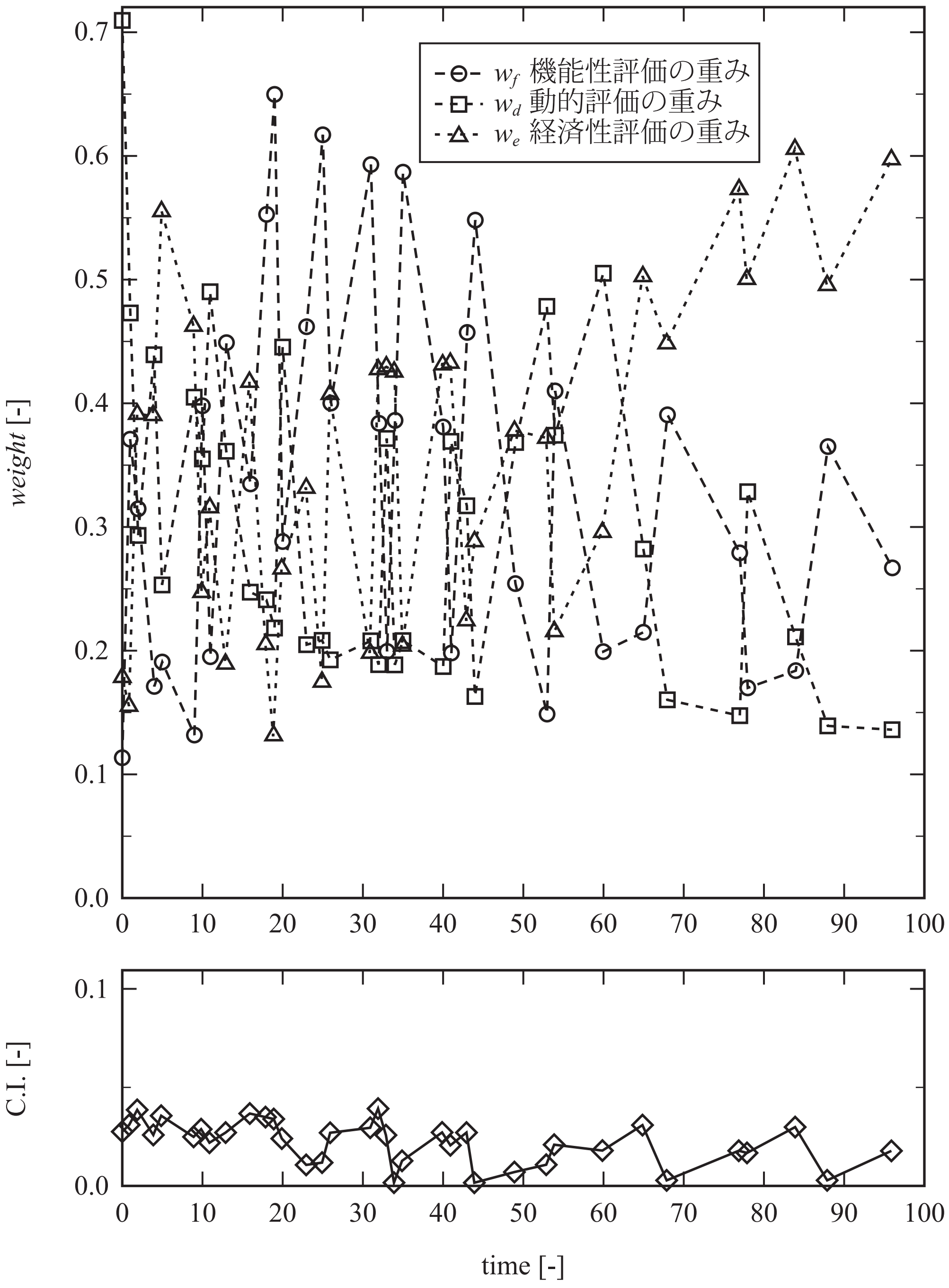


Fig.2.27　運転評価を用いないシミュレーションにおける
評価基準項目の重みおよび整合度の経時変化

2.3 本章の結論

本章では、装置や製品などの要素数の増加に伴って、大規模•複雑化するマイクロ化学プロセスにおいて、飛躍的に増大するプロセスの情報を効率的に管理するために、マルチエージェントを用いた運転操作を検討した。シミュレーションを行なうためのマルチエージェントフレームワークを検討し、装置エージェント、製品エージェント、組立エージェントおよびプロセスマネージャーを用いたシミュレーションシステムの実装を行なった。

マイクロ化学プロセスにおいて、多数の装置モジュールの追加や故障による入れ替えが頻繁に起こった場合でも、マルチエージェントシステムを用いることで、柔軟に対処することが出来た。

また、意思決定における評価項目およびその重みによって、異なる運転結果が得られた。すなわち自律的な運転のためには、化学プロセスの目的にあわせて適切な評価項目を設定し、評価項目の重みを与える必要があることがわかった。

第 3 章
意 思 決 定 機 構 の 高 度 化

エージェント間の相互作用として、協調と競合という相反する効果が挙げられる。マルチエージェントを用いた生物の進化過程のモデル化や社会における国家および企業のモデル化においては、種や国家•企業の生存を懸けた競合戦略や、生存という目的達成のために妥協の結果として協調戦略が論じられることが多い。しかし、妥協による協調は個々のエージェントの利得を上昇させる効果はあっても、必ずしもシステム全体の利得を上昇させるとは限らない。そこで、本研究では、交渉方法や協調方法を検討し、エージェントが主体的に協調戦略を選択することによって、システム全体としての処理能力が向上するような意思決定機構を検討する。具体的には、2 章で構築したマイクロ化学プロセスにおけるマルチエージェントを用いた運転管理システムに対して、製品エージェントと組立エージェント間の交渉による協調機構に確信度を導入し、種々の交渉方法における運転結果の比較を試みる。また、契約ネットプロトコルを用いた装置エージェント間の競合過程に比較優位指標を導入して、エージェントが自発的に協調することによってプロセスの処理能力を向上させる方法を検討する。

3.1 AHP への確信度の導入

社会を複数の意思決定者による決定の複合体とみなし、ゲーム理論に基づいてモ

デル化する研究が数多く行なわれている。ゲーム理論では、自らの利益の最大化を図るように行動するプレイヤー(意思決定および行動の主体)にもたらされる結果が、「自らの行動だけではなく、また、取りまく自然状況だけでもなく、まさに、相手の行動によって変わってくる [64]」ものである。プレイヤーの利益は、他のプレイヤーの利益と相反していることも多く、利害の衝突や当事者間の協調を生むため、「利己的な個人による協調行動の創発や進化 [2]」の説明として用いられることが多い。一方、集団的決定とは複数のプレイヤーの選好に基づいて、集団として 1 つの意思決定を行なうものであり、集団におけるリーダーの選出方法などが例として挙げられる。

AHP は問題を階層的に分割し、評価基準間の一対比較に基づいて評価主体の選好順序を明らかにする方法である。集団的意思決定の場合、各メンバー (評価主体) ごとに立場や価値観の相違が生じることも多いため、メンバー間の一対比較値を 1 つに絞り込む必要がある。AHP では、異なる複数の一対比較値をすり合わせる方法の一つとして、幾何平均がよく用いられる。AHP の一対比較値は 2.1.8 項で述べたように $a_{ji} = 1 / a_{ij}$ および対角成分 $a_{ii} = 1$ とならなければ整合性のある評価が出来ない。ここで、2 つのメンバーの持つ一対比較値 $a_{ij,1}$ と $a_{ij,2}$ から、新たな一対比較値 a_{ij} を計算する方法は、次式で表される。

$$a_{ij} = \sqrt{a_{ij,1} \times a_{ij,2}} \qquad (3.1)$$

この時、a_{ji} は、

$$a_{ji} = \sqrt{a_{ji,1} \times a_{ji,2}} = \sqrt{\frac{1}{a_{ij,1}} \times \frac{1}{a_{ij,2}}} = \frac{1}{a_{ij}} \qquad (3.2)$$

となり、$a_{ji} = 1 / a_{ij}$ の条件が満たされることが分かる。

2 章では、製品エージェントと組立エージェントが異なる AHP の一対比較行列を持ち、両者の値を幾何平均によってすり合わせる交渉方法によって、装置選択問題における意思決定を行なった。しかし、幾何平均によって新たな一対比較値を算出

する方法は、両者の妥協の結果ではなく、また、どちらの選好ともかけ離れてしまう場合があるという批判がある。そこで、本研究では確信度を導入し、エージェントが持つ一対比較値に対する自信の度合いを反映させることとした。

$$a_{ij} = \sqrt[(c_{ij,1}+c_{ij,2})]{a_{ij,1}{}^{c_{ij,1}} \times a_{ij,2}{}^{c_{ij,2}}} \tag{3.3}$$

ここで、$c_{ij,1}$ は、一方のエージェントが持つ一対比較値 $a_{ij,1}$ に対する確信度であり、$c_{ij,2}$ はもう一方のエージェントが一対比較値 $a_{ij,2}$ に対して持つ確信度である。確信度は通常は 1 〜 10 の範囲とし、場合によっては $c_{ij} = 0$ の場合もあるとした。値が大きいほど一対比較値に対するエージェントの自信の度合いが高いとした。

3.1.1 シミュレーション対象および方法

製品エージェントと組立エージェントの交渉方法の違いにより、マイクロ化学プロセスの運転結果を比較するシミュレーションを行なった。シミュレーション対象は 2 章のシミュレーションと同様に混合→反応→分離の 3 つの単位操作からなり、製品の品種により、それぞれの単位操作における使用可能装置タイプが異なる設定である。最大 7 品種のレシピを用い、単位操作ごとに 6 種類の装置タイプ、計 18 種類の装置タイプを設定し、同一の装置タイプでも装置ごとに性能が異なるものとした。

2 章で用いた交渉方法は、製品エージェント、組立エージェントそれぞれが持つ一対比較値の幾何平均をとったものである。言い換えると、$c_{ij,1} = c_{ji,2} = 1$ とみなすことができる。本章では、2 章のシミュレーションと運転結果を比較するために、$c_{ij,1}$ もしくは $c_{ij,2}$ が 0 の交渉方法や、$c_{ij,1}$ および $c_{ij,2}$ に任意の値を設定した交渉方法など次の (i) 〜 (iv) に示す 4 種類の交渉条件で検討を行なった。

(i) 対等な交渉条件

製品エージェントと組立エージェントが持つ一対比較値の幾何平均をとり、装置選択問題の意思決定を行なう。すなわち $c_{ij,1} = c_{ji,2} = 1$ の場合であり、2 章で用いた交渉方法である。

(ii) 組立エージェントが優位な交渉条件

組立エージェントが持つ一対比較値を用いて、装置選択問題の意思決定を行なう。すなわち $c_{ij,1} = 0$ である。

(iii) 製品エージェントが優位な交渉条件

製品エージェントが持つ一対比較値を用いて、装置選択問題の意思決定を行なう。$c_{ij,2} = 0$ である。

(iv) 確信度を用いた交渉条件

任意の値 (1 ～ 10) に設定した確信度を用いて、製品エージェントと組立エージェントの一対比較値の擦り合わせを行う。

3.1.2 シミュレーション結果および考察

(1) 少品種生産シミュレーション

2.2.3 項 (1) と同様に単一品種のみを生産するシミュレーションを行なった。装置エージェントは 12 エージェント、同一品種を生産量の異なるロット番号 1 ～ 10 番の計 10 回受注する。組立エージェントおよびプロセスマネージャーは各 1 エージェントである。製品の生産量および受注時刻 (受注間隔) はランダムであるが、4 種類の交渉条件による運転結果を比較するために、生産量や受注時刻は同一の条件とした。なお、受注時刻になると、製品の注文がプロセスマネージャーに渡されて募集が開始される。事前に各エージェントが、いつど

のような仕事が入ってくるかを知ることは出来ない。

Fig.3.1 に製品の評価値をプロットしたものを示す。横軸は製品のロット番号を示し受注順である。縦軸は製品の総合評価値 $eval_{total}$ を示す。全体的にロット番号が大きくなるに従い評価値が下がっているが、これは運転評価値が摩耗によって時間の経過とともに下がるためと考えられる。本シミュレーションで用いた条件では、すべてのロット番号で組立エージェントが優位な交渉条件 (ii) の方が、製品エージェントが優位な交渉条件 (iii) に比べて評価値が高い結果が得られた。これは、組立エージェントおよび製品エージェントに初期値として与えた AHP の一対比較行列の違いが影響しているものと考えられる。両エージェントが初期値として持つ各評価基準項目に対する一対比較値とその一対比較値に対する確信度の値を Table 3.1–3.4 に示す。Table 3.1 および Table 3.3 より求めた各評価基準項目の重みとその時の整合度 C.I. の値は Table 2.3 と同一である。交渉条件 (ii) では全てのロット番号の製品で、組立エージェントが始めに持っていた一対比較行列を用いて装置選択問題の意思決定を行なっている。また、交渉条件 (iii) では逆に全てのロット番号の製品で、製品エージェントが始めに持っていた一対比較行列を用いて装置選択問題の意思決定を行なっている。なお、製品エージェントの一対比較行列は製品の品種ごとに設定を行なった。従って、単一品種のみを生産する本シミュレーションでは製品エージェントが始めに持っている一対比較行列はすべて同一である。また、対等な交渉条件 (i) と確信度を用いた交渉条件 (iv) では、ロット番号が上がるにつれて、製品エージェントが優位な交渉条件 (iii) の評価値に収束していくことがわかる。これは、単一の品種、すなわち単一の一対比較行列を持つ製品エージェントと組立エージェントが何度も交渉を行なうため、相対的に製品エージェントの持つ選好の

影響が大きくなってしまうためである。しかし、確信度を用いることによって、製品エージェントが持つ選好への収束の速さが、若干遅くなることがわかる。

Fig.3.2 に交渉条件 (iv) において、装置選択問題の意思決定に用いた一対比較行列から求められた各評価基準項目の重み (w_f, w_d, w_e, w_o) と整合度 C.I. を示す。整合度は常に 0.1 以下であり、整合性がとれた評価を行っていることがわかる。交渉条件 (i) の場合のグラフを Fig.2.19 に示した。2 つの交渉条件における重みの変化を比較すると、経済性評価の重み (w_e) と運転評価の重み (w_o) の値は、交渉条件 (i) と (iv) でほぼ同じ変化をとり、かつ、すぐに収束していることがわかる。これは、組立エージェントおよび製品エージェントが始めに持っていた選好における 2 つの評価基準項目に対する重要度がほぼ同じ、すなわち両者のエージェントの意見が一致しているために、交渉の影響が小さいためと考えられる。機能性評価の重み (w_f) と動的評価の重み (w_d) は、交渉条件 (i) に比べ交渉条件 (iv) の方が、変化の割合が小さくなっていることがわかる。本シミュレーションでは、組立エージェントが持つ確信度の値が、製品エージェントが持つ値に比べて大きい。すなわち、組立エージェントの方が若干優位な交渉を行なうことが出来るためである。

(2) 多品種生産シミュレーション

2.2.3 項 (2) と同様に多品種生産シミュレーションを行なった。装置エージェントは 54 エージェント、7 品種、ぞれぞれ生産量の異なる注文を計 35 回受注する。組立エージェントおよびプロセスマネージャーは各 1 エージェントである。

Fig.3.3 に製品の評価値をプロットしたものを示す。横軸は時刻、縦軸は製品の総合評価値 $eval_{total}$ を示す。組立エージェントが優位な交渉条件 (ii) は常

に高い評価値を保ち、かつ値の変動が少ない。逆に製品エージェントが優位な交渉条件 (iii) は、評価値が低く変動が大きいことがわかる。これは、製品エージェントの持つ一対比較行列が製品の品種ごとに設定されており、多品種生産を行なう環境では種々の選好を持つ製品エージェントが次々と投入されるために、評価値の変動が大きいものと考えられる。組立エージェントと製品エージェントが対等な交渉条件 (i) や確信度を用いた交渉条件 (iv) においては、交渉条件 (iii) に比べて変動が穏やかになっていることがわかる。

Fig.3.4 に交渉条件 (iv) において、装置選択問題の意思決定に用いた一対比較行列から求められた各評価基準項目の重み (w_f, w_d, w_e, w_o) と整合度 C.I. を示す。整合度は常に 0.1 以下であり、整合性がとれた評価を行なっていることが確認出来る。重みはw_f 〜 w_oすべての値が大きく乱高下していることがわかる。少品種生産シミュレーションの場合と比べ、品種ごとに重要視している評価基準項目が異なるためである。交渉条件 (i) の場合のグラフは Fig.2.21 に示した。2 つの交渉条件における重みの変化を比較すると、交渉条件 (iv) の方が交渉条件 (iii) に比べて重みの変動が緩やかになっていることがわかる。

以上より、確信度を用いた方が、組立エージェントと製品エージェントが対等に交渉を行なうよりも、確信度を適切に設定することによって評価基準項目の重みの変動を小さくすることが出来るため、多様な製品を受注しても、安定した選好のもとに装置選択問題における意思決定を行なうことが出来ることがわかった。

Table 3.1　組立エージェントが持つ一対比較値

a_{ij}		j			
		機能性評価	動的評価	経済性評価	運転評価
i	機能性評価	**1**	1/5	1/2	1/4
	動的評価	5	**1**	5	1
	経済性評価	2	1/5	**1**	1/4
	運転評価	4	1	4	**1**

Table 3.2　組立エージェントが持つ確信度の値

c_{ij}		j			
		機能性評価	動的評価	経済性評価	運転評価
i	機能性評価	–	10	7	10
	動的評価	10	–	10	10
	経済性評価	7	10	–	10
	運転評価	10	10	10	–

Table 3.3　製品エージェントが持つ一対比較値

a_{ij}		j			
		機能性評価	動的評価	経済性評価	運転評価
i	機能性評価	**1**	5	7	1
	動的評価	1/5	**1**	3	1/2
	経済性評価	1/7	1/3	**1**	1/7
	運転評価	1	2	7	**1**

Table 3.4　製品エージェントが持つ確信度の値

c_{ij}		j			
		機能性評価	動的評価	経済性評価	運転評価
i	機能性評価	–	5	6	8
	動的評価	5	–	6	8
	経済性評価	6	6	–	8
	運転評価	8	8	8	–

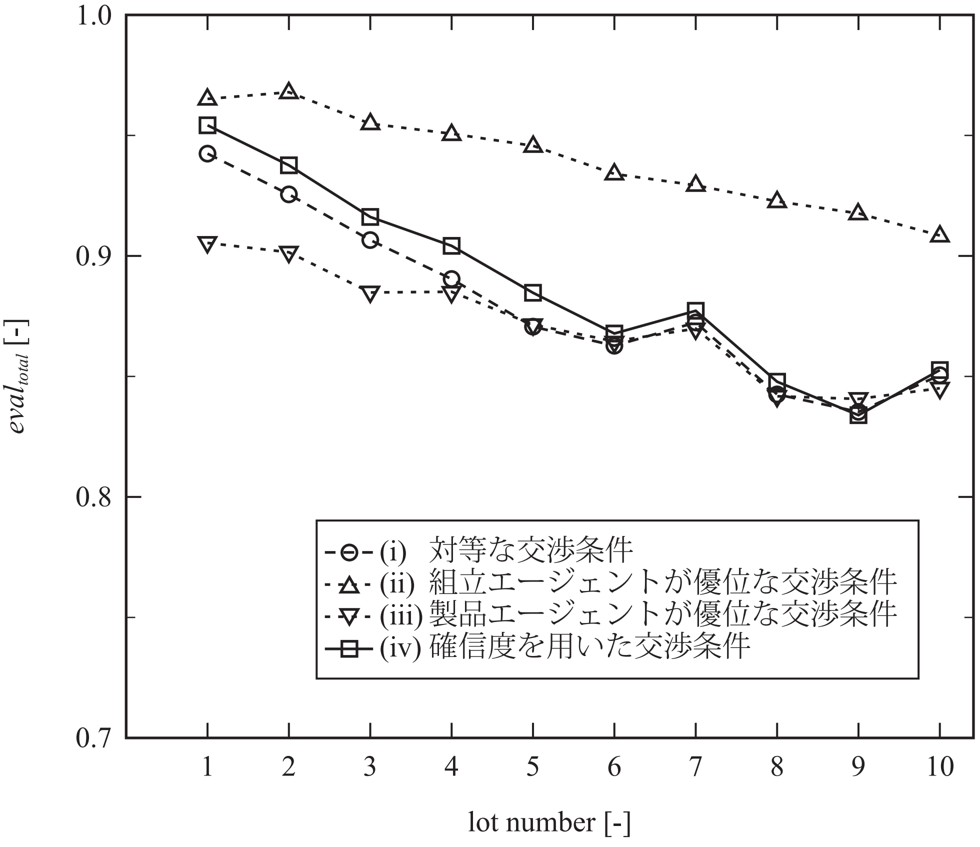


Fig.3.1　少品種生産シミュレーションにおける交渉条件の違いによる製品評価値の比較

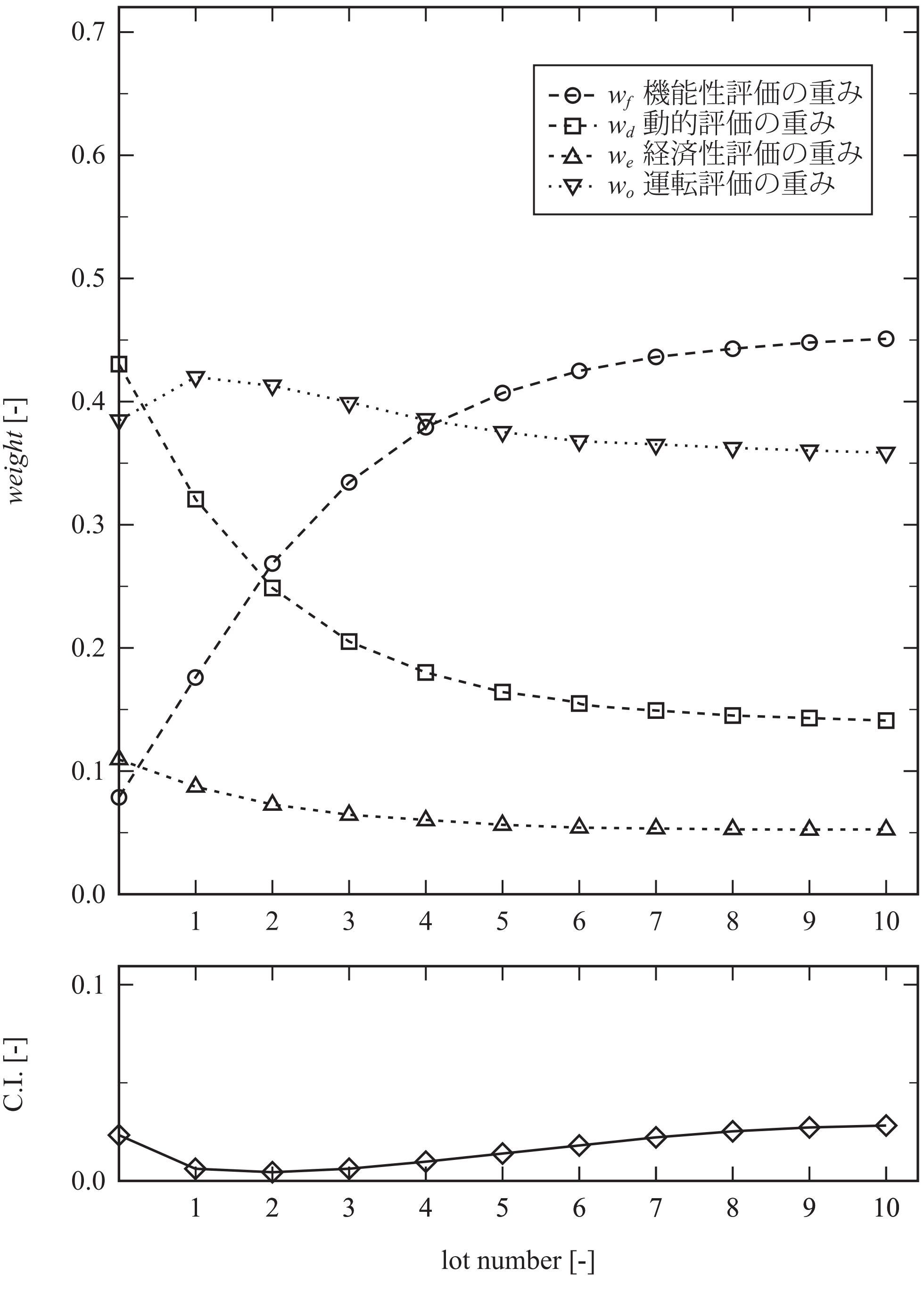


Fig.3.2　確信度を用いた少品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の推移

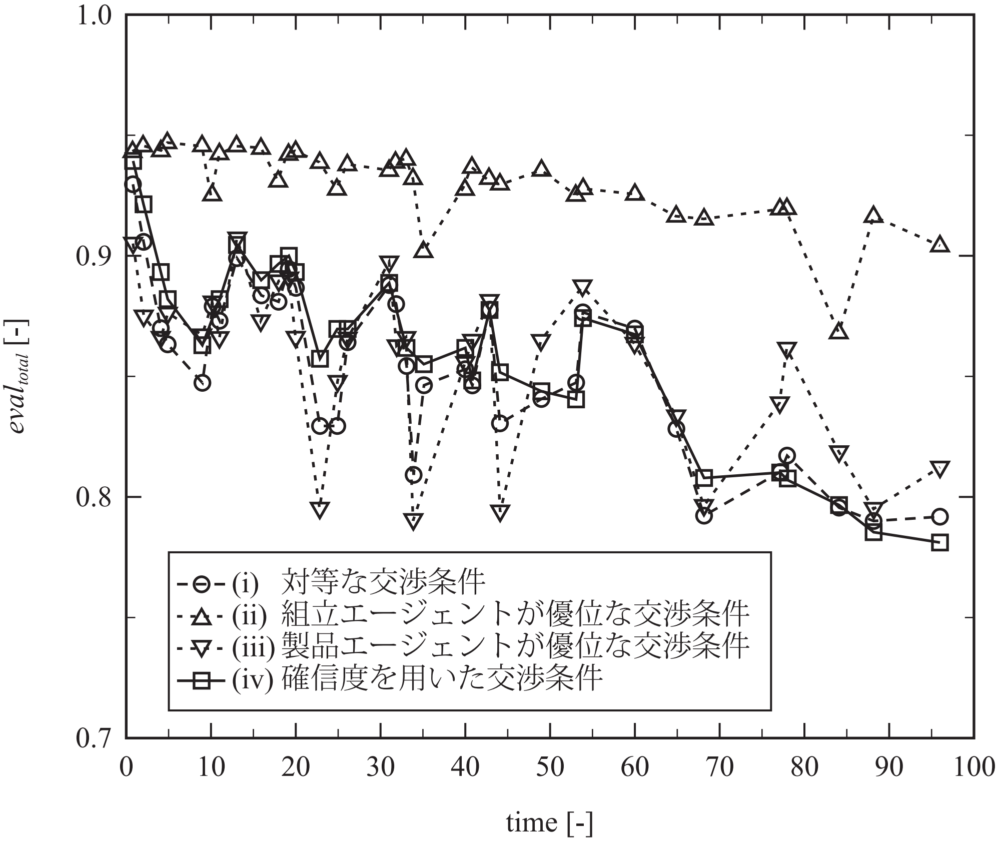


Fig.3.3 多品種生産シミュレーションにおける交渉条件の違いによる製品評価値の比較

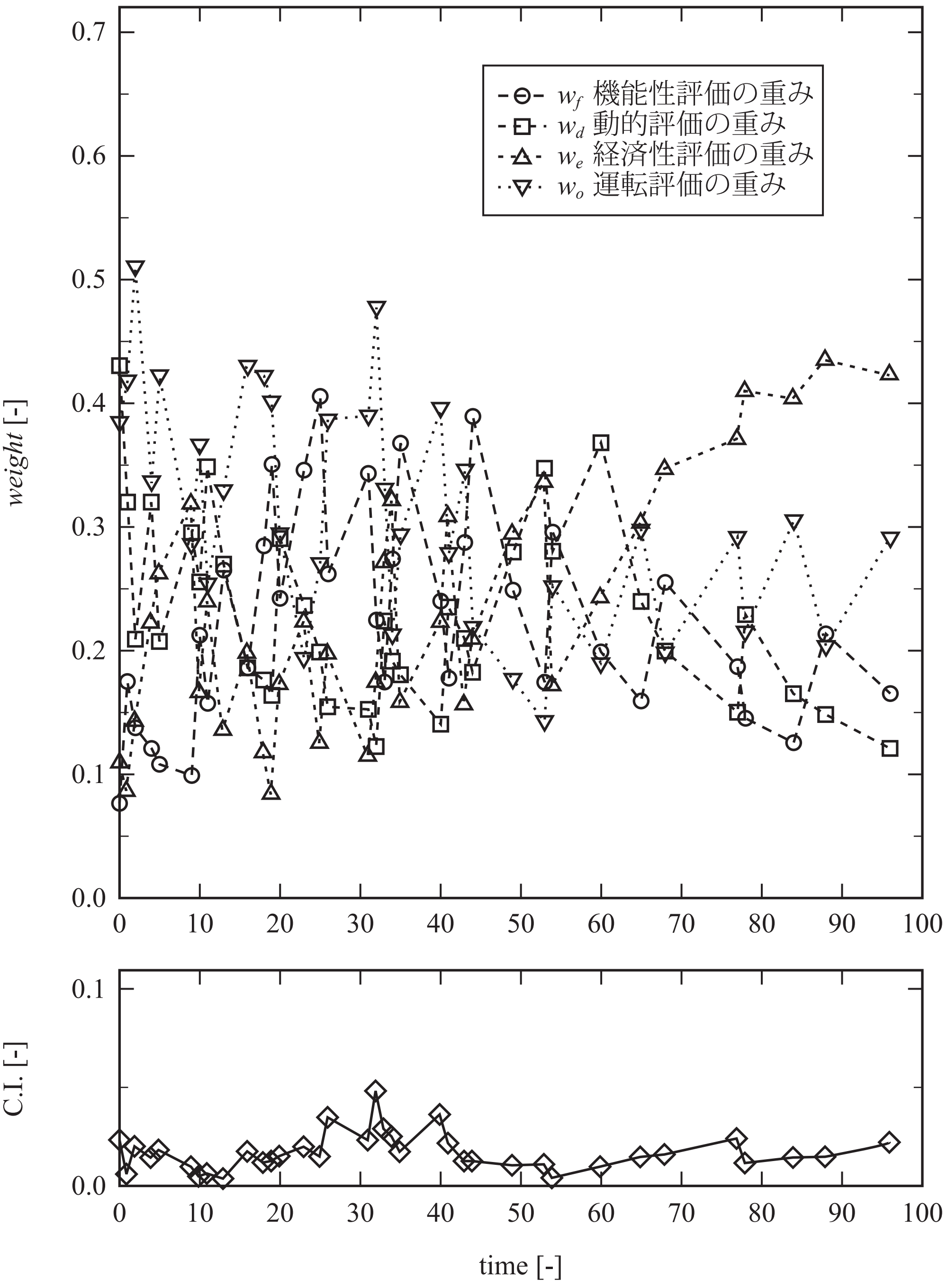


Fig.3.4　確信度を用いた多品種生産シミュレーションにおける
評価基準項目の重みおよび整合度の経時変化

3.2 比較優位指標の導入

国家や企業は、競争力を用いて他の国家や企業といった競争相手を駆逐したり、同盟 (提携) を結ぶことにより、自己の生存をはかることが最大の目的である。しかし、19 世紀にリカードによって唱えられた国際経済における比較優位の貿易理論は絶対的に優位もしくは劣位な立場にある国同士でも、貿易によって双方が益を得ることを示した。比較優位説の詳細は Appendix B に記載した。本節では、マイクロ化学プロセスのマルチエージェントを用いたプロセス運転管理システムにおいて、装置エージェントが募集を受けた仕事を比較優位指標を用いて評価し、応札すべき仕事を意思決定する機構を導入する。

3.2.1 比較優位指標

装置エージェントが製品の生産にかける時間を、生産資源とみなし装置エージェントの比較優位指標を定義する。装置エージェント k の製品 i についての処理速度を $S_{k,\,i}$ とする。また、特定ロットの製品 i の生産量を Q_i とすると、装置エージェント k における製品 i の処理時間 $t_{proc,\,k,\,i}$ は

$$t_{proc,\,k,\,i} = \frac{Q_i}{S_{k,\,i}} \tag{3.4}$$

と表される。処理にかかる時間は、装置エージェント k が投入する生産資源 (労働力) とみなすことができる。装置エージェント k における製品 j 基準の製品 i の比較生産費 $compLabor_{k,\,i,\,j}$ は

$$compLabor_{k,\,i,\,j} = \frac{t_{proc,\,k,\,i}}{t_{proc,\,k,\,j}} \tag{3.5}$$

と表すことが出来る。“製品 i と製品 j にかかる処理時間の比” であり、この値が 1 より小さければ、装置エージェント k は製品 i に対して生産性が高いことを示す。すなわち、“装置エージェント k は製品 i に対して比較優位である” ということが出

来る。式 (3.5) は、2 財のみを比較する場合である。3 財以上の場合、すなわち募集が n ある場合には、任意の 2 製品ごとのすべての組み合わせについて比較生産費 $compLabor_{k,\,i,\,j}$ を計算し、それら値の平均値をとることで装置エージェント k における製品 i に対する比較優位指標 $compAdv_{k,\,i}$ とする。

$$
\begin{aligned}
compAdv_{k,\,i} &= \frac{1}{n-1}\sum_{j=1}^{n} compLabor_{k,\,i,\,j} \\
&= \frac{1}{n-1}\sum_{j=1}^{n} \frac{t_{proc,\,k,\,i}}{t_{proc,\,k,\,j}} \qquad (j \neq i\,,\, n \geq 2)
\end{aligned}
\tag{3.6}
$$

装置エージェント k は、この値が小さいほど製品 i に対して生産性が高いとみなすことが出来る。製品エージェントは、この指標を用いて募集を受けた n の製品の評価を行なう。比較優位指標が最低値を示す製品、すなわち装置エージェント k が最も優位な立場にある製品に対してのみ応札を行なう。

2 章および前節 (3.1) まで用いてきた方法では、装置エージェントはすべての募集に対して即座に応札を行なっていた。しかし、比較優位指標の導入にあたり、一定時間、他のロットの募集メッセージを待つこととした。なお、一定時間経過後も、他の募集メッセージが来なかった場合には、装置エージェントはその募集に対して必ず応札を行なうこととした。他の募集メッセージを待つ時間間隔は、初めに受けた募集メッセージの入札締切時刻までとした。ある装置エージェントにおける募集メッセージと応札の概要を Fig.3.5 に示す。製品 1 〜製品 4 の 4 つの募集メッセージをそれぞれ時刻 $t_{adv,\,1}$ 〜 $t_{adv,\,4}$ に受信し、その入札締切時刻を $t_{close,\,1}$ 〜 $t_{close,\,4}$ とする。装置エージェントは、最初に製品 1 に対する募集メッセージを受け取るが、すぐには応札せずに時刻 $t_{close,\,1}$ まで待機する。その間に製品 2 および製品 3 についての募集メッセージを受け取る。時刻 $t_{close,\,1}$ の時点で、それまでに受信した 3 件の募集メッセージについて比較優位指標を用いて評価を行ない、最も優位な製品に対してのみ入札メッセージを送付する。ここでは、製品 2 に対し

て入札したとする。時刻 $t_{close,2}$ の時点では、すでに製品 2 に対する入札が行なわれているため、新たに評価を行なうことはない。また、時刻 $t_{close,3}$ では未入札の製品 3 と新たに募集メッセージを受け取った製品 4 の 2 件の製品がある。しかし、製品 3 は時刻 $t_{close,1}$ で既に評価が行なわれているため、再度、評価対象とされることはない。従って、装置エージェントは時刻 $t_{close,4}$ まで待機し、他の募集メッセージを受信した場合には同様に比較優位指標を用いて評価•選択を行ない、他の募集メッセージがない場合には製品 4 に対する入札を行なう。

このような方法を用いると、製品の募集に対して 1 件も入札がない場合が考えられる。入札締切時刻の時点で製品エージェントが受信した入札件数が 0 件の場合、製品エージェントは再募集をかけることが出来る。また、再募集の場合には、募集メッセージに設定された募集のプライオリティを上げることにより、確実に受注されるようにした。

3.2.2 シミュレーション対象および方法

装置エージェントに意思決定能力を持たせ、比較優位指標を用いたマイクロ化学プロセスの運転シミュレーションを行なった。シミュレーション対象は 2 章で対象としたシミュレーションと同様に混合→反応→分離の 3 つの単位操作からなり、製品の品種により、それぞれの単位操作における使用可能装置タイプが異なる設定とした。

3.2.3 シミュレーション結果および考察

(1) 2 装置—2 製品シミュレーション

比較優位の貿易理論における 2 国—2 財モデルと同様に、2 装置エージェン

トと 2 つの製品を対象とした。製品 1 および製品 2 に対する装置エージェント A および装置エージェント B の処理速度の値を Table 3.5 に示した。処理速度の値が大きいほど高速であり生産性が高い。すなわち装置エージェント B に比べ装置エージェント A の方が、2 種類の製品ともに生産性が高いため、“装置エージェント A が絶対優位”である。製品 1 の生産量 Q_1 を 3000、製品 2 の生産量 Q_2 を 4000 とする。その時の装置エージェント A および装置エージェント B の処理時間を Table 3.6 に示す。2 財 (製品) の場合、(3.5) 式と (3.6) 式は $n = 2$ の条件で次のように変形することが出来る。

$$compAdv_{k,\,i} = compLabor_{k,\,i,\,j} = \frac{1}{compAdv_{k,\,j}} \tag{3.7}$$

装置エージェント A および装置エージェント B における製品 1 についての比較優位指標は $compAdv_{A,\,1} = 1.25$, $compAdv_{B,\,1} = 0.60$ となる。すなわち装置エージェント A は製品 2 に対して優位性を持ち、装置エージェント B は製品 1 に対して優位性を持つことを意味する。装置エージェントは優位性を持つ仕事にのみ入札を行なうため、製品 1 および製品 2 はそれぞれ 1 つずつの入札を受信する。

Table 3.5　各装置エージェントの各製品についての処理速度

	製品 1	製品 2
装置エージェント A	$S_{A,\,1} = 600$	$S_{A,\,2} = 1000$
装置エージェント B	$S_{B,\,1} = 500$	$S_{B,\,2} = 400$

Table 3.6　各装置エージェントの各製品についての処理時間

	製品 1 ($Q_1 = 3000$)	製品 2 ($Q_2 = 4000$)
装置エージェント A	$t_{proc,\,A,\,1} = 5$	$t_{proc,\,A,\,2} = 4$
装置エージェント B	$t_{proc,\,B,\,1} = 6$	$t_{proc,\,B,\,2} = 10$

その結果、製品エージェントは入札してきた装置エージェントと契約することとなる。

ここで、2 装置—2 製品の条件下における運転結果の組み合わせを検討してみると、Table 3.7 のように 4 パターンのみであることがわかった。時刻 0 から処理を始めた場合の製品 1 および製品 2 の両方の処理が完了する時刻が終了時刻であり、装置エージェント A および装置エージェント B の運転時間の合計を総運転時間として表中に示した。比較優位指標を用いたシミュレーションでは、パターン 1 が選択され本条件下における最短の終了時刻をとったことがわかる。

Table 3.7　2 装置—2 製品における運転結果の組み合わせ

	装置エージェント A	装置エージェント B	終了時刻	総運転時間
パターン 1	製品 2	製品 1	6	4 + 6 = 10
パターン 2	製品 1	製品 2	10	5 + 10 = 15
パターン 3	製品 1, 製品 2	-	9	5 + 4 = 9
パターン 4	-	製品 1, 製品 2	10	6 + 10 = 16

(2) 2 装置—3 製品シミュレーション

次に 2 装置エージェントと 3 つの製品を対象とした。製品 1 ～製品 3 に対する装置エージェント A および装置エージェント B の処理速度の値を Table 3.8 に示す。本条件においても、装置エージェント A の方が絶対優位である。また、製品 1 の生産量 Q_1 を 3000、製品 2 の生産量 Q_2 を 6000、製品 3 の生産量 Q_3 を 5000 とした時の処理時間を Table 3.9 に示す。(3.6) 式を用いて計算した比較優位指標 $compAdv_{k,i}$ を Table 3.10 に示す。シミュレーション結果を Fig.3.6 に示す。Fig.3.6(a) は従来用いてきた即時入札方式による運転結果であり、Fig.3.6(b) は比較優位指標を用いて、装置エージェントが仕事を評価•選択

Table 3.8　各装置エージェントの各製品についての処理速度

	製品 1	製品 2	製品 3
装置エージェント A	$S_{A,1} = 600$	$S_{A,2} = 1000$	$S_{A,3} = 700$
装置エージェント B	$S_{B,1} = 500$	$S_{B,2} = 400$	$S_{B,3} = 300$

Table 3.9　各装置エージェントの各製品についての処理時間

	製品 1 ($Q_1 = 3000$)	製品 2 ($Q_2 = 6000$)	製品 3 ($Q_2 = 5000$)
装置エージェント A	$t_{proc,A,1} = 5$	$t_{proc,A,2} = 6$	$t_{proc,A,3} = 8$
装置エージェント B	$t_{proc,B,1} = 6$	$t_{proc,B,2} = 15$	$t_{proc,B,3} = 17$

Table 3.10　各装置エージェントの各製品についての比較優位指標

	製品 1	製品 2	製品 3
装置エージェント A	$compAdv_{A,1} = 1.354$	$compAdv_{A,2} = 0.613$	$compAdv_{A,3} = 1.280$
装置エージェント B	$compAdv_{B,1} = 0.694$	$compAdv_{B,2} = 0.993$	$compAdv_{B,3} = 1.530$

した場合の結果である。なお、製品 3 は再募集により装置エージェント A が受注した。(a) と (b) を比較すると、(a) では 3 つの製品すべての処理終了時刻が 15、総運転時間が 28 であるのに対し、(b) では処理終了時刻が 14、総運転時間が 20 となり、比較優位指標を用いることで処理終了時刻および総運転時間が短い効率的な運転を行なったことがわかる。ここで、従来の即時入札方式では募集メッセージが送付される順番により、運転結果が異なることがわかった。Fig.3.6(a) では、製品 1 →製品 2 →製品 3 の順に募集した結果である。比較のために即時入札方式で募集順序を変えてシミュレーションを行なった。結果を Fig.3.7 に示す。従来の即時入札方式を用いた場合の運転結果は Fig.3.7(a) 〜 (c) の 3 パターンに分類することが出来る。仕事の割り当てにおける組み合わせパターンとしては、全ての製品を一方の装置エージェントが処理する Fig.3.7(d) のパターンなども考えられるが、本システムで即時入札方式を用いた場合には、処理終了時刻や総運転時間が Fig.3.7(c) よりも悪いパターンが選択されることはなかった。Fig.3.7(a) 〜 (c) の 3 パターンの中では (c) のパターンの効率が悪く、(a) のパターンが処理終了時刻および総運転時間で見た場合の効率が最も良く、対象とする 2 装置—3 製品シミュレーション条件における最適値であり、比較優位指標を用いることで最適値が選択されたことがわかった。本節で導入した比較優位指標を用いることにより、募集順序が前後しても、安定したよい解が得られることがわかった。

以上より、比較優位指標を用いて装置エージェントが入札を行なう仕事を評価•選択することによって、システム全体として処理能力が向上したことがわかる。すなわち、装置エージェントは協調のための妥協戦術を取るのではなく、複数の製品の募集情

報を比較し、自己に優位な選択を行なうことにより、結果的にシステム全体の処理能力向上を図ることが出来た。従って、エージェントが利己的な行動を取ることにより協調効果を出すことが出来た。

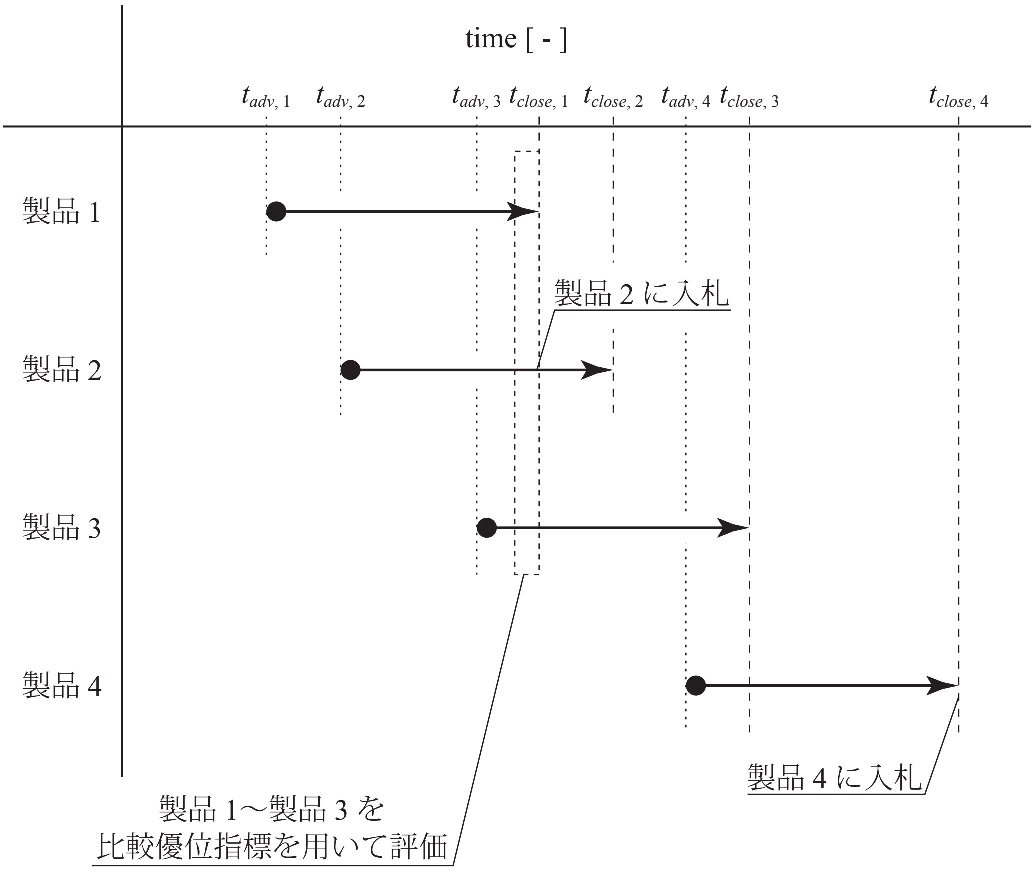
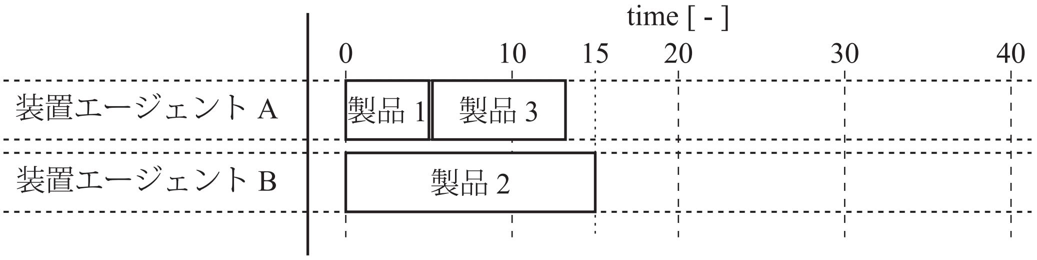
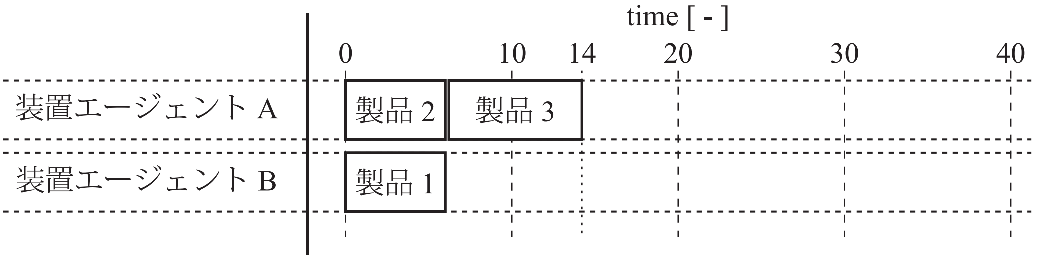


Fig.3.5 複数の募集メッセージに対する入札の様子

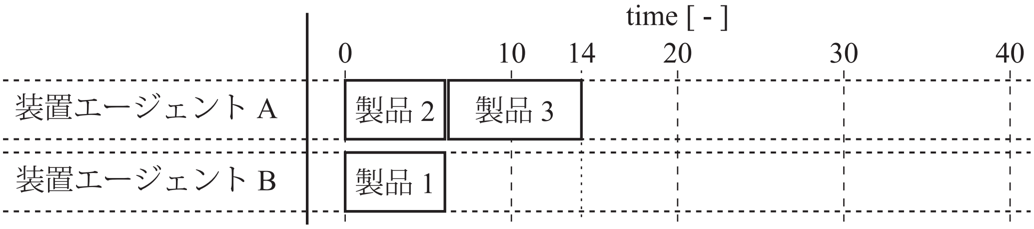


(a) 従来の即時入札方式による運転結果
(処理終了時刻 :15, 総運転時間 :28)

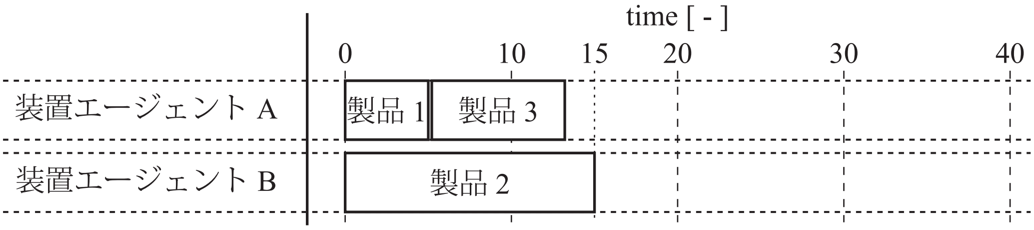


(b) 比較優位指標を用いた運転結果
(処理終了時刻 :14, 総運転時間 :20)

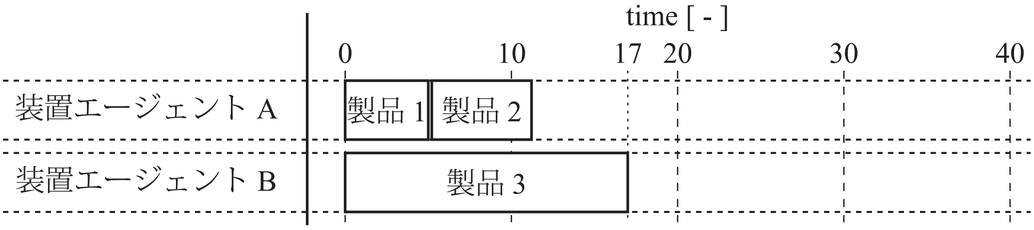
Fig.3.6　装置エージェントの入札方法の違いによる運転結果の比較



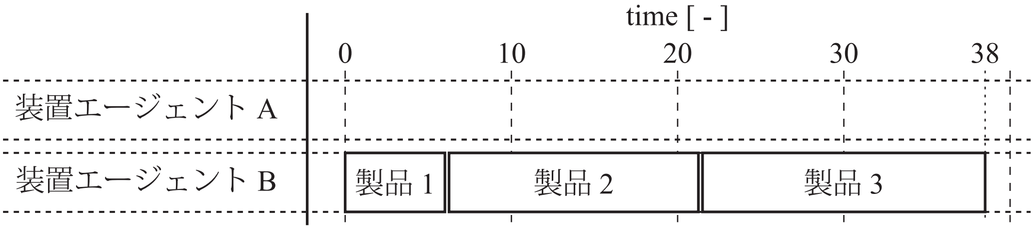
(a) 製品 2 →製品 1 →製品 3 の順に募集した結果
(処理終了時刻 :14, 総運転時間 :20)



(b) 製品 1 →製品 2 →製品 3 の順に募集した結果
(処理終了時刻 :15, 総運転時間 :28)



(c) 製品 1 →製品 3 →製品 2 の順に募集した結果
(処理終了時刻 :17, 総運転時間 :28)



(d) 最悪パターン (処理終了時刻 :38, 総運転時間 :38)

Fig.3.7　募集順序の違いによる運転結果の比較

3.3 本章の結論

本章では、エージェントが主体的に協調行動をとることによって、システム全体としての処理能力向上を目指し、AHP への確信度の導入による交渉方法の改良と、装置エージェントが比較優位指標を用いて募集された仕事を評価•選択することによる、より高度な協調意思決定機構の構築を行なった。

確信度を用いることにより、組立エージェントと製品エージェントが対等に交渉を行なった場合に比べ、確信度の値を適切に設定することによって評価基準項目の重みの変動を小さく出来ることがわかった。従って、多様な製品を受注しても、安定した選好のもとに意思決定を行なうことが出来ることがわかった。

また、装置エージェントが募集された仕事を比較優位指標を用いて評価し、入札すべき仕事を選択することによって、システム全体としての処理能力を向上させることが出来た。特に、本システムでは、他の装置エージェントと交渉を行なわなくても、装置エージェントが利己的にすなわち自己の優位な仕事に対して入札を行なうことにより、システム全体としての処理能力向上という協調効果を得ることが出来た。

第 4 章
データベースからのシミュレーションモデルの動的な取得

現在、盛んに研究•開発が行なわれているマイクロ化学プロセス技術を用いて、新たな化学プロセスを設計するために、知識データベースシステムが必要である。また、知識データベースとマルチエージェントを用いたシミュレーションシステムを接続することにより、種々のマイクロ化学デバイスや操作条件におけるデバイスの性能やプロセス挙動を推算することが出来れば、柔軟な設計システムを構築することが出来る。そこで、本章ではデータベースから種々のシミュレーションモデルを動的に取得してシミュレーションを行なうために、知識データベースの構造や知識データの XML 化を検討する。また、エージェントが種々のシミュレーションモデルを利用するために、マイクロ化学デバイスおよびシミュレーションモデルのクラス構造を検討し、エージェントとシミュレーションモデルのインターフェースについて検討する。

4.1 知識データベース

実用化を目指して、マイクロ化学プロセス技術の研究•開発が盛んに行なわれている。これらの研究開発を支援し、研究者間での円滑な情報交換を促進するために、マイクロ化学プロセス技術•知識を体系的に格納する知識データベース "*MDCOs* (<u>M</u>icro Chemical Technology <u>D</u>atabase for <u>Co</u>llaboration and <u>Co</u>nstruction)" を構築し

た。4.1.1 項ではデータベースの構造について述べ、4.1.2 項では知識データ格納方法について述べる。

4.1.1 知識データベースの構造

マイクロ化学プロセスは、移動現象、化学反応、計測•制御、微細加工 (MEMS) 技術など種々の分野にまたがる技術である。本研究では、研究分野を "マイクロ材料加工"、"マイクロ機構設計"、"マイクロ移動現象"、"マイクロ化学反応"、"マイクロデバイス•ユニット操作"、"マイクロ計測"、"マイクプロセス制御•システム" の 7 分野とし、それぞれの分野の専門家の協力を得て、技術の体系化を図った。これらの多様な研究分野における研究者間で研究開発を支援するために、分野を意識せずに情報を取得出来る検索機能とそれぞれの分野の知識データを体系的にまとめた目次を閲覧するためのカテゴリー閲覧 (一般的にはディレクトリ型とも呼ばれる) 機能を持たせた。Fig.4.1 に *MDCOs* の構造の概要を示す。知識データは XML 化され、ファイルシステム上に保存されている。知識データの XML 化については次項 (4.1.2) で述べる。これらの知識データを効率的に検索するために、リレーショナルデータベースを用いて知識のインデックス化を行なう。また、*MDCOs* エンジン部分はユーザ管理やセッション管理を行なう部分と検索機能を持つ。XML 化された知識は HTML 変換モジュールを通すことにより HTML 文書に成形されるため、ユーザは特別なソフトウェアを必要とせずに一般的な web ブラウジングと同じように PC 上の Web ブラウザを用いて知識の閲覧や検索を行なうことが出来る。

本知識データベース上の *MDCOs* エンジン部、HTML 変換モジュールおよび XML ファイルをリレーショナルデータベースにインデックス化するためのモジュー

ルは、すべて Java Servlet 技術を用いて実装した。Servlet とは、本来、蓄積された情報 (静的コンテンツ) のみを送出する機能しか持たない HTTP サーバ (web サーバとも呼ばれる) において、動的なコンテンツ (ユーザのステータスなどを反映してコンテンツが変化するもの) を生成するためのサーバ側プログラムである。Perl などの開発言語を用いた CGI(Common Gateway Interface) もサーバ側プログラムとしてよく知られている。しかし、CGI は HTTP リクエストごとにプログラムの起動→実行→終了を行なうため実行速度が遅いなどの問題がある。本研究で用いた Servlet では、サービス開始 (サーバ起動) 時に Servlet エンジンと呼ばれる Java プログラムの実行環境を起動し、各プログラムモジュールを必要に応じてメモリに読み込むため高速に動作することが出来る。すなわち HTTP リクエストごとにプログラムを起動するオーバーヘッドがなく、速度面のみならずマシンへの負荷も少ない利点がある。また、Perl 言語に比べてユーザ認証や HTTP セッションの管理も容易に実装することが出来る。Java Servlet 技術を用いることで一般的に得られるこれらの利点以外にも、本研究では、Java 言語でプログラムされたマルチエージェントシミュレーションシステムと接続する場合には、同じ言語で書かれたサーバ側プログラムの方が、スムースな情報交換が可能になるという利点が挙げられる。

4.1.2 知識の XML 化

国内外の文献や特許•研究成果等の種々の知識データを収集し、それらの知識データの再利用性を高めることを目的として XML を用いた知識表現を検討した。Figs.4.2–4.5 に本研究で策定した XML 構造を UML クラス図で示す。Fig.4.2 はトップレベルのエレメントである。ルートノード (最上位階層) である `research`

エレメントは、本 XML フォーマットのバージョン (version) 属性や知識タイプ (type) 属性を持つ。知識タイプは ‘実験 (experiment)’、‘シミュレーション (simulation)’、‘実験およびシミュレーション (expsim)’、‘モデル (model)’ の 4 種類である。ルートノードの下には、ヘッダー情報 (header エレメント) や調査対象•方法 (research-method エレメント)、概要 (abstract エレメント) などのエレメントが記述される。

Fig.4.3 に header エレメント以下の XML 構造を UML クラス図で示す。personnel エレメントは知識の調査者 (又は提供者) であり、personnel/name エレメントに氏名を、personnel/affiliation エレメントに所属機関名を記す。知識の調査者は、連名であることも考慮して、多重度は 1 以上不特定多数となっている。title エレメントは知識の題名であり、日本語または英語もしくはその両方で記述 (lang 属性に ‘jp(日本語)’ または ‘en(英語)’ を設定) することが出来る。research-category エレメントおよび research-subcategory は、研究分野ごとに体系的に整理された目次の ID(カテゴリー ID) を記述する。research-category は ‘主分類’ と呼ばれ、research-subcategory は ‘関連する研究領域’ として表示される。short-abstract エレメントも研究概要である。ルートノード直下にも概要 (/research/abstract エレメント) が存在するが、short-abstract エレメントはより短い概要であり、主に検索結果リスト表示時に使用している。fric-leafid エレメントは “FRIC-ISOP” との接続のための情報である。FRIC-ISOP は web サービスを用いた外部の情報解析プラットフォームである。relate_id エレメントは、本知識に関連する知識 ID を記述することが出来る。

Fig.4.4 に research-method エレメント以下の XML 構造を UML クラス図

で示す。research-method エレメントは、method 属性に 'literature(文献調査)' 又は 'patent(特許調査)'、'report(研究報告)' を指定することにより、知識の主たる調査方法を記述することが出来る。reference エレメントには文献情報を記述する。'title(参考文献の題目)'、'author(参考文献の著者)'、'source(雑誌名、巻、号、ページ)' を記述することが出来る。patent_inf には特許情報を記述する。'title(特許の名称)'、'pat_date(出願日)'、'pat_no_1(公開番号)'、'pat_no_2(公告番号)'、'inventor(発明者)'、'appllicant(出願人)' を記述することが出来る。1つの知識で、複数の参考文献や特許を参照することも可能であるため、reference エレメントおよび patent_inf エレメントの多重度は 0 〜不特定多数とした。

Fig.4.5 に model エレメント以下の XML 構造を UML クラス図で示す。model エレメントは 3 つの子ノードを持ち、それぞれ 'model_1(数値解析モデル)'、'model_2(因果関係•定性モデル)'、'model_3(構造モデル)' の 3 つである。数値解析モデルは、移動現象論や反応速度論などに基づいて立てられた代数方程式や微分方程式、種々の反応温度や流量などの操作条件下において得られた反応収率などのデータである。因果関係•定性モデルは、現象の因果関係や傾向を示し、方程式を立てるには至らないが定性的な関係を表す知識である。構造モデルは、マイクロ化学デバイスの構造や流路形状などの図面や写真を含む知識である。

Fig.4.6 に model ノードの XML 表現の一例を示す。このモデルは実験で得られたデータを表型にまとめたデータであり、HTML 変換モジュールを通すことにより、表に成形して表示することが出来る。Fig.4.7 に HTML 変換モジュールを通して表に成形した例を示す。

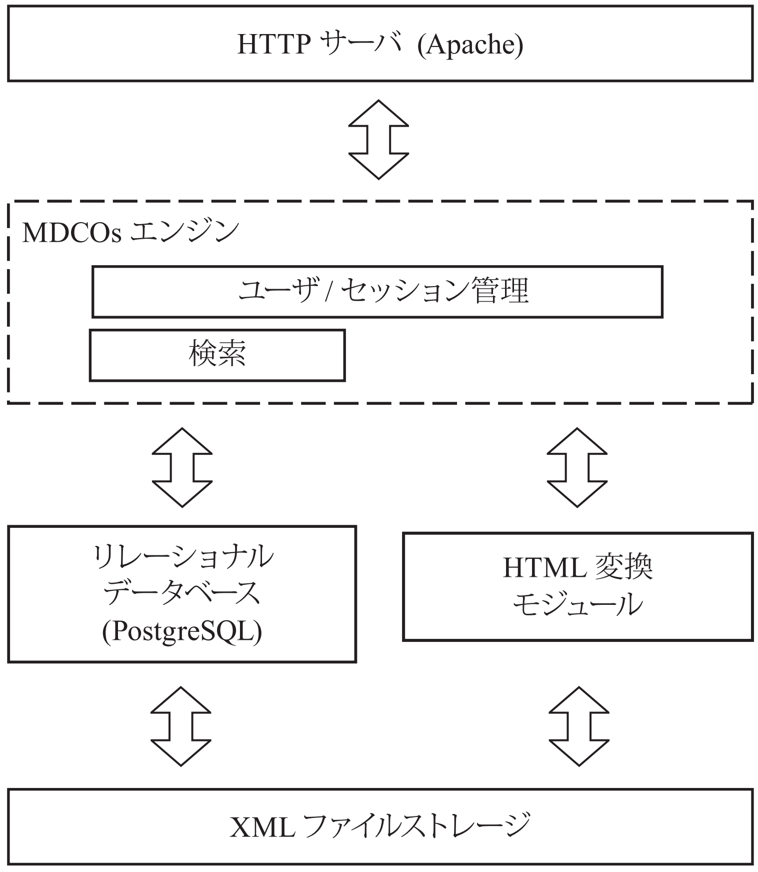


Fig.4.1　*MDCOs* 知識データベースの構造の概略

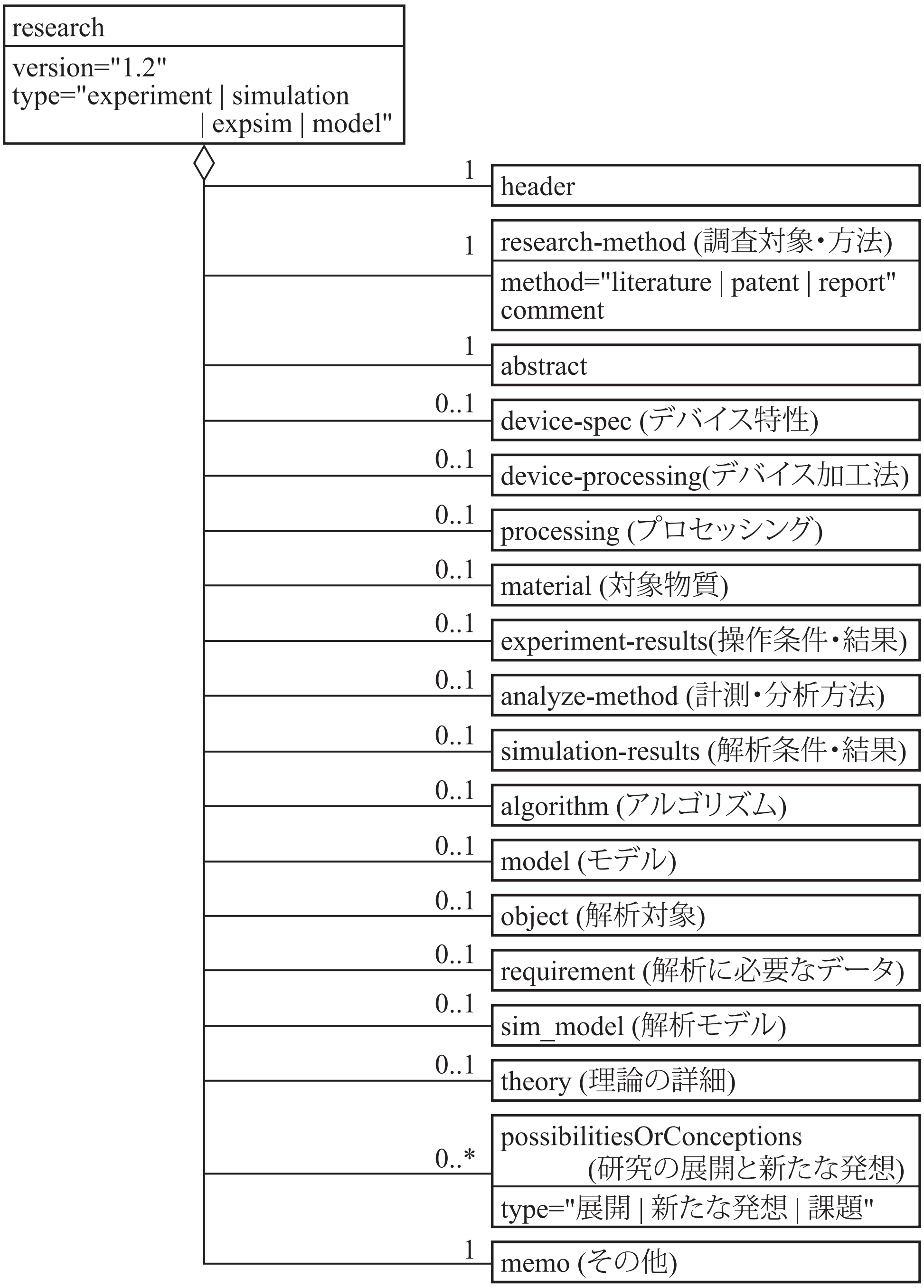


Fig.4.2　知識データの XML 構造 (research エレメント)

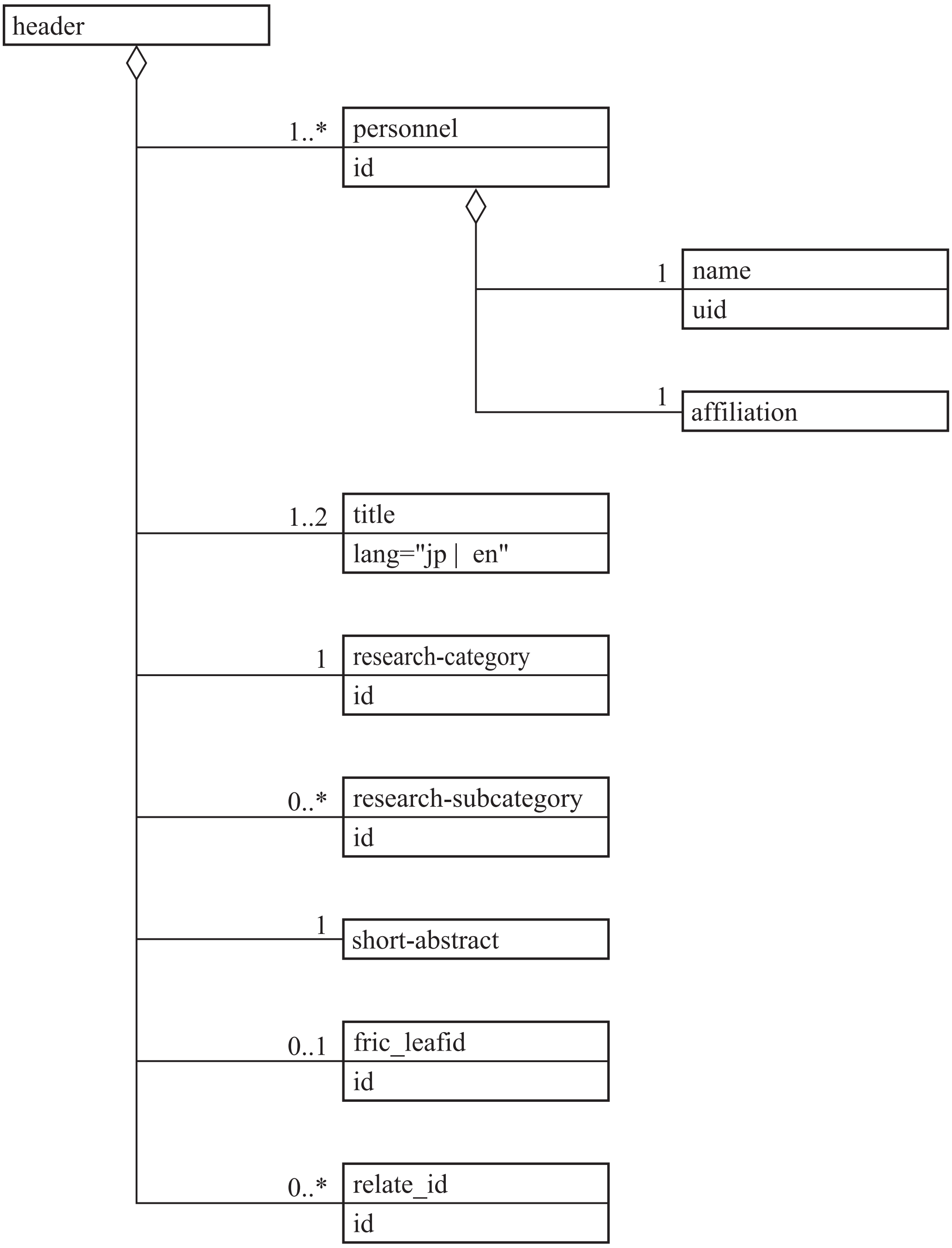


Fig.4.3　知識データの XML 構造 (header エレメント)

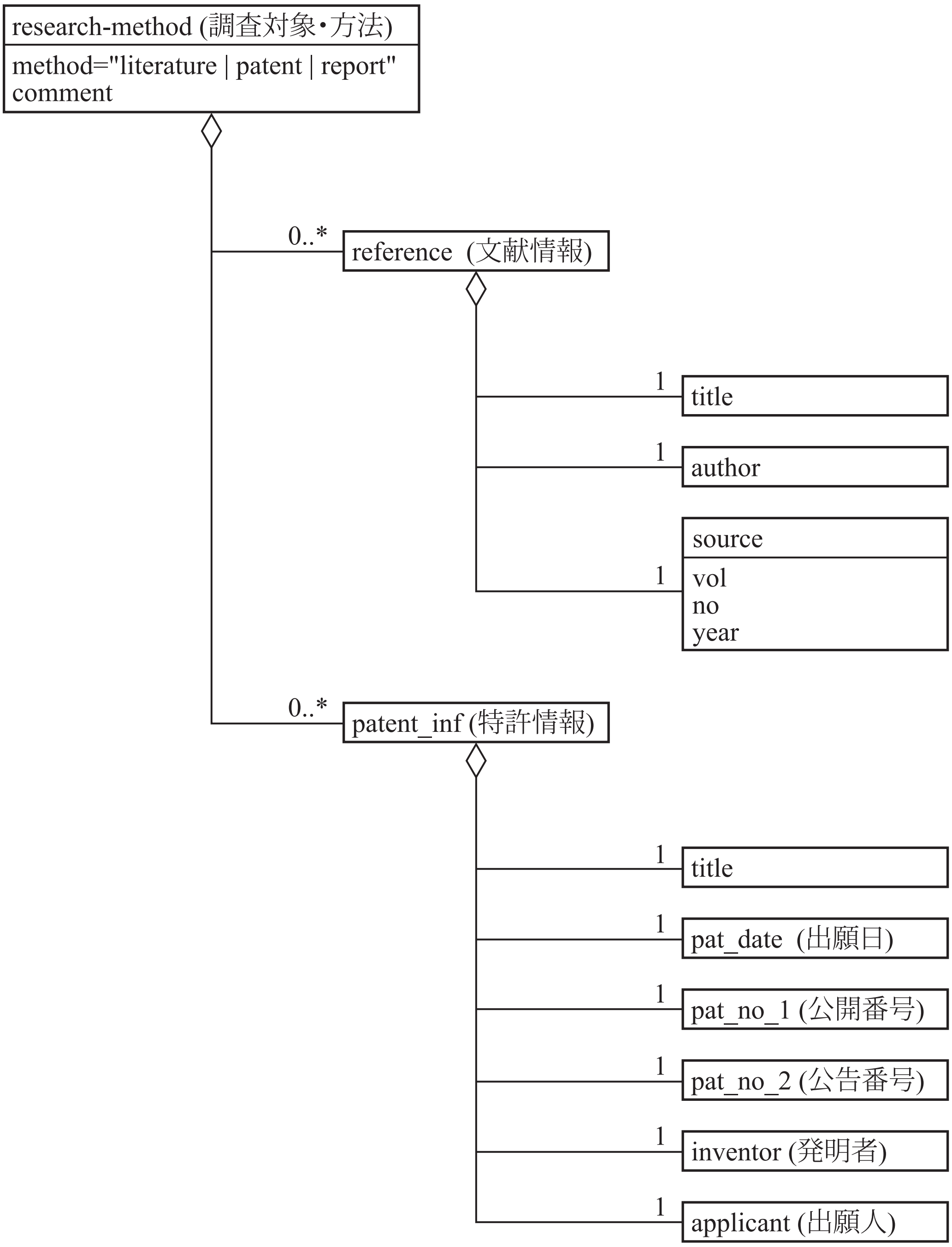


Fig.4.4　知識データの XML 構造 (`research-method` エレメント)

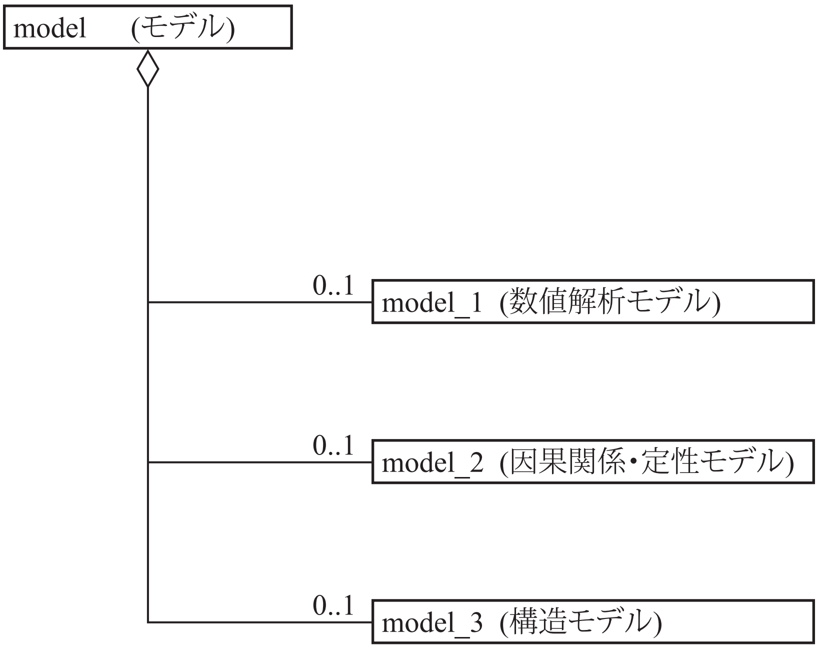
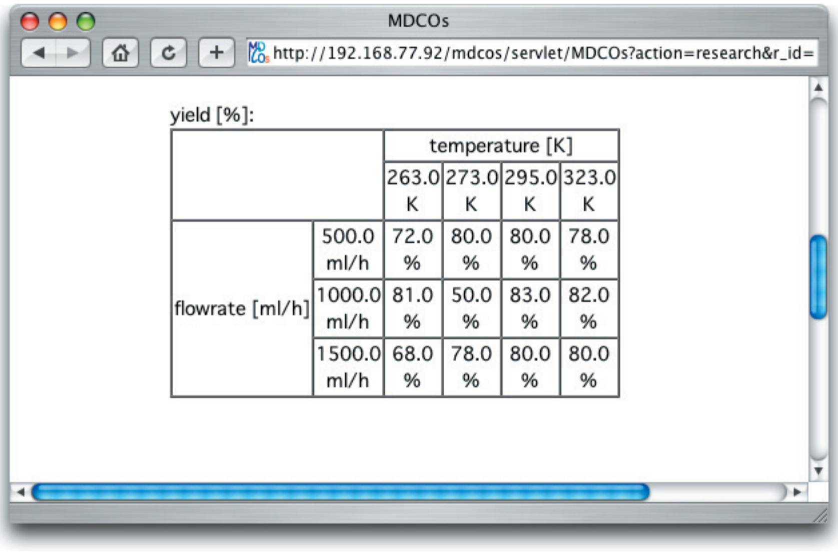


Fig.4.5 知識データの XML 構造 (`model` エレメント)

```
<model>
   <model_1>
      <multi-matrix>
         <parameter id="1">
            <name>temperature</name>
            <unit>K</unit>
            <values>
               <value id="1">263.0</value>
               <value id="2">273.0</value>
               <value id="3">295.0</value>
               <value id="4">323.0</value>
            </values>
         </parameter>
         <parameter id="2">
            <name>flowrate</name>
            <unit>ml/h</unit>
            <values>
               <value id="1">500.0</value>
               <value id="2">1000.0</value>
               <value id="3">1500.0</value>
            </values>
         </parameter>
         <parameter id="3">
            <name>yield</name>
            <unit>%</unit>
         </parameter>
         <matrix>
            <point>
               <parameter id="1" valueid="1"/>
               <parameter id="2" valueid="1"/>
               <parameter id="3" value="72.0"/>
            </point>
                    :
                    :
            <point>
               <parameter id="1" valueid="4"/>
               <parameter id="2" valueid="3"/>
               <parameter id="3" value="80.0"/>
            </point>
         </matrix>
      </multi-matrix>
   </model_1>
</model>
```

Fig.4.6　model ノードの XML 表現の一例



		temperature [K]			
		263.0 K	273.0 K	295.0 K	323.0 K
flowrate [ml/h]	500.0 ml/h	72.0 %	80.0 %	80.0 %	78.0 %
	1000.0 ml/h	81.0 %	50.0 %	83.0 %	82.0 %
	1500.0 ml/h	68.0 %	78.0 %	80.0 %	80.0 %

Fig.4.7　表形式に成形された知識データの一例

4.2 デバイスのクラス化

すでに種々の機能や形状を持つマイクロ化学デバイスが提案されている。また、種々の機能や形状を持つ新規のデバイスが今後も開発されていくことが予想される。そこで、これらのデバイスと装置エージェントを対応させ、種々のマイクロ化学デバイスを用いたシミュレーションを行なうために、マイクロ化学デバイスのクラス構造を検討した。本研究では、2 つのアプローチからデバイスのクラス構造を検討した。4.2.1 項では、デバイスの機能に基づいたクラス化について述べ、4.2.2 項では、デバイスの幾何学的な形状に基づいたクラス化について述べる。

4.2.1 デバイスの機能に基づくクラス化

Fig.4.8 にマイクロ化学デバイスの "機能" に基づいて分類した例を UML クラス図で示す。抽象クラスである 'MicroDevice' クラスは、デバイスの機能に基づいて 'MicroMixer' クラスや 'MicroReactor' クラス、'MicroHeatExchanger' クラスに分類される。さらに 'MicroMixer' クラスは、形状の異なる 'SSIMicroMixer' クラスおよび 'CubicMicroMixer' クラスに分類され、'MicroMixer' クラスの混合機能 (`mix( )`) を継承する。同様に 'MicroHeatExchanger' クラスの熱交換機能 (`heatExchange( )`) は、具象クラス 'CubicMicroHeatExchanger' が継承する。このような機能に基づく分類は、化学装置の分類方法としては一般的な方法だと言える。Fig.4.9 に化学装置便覧 [65] による種々の化学装置の分類の一部を示す。'塔槽類'、'高圧装置' や '真空装置'、'伝熱装置' といった装置の機能に基づく大分類がある。例えば、'伝熱装置' ではその下階層に '熱交換器'、'蒸発缶'、'加熱炉' といった機能に基づく分類がなされる。さらに '熱交換器' の下階層で '多管式熱交換器' や '二重管式熱交換器' などの熱交換器の形状に基づく分類がなされて

いる。

しかし、このような化学装置の機能に基づく分類では、複数の機能を持つ装置が開発されると、それらの機能を併せ持つクラス (分類) を追加する必要がある。Fig.4.9 において‘常圧塔槽類 – 塔類 – 蒸留塔’は蒸留機能 (`distill( )`) を持ち、‘伝熱装置 – 蒸発缶’ は蒸発機能 (`evaporate( )`) を持つ。ここで、‘真空装置 – 真空蒸発缶’ は真空機能 (`evacuate( )`) および蒸発機能、‘真空装置 – 真空蒸留装置’ は真空機能および蒸留機能を持つ。同様に高圧装置も加圧機能 (`pressurize( )`) と他の機能を併せ持つクラスである。Fig.4.8 のマイクロ化学デバイスの分類においても、‘MicroMixerReactor’ クラスは混合機能 (`mix( )`) と反応機能 (`react( )`) を併せ持つクラスである。今後、新たな機能や、機能同士の新たな組み合わせを持つマイクロ化学デバイスが開発された場合、その組み合わせごとに新たな抽象クラスを定義する必要があり、その抽象クラスの下にデバイスの形状に基づいたクラスを定義していくため、クラス構造が複雑化していくことが予想される。

4.2.2 デバイスの形状に基づくクラス化

Fig.4.10 にマイクロ化学デバイスの “幾何学的形状” に基づいて分類した例を UML クラス図で示す。抽象クラス ‘MicroDevice’ クラスは、デバイスの形状に基づいて ‘CubicMicroDevice’ クラスや ‘StackableMicroDevice’ クラス、‘HPLCJointMicroDevice’ クラスに分類される。‘CubicMicroDevice’ クラスは、さらに ‘CubicMicroMixer’ クラスや ‘CubicMicroReactor’ クラスに分類される。これらのマイクロ化学デバイスの機能は、‘*MicroMixer*’ インターフェースや ‘*MicroReactor*’ インターフェースで定義される。具象クラスである ‘CubicMicroMixer’ クラスや

'CubicMicroReactor' クラス、'SSIMicroMixer' クラスは、これらのインターフェースを "実装" することで機能を実現する。また、複数の機能を併せ持つマイクロ化学デバイスは、それに応じて複数のインターフェースを実装すれば良い。例えば、'CubicMicroMixerReactor' クラスは、'*MicroMixer*' インターフェースおよび '*MicroReactor*' インターフェースを実装する。

このように機能をインターフェースとして定義することで、複数の機能を持つマイクロ化学デバイスも容易に表現することが出来る。新たな機能を持つマイクロ化学デバイスが開発された場合には、該当する機能を単一のインターフェースとして定義する。複数の機能同士の新たな組み合わせを持つマイクロ化学デバイスは、インターフェースとして定義した "機能" を複数実装すれば良い。また、新たな形状のマイクロ化学デバイスは、それらの形状に対応した抽象クラスを定義してクラス構造を拡張すれば良い。従って、前項 (4.2.1) で述べたデバイスの機能に基づくクラス化よりも、デバイスの形状に基づいてクラス化した方が、新たな機能や形状の追加、それらの新しい組み合わせを持つマイクロ化学デバイスを容易にかつシンプルに表現することが出来る。以上のことから、本研究ではデバイスの形状に基づいたマイクロ化学デバイスのクラス構造を用いることとする。

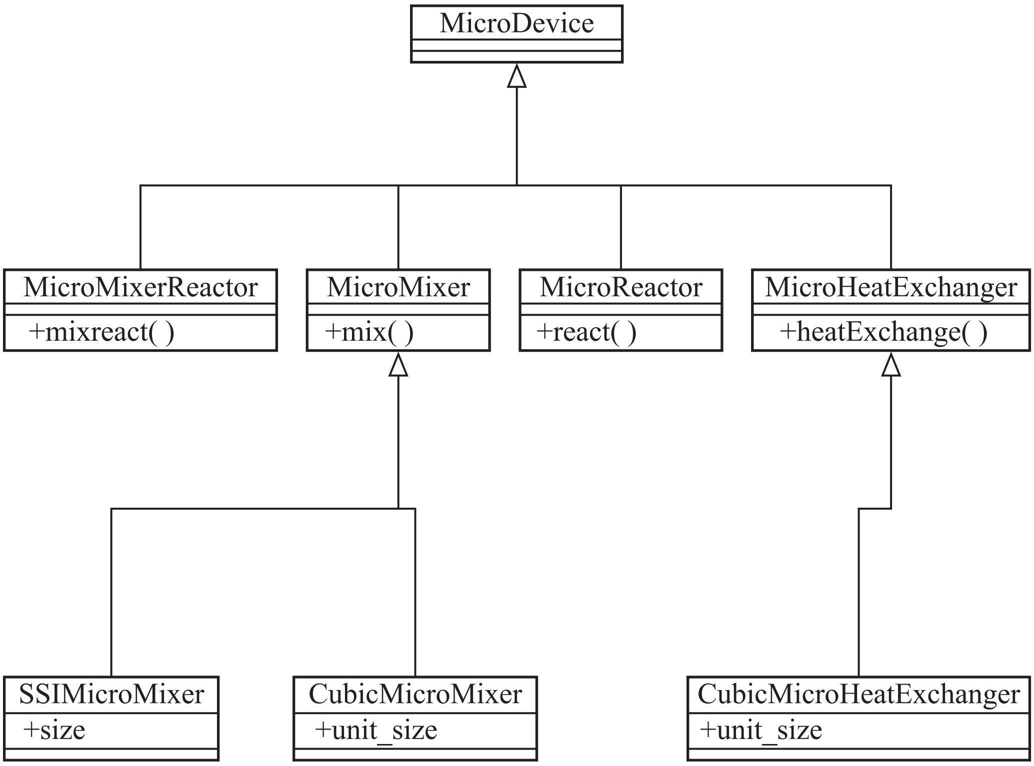


Fig.4.8　マイクロ化学デバイスの機能に基づくクラス図

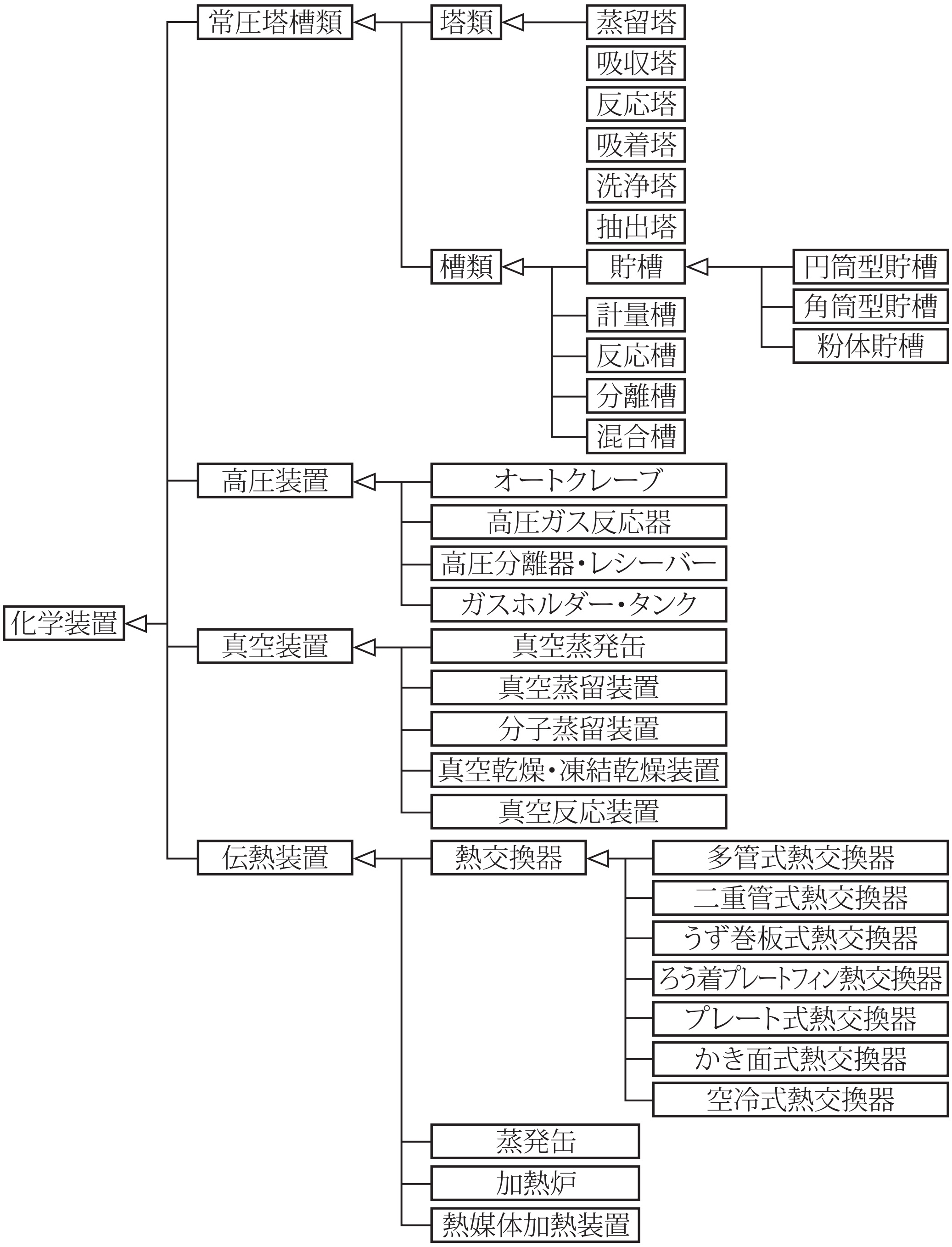


Fig.4.9　化学装置便覧による化学装置の分類の一部

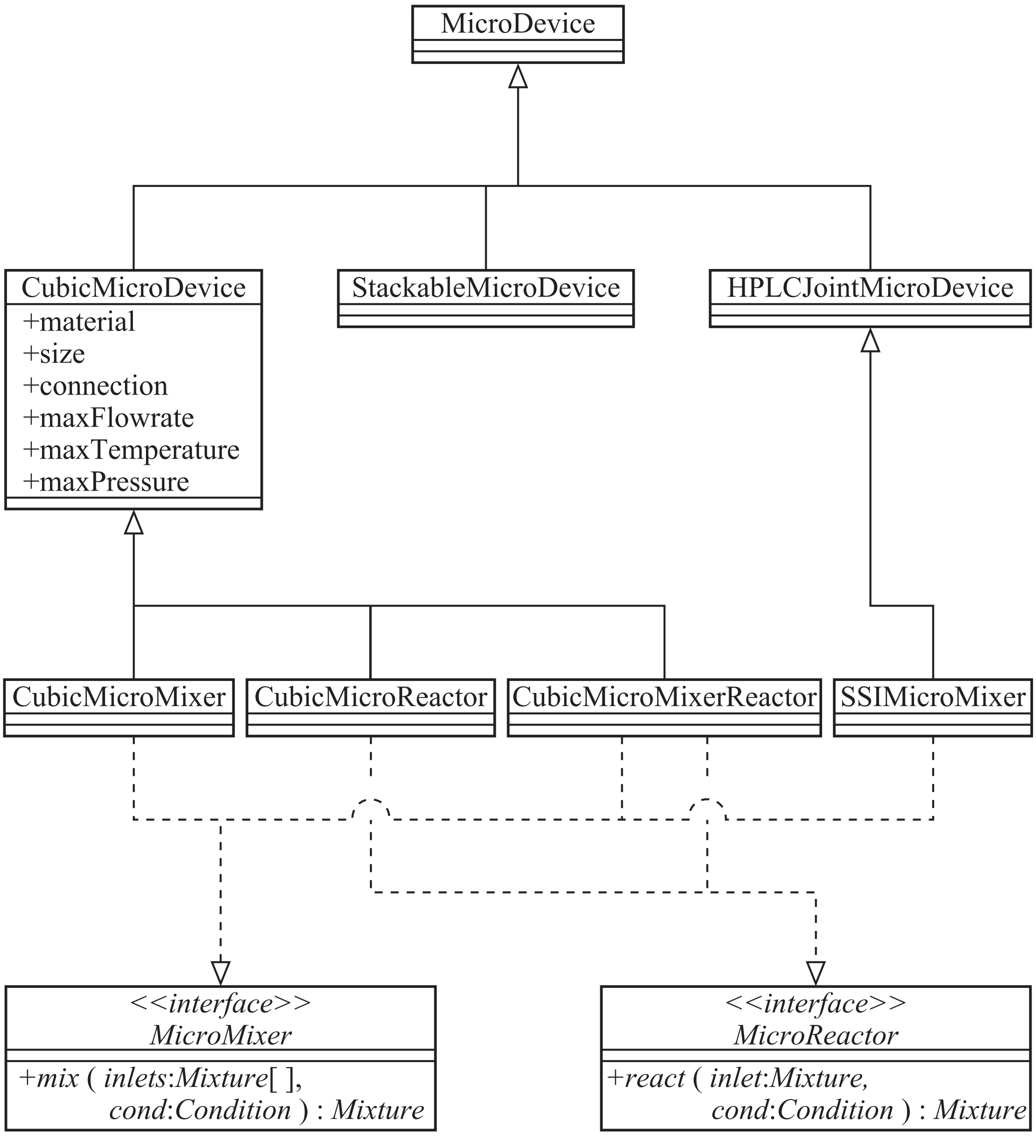


Fig.4.10　マイクロ化学デバイスの装置の幾何学的形状に基づくクラス図

4.3 シミュレーションモデルのクラス化

前節で述べたマイクロ化学デバイスの形状に基づくクラス構造のインターフェースに対応するように、シミュレーションモデルのクラス化を行なった。Fig.4.11 にシミュレーションモデルの UML クラス図を示す。本研究で扱うシミュレーションモデルは、実験データを基に結果を表形式でまとめた表型モデルや、実験結果を実験式にフィッティングしたモデル、現象論に基づく式を用いたモデルなど種々のタイプをモデルとして扱う。Fig.4.11 で示した具象クラス 'TubularReactorTableModel'、'TubularReactorModelA'、'TubularReactorModelB' は、同一の '管型反応器 (tubular reactor)' を対象としたシミュレーションモデルであるため、'*MicroReactorModel*' インターフェースを実装する。ただし、'TubularReactorTableModel' は実験結果を基にして表型モデル ('TableSimModel' クラス) を継承し、'TubularReactorModelA' および 'TubularReactorModelB' は実験式に基づく関数型のシミュレーションモデルであるため、関数型モデル ('AlgebraSimModel' クラス) を継承する。これらの具象クラスが "実装" するインターフェースにおいてメソッドを定義することにより、メソッドの引数や返値を統一することが出来る。具体的には、'*MicroMixerModel*' インターフェースの持つ `mix( )` メソッドの引数は、装置の複数の入口 ('Mixture' クラスの配列変数 `inlets[]`) と操作条件 ('Condition' クラスの変数 `cond`) であり、返値は装置の出口 ('Mixture' クラス) である。'*MicroReactorModel*' インターフェースの持つ `react( )` メソッドの引数は装置の入口 ('Mixture' クラスの変数 `inlet`) と操作変数 ('Condition' クラスの変数 `cond`) であり、返値は装置の出口 ('Mixture' クラス) である。

このようにインターフェースを用いることにより、シミュレーションモデルの中身やモデルタイプの違いを意識することなく、種々のシミュレーションモデルを扱うこと

が出来る。すなわち、これらの '*MicroReactorModel*' インターフェースを実装したクラスは、決められた引数と返値の `react( )` メソッドを必ず持つため、このメソッドを呼び出すことでシミュレーションモデルを用いた計算を行なうことが出来る。したがって、モデルのタイプや計算方法が異なる種々のシミュレーションモデルを、シミュレーションの目的に合わせて動的に取得することが可能となる。また、これらのシミュレーションモデルをデータベースに格納し、実装しているインターフェースで絞り込むことによって容易に検索が可能である。

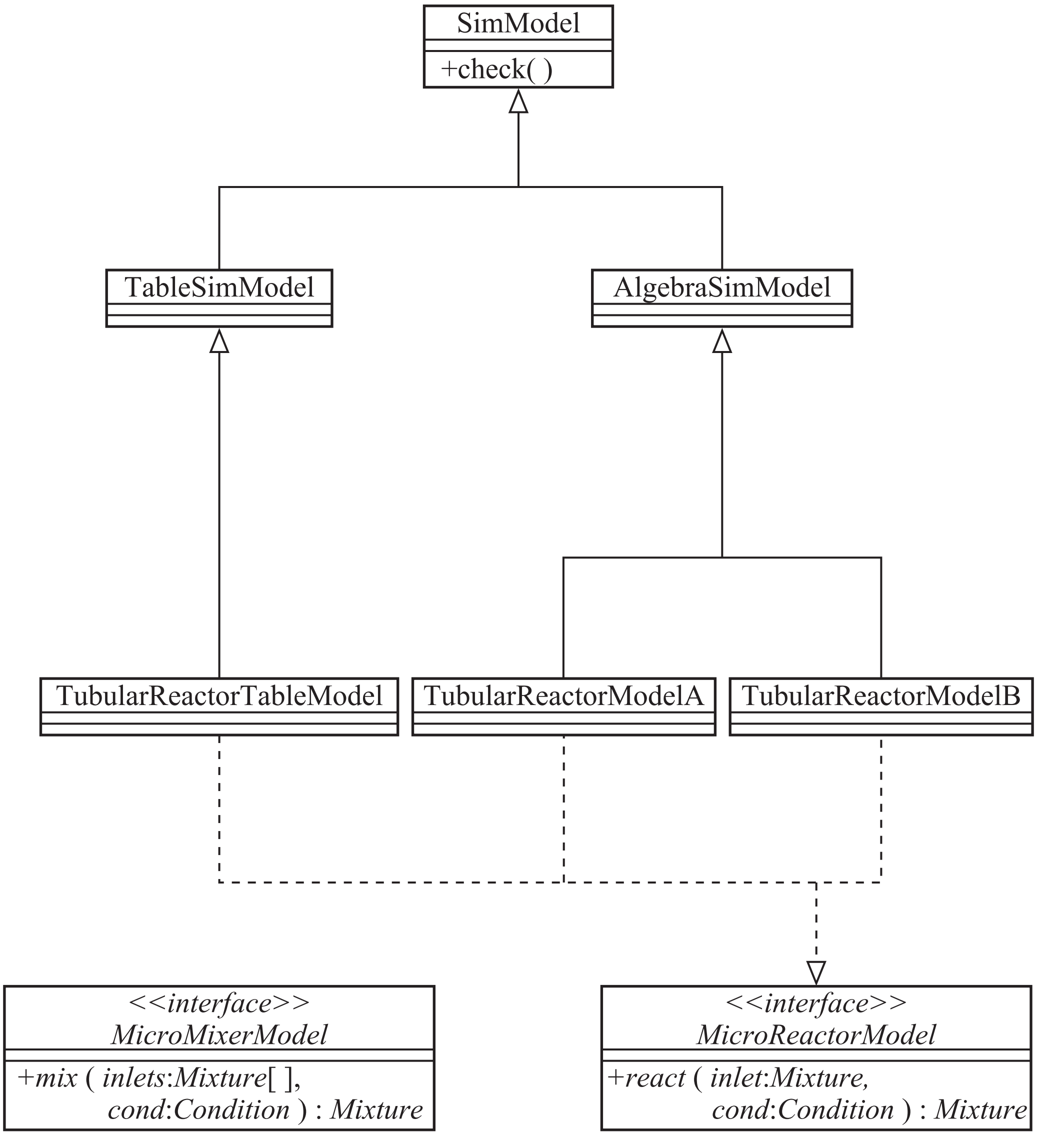


Fig.4.11　シミュレーションモデルのクラス図

4.4 シミュレーションモデルの動的取得シミュレーション

装置エージェントをマイクロ化学デバイスクラス (4.2.2 項) のインスタンスとし、シミュレーションモデル (4.3 節) をデータベースから動的に取得出来ることを検証するために、シミュレーションを行なった。4.4.1 項ではシミュレーションモデル取得のための通信シーケンスについて述べ、4.4.2 項ではデータベース上のシミュレーションモデルを管理するデータベースエージェントについて述べる。4.4.3 項および 4.4.4 項で検証シミュレーションの対象•方法および結果について述べる。

4.4.1 シミュレーションモデル取得のための通信シーケンス

本研究では、シミュレーションシステムがすべて Java 言語で記述されている。従って、シミュレーションモデルの本体である Java クラスファイルをネットワーク上の別のマシンからダウンロードすることにより、シミュレーションモデルの動的な取得が可能となる。クラスファイルの取得の様子を UML シーケンス図で Fig.4.12 に示す。Fig.4.12 (a) は、別マシンにあるクラスを取得し、取得したクラスを用いて計算を行う場合のシーケンスである。シミュレーションモデルは、HTTPd サーバ上に置かれている。別のマシン上に起動された装置エージェントは、ClassLoader オブジェクト (装置エージェント外から、クラスをロードするためのオブジェクトであり、厳密には URLClassLoader) を生成し、この ClassLoader オブジェクトが HTTP プロトコルに従って HTTPd サーバからクラスファイルをダウンロードし、SimModel クラスを生成する。この時、図中の “SimModel” は実際には SimModel クラスの具象クラスのインスタンスであり、4.3 節で述べた “MicroMixerModel” インターフェースや “MicroReactorModel” インターフェースを実装している。従って、これらのインターフェースで定義された ‘計算’ メソッドを

装置エージェントが呼び出すことにより、必要な計算結果を得ることが出来る。比較のため Fig.4.12 (b) に、装置エージェント内にあらかじめクラスファイルを組み込み、組み込まれたシミュレーションモデルを用いる場合のシーケンスを示した。Fig.4.12 (b) での ClassLoader は、厳密には BootStrapClassLoader と呼ばれる Java のシステムレベルのオブジェクトであり、装置エージェントとは無関係に存在している。新しくシミュレーションモデル (のインスタンス) を生成する際は、BootStrapClassLoader を通してインスタンスを得ることが出来る。

ネットワーク上の別マシンからクラスをロード出来ることの検証とそれにかかるオーバーヘッドの測定のために予備実験を行なった。マシン A(MacOS X v10.3.9, 1GHz PowerPC G4, 1GB RAM) およびマシン B(RedHat Linux 7.3, 2GHz Intel Celeron, 512MB RAM) でそれぞれ HTTPd(Apache 1.3.33) を起動し、シミュレーションモデルを格納した。別のマシン C(MacOS X v10.3.9, Dual 1.25GHz PowerPC G4, 2GB RAM) 上で装置エージェントを起動した。マシン A とマシン C の間は 100BASE-TX のネットワークで 1 台のスイッチングハブを介して接続されている。一方、マシン B とマシン C の間は一般のインターネット回線を介して接続されており、複数台のルータおよびゲートウェイを通過する。種々のファイルサイズ (687 Bytes ～ 85,304 Bytes) のシミュレーションモデルを用意し、シミュレーションモデルの生成までにかかる時間を計測した。計測した時間は、Fig.4.12 中の "*loadTime*" に相当する。Fig.4.13 に結果を示す。横軸はクラスファイルのサイズを示し、縦軸はクラスの読み込みにかかる時間を示した。なお、各条件において 5 回測定した平均値である。(a-2) マシン A から取得時と (b) 組み込み型のクラスロード時の時間差は 76.4 ～ 93.6 ms となり、クラスファイルのサイズに依らずほぼ一定の数値を示していることがわかった。マシン A とマシン C の間は 100BASE-

TX の 1 台のハブのみを介して接続されているため通信の輻輳などが起こりにくい理想的な状態である。理論上の通信速度は 100Mbps(=12,800 KB/sec.) であるため、今回のテストに用いた最大 90KB 程度のファイルの理論上の通信時間は 7 ms 程度である。従ってクラスロード時の時間差のうち、7 ms を除いた時間 69.4 〜 86.6 ms が通信時のオーバヘッド (通信セッション確立など) やクラス読み込みにかかる時間である。(a-2) と (b) における時間差がファイルサイズによらずほぼ一定となったのは、Java 言語におけるクラスロード方法が関係あると思われる。Java 言語におけるクラスファイルはプラットフォーム (ハードウェアや OS) から独立した 'Java バイトコード' と呼ばれる形式のファイルである。Java ではクラスのロード時にバイトコードから、プラットフォームに依存した実行形式 (ネイティブコード) に変換する必要がある。この変換作業はクラスファイルの所在に関わらず、必ず行われるため、装置エージェント内にあらかじめクラスファイルが組み込まれていても、ロード時間に相応の時間がかかり、結果的に (a-2) と (b) の時間差がほぼ一定となる結果となった。一方で、(a-1) マシン B からのクラス取得は、複数のハブやルーター、ゲートウェイを介して接続されているため、通信パケット毎に通信経路が変わったり、通信の輻輳が発生することが多く、測定回ごとのロード時間にばらつきが見られた。(a-1) と (b) の時間差は最大で 730 ms 程度であった。

別のマシンから動的にクラスを取得する場合、理想的な通信環境下ではクラスファイルのサイズに依らず、ほぼ一定の通信オーバーヘッドであることがわかった。また、インターネットを介して接続されたマシンからクラスを取得したとしても、数秒でクラスをロード出来ることがわかった。

4.4.2 データベースエージェントの導入

種々のシミュレーションモデルを蓄積したデータベースとシミュレーションシステムとのインターフェースとして、データベースエージェントを導入した。Fig.4.14 にデータベースエージェントを導入したシミュレーションモデルの取得の様子を UML シーケンス図で示した。装置エージェントはデータベースエージェントに対して、シミュレーションモデルの検索条件として、装置タイプや反応系、操作条件の情報を送信する。データベースエージェントは、それに応じてデータベース上のシミュレーションモデル名を返信する。シミュレーションモデル名を受信した装置エージェントは、シミュレーションモデル取得のために URLClassLoader を生成して、HTTPd サーバから該当するクラスファイルを取得することが出来る。

4.4.3 シミュレーション対象および方法

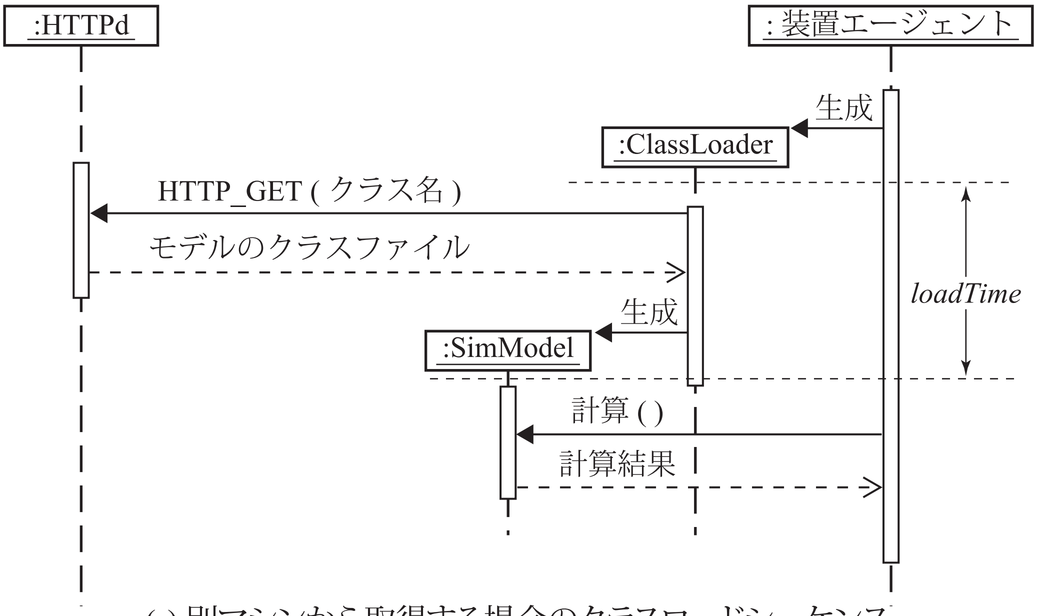
文献 [66] から得られた管型マイクロリアクターにおける実験データに基づいてシミュレーションモデルを用意した。反応器の形状は内径 2 mm、管長 0.7 m の 1 種類のみとした。反応温度ごとに 4 つのシミュレーションモデルに分割し、それぞれの反応温度で 3 段階の流量が設定された表型モデルとした。表型モデルは、データとして入力した条件以外の操作条件では、線形補完により推算を行なうモデルである。データベース上に保存されたデータは、Fig.4.6 に示したフォーマットで保存され、このデータを表型モデルクラスに入力することにより、“管型反応器表型モデル” として使用することが出来る。

4.4.4 シミュレーション結果および考察

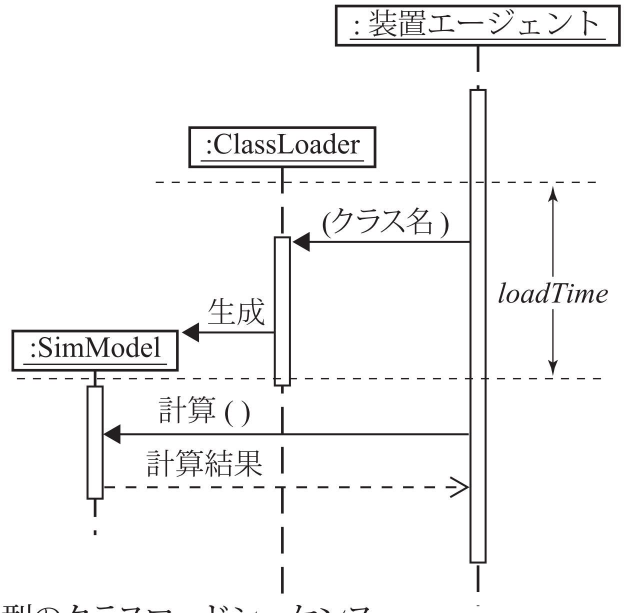
シミュレーションにより推算した管型マイクロリアクターを用いた種々の操作条

件における装置性能 (転化率) を Fig.4.15 に示す。縦軸には装置性能の指標として転化率をとり、横軸には操作条件として流量をとった。反応温度ごとにシミュレーションモデルを分割したため、それぞれのシミュレーションモデルに設定された温度 (261, 273, 295, 323 K) 以外の反応温度を用いることは出来ないが、流量については設定された 3 段階の流量 (500, 1000, 1500 ml/hr) 以外の流量も線形補完により推算が可能である。図中で白抜きの点は、文献から得られたデータである。塗りつぶされた点は、表型モデルを用いて線形補完により推算された値である。この図より、シミュレーションモデルに設定された温度領域に応じて適したシミュレーションモデルが使用されていることがわかる。

また、XML 形式で保存された知識データが、HTML 変換モジュールを通して表に成形されたり、管型反応器表型モデルとしてシミュレーションに用いることが出来、知識データの XML 化によって再利用性を高めることが出来た。



(a) 別マシンから取得する場合のクラスロードシーケンス



(b) 組み込み型のクラスロードシーケンス

Fig.4.12　クラスファイル取得シーケンス

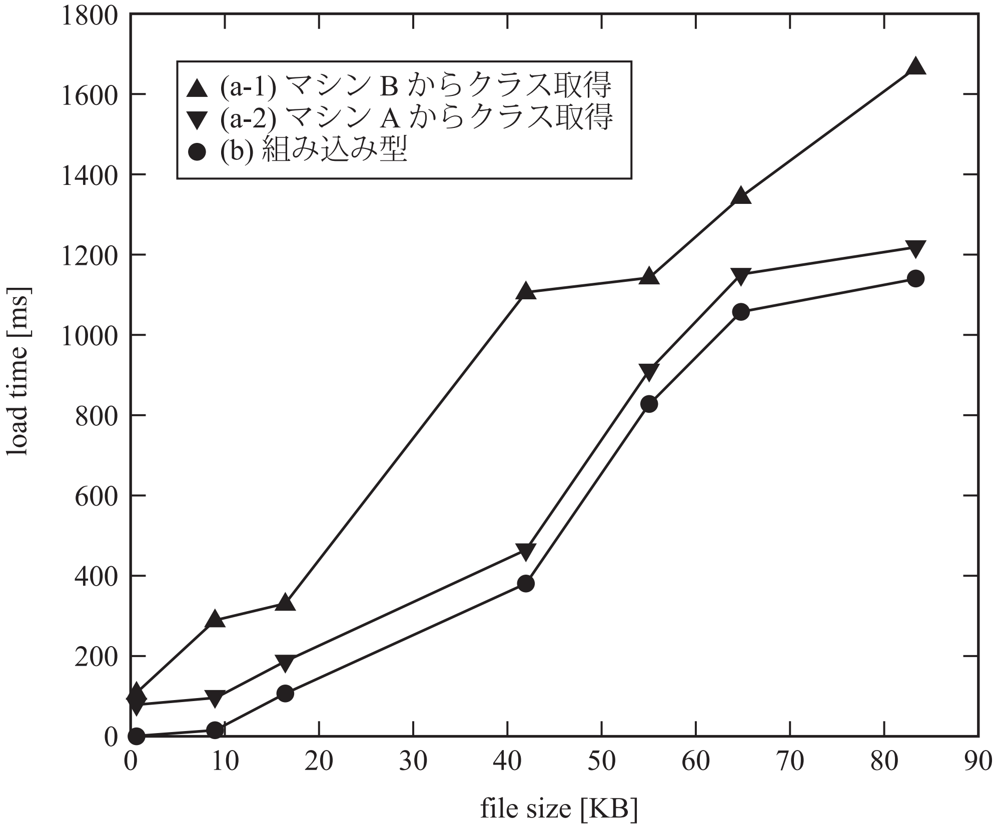


Fig.4.13　異なるファイルサイズおよびネットワーク環境における
クラスファイルロード時間の比較

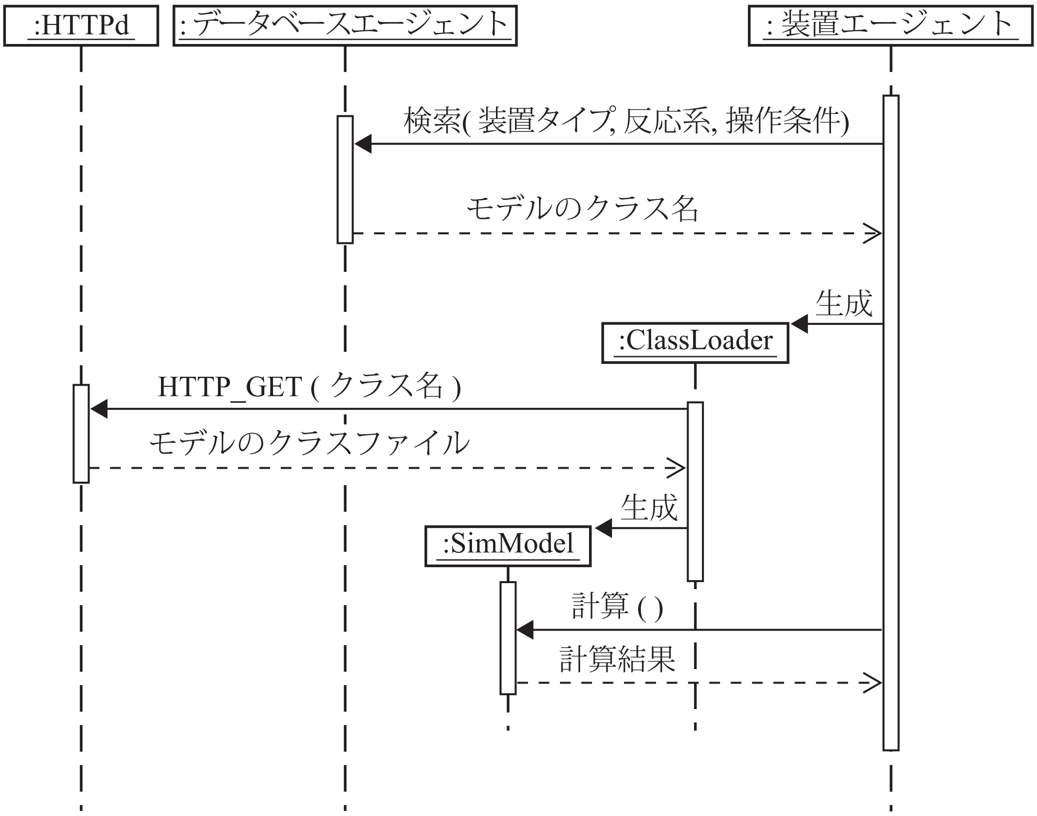


Fig.4.14　データベースエージェントを用いた
シミュレーションモデル取得シーケンス

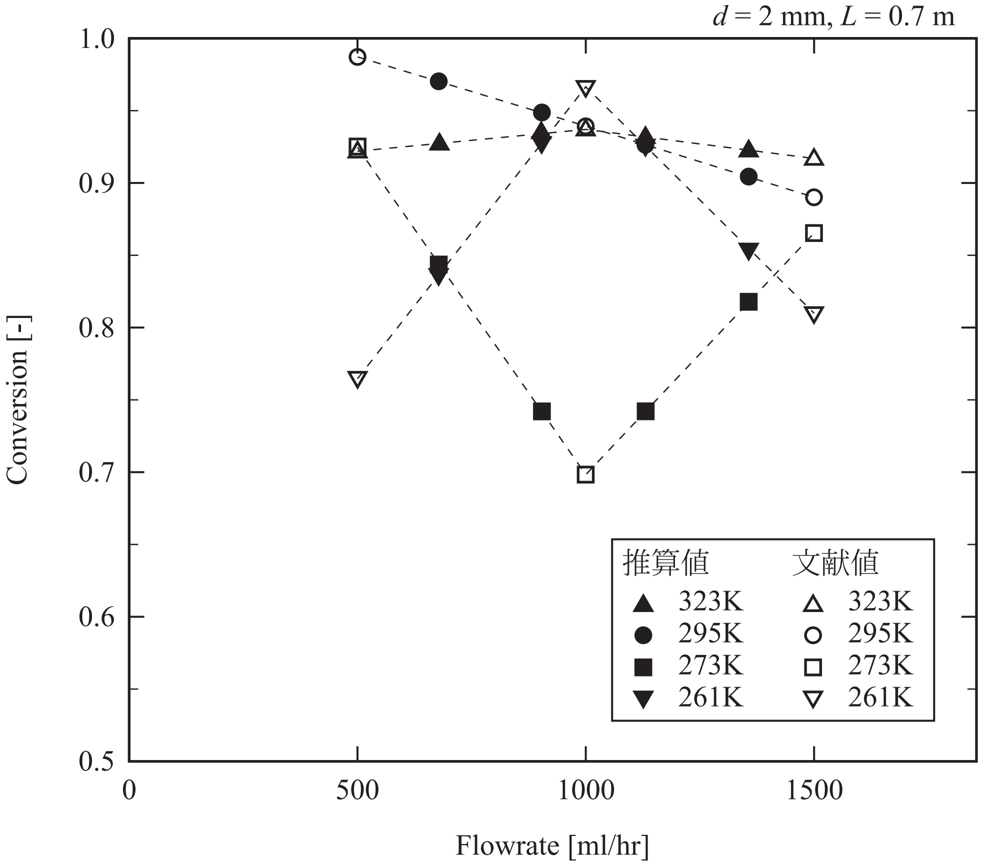


Fig.4.15　表型モデルを用いた種々の操作条件における転化率の推算結果

4.5 本章の結論

本章では、マイクロ化学プロセス技術の研究•開発支援のための知識データベースシステム“*MDCOs*”を構築した。特に、マルチエージェントを用いたシミュレーションシステムとの接続を考慮して、知識データをXML化することとし、そのXMLフォーマットについて検討した。また、知識データベースからシミュレーションモデルを動的に取得し、利用するためのフレームワークとして、装置エージェントに対応するマイクロ化学デバイスのクラス化と、知識データに対応するシミュレーションモデルのクラス化を行なった。知識データベースとマルチエージェントシステムとのインターフェースとして、データベースエージェントを導入した。

知識データをXML化して保存することにより、HTML変換モジュールを通して表に成形しwebブラウザで表示したり、データを表型モデルに取り込みシミュレーションモデルとして利用することが出来た。すなわち、XML化により、知識の再利用性を高めることが出来た。

マイクロ化学デバイスとシミュレーションモデルのクラス化により、関数名や引数•返値を統一的に記述することが出来、動的に取得したシミュレーションモデルを用いて必要な計算を行なうことが出来るようになった。

データベースとの接続によるシミュレーションモデルの動的な取得によって、装置エージェントが保有するもの以外にも種々のシミュレーションモデルを利用することが出来、新たな知識データにも柔軟に対応することが出来るようになった。また、装置エージェントにあらかじめ複数のシミュレーションモデルを組み込むよりも、データベースから動的に取得させることによって、個々の装置エージェントのサイズを小さくすることが出来た。

第 5 章
運転を考慮した設計システム

マイクロ化学プロセスの運転操作を考慮したプロセス設計を行なうために、データベースに蓄積された種々のシミュレーションモデルを評価し、シミュレーションの目的に合わせて適切なシミュレーションモデルを選択する必要がある。そこで、本章では 4 章で構築したシミュレーションモデルをデータベースから動的に取得する機構に、シミュレーションモデルの評価機構を導入し、装置エージェントとデータベースエージェントの協調によって、シミュレーションの目的を考慮した適切なシミュレーションモデルが選択される方法を検討する。

5.1 シミュレーションモデル評価機構の導入

4 章で構築したシミュレーションモデルの動的な取得機構をもとに、データベースに蓄積された種々のシミュレーションモデルから、適切なシミュレーションモデルを選択するために、シミュレーションモデルの評価機構を検討した。

5.1.1 シミュレーションモデル評価機構

一般に化学プロセスの設計は、設計初期にラフなシミュレーションにより種々の装置や操作条件の検討を行ない、ある程度、候補が絞り込まれた設計後期の

段階に詳細なシミュレーションを行なう。従って、設計の段階によりシミュレーション目的が変化していく。具体的には、設計初期では多種の装置や操作条件におけるシミュレーション結果が必要なため、ある程度の計算精度であれば計算が速いシミュレーションモデルが望ましい。一方、設計後期の段階では、装置や操作条件が絞り込まれているため、多少、計算負荷が高くとも計算精度が高いシミュレーションモデルが望まれる。4 章で構築したシミュレーションモデルの動的取得機構では、装置エージェントからのモデル検索条件に合致するシミュレーションモデルをデータベースエージェントが選択•提示していた。従って、モデルの選択にあたり、シミュレーション目的を考慮することは出来ない。そこで、本章ではシミュレーションモデルを評価•選択する機構を導入し、シミュレーション目的を反映させることとした。Fig.5.1 にシミュレーションモデルの評価機構を導入したシミュレーションモデル取得の様子を UML シーケンス図で示した。また、Fig.5.2 にシミュレーションモデル選択における各オブジェクトの振る舞いを UML コラボレーション図で示した。装置エージェントがデータベースエージェントに対して、検索条件を送信すると、データベースエージェントは検索条件に合致するシミュレーションモデルをモデル候補として提示する。この時、各シミュレーションモデルが使用可能か否かは、データベースエージェント側で `check( )` メソッドを呼び出すことにより確認することが出来る。モデル候補として渡された複数のシミュレーションモデルについてシミュレーションの目的を考慮しながら装置エージェントが評価を行なう。選択したシミュレーションモデルを HTTPd サーバからダウンロードして利用することが出来る。

本研究では、シミュレーションの目的を考慮してモデルを選択するために、AHP を用いたシミュレーションモデルの多目的評価を行なうこととした。シミュ

レーションモデルの評価に用いた AHP の構造や評価基準については次項 (5.1.2) で述べる。

5.1.2 シミュレーションモデルの評価基準

Wirth(Pascal 言語の設計者) によれば"アルゴリズム + データ構造 = プログラム" であり、アルゴリズムとはコンピュータを用いて問題解決するために使う手順あるいは手続き [67] である。計算アルゴリズムとデータ構造が定義された計算モジュールをシミュレーションモデルと呼ぶ。すなわち、再利用可能な形で分割された "プログラム" がシミュレーションモデルである。

一般に計算アルゴリズムは "性能"、"一般性"、"利便性"、"報道価値" の大きく 4 つの尺度で評価される [68]。Table 5.1 にアルゴリズムの一般的な評価尺度をまとめた。このうち、"一般性" 評価はアルゴリズムそのものを評価する場合には重要な評価尺度である。しかし、本研究では種々の装置 (すなわち問題例) やパラメータの適用範囲ごとに種々のシミュレーションモデルをデータベースに格納しており、モデルの汎用性などを評価してもそれほど大きな差は生じないと考えられるため、本研究では "一般性" 評価は用いない。また、すでにデータベースに格納されたシミュレーションモデルの評価を行なう場合には、それぞれのシミュレーションモデルの "利便性" 評価にはそれほど差がないと考えられるため、本研究では "利便性" 評価も用いない。さらに、アルゴリズムそのものを学問的もしくは社会的に評価する場合には "報道価値" 評価は重要であるが、データベース上の複数のシミュレーションモデル間では "報道価値" 評価に大きな差がないと思われるため、本研究では "報道価値" 評価も用いない。以上より、本研究では主に "性能" 評価をシミュレーションモデルの評価に用いることとする。

シミュレーションモデルの評価に用いた AHP の構造を Fig.5.3 に示す。評価基準は、計算精度と計算量の 2 項目とし、各項目はそれぞれ次に述べるように評価を行なう。

(1) 計算量評価

計算量とは、計算の速さに関する “時間計算量” と計算時に使用する計算機の記憶容量などに関する “空間計算量” に分類される。Table 5.1 中の “性能—計算の速さ” の評価尺度は、時間計算量を意味し、“性能—メモリ使用量” は空間計算量を意味する。一般的には最悪値解析や確率的解析といった理論的解析法や実験的解析法により、アルゴリズムの計算量評価が行なわれることが多い。しかし、最悪値解析は滅多に起こらない問題例 (= 最悪値) に基づいて評価を行なうため、アルゴリズムの正当な評価ではないという批判がある。例えば、ソートアルゴリズムとしてよく用いられるクイックソート法の最大計算量 (最悪値) を O 記法 (big-O notation) で表記すると $O(n^2)$ であり、それほど高速とは言えない。しかし、クイックソート法の平均計算量は $O(n \log_2 n)$ であり高速なアルゴリズムである。確率的解析は問題例の分布に仮定をおき、得られる解の期待値で評価を行なうものであるが、問題例の分布の仮定が現実からかけ離れているという批判 [68] がある。実験的解析法は、実際のプログラムにランダムな問題例もしくはベンチマーク用の問題を入力してアルゴリズムの評価を行なうものである。しかし、本研究で用いたシミュレーションモデルは、複数のパラメータ推算モデル (e.g. 物質拡散定数計算モデル、反応速度定数計算モデル、反応解析モデル) を含む必要があったり、シミュレーションモデルの引数である運転操作条件により、使用するパラメータ推算モデルが異なったり、

同一のシミュレーションモデルでも計算量が異なるなど、時間計算量を見積もることが難しい。そこで、本研究ではデータベースからシミュレーションモデルをダウンロードする時間や、取得したシミュレーションモデルを記憶容量 (メモリ) に展開する際の空間計算量に直接的に影響を与える ‘シミュレーションモデルの物理的なファイルサイズ’ を計算量の評価項目とした。

(2) 計算精度評価

一般に計算精度も漸近的解析などの理論的解析法や実験的解析法を用いて評価される。漸近的解析 (*O* 記法) を用いて誤差項の評価を行なったり、ベンチマーク問題例を用いた実験的解析法では、既知の最適値と比較することにより解の精度を評価する。しかし、本研究で用いるシミュレーションモデルが対象とするマイクロ化学プロセスのシミュレーションにおいては、既知の最適値や実験値が必ずしも存在するわけではない。そこで、本研究ではシミュレーションモデルに予め設定された適用範囲の中心値付近で、計算精度が最も良くなるものと仮定して、適用範囲の中心値からの変位が小さいほど計算精度が高いとみなし、これを計算精度の評価とした。Fig.5.4 に計算精度評価の概略を示す。横軸は任意の操作変数 x である。あるシミュレーションモデルが $x_{min} \leqq x \leqq x_{max}$ の範囲において適用可能であるとする。本研究では、その中点 x_{mid} において計算精度が最大であるとする。図では縦軸が中点 x_{mid} からの変位を示し、この値が小さいほど計算精度が高いことを示している。

Table 5.1　アルゴリズムの一般的な評価尺度

評価尺度		内　容
性能	計算の速さ	計算時間がかかるか否か。 CPU 時間やメモリアクセス回数、仮想実行時間などで評価される。
	メモリ使用量	計算機の記憶容量をあまり使用しないか否か。アルゴリズムの“軽さ”を表す。
	解の良さ	算出される解が最適値に近いか否か。場合によっては、精度のよい限界値と比較される。
一般性	頑強性	計算の速さや解の良さが問題例の特性に依存せずに安定しているか否か。
	パラメータに対する頑強性	広い範囲のパラメータで安定しているか否か。
	汎用性	広い範囲の問題のクラスもしくは問題例に対して適用可能か否か。
利便性	単純性	解法の理解が容易で記述が簡単か否か。
	実装の容易さ	開発期間が短く、プログラミングが容易か否か。
	拡張性	他の異なる問題に対しても容易に拡張出来るか否か。
	モジュール化の容易さ	アルゴリズムを部品化し、再利用しやすいか否か。
報道価値	新規性	斬新なアイデアが含まれているか否か。
	重要性	重要な問題 (のクラス) を対象としているか否か。

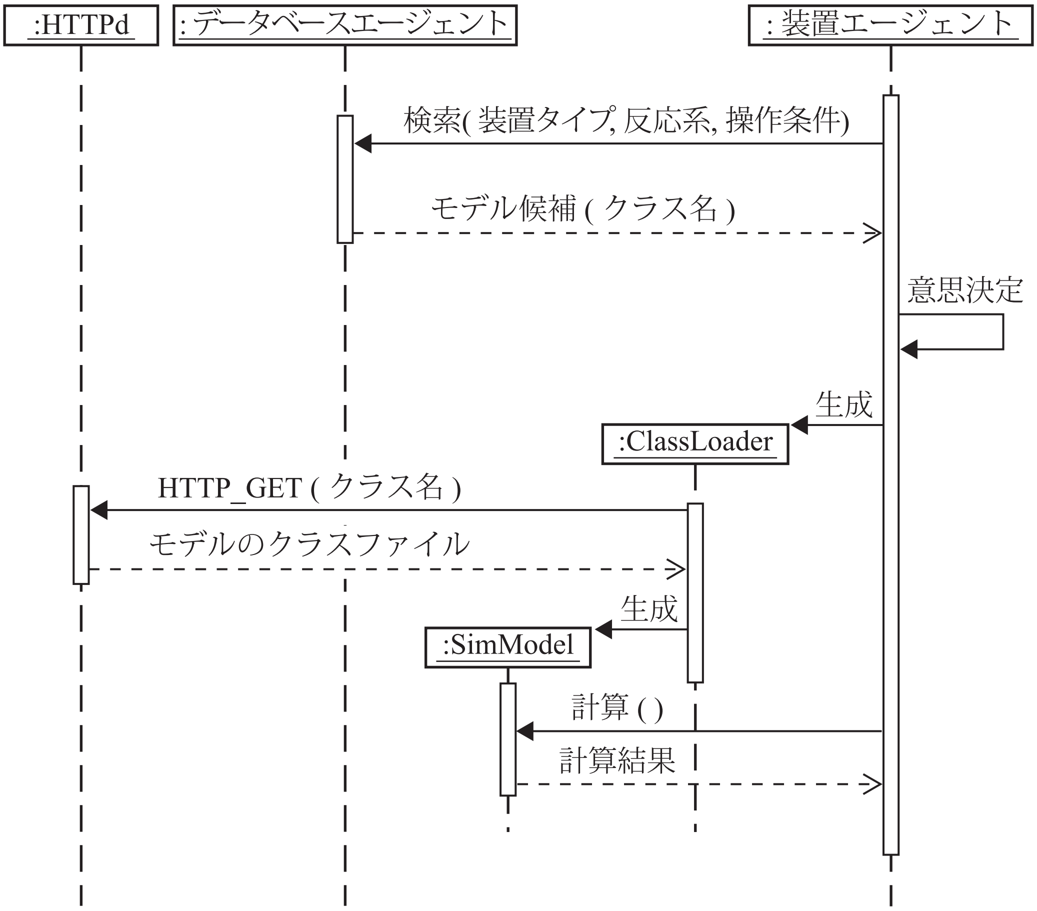


Fig.5.1　シミュレーションモデルの評価機構を導入した
シミュレーションモデル取得シーケンス

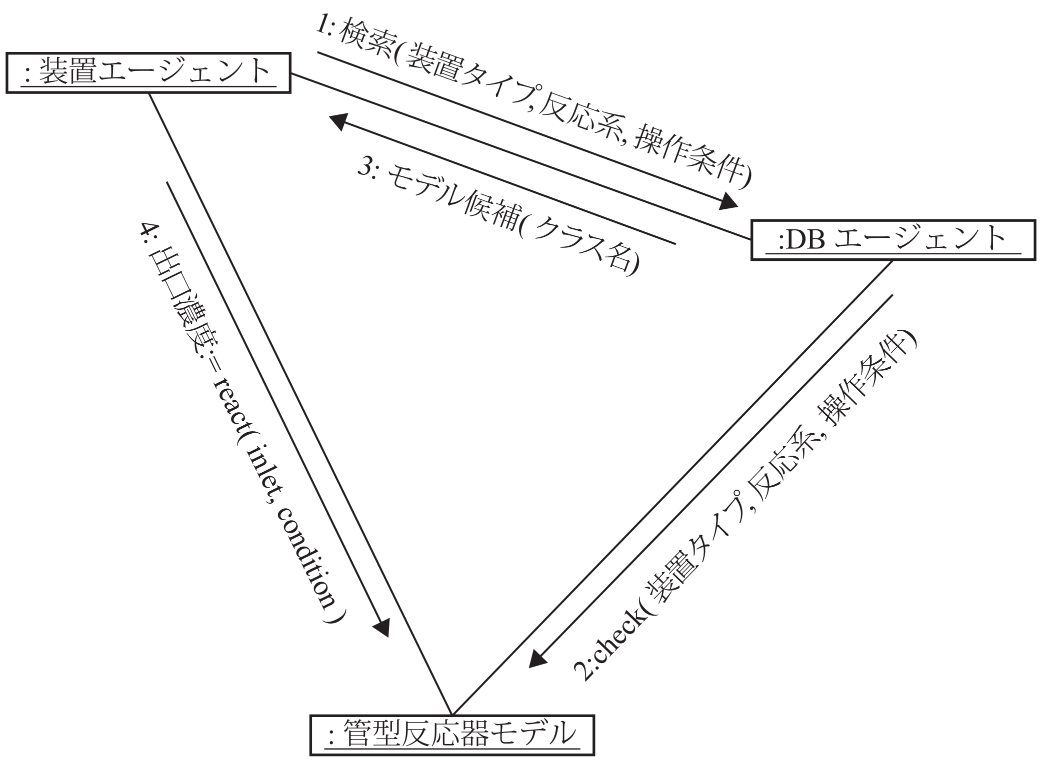


Fig.5.2　シミュレーションモデル選択におけるコラボレーション図

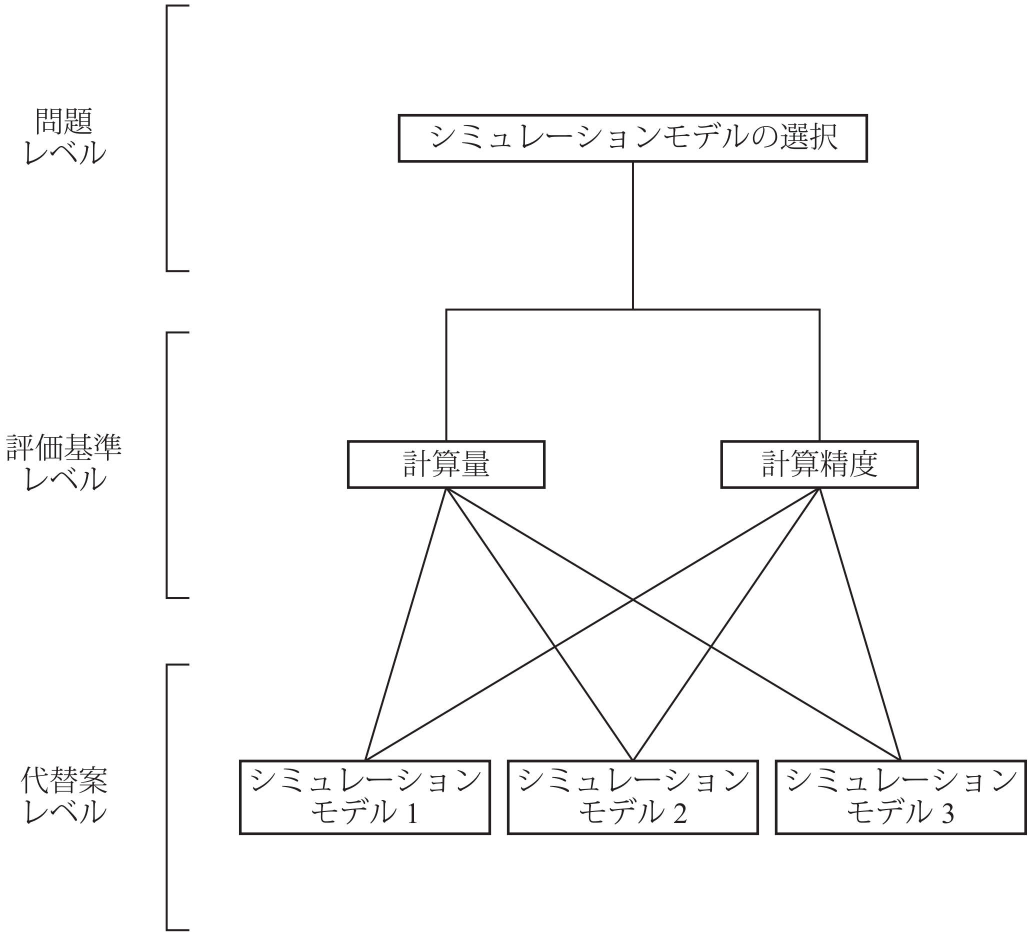


Fig.5.3　シミュレーションモデル評価のための AHP の構造

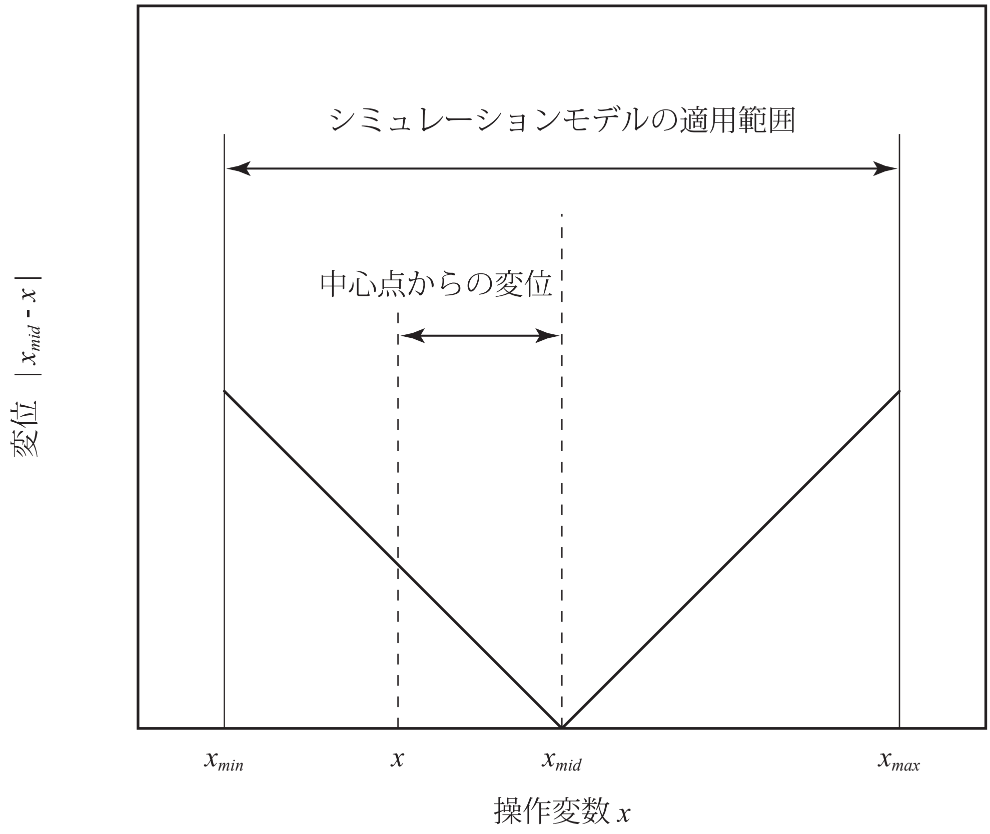


Fig.5.4　計算精度評価の概略

5.2 シミュレーション

5.2.1 シミュレーション対象および方法

“クラリアントプロセス” と呼ばれる逐次反応 [66] を対象として、シミュレーションを行なう。主反応および副反応の反応経路を Fig.5.5 に示す。この反応は反応速度が速く、かつ、反応の進行に伴って反応熱が発生し、爆発的に反応が進行する可能性があるため、マイクロ化学デバイスによる精密な温度制御や反応速度の制御が非常に重要である。本シミュレーションでは、クラリアントプロセスを管型反応器において運転することを検討する。管型反応器は、内径 2 mm、管長 0.7 m の管内部に原料物質を流し、管の外側から一定温度に保つことにより化学反応を制御することが出来る装置である。

管型反応器におけるクラリアントプロセスにおいて、表型モデルと 2 種類の関数型モデルをシミュレーションモデルとして用意した。関数型モデルは流速によりパラメータ推算モデルの一つである物質拡散定数計算モデルが異なるため、操作変数の適用範囲が異なるシミュレーションモデルを 2 つ用意した。これらのモデルの詳細は Appendix C に記載する。本シミュレーションで用いた 3 種類のシミュレーションモデルにおける適用範囲およびファイルサイズを Table 5.2 に示す。ファ

Table 5.2　シミュレーションモデル

	適用範囲 U [m/sec.]	ファイルサイズ [-]
表型モデル	0.04 ～ 0.14	1.0
関数型モデル A	0.11 ～ 0.15	3.0
関数型モデル B	0.03 ～ 0.11	3.0

イルサイズは表型モデルを 1.0 とした時の比で示した。

5.2.2 シミュレーション結果および考察

(1) モデル選択シミュレーション I

種々の運転操作条件において、適切なシミュレーションモデルが選択されることを検証するために、運転操作条件の一つとして流速 U を変えて管型反応器の性能を推算する実験を行った。シミュレーションモデルの評価においては、“計算精度評価” が “計算量評価” よりも「少し重要 (weak importance)」と固定した。流速 U は 0.044 〜 0.140 m/sec. の範囲でシミュレーションを行なった。Fig.5.6 に結果の一例を示す。横軸は流速 U、縦軸は反応転化率を示す。この結果は、単一のシミュレーションモデルを用いて推算したものではない。低流量 (U = 0.044 〜 0.060 m/sec.) 域では関数型モデル B が選択され、中間領域 (U = 0.080 〜 0.100 m/sec.) では表型モデル、高流量 (U = 0.120 〜 0.140 m/sec.) 域では関数型モデル A が選択された結果である。従って、運転操作条件に応じてシミュレーションモデルが評価され、選択されたシミュレーションモデルを用いて装置性能が推算出来たことを示している。

本シミュレーションで用いたシミュレーションモデル評価の計算負荷 ($eval_{simLoad}$) および計算精度 ($eval_{simAcc}$) の評価値の算出には、以下の式を用いた。

$$eval_{simLoad,\,i} = 1 - \frac{K_i}{K_{max}} \tag{5.1}$$

$$eval_{simAcc,\,i} = 1 - \frac{\Delta x_i}{\Delta x_{max}} \qquad (\Delta x_i = | x_{mid,i} - x |) \tag{5.2}$$

シミュレーションモデル i の物理的なファイルサイズを K_i とし、候補となるシミュ

レーションモデルの中で最大のファイルサイズを K_{max} とする。(5.1) 式で規格化することにより、$eval_{simLoad}$ は 0.0 ～ 1.0 となり、ファイルサイズが小さいものほど評価値が大きくなる。従って、本シミュレーションで用いた 3 種類のシミュレーションモデルでは、操作変数 x の値に依らず $eval_{simLoad}$ は一定となる。

また、シミュレーションモデル i の操作変数 x における適用範囲を $x_{min,\,i}$ ～ $x_{max,\,i}$ とし、その中点を $x_{mid,\,i}$ とする。中点からの変位は Δx_i で表され、候補となるシミュレーションモデルの中で最大の変位を持つものを Δx_{max} とする。(5.2) 式で規格化することにより、$eval_{simAcc}$ は 0.0 ～ 1.0 となり、シミュレーションモデルに対する入力値である操作変数 x の値が、各シミュレーションモデルに設定された適用範囲中心点からの変位が小さいものほど、評価値が高くなる。この方法を用いて、本シミュレーションで用いた 3 つのシミュレーションモデルにおける計算精度評価の評価値 $eval_{simAcc}$ を計算した結果を Fig.5.7 に示す。$U < 0.040$ m/sec. や $U > 0.140$ m/sec. の範囲では、適用可能なシミュレーションモデルが 1 つしかないため、$\Delta x_{max} = \Delta x_i$ となり (5.2) 式より評価値 $eval_{simAcc,\,i} = 0.0$ となる。また、5.2.1 項で述べたように、シミュレーションモデルの適用範囲は部分的に重複している。例えば、Fig.5.8 に示すように表型モデルと関数型モデル B の適用範囲内にある点 U_x では、変位の大きさ Δx_i はそれぞれ $0.07 - U_x$ と $0.09 - U_x$ となる。$\Delta x_{max} = 0.09 - U_x$ となるため、表型モデルの計算精度の評価値 $eval_{simAcc,\,table} = 0.0$ となり、関数型モデル B の評価値は

$$eval_{simAcc,\,B} = 1 - \frac{0.07 - x_{max}}{0.09 - x_{max}} \tag{5.3}$$

となるため、評価値の差が非常に大きくなる傾向がある。このように AHP における ‘評価基準レベル’ の特定の評価基準項目における評価値の感度が高い場合、他の評価基準項目の影響が無視されてしまう可能性がある。

(2) モデル選択シミュレーション II

(5.2) 式を用いた計算精度の評価方法では、次のような 2 つの問題点が挙げられる。1 つ目は、中心値からの変位 Δx_i の差が微小であったとしても、評価値 $eval_{simAcc,\,i}$ の差が大きくなってしまう。2 つ目に、Δx_i は操作変数 x の次元およびスケールを持つため、複数の操作変数が適用範囲として設定されている場合には、操作変数ごとの比較が難しい。

そこで、次式のように分布関数を導入して評価値を求めることを検討した。

$$eval_{simAcc,\,i} = \exp\left[-\frac{\Delta x_i^2}{2\sigma^2} \right] \qquad \left(\sigma = \frac{x_{max} - x_{min}}{2} \right) \tag{5.4}$$

上式を用いて 3 つのシミュレーションモデルにおける計算精度評価の評価値を計算した結果を Fig.5.9 に示す。適用範囲の中心点 $x_{mid,\,i}$ で評価値 $eval_{simAcc,\,i} = 1.0$ をとり、適用範囲の端点 $x_{min,\,i}$ および $x_{max,\,i}$ では $eval_{simAcc,\,i} = \exp(-0.5)$ をとる。複数の操作変数がある場合には、各操作変数ごとに (5.4) 式で計算した評価値を平均することで操作変数が持つ次元やスケールを無視することができる。

(5.4) 式を用いて (1) と同様のシミュレーションを行なった結果を Fig.5.10 に示す。Fig.5.6 と比較すると、U = 0.120, 0.125 m/sec. において使用したシミュレーションモデルが異なっていることがわかる。すなわち Fig.5.6 の (5.2) 式を用いたシミュレーションでは ‘関数型モデル A’ が選択されているのに対し、Fig.5.10 の (5.4) 式を用いたシミュレーションでは ‘表型モデル’ が選択されていることがわかる。結果として、Fig.5.6 に比べ Fig.5.10 の方が、転化率が滑らかに変化することがわかった。

(3) 目的変更シミュレーション

シミュレーション目的に応じて、適切なシミュレーションモデルが選択されることを検証するために、シミュレーション目的を変化させてシミュレーションを行なった。管型反応器における操作変数を固定(U = 0.125 m/sec.)し、シミュレーションの目的、すなわちシミュレーションモデルの評価においては、“計算精度評価” が “計算量評価” よりも「かなり重要 (strong importance)」(条件 3-1) および“計算量評価”が“計算精度評価”よりも「かなり重要 (strong importance)」(条件 3-2) とした。この操作条件 (U = 0.125 m/sec.) では、表型モデルと関数型モデル A の適用範囲内であるが、関数型モデル B の適用範囲外である。従って、装置エージェントは表型モデルと関数型モデル A の 2 つをシミュレーションモデル候補として得る。Table 5.3 に結果の一例を示す。計算量評価 $eval_{simLoad}$ および計算精度評価 $eval_{simAcc}$ は、表に示すようにシミュレーションの目的に依存しない。しかし、シミュレーションの目的により評価基準項目の重みの値が条件 (3-1) と条件 (3-2) では逆転するため、総合評価値も表に示すように逆転する結果となった。なお、総合評価値の大小が選好の順序を示し、評価値の差または比の値の大小は選好結果へ影響しない。条件 (3-1) では関数型モデル A

Table 5.3　異なるシミュレーション目的におけるシミュレーションモデル評価の一例

	$eval_{simLoad}$	$eval_{simAcc}$	総合評価値	
			条件 (3-1)	条件 (3-2)
表型モデル	1.00	0.783	0.8190	**0.9637**
関数型モデル A	0.33	0..969	**0.8630**	0.4393
関数型モデル B	(適用範囲外)			

が選択され、条件 (3-2) では表型モデルが選択された。

(4) 装置形状検討シミュレーション

種々の装置形状における装置性能を推算するために、装置形状を変化させたシミュレーションを行なった。管型反応器の形状を表すパラメータの 1 つとして管長 L を変化させてシミュレーションを行なった。操作変数として、U = 0.088, 0.133 m/sec. とし、管長 L を 0.1, 0.3, 0.5, 0.7, 0.9 m の 5 点とした。本シミュレーションでは、文献で用いられた装置と装置形状が異なるため該当する実験データが得られない。さらに装置形状を変化させた実験データが得られていないため、表型モデルを使用することが出来ない。しかし、関数型モデルでは装置形状である管内径 d や管長 L がパラメータとして設定されている。従って、本シミュレーションでは、関数型モデル A と関数型モデル B のみを使用することが出来る。Fig.5.11 に結果の一例を示す。U = 0.083 m/sec. の条件では関数型モデル B を用いて推算され、U = 0.133 m/sec. の条件では関数型モデル A を用いて推算された。管長が長くなるにつれて転化率が上昇していることを示す。このようにシミュレーションモデルを用いることによって、種々の運転操作条件のみではなく、装置形状も変化させたシミュレーションを行なうことが出来るため、運転を考慮した設計に利用することが出来る。

(5) 設計シミュレーション

プロセス設計では、種々の装置を用いて種々の運転操作におけるシミュレーションを行なうことが必要である。そこで、管型反応器の形状として管長の異なるものを 3 種類用意し、それぞれ 3 種類の操作条件 (U = 0.044, 0.088,

0.133 m/sec.) を設定して、設計シミュレーションを行なった。本シミュレーションにおいては、反応器の性能を推算するだけではなく、2 章で述べたシミュレーションフレームワークに従い、入札および意思決定を行なった。装置エージェントの評価における AHP の評価基準項目は Fig.2.13 を用いた。シミュレーションモデルの評価は “計算精度評価” が “計算量評価” よりも「少し重要 (weak importance)」とした。Table 5.4 に本シミュレーションで用いた装置エージェントとそれぞれに設定された装置タイプ (装置形状) および使用可能なシミュレーションモデルを示す。各装置タイプで 3 つの装置エージェントを用意し、それぞれ異なる処理速度を設定した。それぞれの装置の管内径は同一 (d = 0.002 m) なので、処理速度の値から断面平均流速を計算すれば操作変数 U を求めることが出来る。これら 9 つの装置エージェントを用いてシミュレーションを行なった。

製品エージェントは契約ネットプロトコルに従って装置エージェントに対して

Table 5.4　種々の装置形状が設定された装置エージェント

装置タイプ	装置形状		使用可能なシミュレーションモデル	装置エージェント名
	d [m]	L [m]		
管型反応器 A	0.002	0.5	関数型モデル A 関数型モデル B	TubeReactA1 TubeReactA2 TubeReactA3
管型反応器 B	0.002	0.7	表型モデル 関数型モデル A 関数型モデル B	TubeReactB1 TubeReactB2 TubeReactB3
管型反応器 C	0.002	0.9	関数型モデル A 関数型モデル B	TubeReactC1 TubeReactC2 TubeReactC3

募集メッセージを送信する。装置エージェントは、自己の装置タイプとレシピに記載された操作条件をデータベースエージェントに送信し、シミュレーションモデルの検索を依頼する。データベースエージェントはシミュレーションモデル候補を返信するが、返されるシミュレーションモデルの候補は、Table 5.4 に記載されたシミュレーションモデルのうち、それぞれの操作条件が適用範囲内にあるもののみである。例えば、TubeReactorB1 は処理速度 S_{B1} = 500 ml/h のため、断面平均流速を求めると U = 0.044 m/sec. となる。従って、表型モデル及び関数型モデル B のみが適用範囲内にあるため、この 2 つのシミュレーションモデルをデータベースエージェントからモデル候補として提示される。これらのシミュレーションモデル候補を受け取った装置エージェントは、5.1 節およびこれまでに述べたシミュレーションモデルの評価方法に従って評価を行ない、選択したシミュレーションモデルを用いて装置性能の推算を行なう。推算した結果を Table 5.5 に処理精度 $eval_{procAcc}$ として示した。また、各装置エージェントに予め設定された処理速度 S_i および装置エージェントが推算した安全性評価値 $eval_{safety}$ も Table 5.5 に示した。装置エージェントはこれらの評価基準項目の値を組立エージェントに "装置情報" メッセージとして送付する。組立エージェントは、組立時間および組立•分解コストの値を計算する。組立•分解コストは装置タイプに依存するものとして設定した。組立エージェントは、これらの値を追加して、"入札" メッセージとして製品エージェントに送付する。製品エージェントは得られた 9 つの入札を AHP を用いて多目的に評価を行なう。製品エージェントがそれぞれの入札を評価した結果を Table 5.6 に示す。評価値が大きいほど選好の度合いが大きいことを示し、評価値の順に選好の順位も示した。Table 5.5 で示した処理精度評価 $eval_{procAcc}$ は TubeReactorC1 エージェントが最大値を取

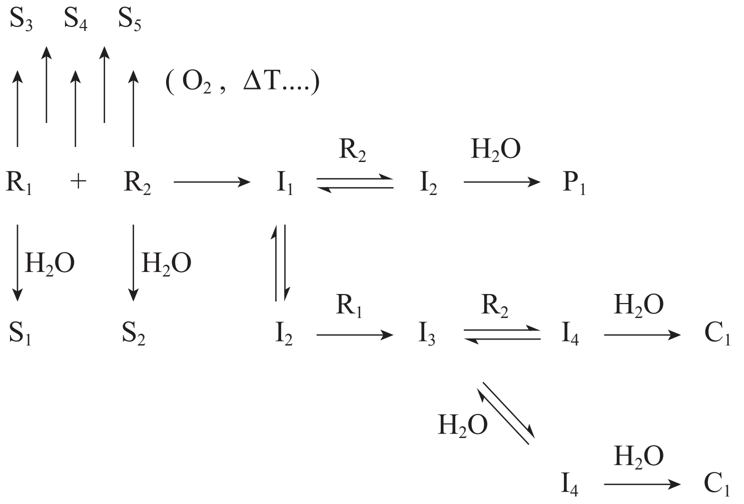
るが、Table 5.6 に示した総合評価では、TubeReactorC1 エージェントは最低値をとっていることが分かる。これは、TubeReactorC1 エージェントは安全性評価や組立•分解コスト評価が他のエージェントに比べて低いためである。一方、処理精度評価 $eval_{procAcc}$ が中程度の TubeReactorB3 エージェントは、総合評価値で最大値をとっていることがわかる。製品エージェントは、Table 5.6 の総合評価値に従って選好の順位を決定し、TubeReactB3 エージェントを選択した。

Table 5.5　各装置エージェントの性能推算値の一例

装置エージェント名	S_i [-]	$eval_{procAcc}$ [-]	$eval_{safety}$ [-]
TubeReactA1	500	0.982	0.959
TubeReactA2	1000	0.866	0.978
TubeReactA3	1500	0.795	0.981
TubeReactB1	500	0.996	0.959
TubeReactB2	1000	0.940	0.978
TubeReactB3	1500	0.891	0.981
TubeReactC1	500	0.999	0.959
TubeReactC2	1000	0.973	0.978
TubeReactC3	1500	0.942	0.981

Table 5.6　製品エージェントによる入札の評価値の一例

入札者 (装置エージェント名)	評価値	選好の順位
TubeReactA1	0.821	7
TubeReactA2	0.841	5
TubeReactA3	0.854	2
TubeReactB1	0.819	8
TubeReactB2	0.841	4
TubeReactB3	0.855	1
TubeReactC1	0.810	9
TubeReactC2	0.833	6
TubeReactC3	0.848	3



R_1, R_2	: reactant
P_1	: product
I_1, I_2, I_3, I_4	: intermediate
S_1, S_2, S_3, S_4, S_5	: side product
C_1	: consecutive product

Fig.5.5　Clariant Process における反応経路

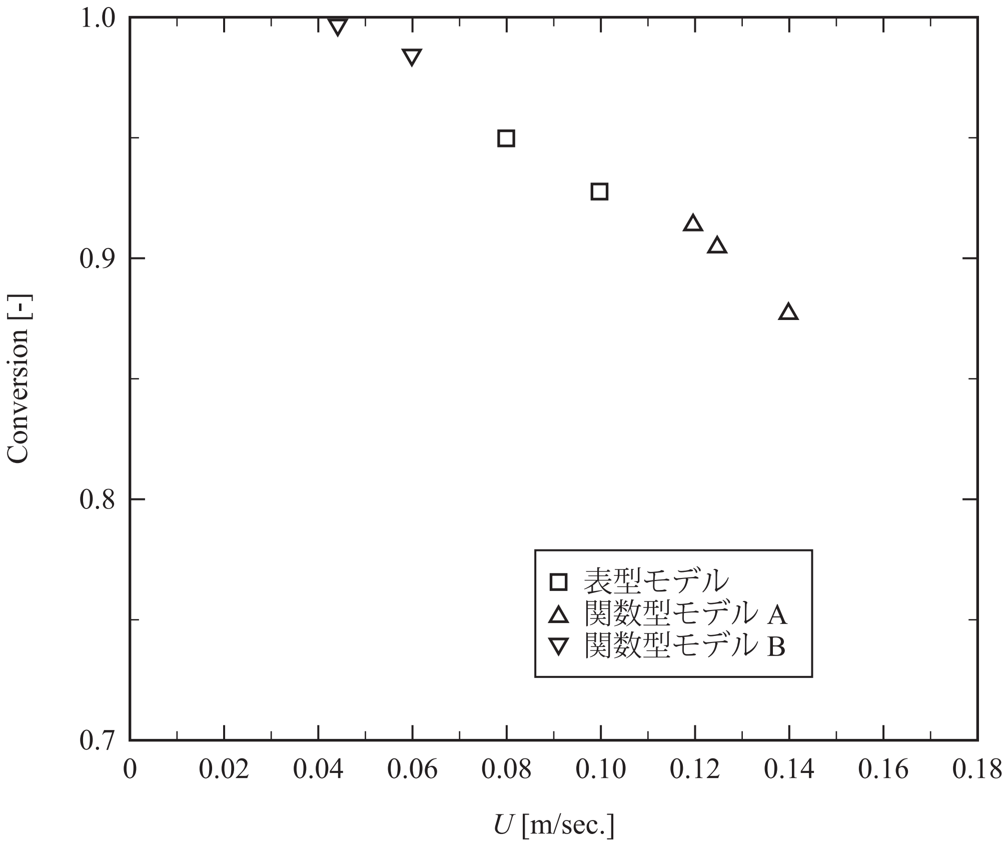


Fig.5.6　モデル選択シミュレーション I における転化率の推算結果

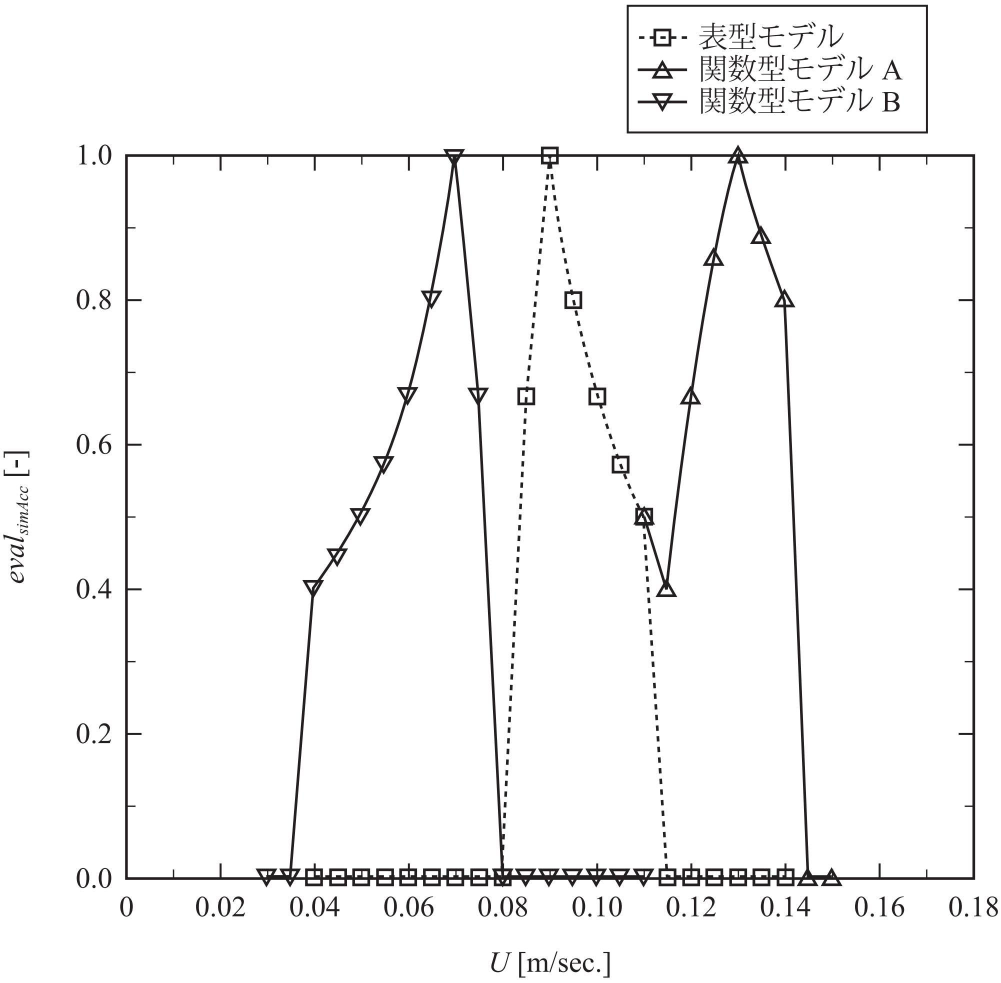


Fig.5.7　種々の運転操作条件における計算精度評価値の変化

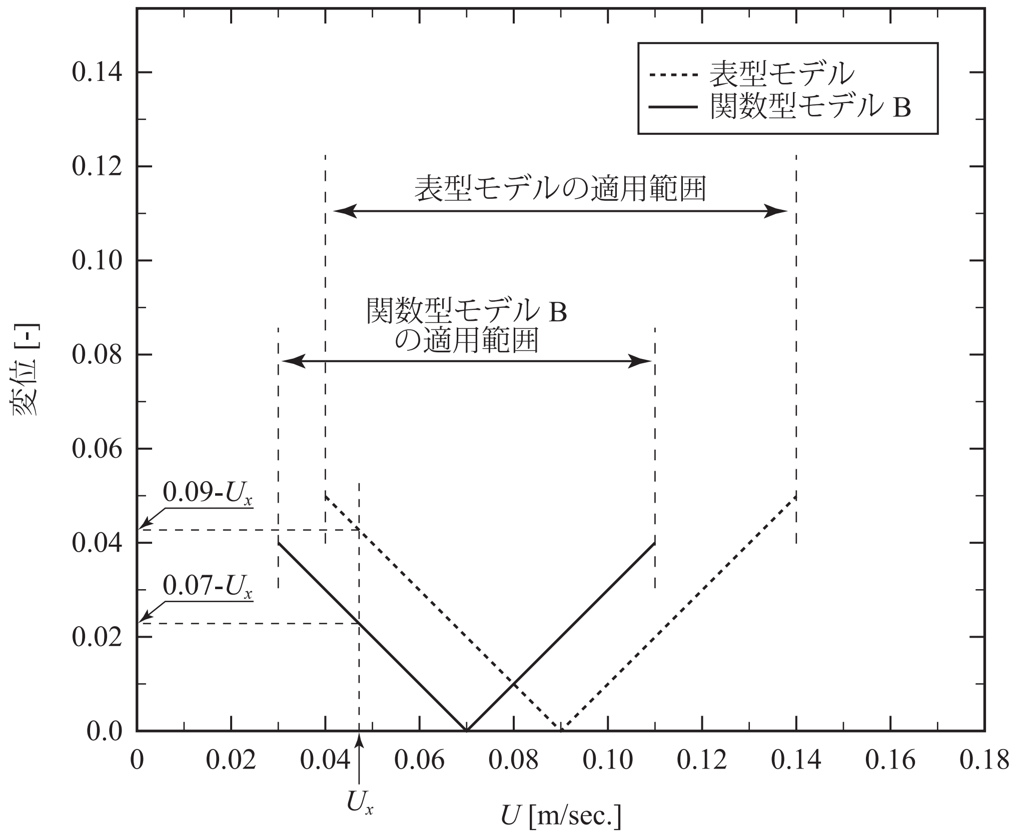


Fig.5.8　表型モデルと関数型モデル B における計算精度評価値の比較

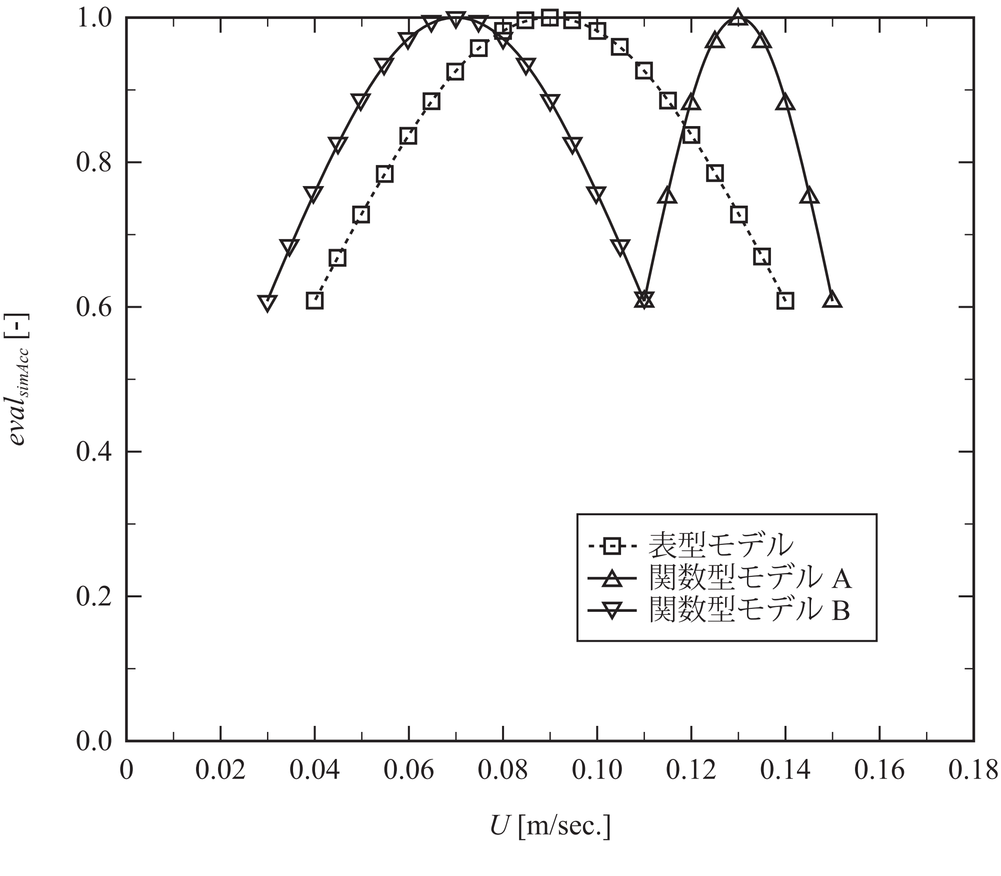


Fig.5.9　種々の運転操作条件における
分布関数を用いた計算精度評価値の変化

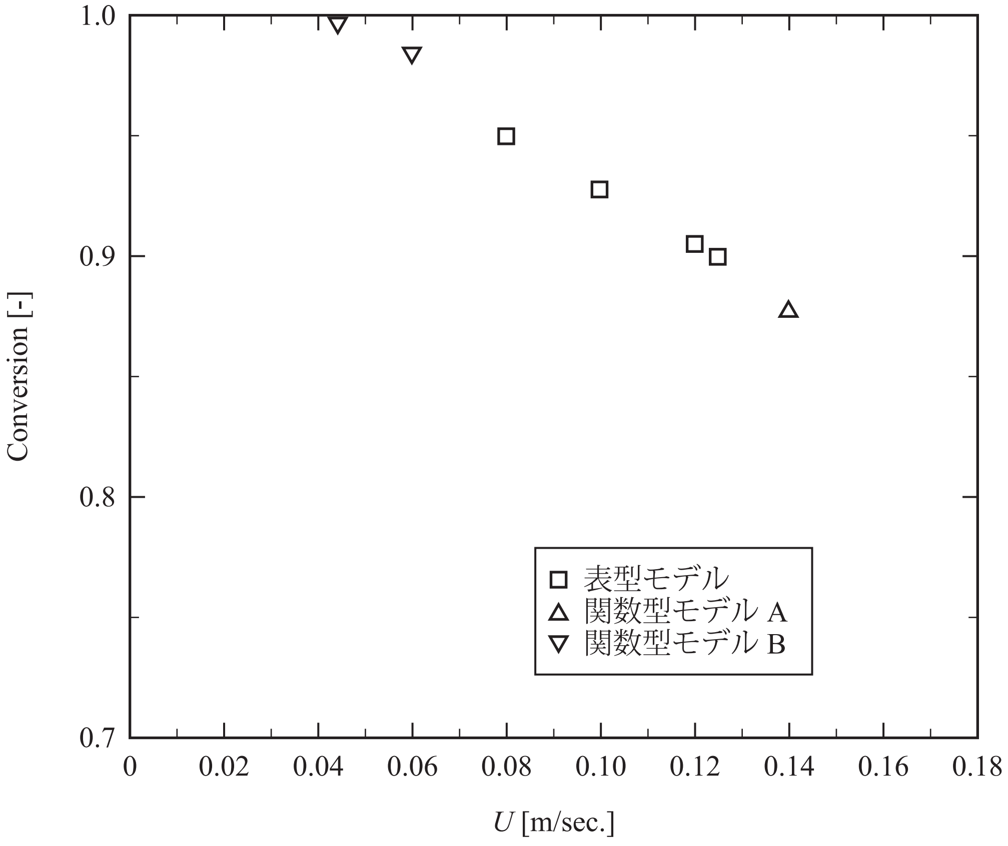


Fig.5.10　モデル選択シミュレーション II における転化率の推算結果

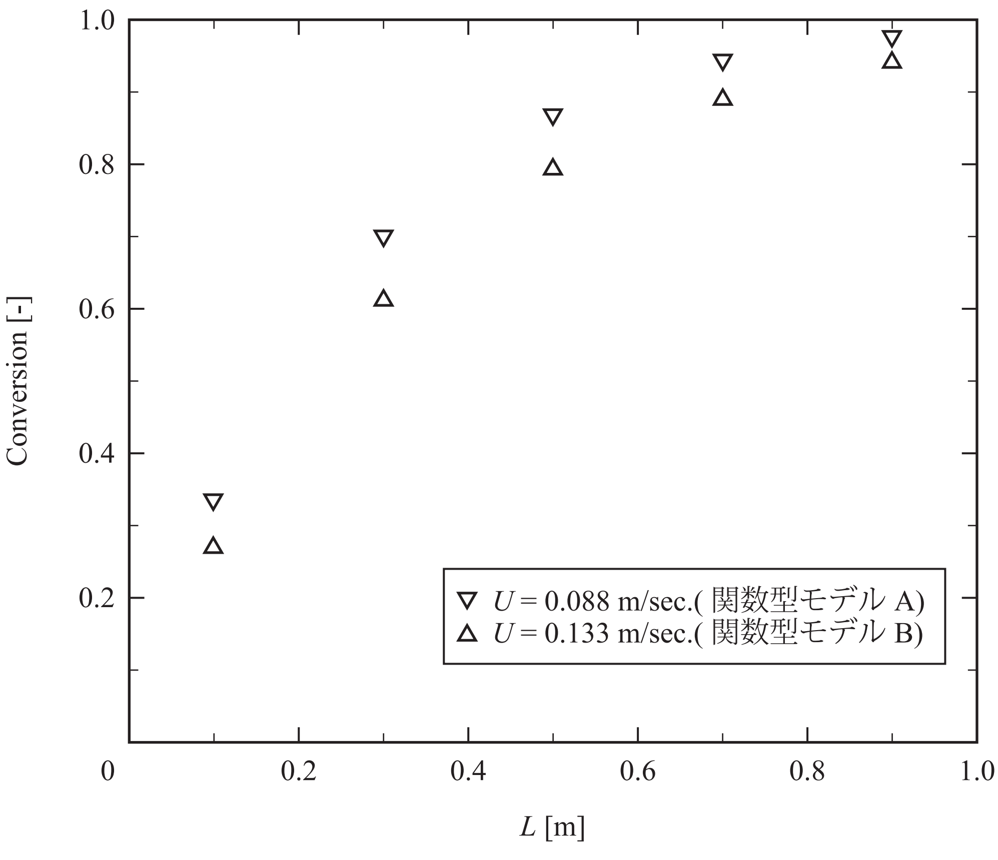


Fig.5.11　装置形状の検討におけるシミュレーション結果の一例

5.3 本章の結論

本章では、マイクロ化学プロセスの運転操作を考慮したプロセス設計を行なうために、4 章で構築したシミュレーションモデルをデータベースから動的に取得する機構に、シミュレーションモデルの評価機構を導入し、シミュレーションの目的に合わせて適切なシミュレーションモデルを選択する機構を導入した。シミュレーションモデルを計算負荷評価と計算精度評価の 2 項目で多目的評価を行ない、設計の進行に伴って変化するシミュレーション目的に応じてシミュレーションモデルを選択することが出来た。

シミュレーションモデルの動的取得機構と評価機構を用いることで、種々の装置や運転操作条件における装置性能や挙動を推算することが出来た。またそれらの結果を AHP を用いて多目的に評価を行なうことによって、装置性能のみではなく経済性や安全性なども考慮したプロセス設計が出来ることを示した。

第 6 章
結　　論

マイクロ化学プロセスの運転•設計システムを対象としてマルチエージェントを用いた効率的で持続可能な情報管理を目指して研究を行なった。具体的には、プロセス運転における評価機構について検討し、AHP への確信度の導入と比較優位指標の導入によりマルチエージェント環境における協調的な動作の発現を試みた。また、知識データベースシステムを構築し、データベースとマルチエージェントシステムを接続して動的にシミュレーションモデルを取得することにより、柔軟なシミュレーションシステムの構築を試みた。また、プロセスの運転を考慮した設計を行なうためにシミュレーションモデルの評価機構を検討し、以下のような結論を得た。

第 1 章「緒論」では、マルチエージェントシステムおよびマイクロ化学プロセスについて概観し、製造業へのエージェントの適用例や意思決定法、知識共有のための知識およびモデルの記述方法について既往の研究を概説して、本研究の目的と意義を述べた。

第 2 章「マルチエージェントを用いた運転操作シミュレーション」では、装置や製品などの要素数の増加に伴い、大規模•複雑化するマイクロ化学プロセスにおいて飛

躍的に増大するプロセス情報を効率的に管理するために、マルチエージェントを用いた運転操作を検討した。シミュレーションを行なうためのマルチエージェントフレームワークを検討し、装置エージェント、製品エージェント、組立エージェントおよびプロセスマネージャーを用いたシミュレーションシステムの実装を行なった。マイクロ化学プロセスにおいて、装置モジュールの追加や故障による入れ替えが頻繁に起こった場合でも、マルチエージェントシステムを用いることで、柔軟に対処することが出来た。また、意思決定における評価項目およびその重みによって、運転結果に違いが見られることがわかった。すなわち自律的な運転のためには、化学プロセスの目的にあわせて適切な評価項目を設定し、それぞれの評価項目に適切な重みを与える必要があることがわかった。

第 3 章「意思決定機構の高度化」では、エージェントが主体的に協調行動をとることによって、システム全体としての処理能力向上を目指し、AHP への確信度の導入による交渉方法の改良と、装置エージェントが比較優位指標を用いて募集された仕事を評価•選択することによる、より高度な協調意思決定機構の構築を行なった。確信度を用いることにより、組立エージェントと製品エージェントが対等に交渉を行なった場合に比べ、確信度の値を適切に設定することによって評価基準項目の重みの変動を小さく出来ることがわかった。従って、多様な製品を受注しても、安定した選好のもとに意思決定を行なうことが出来ることがわかった。また、装置エージェントが募集された仕事を比較優位指標を用いて評価し、入札すべき仕事を選択することによって、システム全体としての処理能力を向上させることが出来た。特に、本システムでは、他の装置エージェントと交渉を行なわなくても、装置エージェントが利己的にすなわち自己の優位な仕事に対して入札を行なうことにより、システム全体としての処理能力

向上という協調効果を得ることが出来た。

第 4 章「データベースからのシミュレーションモデルの動的な取得」では、マイクロ化学プロセス技術の研究•開発支援のために知識データベースシステム "*MDCOs*" を構築した。特に、マルチエージェントを用いたシミュレーションシステムとの接続を考慮して、知識データを XML 化することとし、その XML フォーマットについて検討した。また、知識データベースからシミュレーションモデルを動的に取得し、利用するためのフレームワークとして、装置エージェントに対応するマイクロ化学デバイスのクラス化と、知識データに対応するシミュレーションモデルのクラス化を行なった。知識データベースとマルチエージェントシステムとのインターフェースとして、データベースエージェントを導入した。知識データを XML 化して保存することにより、HTML 変換モジュールを通して表に成形し web ブラウザで表示したり、データを表型モデルに取り込みシミュレーションモデルとして利用することが出来た。すなわち、XML 化により知識の再利用性を高めることが出来た。

マイクロ化学デバイスとシミュレーションモデルのクラス化により、関数名や引数•返値を統一的に記述することが出来、動的に取得したシミュレーションモデルを用いて必要な計算を行なうことが出来るようになった。データベースとの接続によるシミュレーションモデルの動的な取得によって、装置エージェントが保有するもの以外にも種々のシミュレーションモデルを利用することが出来、新たな知識データにも柔軟に対応出来るようになった。

第 5 章「運転を考慮した設計システム」では、マイクロ化学プロセスの運転操作を考慮したプロセス設計を行なうために、4 章で構築したシミュレーションモデルを動

的に取得する機構に、シミュレーションの目的に合わせて適切なシミュレーションモデルを評価•選択する機構を導入した。シミュレーションモデルについて計算負荷評価と計算精度評価の 2 項目で多目的評価を行ない、設計の進行に伴って変化するシミュレーション目的に応じてシミュレーションモデルを選択することが出来た。シミュレーションモデルの動的取得機構と評価機構を用いることで、種々の装置や運転操作条件における装置性能やプロセス挙動を推算することが出来た。また、シミュレーションモデルを用いて推算した結果を評価項目の一つとし、2 章および 3 章で用いた AHP の評価基準を用いて多目的に評価を行なうことにより、装置性能のみではなく経済性や安全性なども考慮したプロセス設計が出来ることを示した。

第 6 章「結論」では、前章までの結果を総括した。

以上、本論文は、マイクロ化学プロセスの運転•設計システムを対象としてマルチエージェントを用いた効率的で持続可能な情報管理を目指して研究を行ない、エージェントの協調的な動作とプロセス運転における評価機構について検討した。また、知識データベースシステムを構築し、マルチエージェントシステムを接続して、エージェントがシミュレーションモデルを動的に取得することにより、プロセスの運転を考慮したプロセス設計を行なうシステムを構築した。

Appendix A　UML

UML とは、Unified Modeling Language の略称であり、オブジェクト指向プログラミングにおけるモデル表記およびモデリング手法として、注目されている手法である。オブジェクト指向以前の開発技法として、POA(Process Oriented Approach= プロセス指向) や DOA(Data Oriented Approach= データ指向) などが用いられて来た。オブジェクト指向では、システムにおける要素をプロセスとデータが一体となったオブジェクトとしてとらえるものである。

このようなオブジェクト指向におけるモデリング手法として、Booch 法 (G. Booch) や OMT 法 (J. Rumbaugh)、OOSE 法 (I. Jacobson) などが代表的な手法として挙げられる。これらのモデリング手法を基に表記法を統一したものが UML であり、1997 年に UML1.0 がリリースされ、それ以後、数々の改良が加えられている。

UML では 10 種類程度 (UML1.5) の表記法が定義されており、静的なオブジェクトの関係を表すクラス図やオブジェクト図、オブジェクト間の相互作用を表すシーケンス図やコラボレーション図の他、ステートチャート図やアクティビティ図などが定義されている。本論文では、クラス図、シーケンス図、コラボレーション図の 3 種類を用いており、以下に 3 種類の表記法を簡単に説明する。

1. UML クラス図

クラス図は静的なクラス構造を表現することが出来る。クラスの継承や集約関係を表すことが出来る。Fig.A-1 に UML クラス図の概略を示す。オブジェクトを四角形で表し四角形の内部を 3 つに区切る。上段をオブジェクト名 (またはクラス名、インターフェース名) とし、中段に属性、下段にメソッドを記述する。属性名およびメソッド名の前の “+” 記号は可視性と呼ばれ、その属性 / メソッドが他のオブジェクトから呼び出しや書き込みが可能であるか否かを示す。“+” は全てのオブジェクトからアクセス可能 (public) であることを示している。本研究では “+” のもののみ記述している。(他に “~” =package, “#” =protected, “-” =private がある)

ClassA は抽象度の高いスーパークラスであり、ClassB は具象クラスである。ClassA は ClassB を ‘汎化’ したものである。逆の見方をすれば、ClassB は ClassA の属性やメソッドを ‘継承’ している。

InterfaceC は ‘インターフェース’ と呼ばれクラスとは区別されている。クラスが外部に対して公開する (外部から呼び出すことが出来る) 仕様を表し、インターフェースにおいてメソッドの呼び出し方 (引数および返値) が定義される。インター

フェースを ‘実装’ したクラス側で、メソッドの具体的な操作を定義する。点線で表す矢印は ‘実現’ 関係を表し、ClassB が InterfaceC を実装していることを示している。なお、クラスは単一のスーパクラスのみを持つことが出来る (多重継承の禁止) が、いくつでもインターフェースを実装することが出来る。

菱形の矢印は ‘集約’ 関係を示す。複数個の ClassPart が、ClassB の一部分であることを示している。なお、多重度 “0..*” は 0 以上の任意の整数を示す。

2. UML シーケンス図

シーケンス図は相互作用図の一つであり、オブジェクト間の動的な関係を時系列で表現することが出来る。Fig.A-2 にシーケンス図の概略を示す。オブジェクトを四角形で表し、オブジェクト名 (インスタンス名) とクラス名を “:” で接続して表記する。具体的なオブジェクト名がない場合には、“:ObjectClass2” のようにオブジェクト名を省略して、クラス名のみを表記することも可能である。点線で表される生存線は各オブジェクトが存在していることを示す。

メッセージは、同期メッセージと呼ばれる返値が来るまで次の処理を待つメッセージと、非同期メッセージに分けられる。また、再帰的に自己を呼び出すことも可能である。

3. UML コラボレーション図

コラボレーション図も相互作用図の一つであり、シーケンス図と同様にオブジェクト間の動的な関係を表現することが出来るが、オブジェクト間の関係に重点を置いた表現方法である。Fig.A-3 にコラボレーション図の概略を示す。オブジェクトの表記法はシーケンス図と同様である。オブジェクトとオブジェクトをリンクと呼ばれる実線で結ぶことにより、2 つのオブジェクト間に相互作用があることを示す。

メッセージの種類は同期メッセージ、同期の返値メッセージ、非同期メッセージであり、表記法もシーケンス図と同様である。ただし、メッセージの時間的な順序を示すためにメッセージ名に順序を付す。Fig.A-3 中の 3 番目のメッセージのように返値とメソッドを “:=” でつなぎ、同期メッセージおよび同期の返値メッセージを 1 本のメッセージとして表記することも可能である。

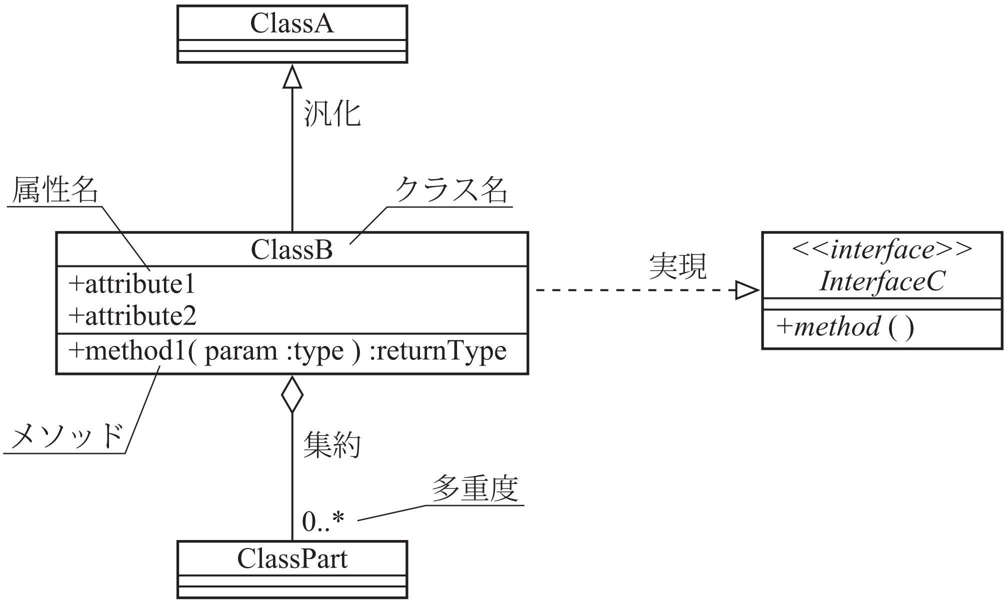


Fig.A-1　UML クラス図

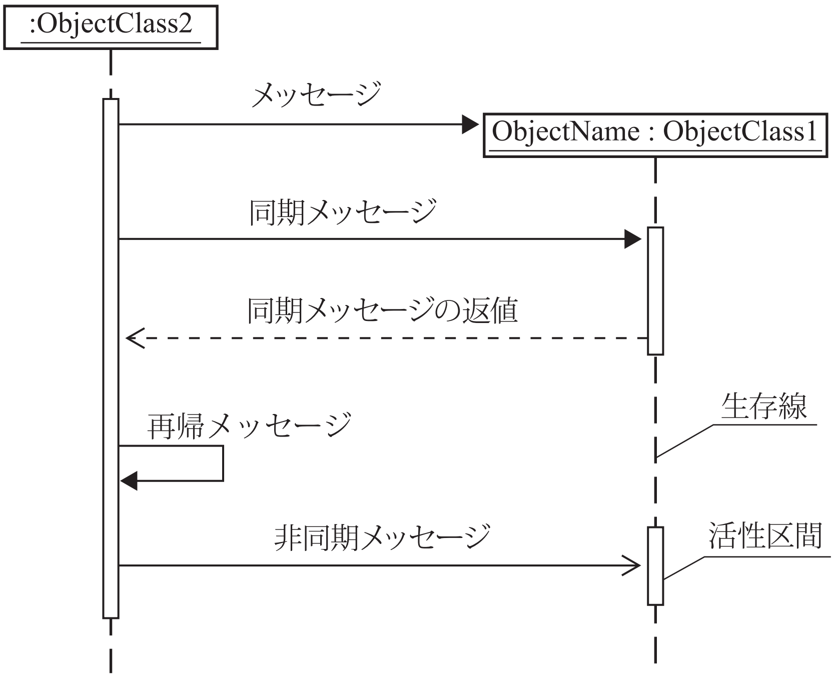


Fig.A-2　UML シーケンス図

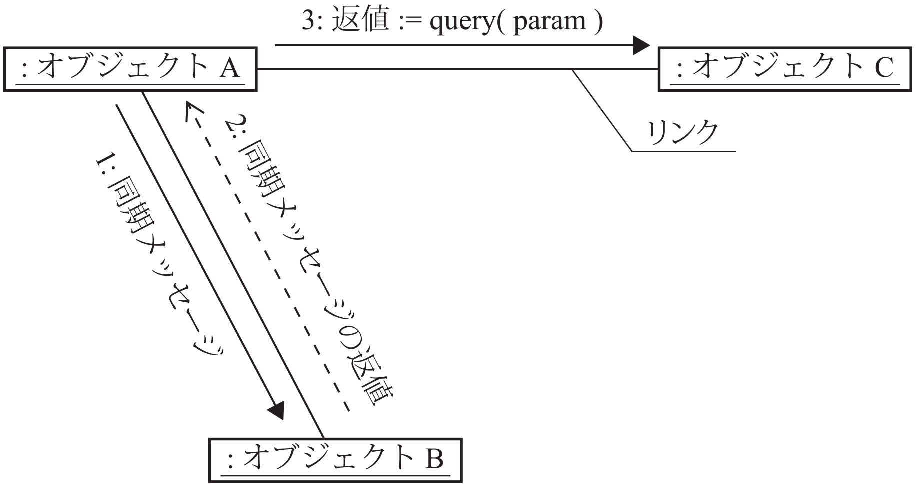


Fig.A-3 UML コラボレーション図

Appendix B　比較優位説

古典的な経済学の学説で "自由貿易の原理" とされる。

「2 つの財 (= 製品) の生産費 (または労働投入量) を比較した値を比較生産費と呼ぶ。比較生産費が自国と外国で異なる時には、"生産費において相対的に優位にある財" に特化して生産•輸出し、これと交換に "相対的に劣位にある財" を輸入することにより、2 国が自給自足するよりも大きな利益を受ける。世界全体としても貿易により利益を受ける。」というものである。この時、以下の前提条件が必要である。

(1) 2 国 2 財モデルである。

(2) 国内的にも国際的にも完全競争が支配する。

(3) 生産要素は国内では自由に移動できるが、国外には移動できない。

(4) 生産要素は完全に利用され、一般的である。

(5) 生産技術は一定とし、生産費も一定 (固定費用) とする。生産量の増減にかかわらず、単位あたりの費用は一定である。

(6) 完全な自由貿易が行なわれ、財の移動に関税や輸送費はかからない。

(7) 債券などの国際取引は全くない。

簡単な例を用いて説明を行なうこととする。イギリスとポルトガルの 2 国のみが存在し、製品として毛織物とワインの 2 財のみがあるとする。労働力としてイギリスは 220、ポルトガルは 170 を有し、毛織物、ワインそれぞれを 1 単位生産するのに必要な労働投入量を Table B-1 のように仮定する。

Table B-1　2 国 2 財モデルにおける労働投入量

	労働投入量 (人日 /1 単位)	
	毛織物	ワイン
イギリス	100	120
ポルトガル	90	80

どちらの製品においてもポルトガルの方が労働投入量が小さい (= 生産効率がよい) ため、ポルトガルが "絶対優位" である。絶対優位である国には貿易によるメリットがないのであろうか?

まず、イギリス、ポルトガルが自給自足を行なったとする。両国が Table B-2 のように労働力を投入したとすると、イギリス、ポルトガルで毛織物およびワインをそれ

それ1単位づつ (合計で毛織物2単位、ワイン2単位) 生産することが出来る。

Table B-2　2国2財モデルにおける自給自足による生産量

	毛織物	ワイン	国別の生産量
イギリス	100 人日	120 人日	2 単位
ポルトガル	90 人日	80 人日	2 単位
財別の生産量	2 単位	2 単位	4 単位

ここで、比較生産費を導入する。比較生産費は、ある国における2製品の生産費 (労働投入量) の比をとることで算出できる。

$$\text{比較生産費} = \frac{\text{製品 1 の労働投入量}}{\text{製品 2 の労働投入量}}$$

この時、比較生産費が1より小なら製品1の方が生産性が高く、
1より大なら製品2の方が生産性が高いことを意味する。

すなわち、“製品1を1単位” 生産するのに犠牲にした “製品2の量 (単位)” とみなすことが出来る。そこでワイン基準で比較生産費を算出すると、

Table B-3　2国2財モデルにおける比較生産費

	比較生産費 (毛織物 / ワイン)
イギリス	100 / 120 = 0.833
ポルトガル	90 / 80 = 1.125

この結果、イギリスは毛織物、ポルトガルはワインに特化した方が良いという指針が得られた。そこで、両国がそれぞれの製品に特化 (分業) して労働力を投入すると、

Table B-4　2国2財モデルにおける分業による生産量

	毛織物	ワイン	国別の生産量
イギリス	220 人日	0	2.2 単位
ポルトガル	0	170 人日	2.125 単位
財別の生産量	2.2 単位	2.125 単位	4.325 単位

となり、自国で消費する量以上の財は貿易によって交換することにより、両国が自給自足を行なうよりも、それぞれの国が利益を受け、また、全体としても生産能力が高まっていることになる。従って、比較生産費に基づいて、自国が相対的に優位にある (比較優位な) 財に特化して生産することにより、生産能力が向上したと言える。

Appendix C　管型反応器モデル

本研究で対象とした管型反応器は、管内 (管内径 $d = 2$ mm、管長 $L = 0.7$ m) に反応物質を流し、管の外部から一定の温度に保つことにより反応を進行させせる装置である。

本研究では、以下のシミュレーションモデルを用意した。

1. 表型モデル

表型モデルは実験データをもとに、線形補完により装置性能を推算するモデルである。実験データとして (反応器温度、流量、転化率) の組が複数組入力される。今回は、反応器温度および流量を操作変数とし、装置の性能として転化率を推算することとした。

2. 関数型モデル

関数型モデルは、円管内で R → I → P の 2 段階の逐次反応が起こると仮定し、数式モデルを用いて装置性能を推算するものである。Fig.C-1 に管型反応器モデルの概略を示す。

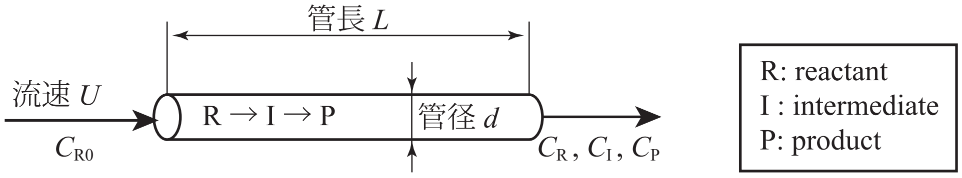


Fig.C-1　管型反応器モデルの概略

ここで C_{R0} は反応器入口における反応物の濃度であり、C_R, C_I, C_P は反応物、中間生成物、生成物の反応器出口におけるそれぞれの濃度である。この時、反応器出口における転化率は次式で表される。

$$conversion = 1 - \frac{C_R}{C_{R0}}$$

$$= 1 - \frac{4a \exp\left[\frac{1}{2}\frac{UL}{\hat{D}}\right]}{(1+a)^2 \exp\left[\frac{a}{2}\frac{UL}{\hat{D}}\right] - (1-a)^2 \exp\left[-\frac{a}{2}\frac{UL}{\hat{D}}\right]} \quad \text{(C.1)}$$

$$a = \sqrt{1 + 4(k_1 \tau)\frac{\hat{D}}{UL}} \quad , \quad \tau = \frac{L}{U} \quad \text{(平均滞留時間)}$$

ここで $\hat{D}$ は軸方向拡散係数 , k_1 は反応速度定数である。

軸方向拡散係数 $\hat{D}$ は操作条件により異なる計算方法を用いる。関数型モデル B の範囲 (低流量域) では次式で表すアリス型の軸拡散モデルにより計算を行なう。

$$\hat{D} = \frac{U^2 d^2}{192 D} \tag{C.2}$$

ここで拡散定数 D は物性値である。

また、関数型モデル A の範囲 (高流量域) では次式で表すテイラー型の軸拡散モデルにより計算を行なう。

$$\hat{D} = D + \frac{U^2 d^2}{192 D} \tag{C.3}$$

参 考 文 献

[1] Russell, S., P. Norvig (古川 康一訳): エージェントアプローチ 人工知能 , 共立出版株式会社 , 1997

[2] 山影 進 , 服部 正太 : コンピュータの中の人口社会 —マルチエージェントシミュレーションモデルと複雑系 , 共立出版株式会社 , 2002

[3] 福田 直樹 , 大囿 忠親 , 新谷 虎松 : モバイルエージェントによる Web ページ構築フレームワーク MiPage の提案 , 人工知能学会論文誌 , **Vol.17**, **No.3**, p.p.348–353, 2002

[4] 伊藤 孝行 , 服部 宏充 , 新谷 虎松 : エージェント間の協調的入札機構に基づく複数オークション入札支援システム BiddingBot, 人工知能学会論文誌 , **Vol.17**, **No.3**, p.p.247–258, 2002

[5] Váncza, J., A. Málkus: An agent model for incentive-based production scheduling, *Computers in Industry*, **Vol.43**, **No.2**, p.p.173–187, 2000

[6] Senin, N., R. Groppetti, D. R. Wallace: Concurrent assembly planning with genetic algorithms, *Robotics and Computer Integrated Manufacturing*, **Vol.16**, p.p.65–72, 2000

[7] Saaty, T. L.: The Analytic Hierarchy Process —Planning, Priority Setting, Resource Allocation, McGraw-Hill, Inc., 1980

[8] 服部 宏充 , 伊藤 孝行 , 新谷 虎松 : エージェント間の交渉における根回しの実現とその応用 , 第 57 回情報処理学会全国大会論文集 (2), 情報処理学会 , p.p.402–403, 1998

[9] Kuroda, C. and M. Ishida: A Proposal for Decentralized Cooperative Decision-Making in Chemical Batch Operation, *Engineering Application of Artificial Intelligence*, **Vol.6**, p.p.399–407, 1993

[10] 石田 愈 , 黒田 千秋 : 人工知能言語によるパイプレス化学生産システムの試作 , 旭硝子財団研究報告 , **Vol.59**, p.p.199–205, 1991

[11] 冨田 守 , 黒田 千秋 , 石田 愈 : レシピを用いたパイプレスバッチプラント用自律分散型運転管理システムの提案 , 化学工学論文集 , **Vol.18**, p.p.629–636, 1992

[12] 黒田 千秋 , 小川 浩平 : マルチエージェント指向の階層化意思決定法 (AHP) とバッチプロセスの柔軟な運転管理 , 第 8 回自律分散システム・シンポジウム , Tokyo, Japan, p.p.67–72, 1996

[13] 片山 謙吾 , 輿石 尚宏 , 成久 洋之 : 強化学習エージェントへの階層化意志決定法の導入—追跡問題を例に—, 人工知能学会論文誌 , **Vol.19**, **No.4**, p.p.279–291, 2004

[14] 木下 栄蔵 : 入門 AHP—決断と合意形成のテクニック, 株式会社 日科技連出版社 , 2000

[15] 和田 英彦 : 次世代生産システムへのエージェントによるアプローチ, 人工知能学会誌 , **Vol.15**, **No.4**, pp. 583–591, 2000

[16] Wada, H., S. Okada: An autonomous agent approach for manufacturing execution control systems, *Integrated Computer-Aided Engineering*, **Vol.9**, **No.3**, pp.251–262, 2002

[17] 和田 英彦 , 櫻庭 祐一 , 根岸 雅子, 山川 利治 : エージェントを用いた加工装置制御システム , 電子情報通信学会論文誌 D-I, **Vol.J84-D-I**, **No.8**, p.p.1283–1293, 2001

[18] Lim, M. K., Z. Zhang: A multi-agent based manufacturing control strategy for responsive manufacturing, *Journal of Materials Processing Technology*, **Vol.139**, p.p.379–384, 2003

[19] Lim, M. K., D. Z. Zhang: An integrated agent-based approach for responsive control of manufacturing resources, *Computers & Industrial Engineering*, **Vol.46**, p.p.221–232, 2004

[20] Ramos, C.: An Architecture and a Negotiation Protocol for the Dynamic Scheduling of Manufacturing Systems, *IEEE International Conference on Robotics and Automation - ICRA1994*, San Diego, U.S.A., p.p.3161–3166, 1994

[21] Ramos, C.: A Holonic Approach for Task Scheduling in Manufacturing Systems, *IEEE International Conference on Robotics and Automation*, Minneapolis, U.S.A., p.p.2511–2516, 1996

[22] Sousa, P., C. Ramos: A distributed architecture and negotiation protocol for scheduling in manufacturing systems, *Computers in Industry*, **Vol.38**, p.p.103–113, 1999

[23] Julka, N., R Srinivasan, I. Karimi: Agent-based supply chain management—1: framework, *Computers and Chemical Engineering*, **Vol.26**, p.p.1755–1769, 2002

[24] Julka, N., I. Karimi, R. Srinivasan: Agent-based supply chain management—2: a refinery application, *Computers and Chemical Engineering*, **Vol. 26**, p.p.1771–1781, 2002

[25] García-Flores, R., X. Z. Wang, G. E. Goltz: Agent-based information flow for process industries' supply chain modelling, *Computers and Chemical Engineering*, **Vol.24**, p.p.1135–1141, 2000

[26] Koshijima, I., K. Niida, T. Umeda: A micro module approach to the design and control of autonomous decentralized chemical plant, *Journal of Process Control*, **Vol.6**, p.p.169–176, 1996

[27] Greenway, G.M., S. J. Haswell, D.O. Morgan, V. Skelton, M. Peter: The use of a novel microreactor for high throughput continuous flow organic synthesis, *Sensors and Actuators B*, **Vol.63**, **No.3**, p.p.153–158, 2000

[28] Löwe, H., W. Ehrfeld: State-of-the-art in microreaction technology: concepts, manufacturing and applications, *Electrochimica Acta*, **Vol.44**, p.p.3679–3689, 1999

[29] Kiwi-Minsker, L., O. Wolfrath, A. Renken: Membrane reactor microstructured by filamentous catalyst, *Chemical Engineering Science*, **Vol.57**, p.p.4947–4953, 2002

[30] Philipps, R., St. Walter, M. A. Liauw: Selective oxidation of isoprene to citraconic anhydride, *Chemical Engineering Journal*, **Vol.90**, p.p.107–116, 2002

[31] Hardt, S., W. Ehrfeld, V. Hessel, K. M. Vanden Bussche: Strategies for size reduction of microreactors by heat transfer enhancement effects, *Chemical Engineering Communications*, **Vol.190**, p.p.540–559, 2003

[32] Jänisch, K., M. Baerns, V. Hessel, W. Ehrfeld, V. Haverkamp, H. Löwe, Ch. Wille, A. Guber: Direct fluorination of toluene using elemental fluorine in gas/liquid microreactors, *Journal of Fluorine Chemistry*, **Vol.105**, **No.1** , p.p.117–128, 2000

[33] Yeong, K.K., A. Gavriilidis, R. Zapf, V. Hessel: Catalyst preparation and deactivation issues for nitrobenzene hydrogenation in a microstructured falling film reactor, *Catalysis Today*, **Vol.81**, p.p.641–651, 2003

[34] Rebrov, E. V., M. H. J. M. de Croon, J. C. Schouten: Design of a microstructured reactor with integrated heat-exchanger for optimum performance of a highly exothermic reaction, *Catalysis Today*, **Vol.69**, p.p.183–192, 2001

[35] Vogel, A., J. W. Schultze: A new microcell for electrochemical surface analysis and reactions, *Electrochimica Acta*, **Vol.44**, p.p.3751–3759, 1999

[36] Ehrfeld, W.: Electrochemistry and microsystems, *Electrochimica Acta*, **Vol.48**, p.p.2857–2868, 2003

[37] Küpper, M., V. Hessel, H. Löwe, W. Stark, J. Kinkel, M. Michel, H. Schmidt-Traub: Micro reactor for electroorganic synthesis in the simulated moving bed-reaction and separation environment, *Electrochimica Acta*, **Vol.48**, p.p.2889–2896, 2003

[38] Gray, B. L., D. Jaeggi, N.J. Mourlas, B. P. van Drieënhuizen, K. R. Williams, N. I. Maluf, G. T. A. Kovacs: Novel interconnection technologies for integrated microfluidic systems, *Sensors and Actuators A*, **Vol.77**, p.p.57–65, 1999

[39] González, C., S. D. Collins, R. L. Smith: Fluidic interconnects for modular assembly of chemical microsystems, *Sensors and Actuators B*, **Vol.49**, p.p.40–45, 1998

[40] 野間口 大 , 吉岡 真治 , 冨山 哲男 : 設計知識管理のための CAD アーキテクチャーの提案 , 人工知能学会論文誌 , **Vol.17**, **No.1**, p.p.104–113, 2002

[41] 野間口 大 , 下村 芳樹 , 冨山 哲男 : 設計者の思考過程のモデルを利用した設計知識管理システム , 人工知能学会論文誌 , **Vol.20**, **No.1**, p.p. 11–24, 2005

[42] Han, C., J. M. Douglas, G. Stephanopoulos: Agent-based Approach to a Design Support System for the Synthesis of Continuous Chemical Processes, *Computers and Chemical Engineering*, **Vol.19**, **Suppl.1**, p.p. S63–S69, 1995

[43] 吉岡 真治 , 冨山 哲男 : 設計支援のための統合モデリング環境の研究—プラガブル・メタモデル機構の提案—, 人工知能学会誌 , **Vol.13**, **No.2**, p.p. 150–157, 1998

[44] Wiedemann, T.: Database oriented modeling with simulation microfunctions, *Proc. of the 1999 Winter Simulation Conference*, Phoenix, U.S.A.,p.p.586–590, 1999

[45] Yang, A., M. Schlüter, B. Bayer, J. Krüger, E. Haberstroh, W. Marquardt: A concise conceptual model for material data and its applications in process engineering, *Computers and Chemical Engineering*, **Vol.27**, p.p.595–609, 2003

[46] Mynors, D. J., P. Broomhead, R. J. Grieve: Web-based material properties for use with metalforming simulation packages, *Journal of Materials Processing Technology*, **Vol.153–154**, p.p.688–693, 2004

[47] 古崎 晃司 , 來村 徳信 , 池田 満 , 溝口 理一郎 :「ロール」および「関係」に関する基礎的考察に基づくオントロジー記述環境の開発 , 人工知能学会論文誌 , **Vol.17**, **No.3**, p.p.196–208, 2002

[48] 石川 誠一 , 久保 成毅 , 古崎 晃司 , 來村 徳信 , 溝口 理一郎 : タスク・ドメインロールに基づくオントロジー構築ガイドシステムの設計と開発 —石油精製プラントを例として—, 人工知能学会論文誌 , **Vol.17**, **No.5**, p.p. 585–597, 2002

[49] 來村 徳信 , 笠井 俊信 , 吉川 真理子 , 高橋 賢 , 古崎 晃司 , 溝口 理一郎 : オントロジーに基づく機能的知識の体系的記述とその機能構造設計支援における利用 , 人工知能学会論文誌 , **Vol.17**, **No.1**, p.p.73–84, 2002

[50] 古崎 晃司 , 來村 徳信 , 佐野 年伸 , 本松 慎一郎 , 石川 誠一 , 溝口 理一郎 : オントロジー構築・利用環境「法造」の開発と利用—実規模プラントのオントロジーを例として—, 人工知能学会論文誌 , **Vol.17**, **No.4**, p.p.407–419, 2002

[51] 來村 徳信 , 溝口 理一郎 : オントロジー工学に基づく機能的知識体系化の枠組み , 人工知能学会論文誌 , **Vol.17**, **No.1**, p.p.61–72, 2002

[52] 亀井 剛次 , 湯川 高志 , 吉田 仙 , 桑原 和宏 : パーソナルレポジトリ間の協調情報検索—RDF を用いたパーソナルエージェントフレームワーク上への実装 , 人工知能学会論文誌 , **Vol.19**, **No.4**, p.p.292–299, 2004

[53] Tang, D.: An agent-based collaborative design system to facilitate active die-maker involvement in stamping part design, *Computers in Industry*, **Vol.54**, p.p.253–271, 2004

[54] Han, S., Y. Kim, T. Lee, T.S. Yoon: A framework of concurrent process engineering with agent-based collaborative design strategies and its application on plant layout problem, *Computers and Chemical Engineering*, **Vol.24**, p.p.1673–1679, 2000

[55] Krink, T.: Cooperation and Selfishness in Strategies for Resource Management, *Spill Science & Technology Bulletin*, **Vol.6**, **No.2**, p.p.165–171, 2000

[56] Krink, T., B. H. Mayoh, Z. Michalewicz: A Patchwork Model for Evolutionary Algorithms with Structured and Variable Size Populations, *Proc. of the Genetic and Evolutionary Computation Conference (GECCO-99)*, Orlando, U.S.A., **Vol.2**, p.p.1321–1328, 1999

[57] 森山 甲一 , 沼尾 正行 : 環境状況に応じて自己の報酬を操作する学習エージェントの構築 , 人工知能学会論文誌 , **Vol.17**, **No.6**, p.p.676–683, 2002

[58] 峯 恒憲 , 松野 大輔 , 雨宮 真人 : エージェントコミュニティを利用した P2P 型情報検索 , 人工知能学会論文誌 , **Vol.19**, **No.5**, p.p.421–428, 2004

[59] 髙玉 圭樹 : マルチエージェント学習—相互作用の謎に迫る—, 株式会社 コロナ社 , 2003

[60] Sen, S., M. Sekaran, J. Hale: Learning to coordinate without sharing information, *Proc. of the National Conference on Artificial Intelligence*, Seattle, U.S.A., p.p.426–431, 1994

[61] ANSI/ISA0S88.01-1995: Batch Control Part 1: Models and Terminology

[62] World Batch Forum: Batch Markup Language BatchML Version 02, World Batch Forum, 2003

[63] 清水 久二 , 福田 隆文 : 機械安全工学 —基礎理論と国際規格 , 養賢堂 , 2000

[64] 松原 望 : 意思決定の基礎 , シリーズ 意思決定の科学 1, 朝倉書店 , 2001

[65] 化学工学協会編 : 化学装置便覧 , 丸善 , 1970

[66] Hessel, V., H. Löwe, C. Hofmann, F. Schönfeld, D. Wehle, B. Werner: Process development of a fast reaction of industrial importance using a caterpillar micromixer/tubular reactor set-up, *in Proc. of 6th International Conference on Microreaction Technology (IMRET6)*, p.p. 39–54, 2002

[67] 有澤 誠 : コンピュータの思考法—計算モデル—, 情報数学の世界 3, 朝倉書店 , 2004

[68] 久保 幹雄 : 組合せ最適化とアルゴリズム , インターネット時代の数学シリーズ 8, 共立出版株式会社 , 2000

Nomenclature

Chapter2

記号	説　明 [単位]　(出現箇所)
$\boldsymbol{A}$	一対比較行列 [-]　(2.1.8)
a_{ij}	評価基準 i と評価基準 j についての一対比較値 [-]　(2.1.8)
C.I.	一対比較値の整合度 [-]　(2.1.8, 2.2.3, 3.1.2)
$Cost_{add,\ i}$	装置エージェント i の追加コスト [-]　(2.2.2)
$Cost_{add,\ max}$	入札した装置エージェント中の最もコストがかかる装置エージェントの追加コスト [-]　(2.2.2)
$Cost_{asmbl,\ i}$	装置エージェント i の組立•分解にかかるコスト [-]　(2.2.2)
$Cost_{asmbl,\ max}$	入札した装置エージェント中の最もコストがかかる装置エージェントの組立•分解にかかるコスト [-]　(2.2.2)
$eval_{addCost,\ i}$	追加コストの評価値 [-]　(2.2.2)
$eval_{asmblCost,\ i}$	組立•分解にかかるコストの評価値 [-]　(2.2.2)
$eval_{asmblTime,\ i}$	装置エージェント i の組立時間の評価値 [-]　(2.2.2)
$eval_{d,\ i}$	装置エージェント i の動的評価値 [-]　(2.2.2)
$eval_{e,\ i}$	装置エージェント i の経済的評価値 [-]　(2.2.2)
$eval_{f,\ i}$	装置エージェント i の機能性評価値 [-]　(2.2.2)
$eval_{o,\ i}$	装置エージェント i の運転評価値 [-]　(2.2.2)
$eval_{procAcc,\ i}$	装置エージェント i の処理精度の評価値 [-]　(2.2.2)
$eval_{procSpeed,\ i}$	装置エージェント i の処理速度の評価値 [-]　(2.2.2)
$eval_{prompt,\ i}$	装置エージェント i の即応性の評価値 [-]　(2.2.2)
$eval_{safety,\ i}$	安全性の評価値 [-]　(2.2.2)
$eval_{total,i}$	装置エージェント i の総合評価値 [-]　(2.2.2)
$F(t)$	不信頼度 [-]　(2.2.2)
$f(t)$	確率密度関数 [-]　(2.2.2)
m	形状パラメータ [-]　(2.2.2)
$R(t)$	信頼度 [-]　(2.2.2)
S_i	装置エージェント i の処理速度 [-]　(2.2.2)
S_{max}	入札した装置エージェント中で最速なものの処理速度 [-]　(2.2.2)

t	時刻 [-] (2.2.2)
$t_{asmblTime,i}$	装置エージェント i の組立•分解にかかる時間 [-] (2.2.2)
$t_{asmblTime,latest}$	入札した装置エージェント中の最も時間を要する装置エージェントの組立•分解にかかる時間 [-] (2.2.2)
$t_{prompt,i}$	処理開始指定時刻と装置エージェント i の作業開始可能時刻との差 [-] (2.2.2)
$t_{prompt,latest}$	入札した装置エージェント中の最も遅い処理開始可能時刻との差 [-] (2.2.2)
$\boldsymbol{w}$	固有ベクトル [-] (2.1.8)
w_d	動的評価の重み [-] (2.2.2, 2.2.3, 3.1.2)
w_e	経済的評価の重み [-] (2.2.2, 2.2.3, 3.1.2)
w_f	機能性評価の重み [-] (2.2.2, 2.2.3, 3.1.2)
w_i	評価基準 i の重み [-] (2.1.8)
w_j	評価基準 j の重み [-] (2.1.8)
w_o	運転評価の重み [-] (2.2.2, 2.2.3, 3.1.2)
α	装置への作業負荷の大きさの尺度 [-] (2.2.2)
γ	位置パラメータ [-] (2.2.2)
η	尺度パラメータ又は特性寿命 [-] (2.2.2)
$\lambda_{\max}$	行列 A の最大固有値 [-] (2.1.8)
$\lambda(t)$	故障率 [-] (2.2.2)

Chapter3

記号	説　明 [単位] (出現箇所)
$a_{ij,\,k}$	エージェント k が持つ評価基準 i と評価基準 j についての一対比較値 [-] (3.1)
C.I.	一対比較値の整合度 [-] (3.1.2)
$c_{ij,\,k}$	エージェント k が持つ一対比較値 $a_{ij,\,k}$ に対する確信度 [-] (3.1)
$compAdv_{k,i}$	装置エージェント k における製品 i に対する比較優位指標 [-] (3.2.1, 3.2.3)
$compLabor_{k,i,j}$	装置エージェント k における製品 j 基準の製品 i の比較生産費 [-] (3.2.1, 3.2.3)
$eval_{total}$	製品の総合評価値 [-] (3.1.2)

Q_i	特定ロットの製品 i の生産量 [-]	(3.2.1, 3.2.3)
$S_{k,i}$	装置エージェント k における製品 i の処理速度 [-]	(3.2.1, 3.2.3)
$t_{adv,\ i}$	製品 i の募集メッセージの受信時刻 [-]	(3.2.1)
$t_{close,\ i}$	製品 i の入札締切時刻 [-]	(3.2.1)
$t_{proc,k,i}$	装置エージェント k における製品 i の処理時間 [-]	(3.2.1, 3.2.3)
w_d	動的評価の重み [-]	(2.2.2, 2.2.3, 3.1.2)
w_e	経済性評価の重み [-]	(2.2.2, 2.2.3, 3.1.2)
w_f	機能性評価の重み [-]	(2.2.2, 2.2.3, 3.1.2)
w_o	運転評価の重み [-]	(2.2.2, 2.2.3, 3.1.2)

Chapter5

記号	説　明 [単位]	(出現箇所)
d	管内径 [mm]	(5.2.2)
$eval_{procAcc}$	処理精度評価値 [-]	(5.2.2)
$eval_{safety}$	安全性評価値 [-]	(5.2.2)
$eval_{simAcc,\ i}$	シミュレーションモデル i の計算精度の評価値 [-]	(5.2.2)
$eval_{simLoad,\ i}$	シミュレーションモデル i の計算負荷の評価値 [-]	(5.2.2)
K_i	シミュレーションモデル i の物理的なファイルサイズ [-]	(5.2.2)
K_{max}	シミュレーションモデル候補中で最大のファイルサイズ [-]	(5.2.2)
L	管長 [m]	(5.2.2)
S_i	装置エージェント i の処理速度 [-]	(5.2.2)
U	流速 [m/sec.]	(5.2.2)
x	任意の操作変数 [-]	5.1.2
x_{max}	任意の操作変数における適用範囲の最大値 [-]	5.1.2
x_{mid}	任意の操作変数における適用範囲の中点 [-]	5.1.2
x_{min}	任意の操作変数における適用範囲の最小値 [-]	5.1.2
Δx_i	操作変数 x の中点 x_{mid} からの変位 [-]	(5.2.2)
Δx_{max}	シミュレーションモデル候補中で最大の変位 [-]	(5.2.2)
σ	標準偏差 [-]	(5.2.2)

謝　　辞

本研究を行なうにあたり、熱心なご指導•ご教授を頂きました黒田 千秋 教授に深く感謝いたします。ご多忙にもかかわらず、本論文の審査を引き受けて下さいました小川 浩平 教授、太田口 和久 教授、渕野 哲郎 助教授、関口 秀俊 助教授に感謝致します。

また、たくさんのアドバイスを頂いた松本 秀行 助手や流体システム工学研究グループの小川 浩平 教授、吉川 史郎 助教授、大川原 真一 助手に感謝致します。
山浦 昭子 事務官および須田 眞理子さんをはじめ、黒田研究室、小川研究室、吉川研究室の皆さんに心から感謝致します。

本研究を進めるにあたり、沢山のディスカッションとアイデアを提供して頂いたマイクロ化学プロセス技術研究組合 (MCPT) の山本 挙さん、長野 みかさん、井上 敬一郎さんをはじめ、組合および関係者の方々に感謝いたします。

本論文の作成にあたり、細やかなチェックとサポートをしてくれた藤岡 沙都子さんに感謝します。

最後に、長い長い学生生活を支えてくれた家族に心から感謝します。ありがとう。