

博士論文

論文題目 組織構造と作業戦略を考慮した作業チーム設計手
法と最適化に関する研究

氏 名 満行 泰河

目次

目次.....	I
図目次.....	V
表目次.....	IX
第1章 序論.....	1
1.1 背景.....	2
1.2 目的.....	3
1.3 本論文の構成.....	4
第2章 関連研究.....	6
2.1 はじめに.....	7
2.2 組織論.....	7
2.3 データ分析に基づいたチームパフォーマンス評価.....	8
2.4 スケジューリングによるチームパフォーマンス評価.....	10
2.5 シミュレーションによるチームパフォーマンス評価.....	13
2.6 組織構造を考慮した作業チーム設計手法.....	14
2.7 本研究の位置づけ.....	20
第3章 提案手法.....	22
3.1 はじめに.....	23
3.2 提案する作業チーム設計手法の概要.....	23
3.2.1 プロジェクトの各段階における作業チーム設計の方針.....	23
3.2.2 作業チームの設計変数決定の流れ.....	25
3.3 チーム作業シミュレーション.....	26
3.3.1 概要.....	26
3.3.2 作業情報.....	28
3.3.3 作業チーム.....	30
3.3.4 チーム作業シミュレーションの流れ.....	35

3.3.5 作業チームのパフォーマンスの評価	45
3.4 手動探索による作業チーム設計手法	47
3.4.1 概要	47
3.4.2 作業チーム手動探索の詳細	47
3.4.3 探索結果を利用した作業チームの決定と分析	50
3.5 遺伝的アルゴリズムによる作業チーム設計支援手法	52
3.5.1 概要	52
3.5.2 組織構造の自動探索手法	53
3.5.3 作業戦略の自動探索手法	57
3.5.4 目的関数の設定に応じた探索計算の詳細	61
3.5.5 探索結果を利用した作業チームの決定と傾向分析	62
第4章 開発した作業チーム設計支援システム	65
4.1 はじめに	66
4.2 作業チーム設計支援システムの基本機能	66
4.2.1 作業情報の入力	68
4.2.2 作業チームの入力	70
4.2.3 作業戦略の入力	75
4.2.4 チーム作業シミュレーション機能	77
4.3 手動探索による作業チーム設計支援機能	80
4.4 遺伝的アルゴリズムによる作業チーム設計支援機能	83
4.4.1 組織構造の設計支援機能	83
4.4.2 作業戦略の設計支援手法	85
第5章 システム開発チームの組織構造設計	89
5.1 はじめに	90
5.2 問題設定	90
5.2.1 ケーススタディで想定する状況	90
5.2.2 提案手法の設定	92
5.3 結果	94
5.3.1 初期作業チーム(T1)	94
5.3.2 各部署に加えるべき人員の検討(T2 – T9)	97

5.3.3 部署間の依存関係についての検討(T10・T12)	100
5.3.4 手動探索の履歴と最終的に設計した作業チーム	106
5.4 まとめ	110
第6章 造船設計チームの組織構造設計	111
6.1 はじめに	112
6.2 問題設定	112
6.2.1 ケーススタディで想定する状況	112
6.2.2 提案手法の設定	115
6.3 結果	119
6.3.1 2つのワークフローを対象とした場合	119
6.3.2 3つのワークフローを対象とした場合	129
6.4 まとめ	142
第7章 複数の造船設計プロジェクトにおけるエフォート率設計	143
7.1 はじめに	144
7.2 問題設定	144
7.2.1 ケーススタディで想定する状況	144
7.2.2 提案手法の設定	147
7.3 結果	150
7.3.1 エフォート率の探索結果	150
7.3.2 人手で考えうる単純な作業戦略案との比較	152
7.4 まとめ	156
第8章 船殻ブロック組立におけるタスク割り振りルール設計	157
8.1 はじめに	158
8.2 問題設定	158
8.2.1 ケーススタディで想定する状況	158
8.2.2 提案手法の設定	161
8.3 結果	163
8.3.1 電力制限の有無によるタスク割り振りルールの比較	163
8.3.2 トラブル考慮の有無によるタスク割り振りルールの比較	170

8.4 まとめ	176
第9章 考察	177
9.1 はじめに	178
9.2 提案する作業チーム設計手法の比較	178
9.3 提案する作業チーム設計手法の有効性	180
9.4 提案する作業チーム設計手法の現場への導入	181
9.4.1 提案手法の適用範囲について	181
9.4.2 現場で求められる困難な作業チーム設計問題への対応	182
第10章 結論	185
10.1 結論	186
10.2 今後の展望	187
謝辞	189
参考文献	192
研究業績	197

図目次

図 1-1 本論文の構成.....	5
図 2-1 CORAS のインターフェイス[19]	9
図 2-2 Dickinson & McIntyre のチームワーク要素モデル[23]	10
図 2-3 PERT のアローダイアグラム[30].....	11
図 2-4 フローショップスケジューリング問題の概要.....	12
図 2-5 ジョブショップスケジューリング問題の概要.....	12
図 2-6 チームパフォーマンス評価モデル[28].....	14
図 2-7 Virtual Design Team モデルの概要[37].....	15
図 2-8 短期型プロジェクトにおけるアメリカと日本の VDT モデル[39].....	16
図 2-9 PMT のコンセプトとインターフェイス[40].....	18
図 2-10 TeamPort のインターフェイス[42]	19
図 3-1 プロジェクトの各段階における作業チーム設計方針の概要	24
図 3-2 チーム作業シミュレーションの概要.....	26
図 3-3 本研究における組織構造のモデル化.....	33
図 3-4 チーム作業シミュレーションのフロー	39
図 3-5 優先度に基づいたタスク割り振りの流れ.....	42
図 3-6 チーム作業シミュレーション結果の出力例	45
図 3-7 手動探索による作業チーム設計手法の概要	47
図 3-8 作業チーム手動探索の流れ.....	48
図 3-9 探索結果をもとにした最終設計案決定の流れ.....	50
図 3-10 遺伝的アルゴリズムによる作業チーム設計支援手法の流れ	52
図 3-11 組織構造の設計支援手法の概要	54
図 3-12 組織構造を表す遺伝子のコード化	55
図 3-13 スキル情報に基づく組織構造情報のフィルタリング	56
図 3-14 作業戦略の設計支援手法概要	57
図 3-15 エフォート率の探索アルゴリズムにおける遺伝子のコード化[46].....	59
図 3-16 タスク割り振りルールを表す遺伝子のコード化	60
図 3-17 提案手法による自動探索計算の詳細な流れ	62

図 3-18 探索結果から最終的な作業チーム設計案を選択する流れ	63
図 3-19 シナリオの変化に対するタスク割り振りルールの変化分析手法	64
図 4-1 作業チーム設計支援システムの基本機能の概要	67
図 4-2 ワークフロー入力画面	68
図 4-3 ワークフロー保存ファイルの例	69
図 4-4 作業実施における制約の入力画面	70
図 4-5 ワークフローフォルダの選択	71
図 4-6 人員と工作機械情報の入力画面	72
図 4-7 部署間の依存関係情報の入力画面	73
図 4-8 組織構造情報保存ファイルの例	74
図 4-9 各ワークフローへのエフォート率の入力画面	75
図 4-10 タスク割り振りルールの入力画面	76
図 4-11 チーム作業シミュレーション設定の入力画面	77
図 4-12 チーム作業シミュレーション実行中の画面	79
図 4-13 ガントチャートの例	79
図 4-14 手動探索による作業チーム設計支援機能の概要	80
図 4-15 手動探索に関するメモの入力画面	81
図 4-16 手動探索の履歴が記録された CSV ファイル	82
図 4-17 組織構造の設計支援機能の概要	83
図 4-18 組織構造の設計支援機能実行に出力されるファイルなど	84
図 4-19 エフォート率の設計支援機能の概要	85
図 4-20 エフォート率の設計支援機能実行後の画面	86
図 4-21 タスク割り振りルールの設計支援機能の概要	87
図 4-22 タスク割り振りルールの設計支援機能実行に出力されるファイル	88
図 5-1 対象とするシステム開発業務を表すワークフロー	91
図 5-2 初期作業チーム(T1)の組織構造	95
図 5-3 作業チーム(T1)のシミュレーション結果例	96
図 5-4 作業チーム(T2 - T7)のチーム作業シミュレーション結果例	98
図 5-5 作業チーム(T7)までのパフォーマンス変化	98
図 5-6 作業チーム(T8- T9)のチーム作業シミュレーション結果例	99
図 5-7 作業チーム(T9)までのパフォーマンス変化	100

図 5-8 作業チーム(T10)の組織構造	101
図 5-9 作業チーム(T10)のチーム作業シミュレーション結果例	102
図 5-10 作業チーム(T10)までのパフォーマンス変化.....	103
図 5-11 作業チーム(T11 と T12)の組織構造	104
図 5-12 作業チーム(T11 と T12)のチーム作業シミュレーション結果例.....	105
図 5-13 作業チーム(T12)までのパフォーマンス変化.....	106
図 5-14 最終的に設計された作業チーム(T16)	108
図 5-15 作業チーム(T16)のチーム作業シミュレーション結果例	109
図 5-16 設計作業の繰り返しによるパフォーマンスの変化.....	109
図 6-1 1つの造船設計プロジェクトを表すワークフロー	113
図 6-2 提案手法による探索結果(Case 1-1)	120
図 6-3 パレートフロントを構成する組織構造の比較(Case 1-1)	121
図 6-4 Case1-1 で所要時間の最も短かった組織構造(PO11-1)	122
図 6-5 PO11-1 におけるチーム作業シミュレーション結果の一例	122
図 6-6 Case1-2 で合計コストの最も少なかった組織構造(PO11-4)	123
図 6-7 PO11-4 におけるチーム作業シミュレーション結果の一例	123
図 6-8 提案手法による探索結果(Case 1-2)	124
図 6-9 パレートフロントを構成する組織構造の比較(Case 1-2)	125
図 6-10 Case1-2 で所要時間の最も短かった組織構造(PO12-1)	126
図 6-11 PO12-1 におけるチーム作業シミュレーション結果の一例.....	126
図 6-12 Case1-2 で合計コストの最も少なかった組織構造(PO12-15)	127
図 6-13 PO12-15 におけるチーム作業シミュレーション結果の一例	127
図 6-14 Case1-1 と Case1-2 で選択された組織構造の評価結果.....	128
図 6-15 提案手法による探索結果(Case 2-1)	129
図 6-16 パレートフロントを構成する組織構造の比較(Case2-1)	130
図 6-17 Case2-1 で所要時間の最も短かった組織構造(PO21-1)	131
図 6-18 PO21-1 におけるチーム作業シミュレーション結果の一例	131
図 6-19 Case2-1 で合計コストの最も少なかった組織構造(PO21-7)	132
図 6-20 PO21-7 におけるチーム作業シミュレーション結果の一例	132
図 6-21 提案手法による探索結果(Case 2-2)	133
図 6-22 パレートフロントを構成する組織構造の比較(Case2-2)	134

図 6-23 Case2-2 で所要時間の最も短かった組織構造(PO22-1)	135
図 6-24 PO22-1 におけるチーム作業シミュレーション結果の一例	135
図 6-25 Case2-2 で合計コストの最も少なかった組織構造(PO22-14)	136
図 6-26 PO22-14 におけるチーム作業シミュレーション結果の一例	136
図 6-27 提案手法による探索結果(Case 2-3)	137
図 6-28 パレートフロントを構成する組織構造の比較(Case2-3)	138
図 6-29 Case2-3 で所要時間の最も短かった組織構造(PO23-1)	139
図 6-30 PO23-1 におけるチーム作業シミュレーション結果の一例	139
図 6-31 Case2-3 で合計コストの最も少なかった組織構造(PO23-13)	140
図 6-32 PO23-13 におけるチーム作業シミュレーション結果の一例	140
図 6-33 Case2-1 と Case2-2 と Case2-3 で選択された組織構造の評価結果.....	141
図 7-1 1つの造船設計プロジェクトを表すワークフロー	145
図 7-2 造船設計作業チームの人員に関する依存関係情報	147
図 7-3 算出されたエフォート率案によるシミュレーション結果	151
図 7-4 タスクの掛け持ちを禁止した場合のシミュレーション結果	153
図 7-5 エフォート率を均等に配分した場合のシミュレーション結果	154
図 7-6 各ケーススタディにおける単位時間ごとの作業人員数の推移	155
図 8-1 1枚の船殻ブロック組み立て工程を示すワークフロー	159
図 8-2 ブロック組立作業チームの人員に関する依存関係情報	160
図 8-3 Case1 のチーム作業シミュレーション結果	164
図 8-4 Case2 のチーム作業シミュレーション結果	164
図 8-5 Case1 と Case2 での使用電力量の推移	165
図 8-6 Case3 のチーム作業シミュレーション結果	167
図 8-7 Case4 のチーム作業シミュレーション結果	167
図 8-8 Case3 と Case4 での使用電力量の推移	168
図 8-9 Case5 におけるタスク割り振りルールの探索結果	171
図 8-10 Case6 におけるタスク割り振りルールの探索結果	173
図 8-11 電力制限の有無によるパレートフロントの比較	175
図 8-12 得られたタスク割り振りルールによるシミュレーション結果の範囲 ..	175

表目次

表 3-1	タスクの属性.....	29
表 3-1	組織情報の例.....	31
表 3-2	組織構造によって定義された権限によってスキルが限定された例.....	33
表 3-3	本研究で採用したディスパッチングルール.....	35
表 3-4	チーム作業シミュレーションの説明で使用する記号.....	36
表 3-5	作業チーム設計における設計変数の候補.....	49
表 3-6	手動探索における作業チーム設計の履歴.....	51
表 4-1	作業チーム設計支援システムの対象に関する情報.....	66
表 5-1	IT システム開発チームの候補人員に関する情報.....	92
表 5-2	第 5 章における提案手法の設定.....	93
表 5-3	第 5 章における作業チームの設計変数と固定する変数.....	94
表 5-4	初期作業チーム(T1)内の人員に関する情報.....	95
表 5-5	作業チーム(T10)内の人員に関する情報.....	101
表 5-6	作業チーム(T11)内の人員に関する情報.....	104
表 5-7	作業チーム(T12)内の人員に関する情報.....	105
表 5-8	組織構造の手動探索履歴.....	107
表 5-9	作業チーム(T16)内の人員に関する情報.....	108
表 6-1	造船設計組織の人員に関する情報.....	114
表 6-2	第 6 章における作業情報の設定.....	116
表 6-3	第 6 章における提案手法の設定.....	117
表 6-4	第 6 章における作業チームの設計変数と固定する変数.....	118
表 7-1	造船設計作業チームの人員に関するスキルセット情報.....	146
表 7-2	第 7 章における提案手法の設定.....	148
表 7-3	第 7 章における作業チームの設計変数と固定する変数.....	149
表 7-4	算出されたエフォート率案.....	150
表 8-1	船殻ブロック組み立て作業チームの人員と工作機械に関する情報.....	160
表 8-2	第 8 章における探索手法の設定.....	162
表 8-3	各シナリオの設定.....	162

表 8-4 第 8 章における作業チームの設計変数と固定する変数	163
表 8-5 Case 1 と Case 2 のタスク割り振りルールの変化分析.....	166
表 8-6 Case 3 と Case 4 のタスク割り振りルールの変化分析.....	169
表 8-7 電力消費量ピーク時に使用されている工作機械.....	170
表 8-8 Case3 と比較した場合の Case 5 のタスク割り振りルールの変化分析...	172
表 8-9 Case4 と比較した場合の Case 6 のタスク割り振りルールの変化分析...	174
表 9-1 提案手法内におけるそれぞれの作業チーム設計手法の比較	178

第1章 序論

1.1 背景.....	2
1.2 目的.....	3
1.3 本論文の構成.....	4

1.1 背景

組織やチームのパフォーマンス向上は、活動を行う人間集団にとって重要な課題である。企業への IT システムの導入やトヨタ生産方式に代表される業務改善はそのような取り組みの代表的な例として挙げられる。

製造業や軍事関係などの、特に高度な安全性と信頼性が求められる現場においては、対象とするプロジェクトに関して、専門知識や技術を持った組織内の複数の人員が 1 つのチームとなってプロジェクトに取り組む必要がある。これは、遂行するプロジェクト内の各タスクで多様なスキルや判断・経験を必要とする場合、熟練者 1 人の判断よりも共通の目的を持った作業チームの判断・経験に従った方が高い業績を上げることができると知られているからである [1][2]。

応用行動科学分野の組織行動学によると、チームとは「明確な業績目標があり、それに向けて素早く行動、実施、方向転換することができ、協調を通じてプラスの相乗効果を生み出すことが可能な集団」と定義することが出来る。また、チームを構成するメンバーに必要な能力として、①技術的専門スキル、②問題解決および意志決定のスキル、③対人的スキルが挙げられる[3]。

このようなチームで行う業務では、日々試行錯誤によるパフォーマンス向上への取り組みが行なわれている[4]。原子力プラントのコントロールルーム、航空機のcockpit、大型船舶のブリッジなどの作業現場では、新しいシステムの導入や作業マニュアルの改善といった手法を導入することによるチームパフォーマンスの改善が試みられている。一方で、組織内での動的なタスク分担を必要とするプロジェクトでは、実際にプロジェクトに取り組む前にそのプロジェクトの性質やプロジェクト期間などの様々な情報を考慮して、適切に作業チームを設計する必要がある。

一般に、プロジェクトを適切に運営・実施するためには、プロジェクトの進め方とプロジェクト内における各タスクに対する必要なスキルを考慮して、組織内の人員や工作機械などの機材を適切に配置する必要がある[5]。加えて、トラブルなどの不確実性が高いプロジェクトでは、単に作業を実施するだけでなく、トラブルに対応するために上司や上流工程の担当者との調整を含めた作業チームの組織構造まで考慮して作業チームの設計を行う必要がある。さらに、人員や機械だけでなく、作業チーム全体として実際にどのようにプロジェクトに取り組むかを

表す作業戦略の違いによって、作業チームのパフォーマンスは大きく依存する。

製造業などの生産現場でも同様の理由により、専門知識・技術を持ったそれぞれの人員がチームとなって取り組む業務やプロジェクトが多くの割合を占めている。特に、造船・航空・宇宙産業などの大規模な構造物を対象にしたプロジェクトでは、企業の中の大勢の人員と数多くの工作機械が連携して作業チームとして取り組む必要がある。しかしながら、組織構造や作業戦略などの含めた作業チーム設計を行う場合は、考慮すべき設計変数が大量に存在するため定量的な評価に基づいた適切な作業チームの設計を行うことは難しく、現場では人材管理システムやスケジューリングソフトなどの様々な IT システムを用いつつも、設計者の勘と経験に基づいた作業チーム設計を行っているのが現状である。また、IT 企業などで頻繁に行われているシステム開発を対象とするプロジェクトでも、上流工程と下流工程の間で異なる部署間の連携や外部の会社との連携が必要となることが多く、どのような組織構造でプロジェクトに取り組むべきかや、どのような作業戦略でプロジェクトを運営すべきかについては、対象とするプロジェクト管理者の勘と経験によって決定しているのが現状である。

1.2 目的

本研究は、組織構造と作業戦略を考慮した作業チームの設計手法を提案することを目的とする。具体的には、作業チーム内でのタスク割り振りと共同作業やチーム内でのトラブル対応による遅延などを考慮したチーム作業シミュレーションを利用して、開発した作業チーム設計支援システム上で設計と評価を繰り返すことで、適切に作業チームの設計を行う手法を提案する。また、組織構造や作業戦略などの人手で決定することが難しい作業チームの各設計変数に対して、遺伝的アルゴリズムを用いて作業チームの設計を支援する手法を提案する。

また、システム開発プロセスと造船業の設計・生産プロセスを対象に提案手法を適用し、開発した作業チーム設計支援システムと遺伝的アルゴリズムを用いた作業チームの各設計変数に対する設計支援手法が有用であることを示す。

1.3 本論文の構成

本論文の構成を図 1-1 に示す。

第 2 章では、本研究と関連する知識や研究事例を挙げ、本研究の位置づけについて述べる。

第 3 章では、作業チーム設計手法と、各設計変数に対する遺伝的アルゴリズムを用いた作業チーム設計支援手法について述べる。

第 4 章では、前章で提案した手法を実装した作業チーム設計支援システムの詳細について述べる。

第 5 章から第 8 章は、提案手法と開発した作業チーム設計支援システムを用いたケーススタディについて述べる。

第 5 章では、開発した作業チーム設計支援システム上で、IT システム開発チームの設計を行う。

第 6 章では、組織構造を考慮した作業チーム設計支援手法を船舶基本設計プロセスに適用し、提案手法の評価と組織構造の分析を行う。

第 7 章では、作業チーム内の各人員のエフォート率設計手法を船舶基本設計プロセスに適用し、提案手法の評価を行う。

第 8 章では、電力ピークカットを考慮した船殻ブロック組立プロセスを対象にタスク割り振りルール of 自動設計手法を適用し、提案手法の評価と電力制限の有無によるチームパフォーマンスの評価を行う。

第 9 章では、ケーススタディ全体を通して提案手法の有効性について考察する。

第 10 章では、結論と今後の展望について述べる。

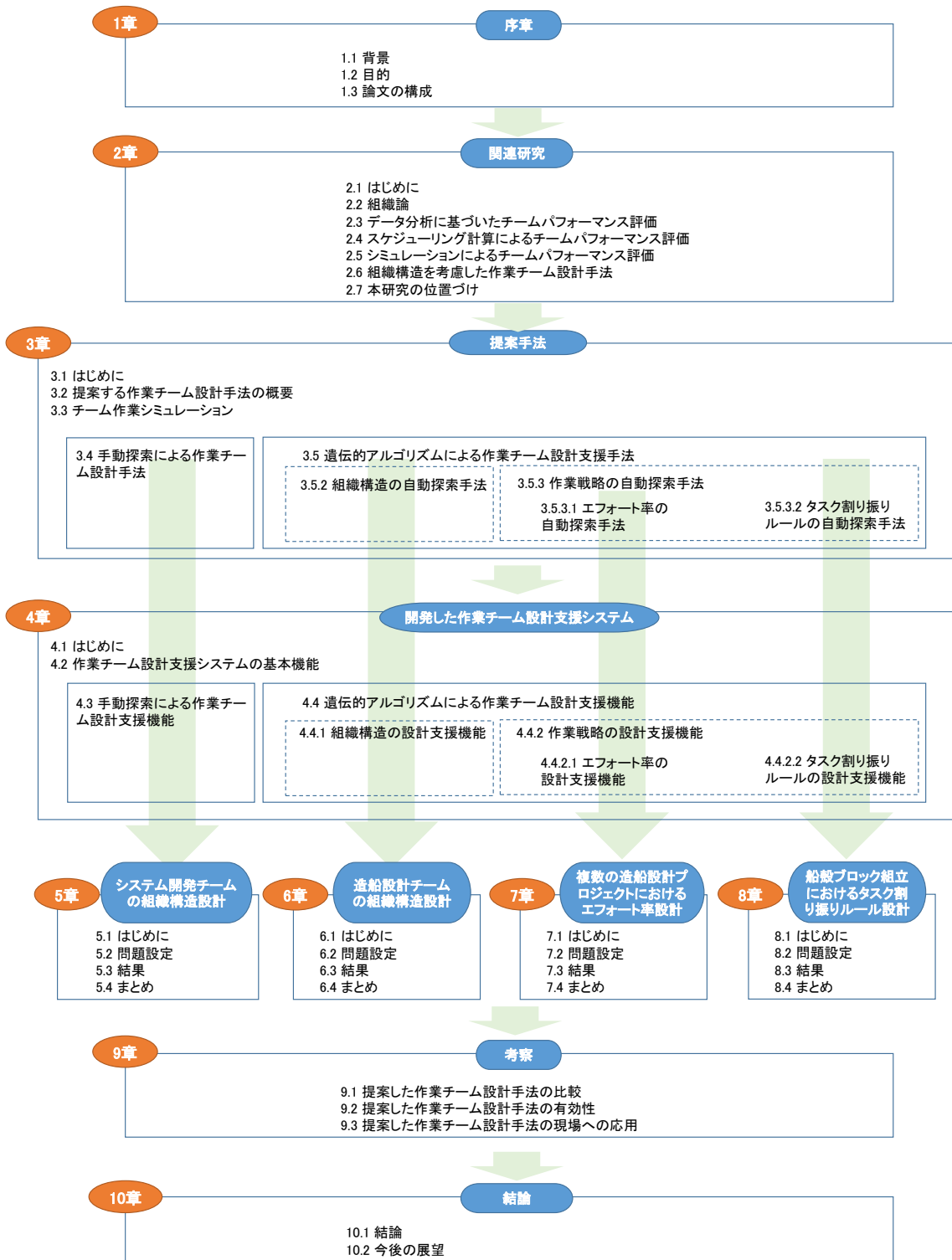


図 1-1 本論文の構成

第2章 関連研究

2.1 はじめに	7
2.2 組織論	7
2.3 データ分析に基づいたチームパフォーマンス評価	8
2.4 スケジューリングによるチームパフォーマンス評価	10
2.5 シミュレーションによるチームパフォーマンス評価	13
2.6 組織構造を考慮した作業チーム設計手法	14
2.7 本研究の位置づけ	20

2.1 はじめに

本章では、作業チーム設計に関する既存研究について述べる。組織論、チームパフォーマンスやチーム間での関わり合いをデータ分析に基づいて行う研究、シミュレーションによるチームパフォーマンスの評価手法と作業チーム設計の研究事例について概観する。

2.2 組織論

チームや組織のあるべき姿に関しては、社会学や経営学などの分野で組織論[6]として1900年初頭から盛んに議論がされている。特に、より良い組織構造に関しては、組織論の中でも時代によって様々な提言がなされている。Weber[7]は、組織とは「構造・階層・権限・役割」によって定義されると述べている。Bernard[8]は、人間の意思決定が集合することによって制約が生まれ、協調作業をすることによって組織が生まれると論じている。また、組織が成立する要素として「共通目的・貢献意欲・コミュニケーション」が必要であると定義している。Simon[9]らは、組織とは「役割分担・目標・報償・行動」によって決定するものと捉え、Bernardの考え方をさらに発展させて組織論という学問分野を確立させた。1990年代後半になると、組織が直面する状況は多種多様かつ不確実であるため唯一最良の組織構造があるわけではないというコンティンジェンシー理論[10]が提唱され、組織とは統制範囲やルールが存在するサブシステムごとに管理されたものであると捉えられている[11]。

以上より、組織構造に関しては時代が進むにつれて固定的な中央集権型から、個別の状況に適応する分散型の組織構造に変化していると言える。このため、対象の業務やプロジェクトに応じた作業チームや組織の設計と定量的なパフォーマンス評価を行う技術の開発が期待されている。

2.3 データ分析に基づいたチームパフォーマンス評価

生産現場におけるパフォーマンス評価手法の代表的な例として、業務改善手法の総称である IE(Industrial Engineering)[12][13][14]が挙げられる。IEとは、それぞれの職種で実際に高い業績を出している熟練者に共通してみられる行動特性を動作分析によって算出し、模範的な行動を導き出そうとする手法である。標準作業・時間の設定を行うことで作業者に標準作業量を設定し作業の管理を容易にする手法であるテーラーの科学的管理法[15]が発端であると言われており、現在では基本動作の大きさとなる移動距離・動作の難易度と時間値表を用いて作業時間を設定する方法であるサブリック分析[15]や MTM(Method Time Measurement)法[16]に代表される動作分析と、全体の工程をフロー形式で表す工程表を作成し、スループット時間の短縮などを目指す工程分析の2つに大別することが可能である。

標準的な IE は個別の作業員個人を対象に行うものであるが、現在では IE の分析方法を応用して様々な分野でチームパフォーマンスの分析と評価が行われている。特に原子力分野では、10年以上前から安全性の追求を目的としてチームワークに着目したプラント操業のパフォーマンス評価に関する研究が行われている。佐相[17][18]らは多重故障に対応するためのチーム行動の特性を把握するために、「ゴール・サブゴール・ストラテジー」という概念を導入し、それらを基準に観察やアンケートによるチーム作業の分析を行うことで、チームの直面している状況によって各運転員の発声数と発声の難易度に違いがみられることを明らかにした。また、安藤ら[19]は船舶ブリッジにおけるチーム作業のビデオや音声を記録し、対話の定量的な分析を行なっている。シミュレータトレーニングから得られるビデオ・音声・シミュレータのログといった多様な時系列データを取り扱う事のできる CORAS (COoperation Record and Analysis System)を開発し、CORAS を用いることで船舶ブリッジチームにおける協調作業のパターンを整理し、ブリッジ業務における役割分担と協同の重要性を知見として得ることができている。CORAS のインターフェイスを図 2-1 に示す。CORAS は人間の行動を時系列データの探索によって分析する手法である ESDA (Exploratory Sequential Data Analysis) [20][21]を支援するシステムであり、図 2-1 の左側に示すビデオデータや右側の航海情報などのマルチメディアを含めた時系列データに対し、状況や対話などの情報を表すアノテーションやコードの付加を行うことで、それらのデータを構造化し分析の効率化を図ることが可能である。

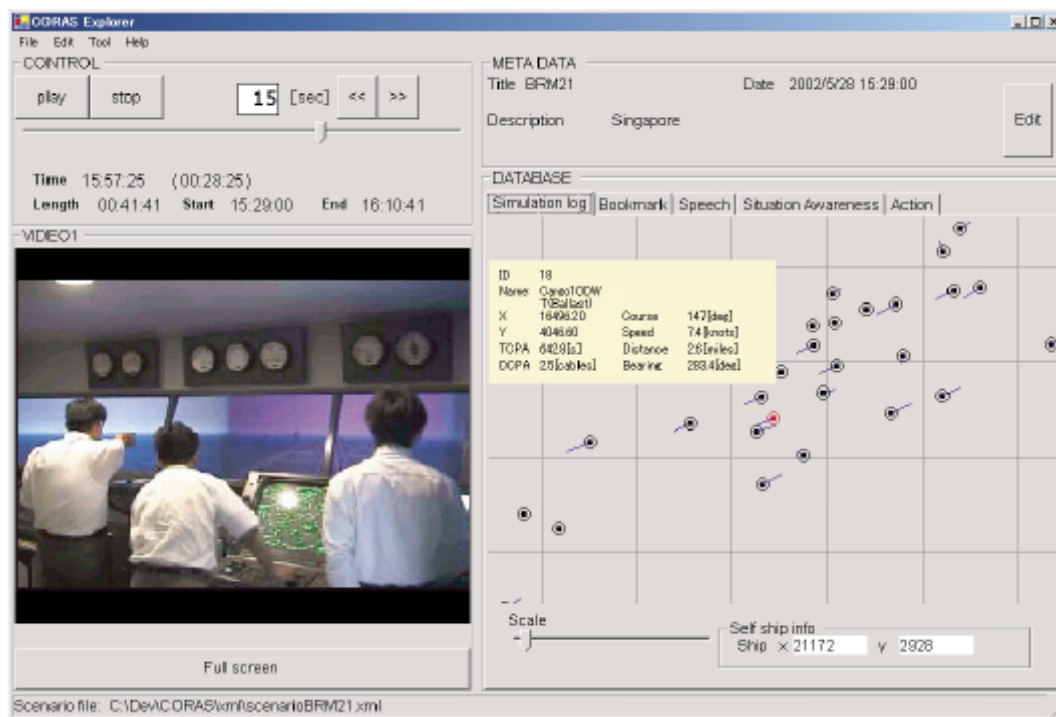


図 2-1 CORAS のインターフェイス[19]

また、満行ら[22]は製鉄所の連続鋳造業務を対象として、作業チームの各人員の動線データをチーム単位で分析・比較することにより、トラブルなどの通常の工程とは逸脱した時間帯を抽出可能であることを示している。

生産現場以外におけるチーム作業分析の研究では、三沢ら[23]が Dickinson & McIntyre のチームワーク要素モデルを元にした、アンケートによる看護師チームのチームワーク測定指標として TMN (Teamwork Measure for Nursing teams) を提案し、現状との比較による TMN の妥当性を示した。Dickinson & McIntyre のチームワーク要素モデルを図 2-2 に示す。Dickinson & McIntyre のチームワーク要素モデルでは、チームワークの行動変数とその基盤となる心理的変数の両方を考慮し、重要な要素として図 2-2 の 7 つの要素を挙げている。このモデルを使用する際には、メンバーをチームに関するインフォーマントとし、チームワークに関する全体的な評価を求める手法が一般的に使われている。

上記の研究例で示した通り、チームワークを基準にしたパフォーマンス評価に関しては、

観察者や分析者の主観でチームワークの評価指標を設計し、それらに必要なデータを取得するのが一般的であると言える。

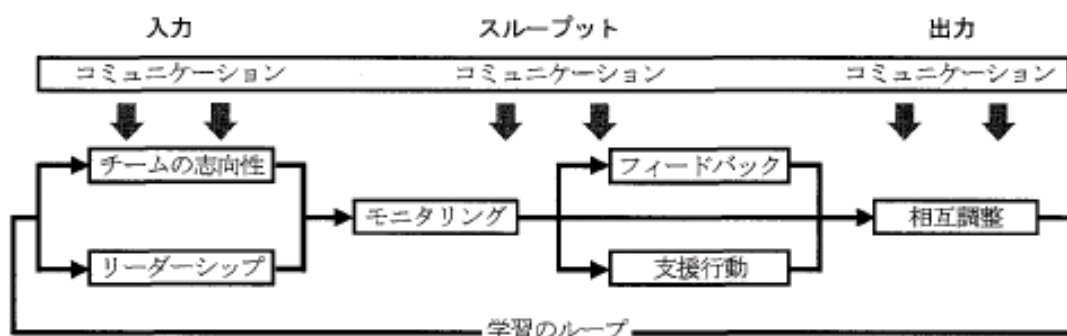


図 2-2 Dickinson & McIntyre のチームワーク要素モデル[23]

2.4 スケジューリングによるチームパフォーマンス評価

実際のプロジェクトや業務遂行時に発生するデータからチームパフォーマンスを評価するボトムアップ的なアプローチだけでなく、作業チームをモデル化しスケジューリング計算を行うことでチームパフォーマンスを評価するトップダウン的なアプローチも多くの分野で行われてきた。

対象とするプロジェクト内の各作業の実施に対して、必要となるリソースが制限なく使用可能な場合のスケジュールは、PERT/CPM (Program Evaluation and Review Technique/Critical Path Method) [30]と呼ばれる手法により最適解を求めることができる。PERT は、1950 年代後半にアメリカ海軍によって開発された、オペレーションズ・リサーチの分野の手法である。PERT では、プロジェクト全体を構成する各作業の相互依存関係をネットワーク図にすることで、各作業の開始・完了時刻や余裕時間、プロジェクト全体の所要時間を求めたり、クリティカルパスを明らかにしたりする。ネットワーク図の作成にはアローダイアグラムという表記法を用いるのが特徴である。アローダイアグラムでは、リンク矢印(アロー)がタスクを表し、ノードがタスクの開始点・終了点を表すという点がワークフローと大きく異なる。CPM(Critical Path

Method)は, PERT で求められた所要時間をさらに早める必要がある場合に用いられる. 各作業で追加費用をかけると所要時間が短縮できるとしたとき, 費用増加を最小にするにはどの作業を短縮すればよいかという最適化問題を解くことができる.

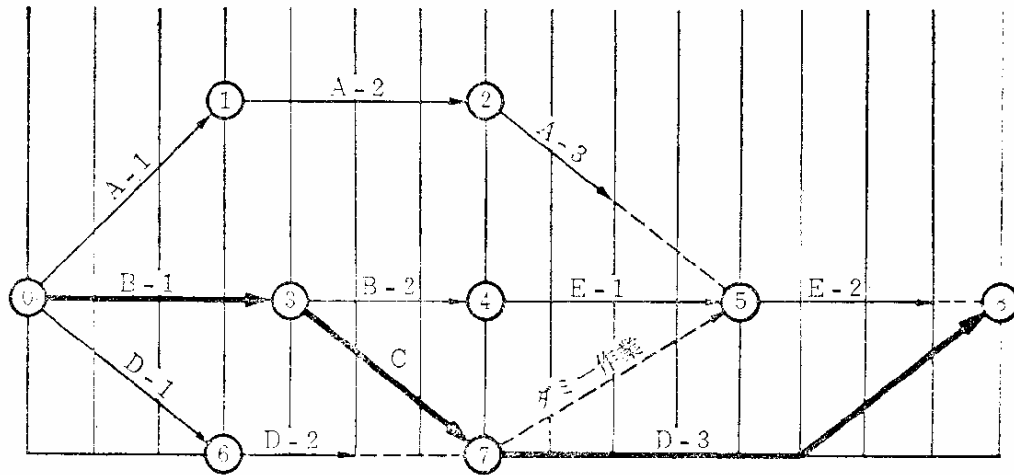


図 2-3 PERT のアローダイアグラム[30]

しかしながら PERT/CPM は作業に対して利用できるリソースに制限がない場合は万能な手法であるが, 現実問題では作業に対して利用できるリソースは有限である場合が多い. それらの問題に対応する方法として, フローショップスケジューリング問題やジョブショップスケジューリング問題[31]などが有名である.

フローショップスケジューリング問題の概要図を図 2-4 に示す. 1つの物的システムで生産が予定されている製品の加工順序に従ってリソースを配列し, それを用いて加工対象物を生産する場合のシステムの構成をフローショップと呼ぶ. このようなフローショップを構成する各リソースでの加工時間を考慮して加工対象物の投入順序を決定する問題はフローショップスケジューリング問題に分類される.

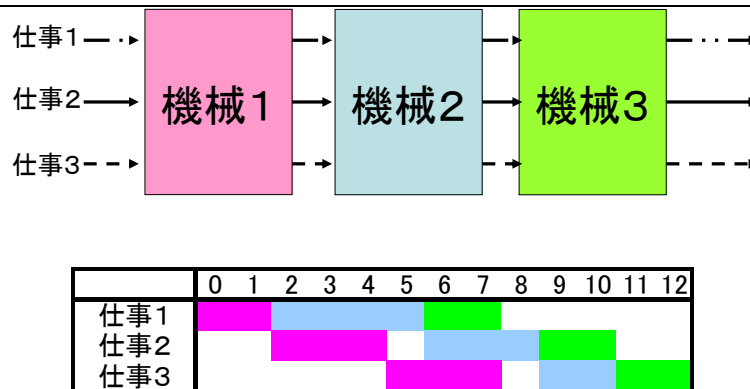


図 2-4 フローショップスケジューリング問題の概要

ジョブショップスケジューリング問題の概要を図 2-5 に示す. 加工対象物によって加工順序が異なる場合, フローショップのようなリソース配列は使用できない. そのため, 加工対象物はそれぞれの加工順序に従ってリソースに送られ, 加工が行われる. この場合, リソースごとに生産順序を決定する必要がある. このような問題はジョブショップスケジューリング問題に分類される.

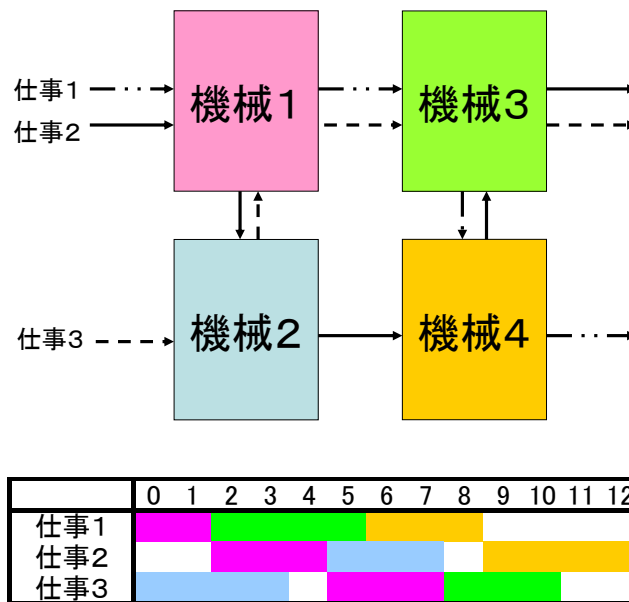


図 2-5 ジョブショップスケジューリング問題の概要

また、それらを一般化した資源制約付きプロジェクトスケジューリング問題(RCPSP (Resource-Constrained Project Scheduling Problem))[32][33][34]などの様々な定式化が行われている。例えば、Yamatoら[35]は、造船所の船殻ブロック組み立て工程を対象に、遺伝的アルゴリズムを用いた最適化スケジューリング技術を用いて、作業チームに属する新人への教育がどの程度効果を与えるかを定量的に測定する手法を提案している。

2.5 シミュレーションによるチームパフォーマンス評価

作業チームのパターン化されたタスクをタスクネットワークや DFM(Dynamic Flowgraph Methodology)[24]といった一種のワークフローの形で基準となるチーム作業を定義し、それぞれのタスクの終了時間を測定し、タスクを最小単位としてチーム作業を評価する取り組みが数多く行なわれている。例えば、Laughery[25]はタスクネットワークシミュレーションを用いて原子力プラントオペレータのワークロードを予測した。また、Yowら[26]はシミュレータ実験から得られたデータを用いることで、原子力プラントにおいて行なわれるパターン化されたオペレーションの終了時刻などがある程度予測できることを示している。また、Kontogiannis[27]はタスクネットワークと認知モデルを、数学的な離散事象システムをモデル化するための手法の一つである色つきペトリネットによって表現することで、タスクネットワークを用いながら組織内での動的なタスク分担といったチーム行動を表現し、常にメンバーの仕事量をモニタリングしながら適切なタスク分担を行うことで仕事量の増加に柔軟に対応する組織のシミュレーションを実現している。さらに、角田ら[28][29]はタスクネットワークシミュレーションを用いて船舶のブリッジ業務におけるトレーニングシナリオを定量的に評価する研究を行なっている。角田らが提案するチームパフォーマンス評価モデルを図 2-6 に示す。チームパフォーマンス評価モデルは、タスクネットワークモデルとタスクネットワークを制御する他のモデルから成り、3Dアニメーションを利用したインタビューなどを行いながらタスクネットワークの修正を行う。最終的には、状況認識の観点からチームのパフォーマンス評価を行う。

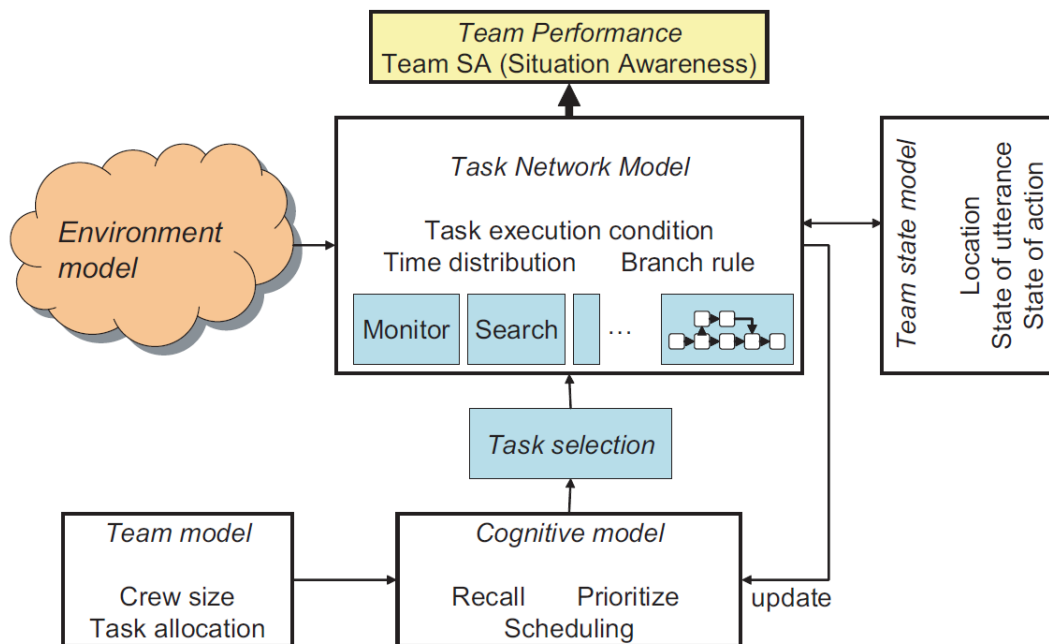


図 2-6 チームパフォーマンス評価モデル[28]

2.6 組織構造を考慮した作業チーム設計手法

作業チームのパフォーマンス評価を行う際に、作業チームの人員間の関係を表す組織構造に着目してパフォーマンスの評価やどのような作業チームを設計すればよいかを検討するための研究も盛んに行われている。Levittら[36]やJin[37][38]らは、Virtual Design Team(VDT)という組織構造における設計者間の調整を考慮して設計プロセスを進めるシミュレーションモデルを提案している。VDTでは、人員間の大まかな連絡経路を組織構造によって定義し、それぞれの人員の作業実施のモデル化に基づいてシミュレーションを行う。また、対象とするタスクネットワークに含まれる各タスクの属性として「Flexibility・Complexity・Uncertainly・Interdependence strength」を導入することで、組織構造内で行われる人員間のコミュニケーションなどを定量的に評価することが可能である。

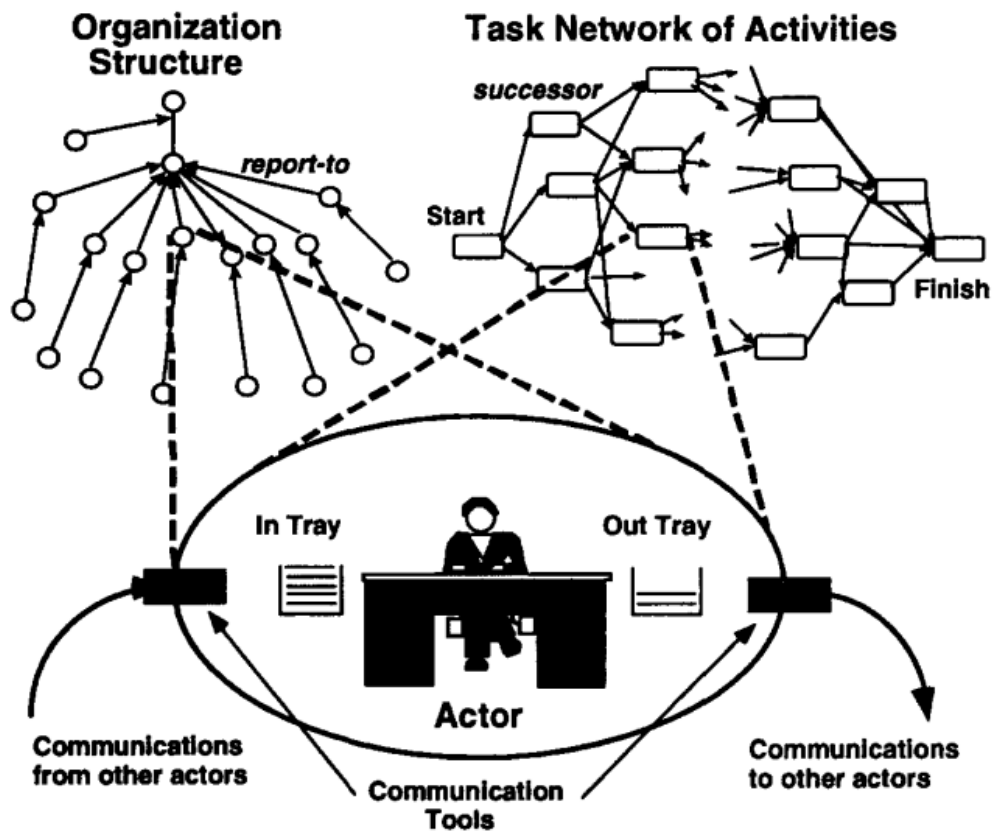


図 2-7 Virtual Design Team モデルの概要[37]

Virtual Design Team は、組織構造を評価するための基本コンセプトとして様々な分野で応用されている。例えば、Horii ら[39]はアメリカと日本の組織構造や個人の振る舞いなどの違いに関して定性的な分析を行い、日本ではアメリカに比べて組織構造の階層が深いことを言及している。さらに、その分析結果をもとにして、アメリカと日本の Virtual Design Team モデルを図 2-8 のように構築し、シミュレーションによる定量的な分析を用いて日本とアメリカの短期型プロジェクト実施に関する特徴や違いを作業量やリスクの観点から分析している。

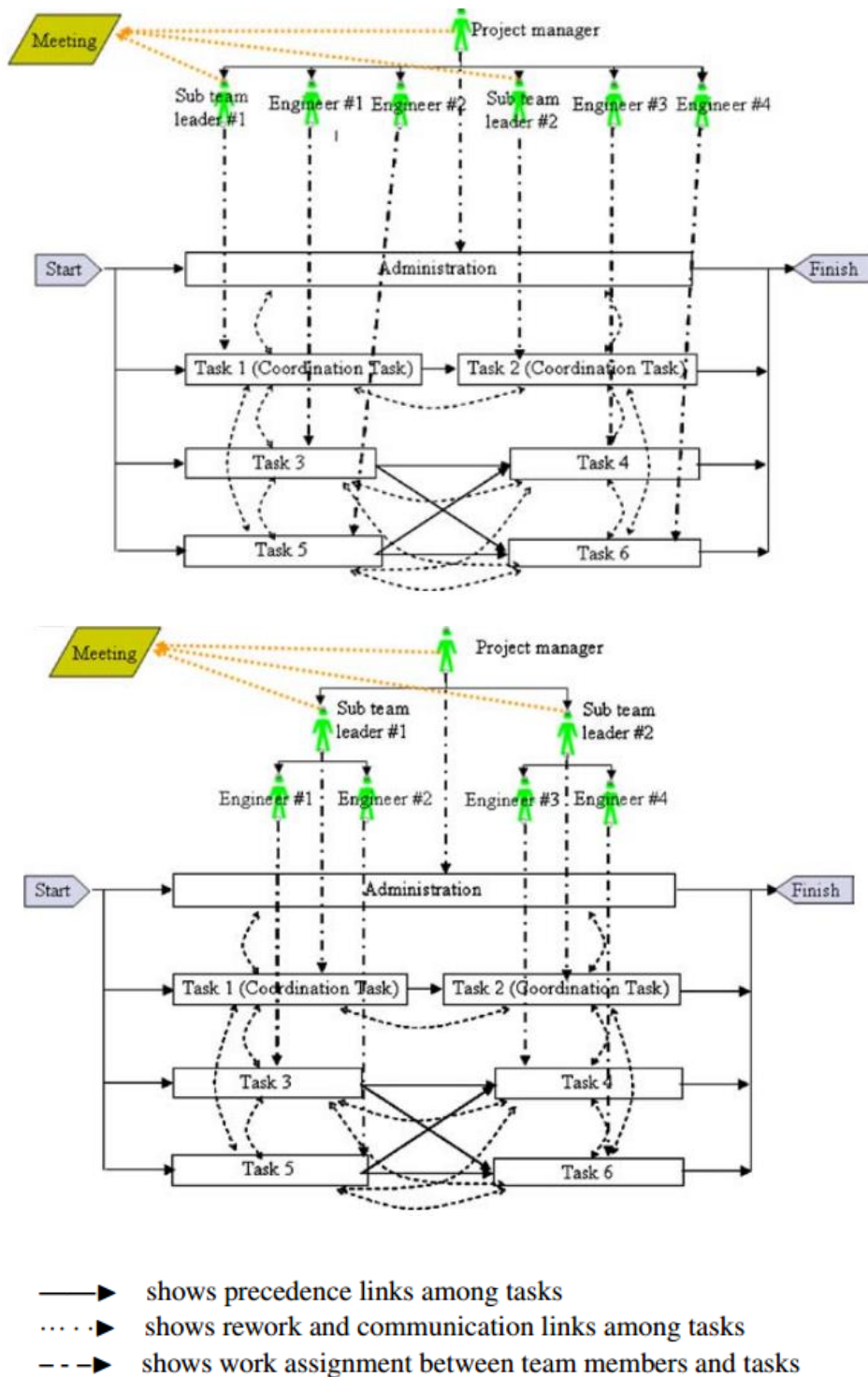


図 2-8 短期型プロジェクトにおけるアメリカと日本の VDT モデル[39]

また, Suzukiら[40]は, Virtual Design Team のコンセプトを実装したシミュレーションツールである Process Management Tool(PMT)を開発し, 金型メーカーなどの製造業を対象にその効果を実証している. PMT のコンセプトとインターフェイスを図 2-9 に示す. PMT では「Client・Process・Organization・Resource」の4つのモデルから構成され, Process モデルを軸として他の3つのモデルとの関わりを詳細に定義することで, 信頼性の高いシミュレーションを行うことが可能である.

近年では, Moser[41]が複数の国の企業間で協力して行われる国際プロジェクトを対象に現実性の高いシミュレーションを行うためのモデル化とプロジェクトデザインの方法論を提案している. Moser は, 独自に開発したシミュレーションツールである TeamPort[42]とプロジェクトデザインの方法論を用いて, 航空宇宙業界や自動車業界などの様々な業種でその効果を立証している. TeamPort のインターフェイスを図 2-10 に示す. TeamPort 上では, 「Product・Activity・Team・Phase」のモデルでプロジェクトを定義する. Team の作業位置を地図上から選択することで, 定義したモデルを用いてシミュレーションを行う際にチーム間の距離や時差・言語の違いなどから発生する障害などを考慮してシミュレーションを行うことが可能である.

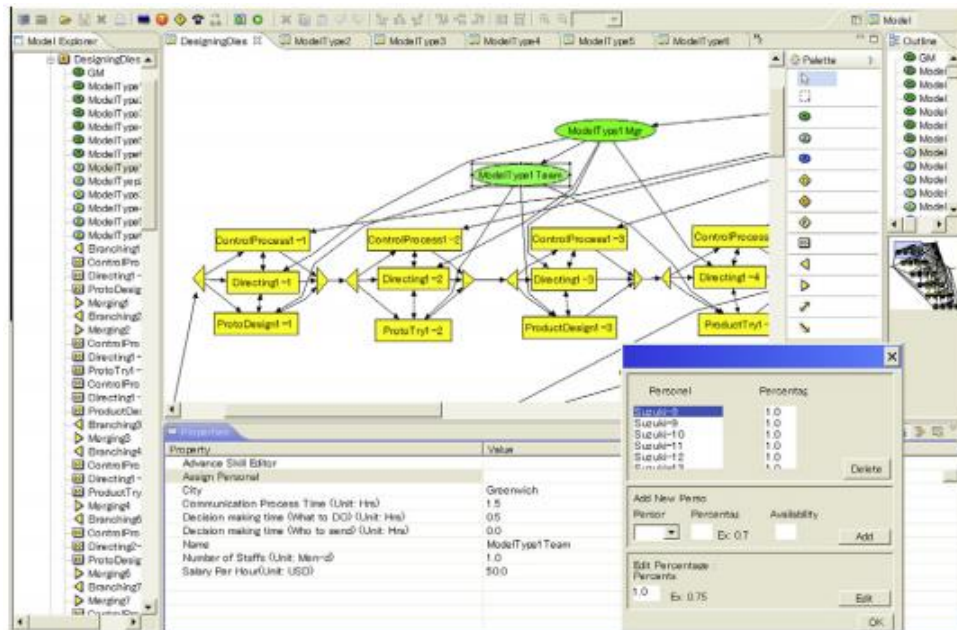
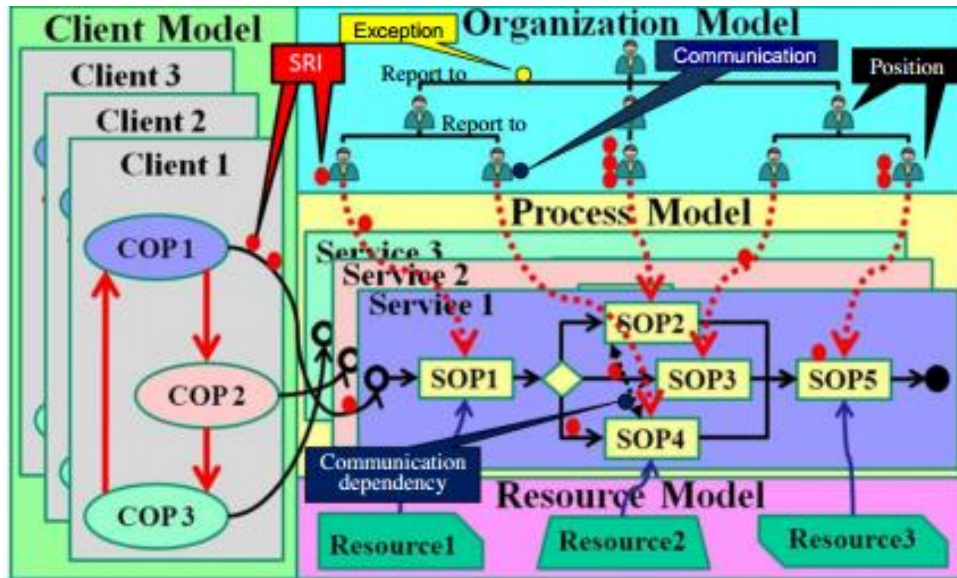


図 2-9 PMT のコンセプトとインターフェイス[40]

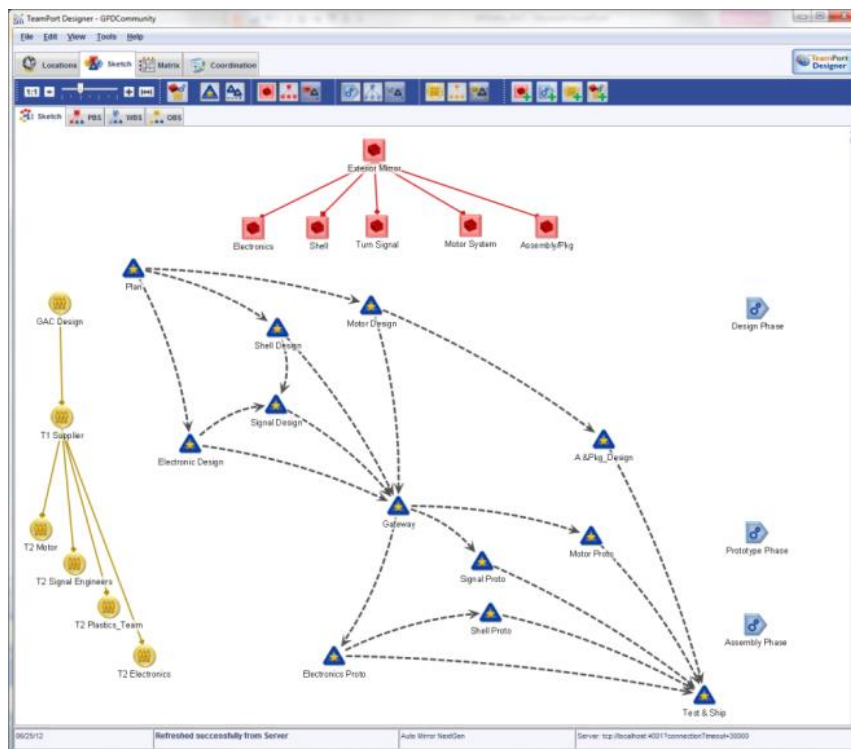
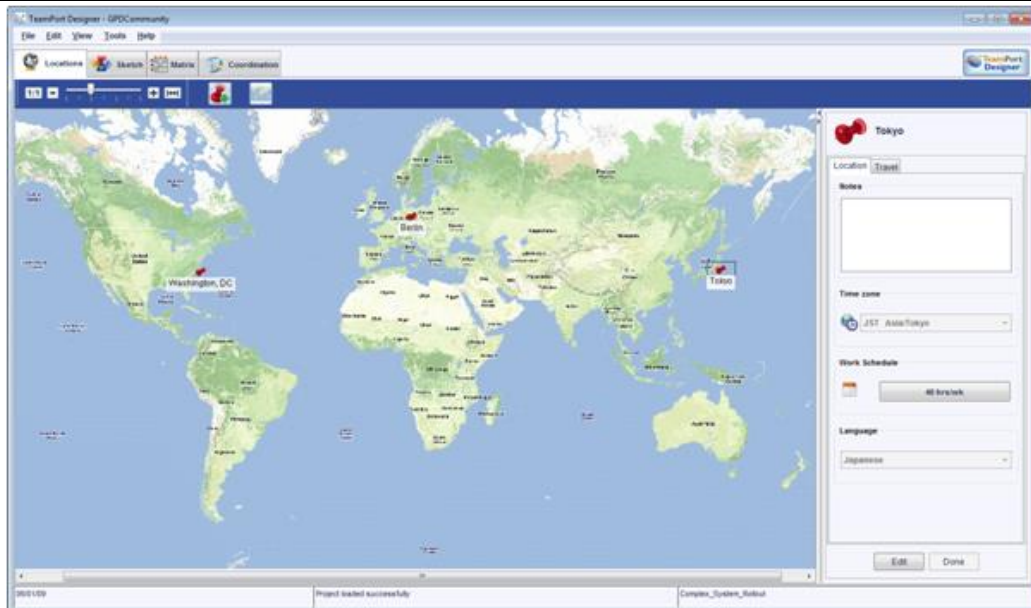


図 2-10 TeamPort のインターフェイス[42]

2.7 本研究の位置づけ

本章では、組織論、チームパフォーマンスやチーム間での関わり合いをデータ分析に基づいて行う研究、スケジューリング計算やシミュレーションによるチームパフォーマンスの評価手法と、組織構造に着目した作業チーム設計の研究事例について概観した。

2.5 や 2.6 で紹介した研究事例では、不確かさを含めたプロジェクトの詳細なシミュレーションモデルを定義し、シミュレーションによって定量的に評価を行うことを繰り返すことで、作業チーム設計を行っており、本研究も同様の手順を踏むことで適切な作業チーム設計を目指す。

これまでに、2.5 や 2.6 で紹介した研究事例のように、作業チームのパフォーマンス評価と作業チーム設計に関する様々な手法が提案されてきた。作業チーム設計の方針については様々なものが考えられるが、以下の 2 点に大別可能であると考える。

- 静的な状況に対して最適化した作業チーム設計
- 動的な状況に対応可能な作業チーム設計

プロセスや作業チームの不確かさを考慮しない静的な状況を想定した作業チーム設計では、対象とするプロジェクトのプロセスを正確に記述することができれば、目的に応じた最適な作業チームの設計は理論上可能である。ただし、作業チームの設計変数は複数存在し、それぞれの設計変数における解空間は広いいため、設計者が適切に作業チームの設計を行うことは非常に難しい。

一方で、プロジェクト実施中のトラブル対応などの、刻一刻と動的に状況が変化する不確かな環境に対応可能な作業チーム設計では、どのような不確かさを想定するかによって作業チーム設計の方針が変わってくる。プロジェクトの進捗が少し滞るようなトラブルなどの不確かさを考慮する際には、静的な状況に対する作業チーム設計の方針にトラブルを考慮する工夫を加えることで対応可能である。しかしながら、プロジェクトの方針が大きく変更するようなトラブルなどの不確かさを考慮する場合には、大きなトラブルに対応するためのシナリオを明確に定義して、シナリオごとにシミュレーションを行うことで作業チームの評価を行う必要がある。

また、作業チームや組織を対象としたシミュレーションでは、人員へのタスクの割り振りや状態の更新を繰り返すことでシミュレーション計算を行うが、タスクの割り振りのルールに関して

はシミュレーションソフトごとにルールベースで不変なものとして定義されているのが一般的である。しかしながら、実際の業務ではタスクの割り振りルールや人員の各プロジェクトへのエフォート率などに代表される作業戦略を、対象とするプロジェクトに応じて変更しながら取り組んでおり、2.5 や 2.6 で紹介した研究事例では作業戦略は作業チーム設計の対象外となっている。

このような観点から、本研究の新規性は以下の 2 点である。

- 現実的な作業チーム設計を行うための、作業戦略(タスク割り振りルールや各プロジェクトへのエフォート率)を含めた作業チーム設計手法の提案
- 設計者の負担を軽減するための、遺伝的アルゴリズムを利用した適切な作業チーム設計支援手法の提案

第3章 提案手法

3.1 はじめに.....	23
3.2 提案する作業チーム設計手法の概要.....	23
3.2.1 プロジェクトの各段階における作業チーム設計の方針.....	23
3.2.2 作業チームの設計変数決定の流れ.....	25
3.3 チーム作業シミュレーション.....	26
3.3.1 概要.....	26
3.3.2 作業情報.....	28
3.3.3 作業チーム.....	30
3.3.4 チーム作業シミュレーションの流れ.....	35
3.3.5 作業チームのパフォーマンスの評価.....	45
3.4 手動探索による作業チーム設計手法.....	47
3.4.1 概要.....	47
3.4.2 作業チーム手動探索の詳細.....	47
3.4.3 探索結果を利用した作業チームの決定と分析.....	50
3.5 遺伝的アルゴリズムによる作業チーム設計支援手法.....	52
3.5.1 概要.....	52
3.5.2 組織構造の自動探索手法.....	53
3.5.3 作業戦略の自動探索手法.....	57
3.5.4 目的関数の設定に応じた探索計算の詳細.....	61
3.5.5 探索結果を利用した作業チームの決定と傾向分析.....	62

3.1 はじめに

本章では, 提案する作業チーム設計の流れと, 作業チーム設計に必要となるチーム作業シミュレーション, 設計の流れの各ステップにおける設計変数を適切に決定するための手法として, 手動探索による作業チーム設計手法と遺伝的アルゴリズムによる作業チーム設計支援手法について説明する.

3.2 提案する作業チーム設計手法の概要

3.2.1 プロジェクトの各段階における作業チーム設計の方針

提案手法では, プロジェクトの各段階において作業チーム設計の方針を変更して対応する. プロジェクトの各段階における作業チーム設計方針の概要を図 3-1 に示す.

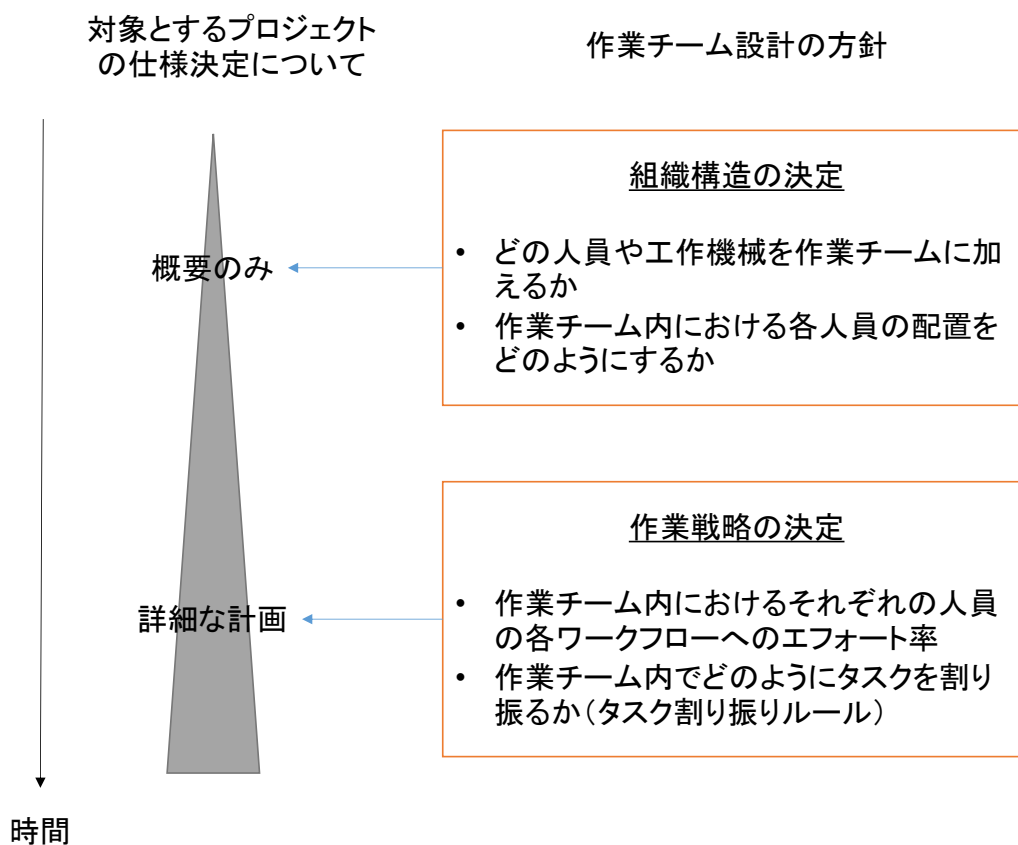


図 3-1 プロジェクトの各段階における作業チーム設計方針の概要

まず、プロジェクトの大まかな概要のみが決定している段階では、そのプロジェクトに対して①どの人員や工作機械を作業チームに加えるかと、②作業チーム内に採用した人員をどのように配置するかによって表される組織構造を決定する。例えば、造船設計プロジェクトにおいて、設計要目の決定する順番のみが決まっており、各設計項目決定に必要とする工数などの詳細な情報は決まっていない段階では、造船設計プロジェクトに誰を採用するかや、採用した人員間の関係をどのようにするかを決定する。

次に、プロジェクトの詳細な仕様が決定している段階では、前段階で決定した組織構造を持つ作業チームが、対象とするプロジェクトに対してどのように取り組むべきかを表す作業戦略を決定する。例えば、造船設計プロジェクトにおいて、設計項目に必要とする目安の工数や詳細な作業手順などが決定した段階では、作業チーム内でどのようにタスクを割り振るか

や、担当するタスクに対して人員がどのように取り組むかなどを決定する。なお、本研究では、作業戦略として以下の2つの項目を決定するものとする。

- 人員の各プロジェクトに対するエフォート率
- 作業チーム内でどのようにタスクを割り振るかを表すタスク割り振りルール

3.2.2 作業チームの設計変数決定の流れ

プロジェクトの各段階において対象とする作業チームの設計変数を決定した後は、設計変数の値を変更することで作業チームの設計案を作成し、作成した作業チームの設計案を評価することを繰り返すことで、適切な作業チームの設計変数を求めることを目指す。

なお、本研究では、図 3-1 の各ステップで初めから組織構造や作業戦略を1つに決定するのではなく、まずは適切な組織構造案や作業戦略案を複数抽出することを想定している。作業チームの設計を行う際には、各ステップでプロジェクトの取り巻く環境が変化することが想定されるが、複数の適切な組織構造案や作業戦略案を保持しておくことで、状況の変化に応じて柔軟に組織構造と作業戦略の適切な組み合わせを選択可能にするためである。

本研究では、作業チーム設計案の良し悪しを評価するために、3.3 のチーム作業シミュレーションを利用することで、どの作業チーム設計案が良いかを定量的に判断する。また、本研究では、3.4 の手動探索における作業チーム設計手法と、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法という2つの手法のいずれかを選択して、作業チーム設計案の作成と評価を繰り返し行う。

3.3 チーム作業シミュレーション

3.3.1 概要

チーム作業シミュレーションの概要を図 3-2 に示す。チーム作業シミュレーションの目的は、作業チーム設計案のパフォーマンスを定量的に評価することである。

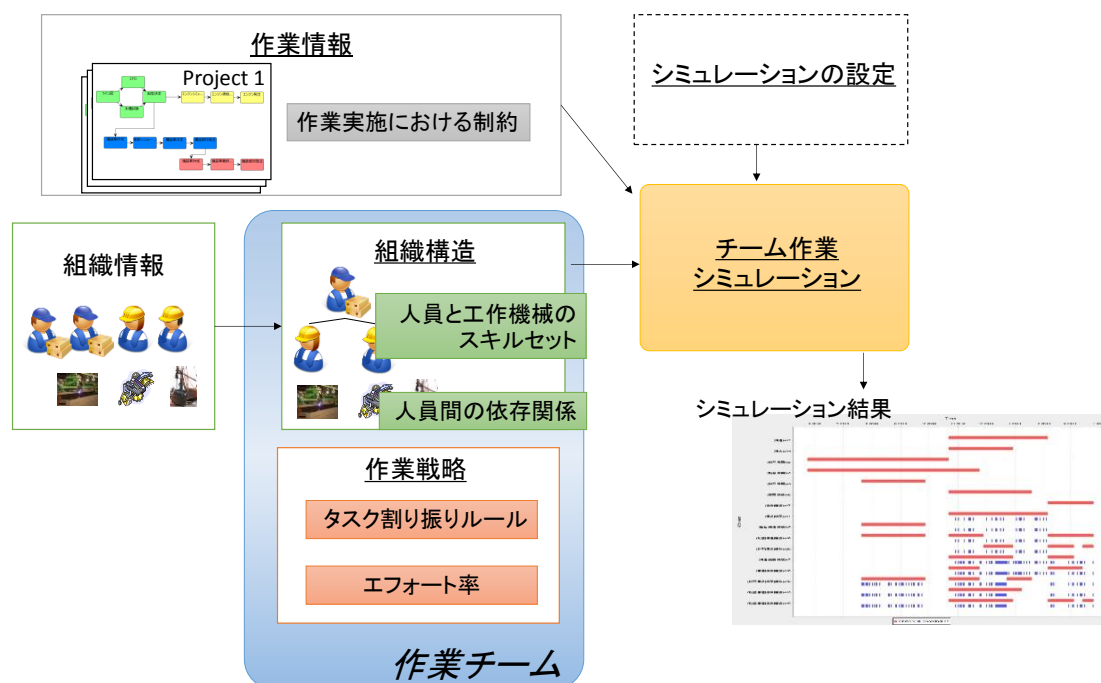


図 3-2 チーム作業シミュレーションの概要

以降では、チーム作業シミュレーションの流れについて説明する。

まず、分析の対象となる作業内容を作業情報として定義する。作業情報は、作業の実施方法を表すワークフロー群と、作業実施における制約からなる。

次に、作業チームの設定を行う。作業チームは、組織構造と作業戦略によって構成される。本研究では、作業チームを設計する際には、組織内の各人員や工作機械に関するスキルとコストの情報が登録されている組織情報をあらかじめ保持しているものとする。保持して

いる組織情報から作業チームの人員を選択し、選択した人員間の依存関係を決定することで作業チームの組織構造を定義する。作業戦略は、タスク割り振りルールと、それぞれの人員の各ワークフローに対するエフォート率によって構成される。具体的には、各タスクを人員や工作機械に割り振る際のルールであるディスパッチングルールの重みベクトルによってタスク割り振りルールを表現し、対象のワークフローが複数存在する場合はそれぞれのワークフローへのエフォート率も各人員に対して定義する。なお、本研究におけるエフォート率とは、複数のタスクを掛け持ちした際における各タスクへの作業量の分散を定義する値のことを指す。

以上の入力情報をもとに、チーム作業シミュレーションを行うことで設計した作業チームのパフォーマンスを評価する。チーム作業シミュレーションを行う際には、分析の対象や目的に応じてあらかじめ工作機械を考慮するかどうかやタスクの掛け持ちを許可するかどうか、トラブルとその調整を考慮するかどうかを設定する。チーム作業シミュレーションは、上記の設定内容と作業戦略に基づいて、①タスクを作業チームの人員と工作機械に割り振り、②各タスクの状態を更新する、という2つの操作をすべてのタスクが終了するまで繰り返す。チーム作業シミュレーションの結果として、作業チームの各人員がどの時間にどのタスクを担当したかを表すガントチャートを出力する。さらに、出力されたガントチャートに基づいて、所要時間と合計コストに着目して設定した作業チームのパフォーマンスを評価する。

3.3.2 作業情報

作業情報は、ワークフロー群と、作業実施における制約の 2 つによって構成される。

3.3.2.1 ワークフロー

ワークフローは作業チームが実施する作業内容を表すものであり、複数のタスク情報とそれらの前後関係の情報によって構成される。1 つのタスクが持つ属性を表 3-1 に示す。

チーム作業シミュレーション前には、タスクの名前や工数などの情報を定義する。タスクを担当するために必要なスキルに関しては、組織情報におけるスキルの名前と対応させて記述することで、タスクを担当可能な人員や工作機械を限定することが可能になる。なお、複数のスキルを必要とするタスクを定義する場合には、タスクの名前の付け方にルールを設けることで対応が可能であるが(例:タスクの名前を【スキル 1|スキル 2】にする)、基本的には複数のスキルを必要とするタスクを考慮せず、可能な限り個別のタスクに分解する方針を取る。

チーム作業シミュレーション中に更新される属性に関しては、単位時間ごとにタスクの状況を更新し、タスクの残り工数が 0 になったらタスクの状態を **Completed** にするなど、必要に応じて更新していく。

表 3-1 タスクの属性

	属性名	説明
チーム作業シミュレーション前に定義する属性	タスク ID	タスクの ID. 同一ワークフロー内に同じ名前のタスクが複数存在する際には, タスク ID によって区別する.
	ワークフローID	タスクが所属するワークフローの ID.
	ワークフロー締切	ワークフローの終了時刻に期限がある場合に設定する
	名前	タスクの名前.
	先行タスク ID	このタスクを実行するために終了しておく必要のあるタスクの ID. 複数のタスク ID を登録可能.
	工数	タスクを終了ための作業量を表す.
	最大同時担当者数	同時刻にタスクを担当できる人員の最大数.
	トラブル率	単位時間あたりにトラブルが発生する確率.
	担当部署	タスクを担当する部署が決まっているときなどは, あらかじめ担当部署を登録しておく.
チーム作業シミュレーション中に更新される属性	残り工数	現在のタスクが終了するための残り工数を表す.
	担当者	時刻ごとの, 担当した人員と工作機械の情報を表す.
	状態	None : 作業不可 Ready : 作業待機 OnWork : 作業中 Completed : 作業終了
	最早開始時刻	ワークフローが順調に進んだ場合における作業を最も早く着手できる時刻. 時刻ごとに PERT によって計算し, 更新する.
	最遅開始時刻	ワークフローを順調に終えるために, 遅くてもこの時間までには作業を開始しなければならない時刻. 時刻ごとに PERT によって計算し, 更新する.
	開始時刻	タスクの状態が OnWork になった時刻を表す.
	終了時刻	タスクの状態が Completed になった時刻を表す.

3.3.2.2 作業実施における制約

一般に、作業実施において考慮すべき制約は対象とする業務や対象とする現場によって異なるが、ワークフローや作業チームに依存しない範囲の制約は、内的環境による制約と外的環境による制約に大別可能である。

内的環境による制約の例としては、作業場所や作業面積による制約が挙げられる。例えば、造船所などの大きな部材を取り扱う業務では、工場の面積の関係で大きな部材を複数個同時に製作することができない状況が想定される。そこで、本論文では生産現場における部材の置き場所の面積を考慮して、「同時に実行可能なワークフローの最大数」を制約として取り入れている。また、作業場所による制限として作業場所内での移動も考えられるが、本論文では小規模の作業場所において1時間~1日を単位時間としてチーム作業シミュレーションによる分析を行うため、移動に関しては考慮していない。

外的環境による制約の例としては、法律による制約が挙げられる。例えば、経済産業省は2011年に発生した東日本大震災の影響で、夏期の電力需要が高い時期に前年比で使用最大電力の15%削減目標を工場に対して設定している。そこで、本論文では生産現場における電力ピークカットを考慮するために、「単位時間あたりに使用可能な電力使用量の最大値」を制約として取り入れている。

なお、上記に挙げた作業実施における制約の具体例は、本論文のケーススタディで必要なために導入した制約であり、他の業種を対象とする際には、対象の業種の性質に応じて臨機応変に変更する必要がある。

3.3.3 作業チーム

次に、設計対象である作業チームについて説明する。作業チームは、組織情報をもとに作成される組織構造と作業戦略によって構成される。

3.3.3.1 組織情報

作業チームに加える人員と工作機械を選択する際には、組織内の各人員や工作機械に関するスキルとコストの情報が登録されている組織情報をあらかじめ保持しているものとする。組織情報の例を表 3-1 に示す。通常の企業や組織では、人員と工作機械は独立して管理さ

れているが、本論文では説明を簡略化するために同時に管理することを想定する。

組織情報に登録されている人員と工作機械は、種別・コスト・スキル・電力の4つの属性で定義される。種別に関しては、1が人員を、-1が工作機械を表す。コストに関しては、単位時間当たりの固定給と変動給を設定することで、正規社員と非正規社員の区別や工作機械の減価償却などの考慮が可能になる。スキルに関しては、3.3.4のチーム作業シミュレーション上でタスクの掛け持ちがない場合における単位時間あたりの作業進捗量を表す。スキルの値とトラブル率によって、熟練者と非熟練者を区別することが可能である。また、種別が-1、すなわち工作機械の場合には、工作機械が使用された時の単位時間当たりの電力使用量を別途定義する。

表 3-1 組織情報の例

ID	種別	コスト [円/Time]		スキル			トラブル率	電力[kW/Time]
		固定給	変動給	Skill A	Skill B	Skill C		
W1	1	2500	0	1	1	0	0.1	-
W2	1	0	1800	0	0.5	1	0.3	-
F1	-1	700	150	1	0	1	0.1	80
F2	-1	250	1000	0	1	0	0.1	120

3.3.3.2 組織構造

本研究では、組織情報から人員と工作機械を抽出し、依存関係を決定することで組織構造を定義する。具体的な設計変数は以下の2つである。

- 作業チームに加える人員と工作機械
- 作業チーム内での人員間の依存関係

作業チームに加える人員と工作機械に関しては、組織情報から実際に作業を行う人員と工作機械を選択することで定義する。なお、工作機械をチーム作業シミュレーションで考慮しない場合には、工作機械を選択する必要はない。

作業チーム内での人員間の依存関係に関しては、作業チーム内の人員に関して上司

や部下といった関係を定義する。チームで何らかのプロジェクトを行う際には、単にチームの人間が個別に与えられた仕事を行うだけでなく、トラブルなどが起きた際には必要に応じて組織構造の中で上司や部下との調整やコミュニケーションを行いながら作業を進めていく。本研究では、作業チームの人員間の依存関係を含めた組織構造の概念を、先行研究[36][37][38][39][40][41]の考え方をもとにして、図 3-3 と表 3-2 のようにモデル化した。なお、本研究では遅延が発生する程度の小さなトラブルのみを、チーム作業シミュレーションで扱うものとする。

作業チーム内の各人員は、いずれかの部署に所属する。まず、組織構造内の末端の部署を定義する。末端の部署には各タスクに関して権限を持つか否かの情報が、各タスクの所属部署属性に記載されている情報に応じて自動的に定義される。図 3-3 では、「部署 1」は「Task 1」に対する権限を、「部署 2」は「Task 2」に対する権限を、「部署 3」は「Task 3」に対する権限を持つものとする。次に、末端部以外の部署を定義する。末端部以外の部署は必ずいずれかの部署と依存関係を持つものとし、一つ下の部署の権限を引き継ぐことが可能である。例としては、「部署 1|2」に属する人員は Task 3 に対するスキルは保持しているが、Task 3 に対する権限を持っていないため担当することはできない。表 3-2 では権限によって限定されるスキルを黒塗りで表現している。

また、図 3-3 において、部署 2 の人員が Task 2 を担当中のある時間でトラブルが発生した場合、その時間だけ直属の親の部署に所属する上司に「調整(Task 2)」という調整タスクが発生する。ただし、「調整(Task 2)」を担当できるのは Task 2 のスキルを持つ上司に限定される。また、図 3-3 において Task 2 を担当している「部署 1|2」の人員や Task 3 を担当している「部署 3」の人員のように上司がいない状態でトラブルが発生した場合には、その時間での彼らの作業進捗が 0 となる。なお、工作機械に関しては工作機械間での調整などが行われることはないため、依存関係の定義は行わず、また、担当部署に関係なくスキルを持つすべてのタスクを担当可能であるものとする。

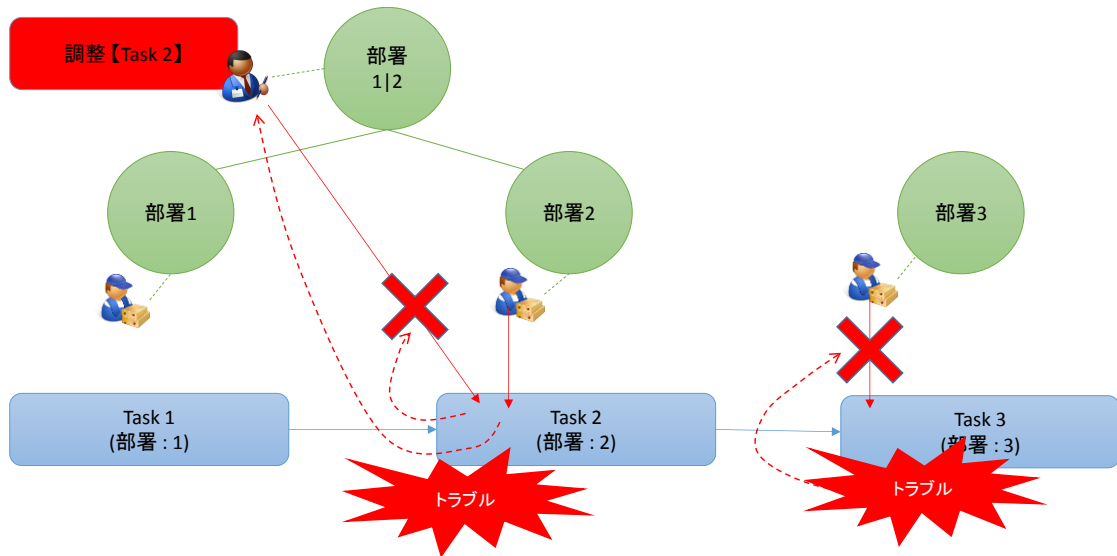


図 3-3 本研究における組織構造のモデル化

表 3-2 組織構造によって定義された権限によってスキルが限定された例

名前	所属部署	Task 1	Task 2	Task 3
A	部署 1 2	1	1	1
B	部署 1	1	0	1
C	部署 2	1	1	0
D	部署 3	0	0	1

3.3.3.3 作業戦略

作業戦略は、作業チームが作業を行う際の人員へのタスクの配分と、どのように作業に取り組むかを定義するものである。具体的な設計変数は以下の2つである。

- タスク割り振りルール
- 各ワークフローへのエフォート率

本研究では、ディスパッチングルール[43]の重みベクトル w によってタスク割り振りルールを表現する。本研究で用いたディスパッチングルールを表 3-3 に示す。本研究では合計9つのディスパッチングルールを採用し、タスク割り振りルール w は式(3.1)で示すように9つのディスパッチングルールの重みベクトルによって表現される。

人員の各ワークフローへのエフォート率に関しては、対象のチーム作業で複数のワークフローを扱う、かつチーム作業シミュレーションで同時に複数の作業を担当可能とする設定の場合において、選択した作業チームの人員ごとに独立して定義する。エフォート率は、チーム作業シミュレーション上である人員が複数のタスクを同時に担当する場合におけるそれぞれのタスクに対する作業量を定義するための変数である。なお、工作機械に関しては、同時に複数のタスクを担当することは考慮しないため、各ワークフローへのエフォート率を定義しない。

表 3-3 本研究で採用したディスパッチングルール

ルール名	変数	説明
EDD	w_{EDD}	Earliest Due Date: 納期に最も近いタスクを優先する.
FIFO	w_{FIFO}	First In First Out: 先行作業がすべて完了し, 作業開始可能になった時間の早いタスクを優先する.
SPT	w_{SPT}	Shortest Processing Time: 工数が最小のタスクを優先する.
EST	w_{EST}	Earliest Start Time: 最早開始時間と最遅開始時間の平均が早いタスクを優先する.
TSLACK	w_{TSLACK}	Total SLACK time: クリティカルパス上に近いタスクを優先する.
SPN	w_{SPN}	Skilled Person Number: その作業に対する能力を持つ作業者の人数が少ないタスクを優先する.
SSP	w_{SSP}	Sum Skill Point: 持っている能力の少ない作業員・工作機械を優先する.
VC	w_{VC}	Variable Cost: 単位時間当たりコストの小さい作業員・工作機械を優先する.
EC	w_{EC}	Electric power Consumption: 単位時間あたりの消費電力が小さい工作機械を優先する.

$$w = \{w_{EDD}, w_{FIFO}, w_{SPT}, w_{EST}, w_{TSLACK}, w_{SPN}, w_{SSP}, w_{VC}, w_{EC}\} \quad (3.1)$$

3.3.4 チーム作業シミュレーションの流れ

作業情報と作業チーム情報を入力として, チーム作業シミュレーションを行いチーム作業の結果を示すガントチャートを出力する. チーム作業シミュレーションの説明に使用する記号を表 3-4 に示す. 本研究は, 資源制約付きプロジェクトスケジューリング問題 (RCPSP : Resource-Constrained Project Scheduling Problem) [32][33][34]に基づいたチーム作業のモデル化を行い, タスクの割り振りと状態の更新を繰り返しながらチーム作業のシミュレーションを行う手法を説明する.

表 3-4 チーム作業シミュレーションの説明で使用する記号

変数	説明
t	チーム作業シミュレーション計算内での時刻 ($t \in \mathbf{T}$)
w	ワークフローID ($w \in \mathbf{W}$)
j	タスク ID ($j \in \mathbf{J}$)
m	人員 ID ($m \in \mathbf{M}$)
f	工作機械 ID ($f \in \mathbf{F}$)
h_j	タスク j に対する任意の先行タスク ID ($h_j \in \mathbf{H}_j$)
$d_j(t)$	時刻 t におけるタスク j の残り工数
$MaxNumP_j$	タスク j の最大同時担当者数
$x_j(t)$	タスク j が時刻 t に終了したかどうかを表す変数
$Limit_j$	タスク j が所属するワークフローの締め切り時刻
tR_j	タスク j のトラブル率
$ES_j(t), LS_j(t)$	時刻 t におけるタスク j の最早開始時刻, 最遅開始時刻
cc_m, vc_m	人員 m の固定給, 変動給
$rM_{j,m}$	タスク j における人員 m のスキル値
$eR_{m,w}$	人員 m のワークフロー w に対するエフォート率
tR_m	人員 m のトラブル率
cc_f, vc_f	工作機械 f の固定給, 変動給
$rF_{j,f}$	タスク j における工作機械 f のスキル値
tR_f	工作機械 f のトラブル率
ec_f	工作機械 f を使用した際に消費される単位時間当たりの電力量
$p_{j,m}(t)$	時刻 t における人員 w がタスク j のみを担当した場合の作業進捗量
$pM_{j,m}(t)$	時刻 t に人員 m がタスク j を担当しているかどうかを表す変数
$pW_{w,m}(t)$	時刻 t に人員 m がワークフロー w に属するタスクを担当しているかどうかを表す変数
$pF_{j,f}(t)$	時刻 t に工作機械 f がタスク j を担当しているかどうかを表す変数
$MaxNumW$	単位時間において同時に実行可能なワークフローの最大数
$MaxE$	単位時間あたりに使用可能な電力の最大値

3.3.4.1 チーム作業シミュレーションの設定

対象のプロジェクトや業務の属性によってチーム作業の取り組み方は異なる。本研究では、以下の 3 項目に関して考慮するか否かをチーム作業シミュレーションを行う前に設定する。以下の 3 項目の設定は、チーム作業シミュレーションのフローにおける各ステップでの操作に用いる。

- 各人員の同時刻におけるタスクの掛け持ち
- タスク実行における工作機械の考慮
- 各タスクでランダムに発生するトラブルとその調整の考慮

3.3.4.2 チーム作業シミュレーションで扱う問題の定式化

チーム作業シミュレーションで扱う問題の定式化について説明する。本節で定式化した条件下において、チーム内でのタスクの割り振りと調整・実行を繰り返すことになる。

対象とする作業情報を定義した式を以下に示す。式(3.2)はタスクの終了時刻は一意に定まることを示している。また、式(3.3)はタスク間の前後関係というワークフローの制約を、式(3.4)は各タスクにおける同時刻の最大担当者数の制約を示している。

また、式(3.5)は同時実行ワークフロー数に関する制限を、式(3.6)は単位時間あたりに使用可能な電力使用量に関する制限を示している。

$$\sum_t x_j(t) = 1 \quad (x_j(t) \in \{0,1\}) \quad (3.2)$$

$$\sum_t t \cdot x_h(t) \leq \sum_t (t - d_j(0)) \cdot x_j(t) \quad (3.3)$$

$$\sum_m pM_{j,m}(t) \leq \text{MaxNumP}_j \quad (pM_{j,m}(t) \in \{0,1\}) \quad (3.4)$$

$$\sum_w \sigma \left(\sum_m pW_{w,m}(t) \right) \leq \text{MaxNumW} \quad (pW_{w,m}(t) \in \{0,1\}, \sigma(a) = \begin{cases} 0 & (a = 0) \\ 1 & (a \neq 0) \end{cases}) \quad (3.5)$$

$$\sum_f pF_{j,f}(t) \cdot ec_f \leq \text{MaxE} \quad (3.6)$$

次に、設計対象となる作業チームの制約について定義した式を以下に示す。

式(3.7)と式(3.8)は, それぞれの人員・工作機械のスキル値は 0 から 1 までの実数で定義されることを示している.

$$rM_{j,m} = [0,1] \quad (3.7)$$

$$rF_{j,f} = [0,1] \quad (3.8)$$

タスク割り振りに関しては, 人員のタスク掛け持ちを許可しない場合は, 式(3.9)と式(3.10)によって同時刻にそれぞれの人員または工作機械が 2 つ以上のタスクを担当してはいけない制約を導入する. 一方で, 人員のタスク掛け持ちを許可する場合には式(3.10)のみを用いて, 工作機械のみ 2 つ以上のタスクを担当してはいけない制約を導入する.

$$\sum_j pM_{j,m}(t) \leq 1 \quad (3.9)$$

$$\sum_f pF_{j,f}(t) \leq 1 \quad (3.10)$$

上記で導入した制約を守りながらタスクの割り振りをを行い, 割り振られた担当タスクに応じてそれぞれの人員と工作機械が単位時間ごとにタスクの残り工数を減らす, という 2 つの操作をチーム作業シミュレーションでは繰り返し行う.

3.3.4.3 チーム作業シミュレーションのフロー

チーム作業シミュレーションのフローを図 3-4 に示す。

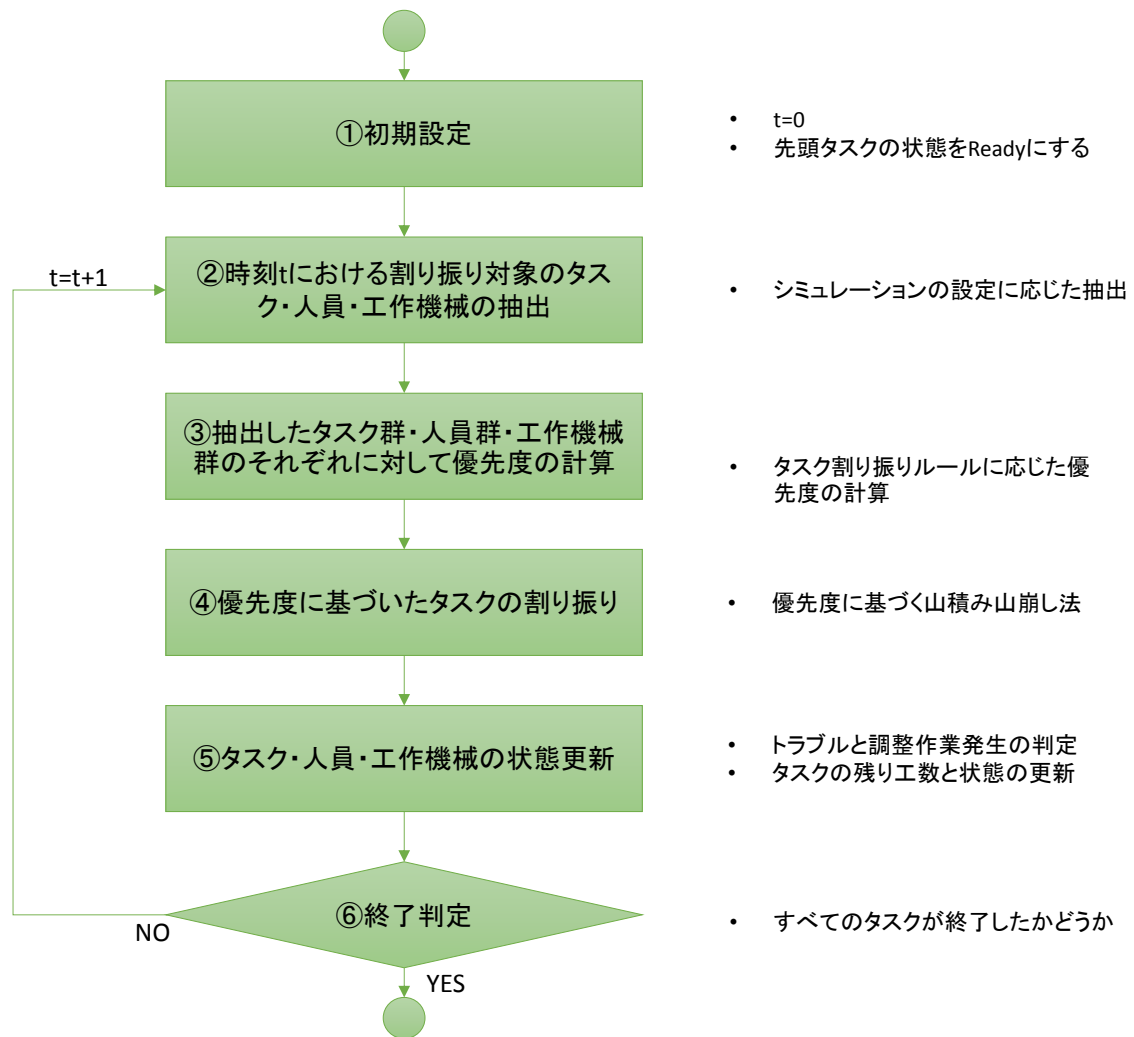


図 3-4 チーム作業シミュレーションのフロー

① 初期設定

初期設定として、チーム作業シミュレーションにおける現在の時刻 t を $t=0$ とし、対象のワークフロー群における先頭タスクの状態を Ready、それ以外のタスクの状態を None に設定する。各ワークフローにおける先頭タスクかどうかの判断方法としては、タスクの属性「先行タスク ID」に何も登録されていないタスクを先頭タスクと判断する。

② 割り振り対象のタスク・人員・工作機械の抽出

時刻 t において割り振り対象となるタスク群・人員群・工作機械群を抽出する。割り振り対象となるタスク群に関しては、「タスクの状態が **Ready**」または「タスクの状態が **OnWork** かつ時刻 $t-1$ における担当者の数が最大同時担当者数を下回っている」のいずれかを満たすタスクを抽出する。また、タスク割り振りの対象となる人員群・工作機械群の抽出方法に関しては、チーム作業シミュレーションの設定によって異なる。人員のタスク掛け持ちを禁止する場合にはこの時点でタスクを担当していない人員を、人員のタスク掛け持ちを許可する場合にはすべての人員を割り振り対象の人員群として抽出する。また、工作機械に関しては、タスク実行において工作機械を考慮する場合にはこの時点でタスクを担当していない工作機械を抽出する。また、タスク実行において工作機械を考慮しない場合には割り振り対象の工作機械を抽出せず、以降の操作では人員に関してのみを対象とする。

③ 抽出したタスク群・人員群・工作機械群のそれぞれに対して優先度の計算

抽出したタスク群・人員群・工作機械群のそれぞれに対してタスク割り振りルール \mathbf{w} に応じて優先度 $\mathbf{F}(t)$ を計算する(式(3.11))。抽出したタスク群の優先度は式(3.12)、人員群の優先度は式(3.13)、工作機械群の優先度は式(3.14)を用いてそれぞれ計算を行う。

$$\mathbf{F}(t) = \{f_j(t), f_m(t), f_f(t)\} \quad (3.11)$$

$$f_j(t) = w_{EDD} \cdot EDD_j(t) + w_{FIFO} \cdot FIFO_j(t) + w_{SPT} \cdot SPT_j(t) + w_{EST} \cdot EST_j(t) + w_{TSLACK} \cdot TSLACK_j(t) + w_{SPN} \cdot SPN_j(t) \quad (3.12)$$

$$f_m(t) = w_{SSP} \cdot SSP_m(t) + w_{VC} \cdot VC_m(t) \quad (3.13)$$

$$f_f(t) = w_{SSP} \cdot SSP_f(t) + w_{VC} \cdot VC_f(t) + w_{EC} \cdot EC_f(t) \quad (3.14)$$

式(3.12)・式(3.13)・式(3.14)では、3.3.3.3で説明した各ディスパッチングルールに基づいて算出される値とタスク割り振りルール(各ディスパッチングルールの重みベクトル)によって算出される。各ディスパッチングルールに基づいて算出される値は以下の式によって算出される。

$$EDD_j(t) = Limit_j \quad (3.15)$$

$$FIFO_j(t) = \max(t \mid x_{h_j}(t) = 1) \quad (3.16)$$

$$SPT_j(t) = d_j(t) \quad (3.17)$$

$$EST_j(t) = ES_j + LS_j \quad (3.18)$$

$$TSLACK_j(t) = ES_j - LS_j \quad (3.19)$$

$$SPN_j(t) = \sum_m rM_{j,m} \quad (3.20)$$

$$SSP_m(t) = \sum_j rM_{j,m} \quad (3.21)$$

$$VC_m(t) = VC_m \quad (3.22)$$

$$SSP_f(t) = \sum_j rF_{j,f} \quad (3.23)$$

$$VC_f(t) = VC_f \quad (3.24)$$

$$EC_f(t) = EC_f \quad (3.25)$$

④ 優先度に基づいたタスクの割り振り

式(3.11)で計算した優先度に基づいて、3.3.4.2 のタスク割り振りの制約の中でタスクと人員・工作機械の山積み・山崩し法によって割り振りを行う。タスク割り振りの流れを図 3-5 に示す。図 3-5 では、あるタスクに対してどの人員と工作機械を割り当てるかを、優先度に基づいて決定するという操作を繰り返し行っている。ただし、チーム作業シミュレーションの設定で工作機械を考慮しないとした場合は、図 3-5 の工作機械の部分を見捨てる、単に人員との関係だけでタスクの割り振りを決定する。

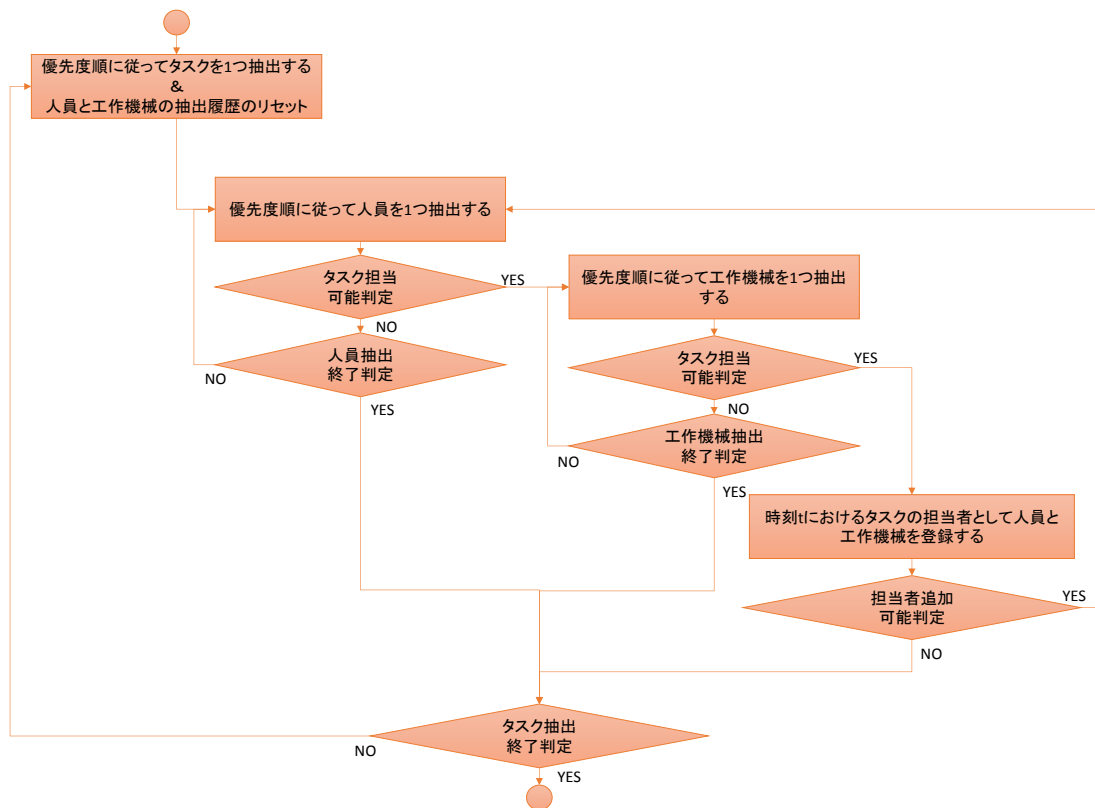


図 3-5 優先度に基づいたタスク割り振りの流れ

人員・工作機械がタスクを担当可能かどうかは、以下の3項目のすべてを満たす場合に可能であると判断する。

- タスクを担当した場合に、設定したタスクの掛け持ちに関する制約を破らない (式(3.9)と式(3.10))
- 人員の担当するタスクの所属するワークフローへのエフォート率が0よりも大きい (式(3.26))
- 人員・工作機械のタスクに対するスキル値が0よりも大きい (式(3.27)と式(3.28))

$$eR_{m,w(j)} > 0 \quad (3.26)$$

$$rM_{j,m} > 0 \quad (3.27)$$

$$rF_{j,f} > 0 \quad (3.28)$$

⑤ タスク・人員・工作機械の状態更新

割り振りの結果に従ってタスクの残り工数を減らし、必要に応じてトラブル発生判定とタスク・人員・工作機械の状態を更新する。

まず、新たに担当者が割り振られたタスクの状態が **Ready** である場合は、タスクの状態を **OnWork** に変更する。

この時点で、チーム作業シミュレーション上でトラブルを考慮すると設定した場合のみ、タスクの状態が **OnWork** であるすべてのタスクに対してトラブル発生の判定を行う。トラブル発生の判定は、ランダム変数である $rand_j(t)$ の値を更新した後に、タスク j に関して時刻 t における担当者 m と工作機械 f のすべての組み合わせのうち、少なくとも一つの組み合わせが式(3.29)を満たす場合にトラブルが発生したと判断する。ただし、チーム作業シミュレーションで工作機械を考慮しない場合は式(3.30)によって判定する。

$$rand_j(t) \geq (1 - tR_j)(1 - tR_m)(1 - tR_f) \quad (3.29)$$

$$rand_j(t) \geq (1 - tR_j)(1 - tR_m) \quad (3.30)$$

あるタスクに対してトラブルが発生したと判断された場合、担当している人員のそれぞれで個別のトラブルが発生したものとし、個別のトラブルに対する調整を作業チーム内で行える状況かどうかのチェックを行う。具体的には、あるタスク j を担当している人員 m が個別に持つトラブルに対して、以下の条件を満たす人員が存在する場合にそのトラブルに対して調整を行うことができる状況であると判断する。

- 担当者の所属する部署と直接の依存関係を持つ上部の部署に所属している
- そのタスクに対するスキルを保持している

該当する人員が存在しない場合は、トラブル対応が不可能であるため、その時刻において人員 m の担当しているすべてのタスクの作業進捗量をゼロとする。一方、該当する人員が存在する場合、該当する人員すべてに調整タスクが割り振られる。なお、調整タスクはどのような状況においても単位時間内に終了できるものとする。

次に、タスクの残り工数を更新する。時刻 t において人員 m が工作機械 f を使ってタスク j のみを行う場合の作業進捗量は式(3.31)で表される。一方で、チーム作業シミュレー

シヨンの設定で工作機械を考慮しない場合の作業進捗量は式(3.32)で表される.

$$p_{j,m}(t) = (pM_{j,m}(t) \cdot rM_{j,m}) \cdot (pF_{j,f}(t) \cdot rF_{j,f}) \quad (3.31)$$

$$p_{j,m}(t) = pM_{j,m}(t) \cdot rM_{j,m} \quad (3.32)$$

人員のタスク掛け持ちを許可しない場合は式(3.33), タスク掛け持ちを許可する場合は式(3.34)によって各タスクの残り工数を更新していく. 式(3.34)では, 複数のタスクを掛け持つそれぞれの人員が, エフォート率に従って単位時間内での作業量を分担しつつそれぞれのタスクに取り組むことを示している.

$$d_j(t) = d_j(t-1) - \sum_m p_{j,m}(t) \quad (3.33)$$

$$d_j(t) = d_j(t-1) - \sum_m \left(\frac{eR_{m,w}}{\sum_w pW_{m,w} \cdot eR_{m,w}} \cdot \frac{1}{\sum_{j \in w} pM_{j,m}(t)} \cdot p_{j,m}(t) \right) \quad (3.34)$$

最後に, タスク・人員・工作機械の状態を更新する. 具体的には, 以下の 2 つの操作を該当するすべてのタスクに関して行うことで, ワークフローの状態を更新する.

- この時点でタスクの状態が **OnWork** かつタスクの残り工数が 0 以下のすべてのタスクに対して, タスクの状態を **Completed** に変更する.
- 上の操作の後, タスクの状態が **None** かつすべての先行タスクが **Completed** のタスクに対して, タスクの状態を **Ready** に変更する.

⑥ 終了判定

上記の操作が終わった後, すべてのタスクの状態が **Completed** であるかどうかをチェックし, 1 つでも異なる場合は $t = t + 1$ として②の割り振り対象のタスク・人員・工作機械の抽出に戻る. 一方, すべてのタスクの状態が **Completed** である場合には, この時点でチーム作業シミュレーションを終了する.

3.3.5 作業チームのパフォーマンスの評価

チーム作業シミュレーション結果は、各時刻における人員・工作機械の担当タスクに関する情報がガントチャートの形式で図 3-6 のように出力される。図 3-6 では、縦軸が人員を、横軸が時間を表し、各人員がどの時間にタスクを担当していたかを赤色で、どの時間に調整作業を行っていたかを青色で表している。また、同時に各人員や工作機械が費やしたコストについても計算される。図 3-6 の枠内である人員における固定給と変動給が発生する時間帯を表示している。本研究では、初めてタスクや調整作業を担当した時間から最後にタスクや調整作業を担当した時間の間、タスクを担当していない時間帯でも固定給は発生しており、変動給はタスクを担当していない時間帯では発生していないものとする。

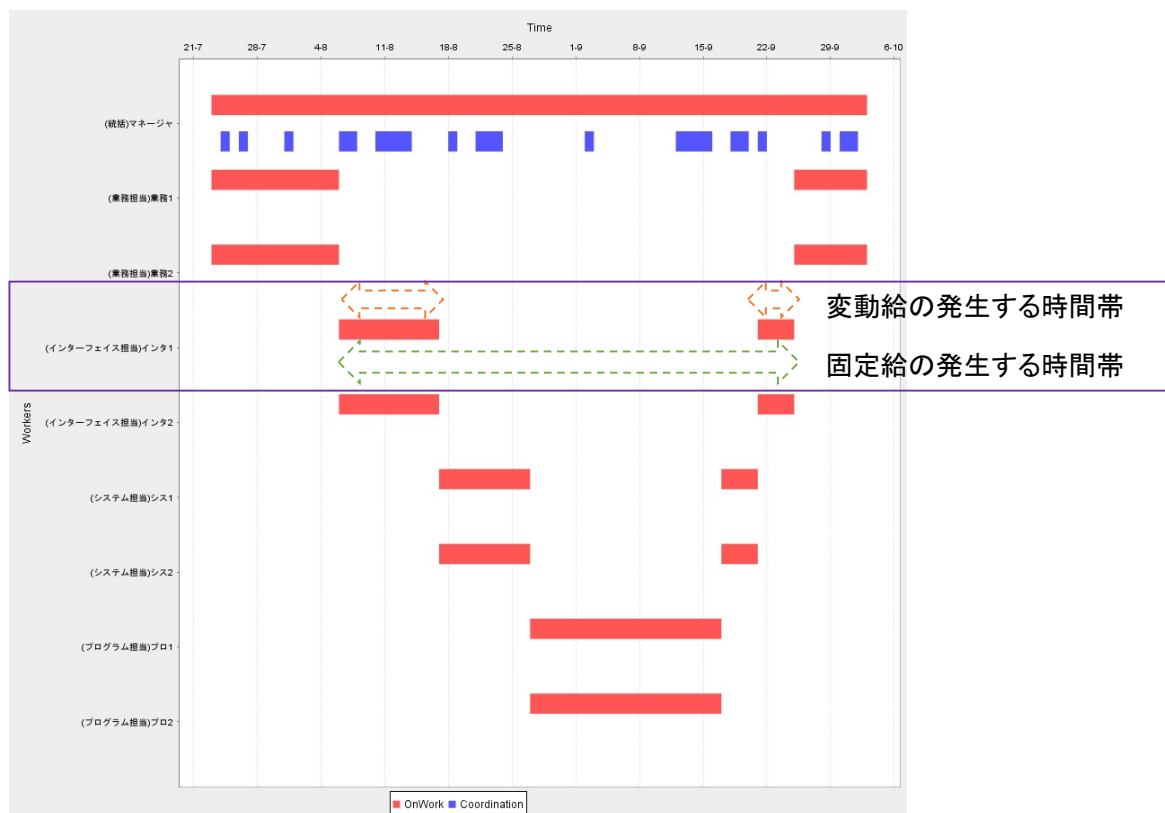


図 3-6 チーム作業シミュレーション結果の出力例

チーム作業シミュレーションの結果をもとに、対象とする作業情報に対する作業チーム

のパフォーマンスを評価する. 本研究では, チームパフォーマンスの評価指標として以下の2つを採用した.

- 所要時間 T
- 合計コスト $TotalCost$

所要時間に関しては, チーム作業シミュレーション終了時における t が所要時間となる. すなわち, すべてのワークフローを終了するためにかかった時間のことを指す.

合計コストは, すべてのワークフローを終了するために必要とした費用の合計を指す. チーム作業シミュレーションにおいて, それぞれの人員と工作機械に費やした個別の費用は式(3.35)と式(3.36)によって計算され, 結果として合計コストは式(3.38)で定義することが可能である. 式(3.35)と式(3.36)では, 初めてタスクを担当した時間から最後にタスクを担当した時間の間, タスクを担当していない時間帯でも固定給は発生しており, 変動給はタスクを担当していない時間帯では発生していないとして計算を行っている.

$$totalCost_m = \sum_{t=startT_m}^{finishT_m} cc_m + \sum_{t=startT_m}^{finishT_m} E \left(\sum_j pM_{j,m}(t) \right) \cdot vc_m \quad (3.35)$$

$$totalCost_f = \sum_{t=startT_f}^{finishT_f} cc_f + \sum_{t=startT_f}^{finishT_f} E \left(\sum_j pM_{j,f}(t) \right) \cdot vc_f \quad (3.36)$$

$$E(x) = \begin{cases} 0 & (x \leq 0) \\ 1 & (x > 0) \end{cases} \quad (3.37)$$

$$TotalCost = \sum_m totalCost_m + \sum_f totalCost_f \quad (3.38)$$

3.4 手動探索による作業チーム設計手法

3.4.1 概要

本節で提案する手動探索による作業チーム設計手法の概要を図 3-7 に示す。プロジェクトの段階に応じて設計対象として指定した作業チームの設計変数を変更しながら作業チームの設計案を作成し、チーム作業シミュレーションによって作業チーム設計案を評価することを繰り返すことで、より良い作業チームを設計することを目指す手法である。

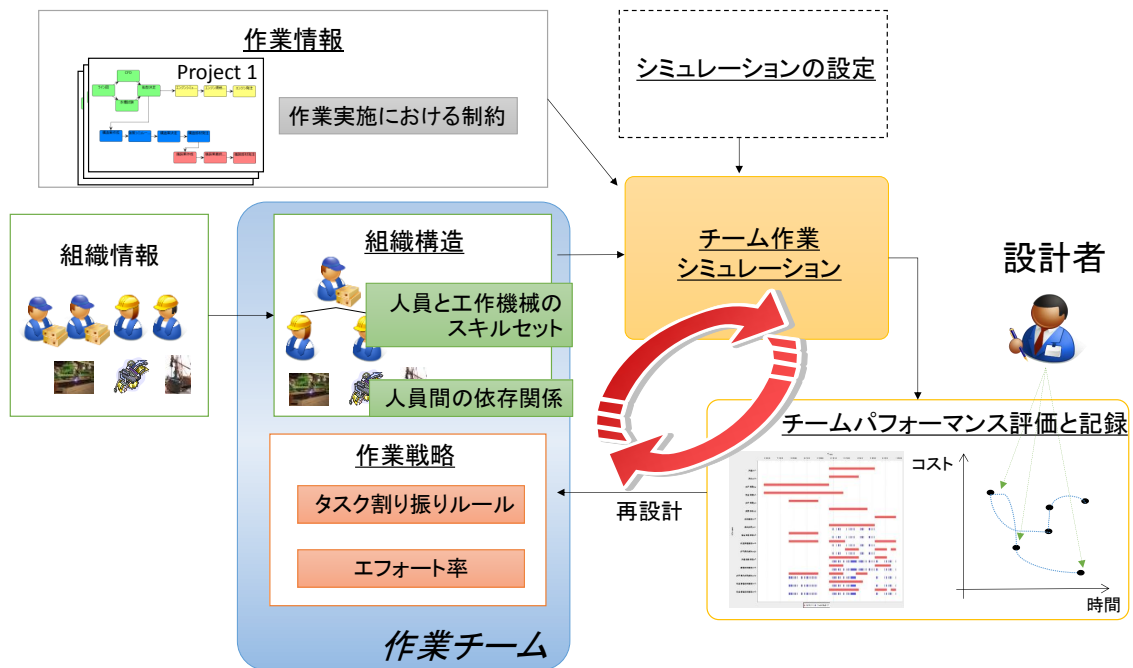


図 3-7 手動探索による作業チーム設計手法の概要

3.4.2 作業チーム手動探索の詳細

本研究で提案する作業チーム設計の流れを図 3-8 に示す。

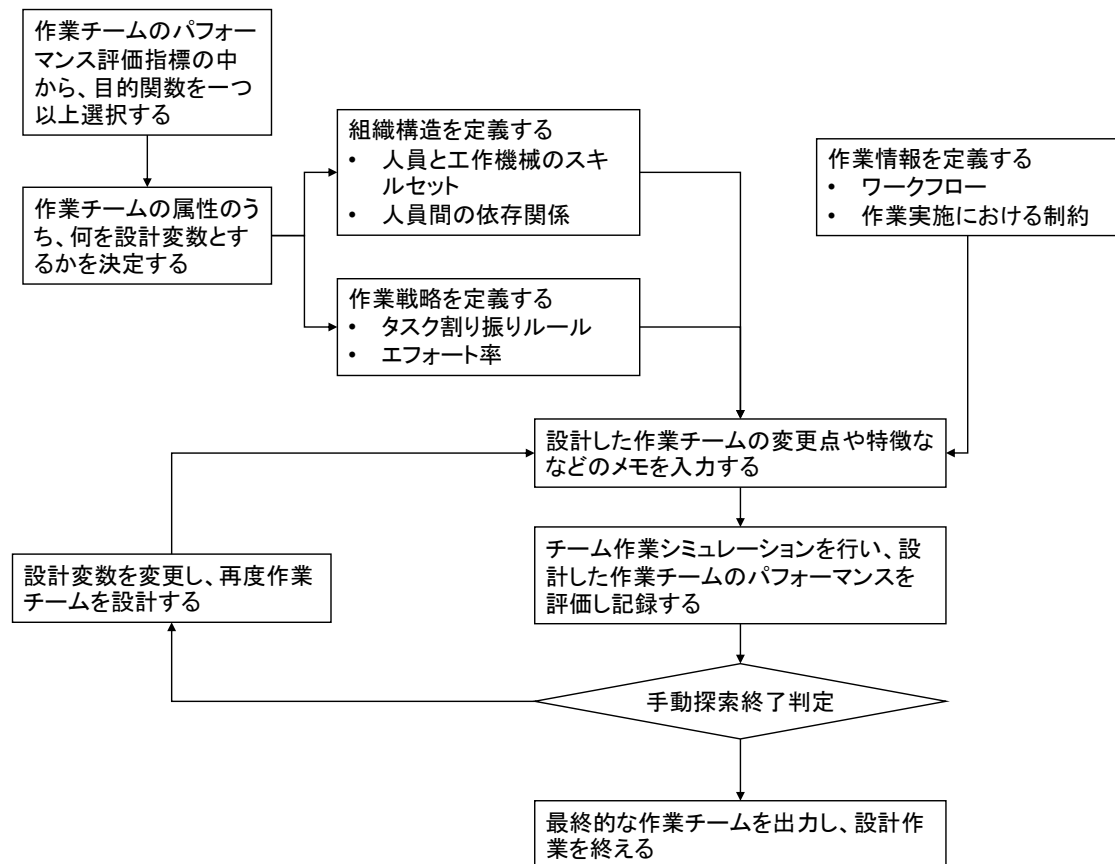


図 3-8 作業チーム手動探索の流れ

提案手法では、3.3.5 の作業チームのパフォーマンス評価指標である「所要時間」と「合計コスト」から、対象のプロジェクトに応じて必要な目的関数を選択して探索を開始する。また、3.3.4.1 のチーム作業シミュレーションの設定でトラブルを考慮とした場合については、各タスクのトラブル率に従って確率的にトラブルが発生するためチーム作業シミュレーションの結果が一定ではない。その場合には、1つの作業チーム設計案につき複数回のチーム作業シミュレーションを行い、「所要時間」と「合計コスト」のそれぞれについての「平均」や「標準偏差」を目的関数として採用することも考えられる。

次に、作業チームの設計変数のうち何を設計対象とするかを決定する。作業チームの設計を行う際の設計変数の候補を表 3-5 に示す。プロジェクトの段階に応じて表 3-5 のどの項目を設計変数とするかを選択し、手動探索を繰り返す際には選択した設計変数以外は固定

した状態で探索を行う。

表 3-5 作業チーム設計における設計変数の候補

設計変数	設計変数の設定に関する説明
作業チームに加える人員と工作機械	組織情報から作業チームに加えるべき人員・工作機械を選択する。
人員間の依存関係	チーム作業シミュレーションの設定でトラブルを考慮する場合にのみ結果に影響を与える。
各ワークフローへのエフォート率	作業情報内のそれぞれのワークフローに対するエフォート率を、それぞれの人員に対して個別に設定する
タスク割り振りルール	作業戦略内の各ディスパッチングルールに対する重みベクトルの値を設定する。

対象の作業情報に対して作業チームを設計し、チーム作業シミュレーションを行い、その結果をもとに作業チームを再設計することを繰り返すことで、適切な作業チームを設計する。

手動探索を繰り返し行う中で作業チーム再設計案を検討する際には図 3-9 に示すように、採用した目的関数の値と、過去に記録した作業チーム設計案をもとにして、選択した設計変数の値を変更することにより次の作業チーム設計を行う。手動探索を終了する条件としては以下の 2 つの条件のいずれかが考えられ、対象とする問題や状況に応じて適宜選択する必要がある。

- 作業チームのパフォーマンスが設計者の要求を満たしている場合
- 探索の結果、これ以上過去に探索した結果よりもパフォーマンスの高い作業チームを設計することができないと判断した場合

また、チーム作業シミュレーションの前には、今回設計した作業チームの変更点や特徴などのメモを入力し、前回の設計案と比べてどのような変更が加えられたのかを設計案と関連付けながら作業チームの設計を行う。

3.4.3 探索結果を利用した作業チームの決定と分析

3.4.2 の手動探索を終了した後、探索結果から最終的な作業チーム設計案を選択する流れを図 3-9 に示す。手動探索の終了後は、探索結果から設定した目的関数に対してパレートフロントを構成する作業チーム設計案を抽出する。最終的には、チーム作業シミュレーションでは考慮できなかった対象業務に関する知識(例: 人員 A は一定期間だけ作業不可)などの様々な項目を考慮して、パレートフロントを構成する作業チーム設計案の中から最終的な作業チーム案を選択する。

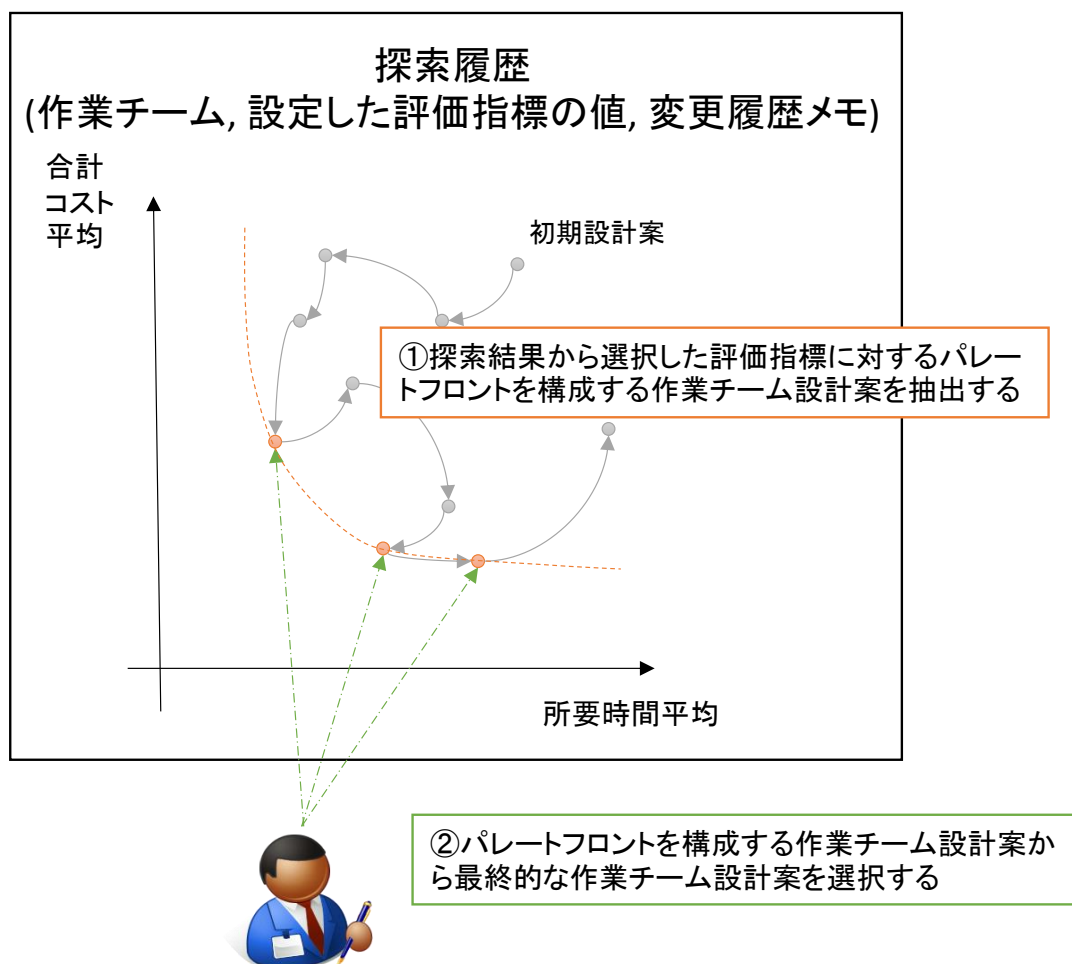


図 3-9 探索結果をもとにした最終設計案決定の流れ

また、手動探索によって、作業チーム設計案の情報とチーム作業シミュレーション結果の評価値に関する情報に加えて、設計した作業チームの変更点や特徴などのメモが設計の履歴として保存されている。表 3-6 に手動探索における作業チーム設計の履歴の例を示す。表 3-6 の 1 行が 1 つの作業チーム設計案の情報を表しており、メモ欄に関しては前回の作業チーム設計案からの変更点に関する情報が記載されている。このメモ欄と作業チームのパフォーマンス評価結果を比較しながら分析することで、どのような設計変更案が効果があったのかなどを振り返ることが容易になる。

表 3-6 手動探索における作業チーム設計の履歴

日付	結果ファイル	所要時間	合計コスト	メモ
2013/7/23 15:55	C:\Users\	69.05	4807.87	初期状態
2013/7/23 15:56	C:\Users\	67.58	4739.96	プログラム担当にアルバイトを一人追加
2013/7/23 15:58	C:\Users\	66.74	4696.94	更にもう一人追加
2013/7/23 15:59	C:\Users\	65.65	4640.83	更にもう一人追加
2013/7/23 16:00	C:\Users\	64.97	4631.29	更にもう一人追加
2013/7/23 16:01	C:\Users\	63.83	4523.85	更にもう一人追加
2013/7/23 16:03	C:\Users\	63.29	4505.01	更にもう一人追加
2013/7/23 16:05	C:\Users\	63.23	4526.99	システム担当にアルバイトを一人追加
2013/7/23 16:07	C:\Users\	62.83	4544.79	インターフェイス担当にアルバイトを一人追加
2013/7/23 16:24	C:\Users\	60.47	4271.99	システム担当にプログラム担当のスキルも追加した
2013/7/23 16:30	C:\Users\	58.38	4092.26	インターフェイス担当にシステム担当のスキルを追加
2013/7/23 16:34	C:\Users\	60.95	4290.97	インターフェイス担当にプログラム担当のスキルを追加&システム担当スキルを削除
2013/7/23 16:38	C:\Users\	54.47	3861.91	一個前に戻る+業務担当にインターフェイス担当のスキルを追加
2013/7/23 16:39	C:\Users\	54.91	3884.77	統括とインターフェイス担当の上下関係を削除
2013/7/23 16:41	C:\Users\	54.8	3885.56	一個前に戻る+統括とシステム担当の上下関係を削除

3.5 遺伝的アルゴリズムによる作業チーム設計支援手法

3.5.1 概要

あるプロジェクトを実施するための作業チームを設計する際には、組織内のすべての人員が登録された組織情報から作業チームに加えるべき人員を適切に選択する必要があるが、組織情報に登録された人員の数が多い場合には、プロジェクトに応じて人手で作業チームに加えるべき人員を適切に選択することは非常に負担がかかる。また、図 3-3 のように、作業チームを設計する際に作業チーム内のそれぞれの人員の依存関係まで考慮する場合は、単に作業チームに加えるべき人員を選択するよりも非常に困難な作業となる。また、作業戦略に関しては、本研究ではエフォート率とタスク割り振りルールを設計の対象としているが、どちらも実数によって構成されるため、手動で適切な値を設定するのは非常に困難である。

本節では、遺伝的アルゴリズムを用いて、3.4 の手動探索による作業チーム設計手法における探索を自動化することで、作業チームにおける組織構造と作業戦略の設計を支援する手法を提案する。提案手法では、設計対象が組織構造と作業戦略のいずれの場合においても、図 3-10 の流れに沿って作業チームの設計を行う。本論文で扱う問題は、複数の目的関数を満足する解を求める多目的最適化問題と捉えることが可能であり、提案手法では多目的最適化を行う代表的な手法である MOGA(Multi-Objective Genetic Algorithm)[44][45]を用いて作業チームの探索を行う。



図 3-10 遺伝的アルゴリズムによる作業チーム設計支援手法の流れ

以降では、3.5.2 で設計対象が組織構造の場合における自動探索手法と遺伝子のコード化について、3.5.3 で設計対象が作業戦略の場合における自動探索手法と遺伝子のコード化について説明する。また、3.5.4 で設定した目的関数に応じた遺伝的アルゴリズムにおける探索について説明し、3.5.5 で探索結果をもとにした作業チームの傾向分析手法と作業チームの最終決定について述べる。

3.5.2 組織構造の自動探索手法

組織構造の自動探索手法では、組織情報に登録されているすべての人員から、作業チームに加えるべき人員とそれらの依存関係を表す組織構造を遺伝的アルゴリズムによって探索する。ただし、提案手法は図 3-3 の組織構造のモデル化を前提に構築されたアルゴリズムである。また、提案手法を実施する際には、作業情報内のワークフローにおける各タスクには担当部署名がすべて登録されていることを前提としている。

組織構造の自動探索手法を含めた組織構造の設計支援手法の概要を図 3-11 に示す。提案手法では、プロジェクトの内容を表すワークフローと、作業チーム設計の対象となる組織情報を入力とする。組織の人員表と 1 つの遺伝子によって作業チームの組織構造を表現することができるように遺伝子型を設計し、組織構造とワークフローからトラブルと調整を考慮したチーム作業シミュレーションによって組織構造を含めた作業チームのパフォーマンスを評価することを繰り返し行う。

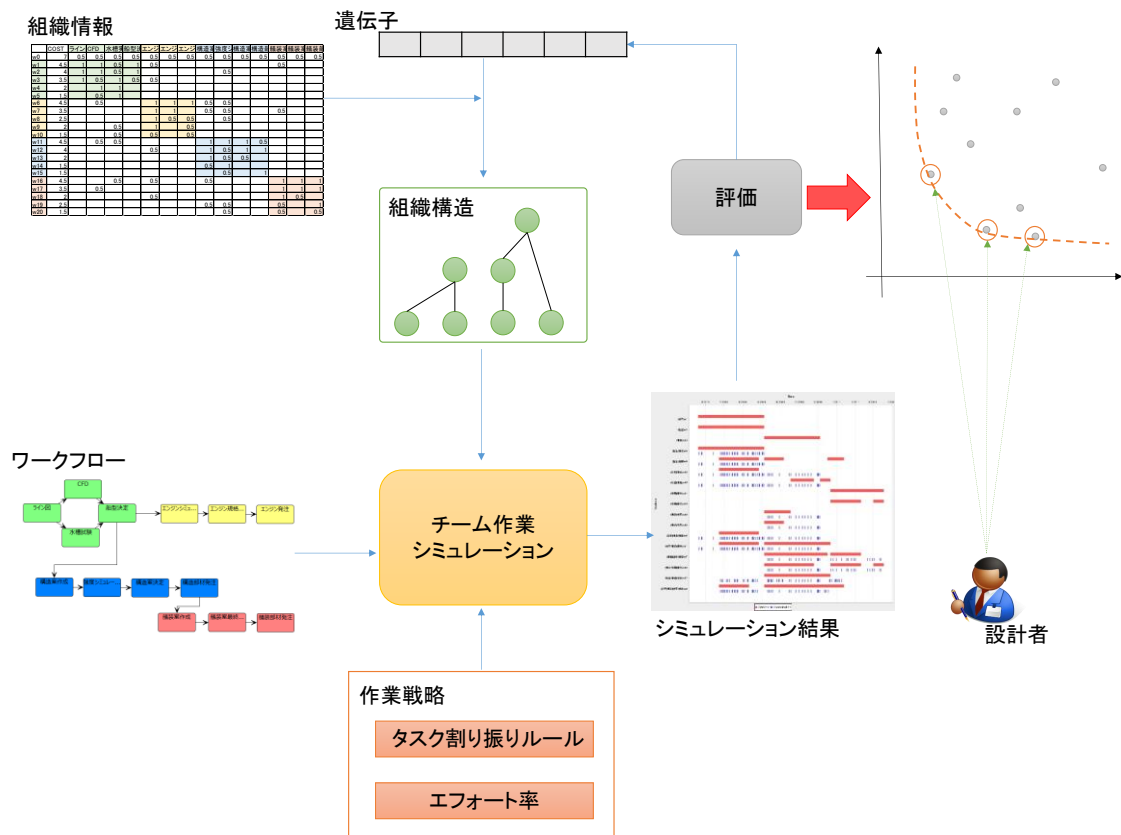


図 3-11 組織構造の設計支援手法の概要

遺伝的アルゴリズムを用いる際の遺伝子のコード化手法を図 3-12 に示す。まず、対象のワークフローに登録されている N_D 個の担当部署群を抽出する。組織情報に登録されているの人員の数が N_M 人である場合、 $N_D \cdot N_M$ 個の「0」または「1」を取る変数によって 1 つの遺伝子が構成される。人員 1 人当たり N_D 個の「0」または「1」を取る変数によってその人員の所属部署を表している。図 3-12 を例にすると、ある人員が「部署 3」に所属する場合は「001」、部署 1|2」に所属する場合は「110」、どこにも所属せずこのプロジェクトには関与しない場合は「000」と記述される。

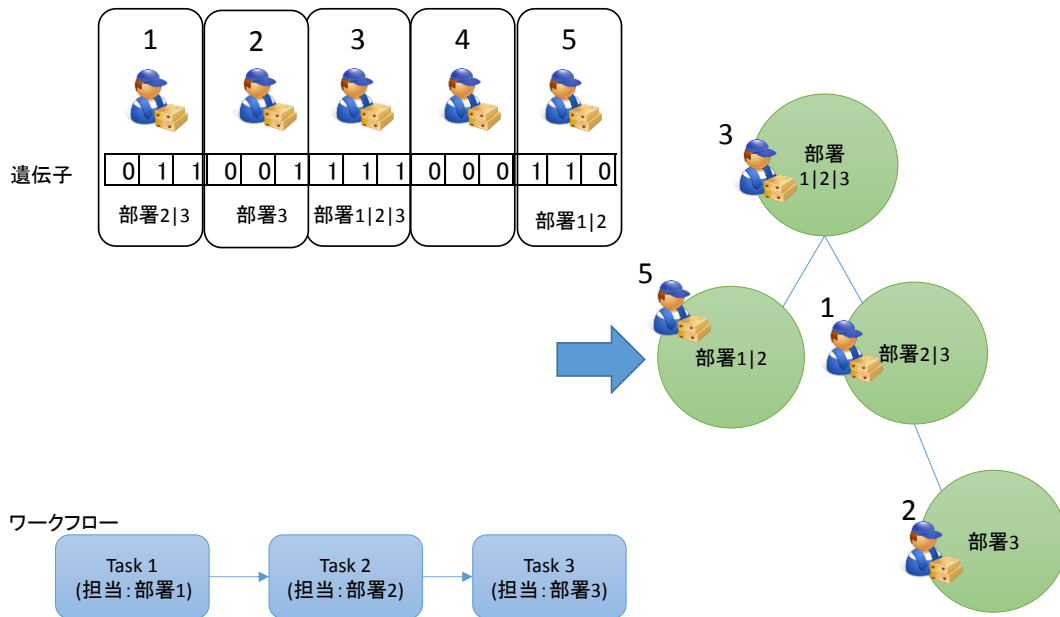


図 3-12 組織構造を表す遺伝子のコード化

ただし、図 3-12 の遺伝子コードによって表される組織構造をそのまま用いると、所属する部署の担当スキルを持たない人員が発生する可能性がある。そこで、本研究では図 3-13 で示すフィルタリングの方法を用いて補正した組織構造を用いてチーム作業シミュレーションを行う。図 3-13 では、組織情報に記載されているスキル情報を用いて、それぞれの人員がその部署に所属することは意味があるのかどうかを判定し、意味がない所属情報をそぎ落とす操作を行っている。例えば図 3-13 では、フィルタリング前は「部署 1|2」に所属していた人員が、部署 1 に関するスキルを持たないためフィルタリング後には「部署 2」に所属が変更していることが分かる。このフィルタリングを行うことで、現実にはあり得ない人員配置案を削除することが可能である。

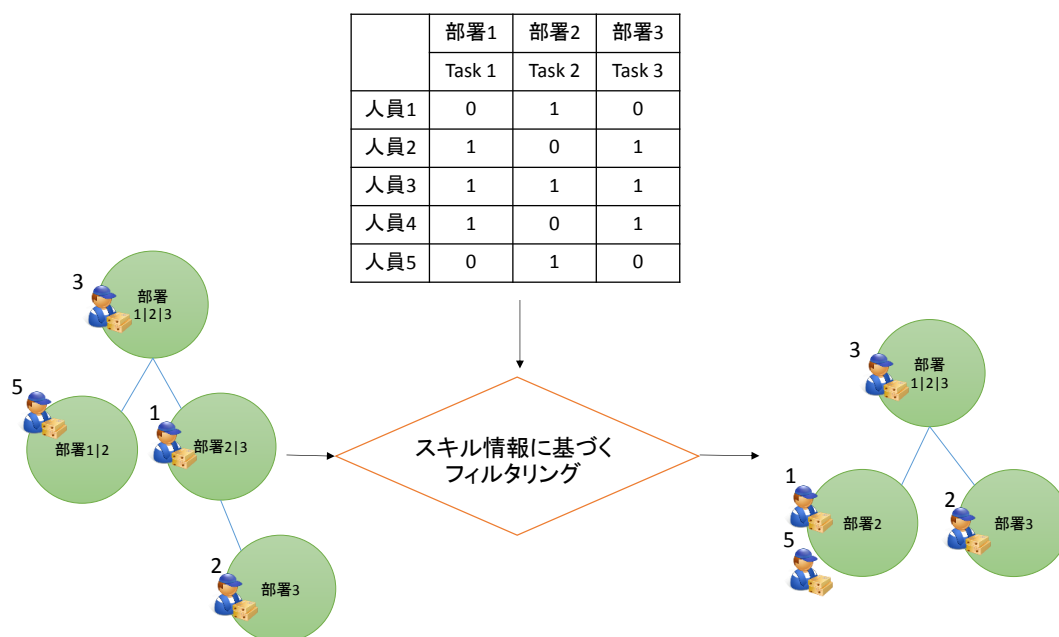


図 3-13 スキル情報に基づく組織構造情報のフィルタリング

上記でコード化された遺伝子とフィルタリングによって生成される組織構造を用いて、通常の遺伝的アルゴリズムと同様に交叉や突然変異による遺伝子群の操作とチーム作業シミュレーションを繰り返すことで、適切な組織構造の候補を自動的に探索することが可能になる。

3.5.3 作業戦略の自動探索手法

本節では、作業戦略の自動探索手法についてエフォート率の自動探索手法とタスク割り振りルール of 自動探索手法に分けて説明する。作業戦略の自動探索手法を含めた作業戦略設計支援手法の概要を図 3-14 に示す。

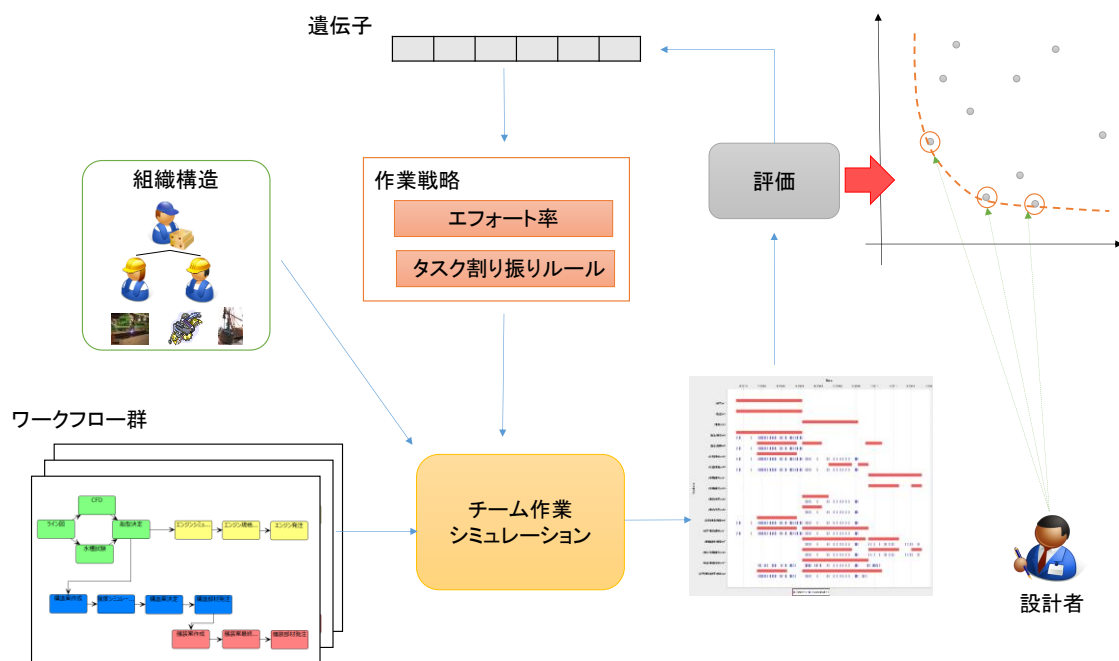


図 3-14 作業戦略の設計支援手法概要

作業戦略の自動探索手法では、作業チームの組織構造とプロジェクトの内容を表す複数のワークフローを入力とする。3.5.2 の組織構造の自動探索手法では、組織構造の構成要素である「作業チームに加えるべき人員の選定」と「作業チーム内人員の配置」を同時に探索する手法を採用した。しかしながら、作業戦略に関しては作業チーム内の人員個別の属性であるエフォート率と、作業チーム全体としての属性であるタスク割り振りルールという異なるため、作業戦略に関してはエフォート率の自動探索手法とタスク割り振りルールの自動探索手法の2つの手法を別々に使用することで対応する。

エフォート率を設計対象とする場合は、作業チーム内の各人員におけるエフォート率を1つの遺伝子に表現できるように遺伝子型を設計し、チーム作業シミュレーションによって作業

チーム内の各人員におけるエフォート率を含めた作業チームのパフォーマンスを評価することを繰り返し行う。一方で、タスク割り振りルールを設計対象とする場合は、作業チームにおけるタスク割り振りルールを1つの遺伝子に表現できるように遺伝子型を設計し、チーム作業シミュレーションによって作業チームにおけるタスク割り振りルールを含めた作業チームのパフォーマンスを評価することを繰り返し行う。

3.5.3.1 エフォート率自動探索手法

エフォート率の自動探索手法では、作業チームに採用されたすべての人員に対して、プロジェクトに登録されている各ワークフローに対してどの程度関わるかを表すエフォート率を遺伝的アルゴリズムによって探索する。提案手法は、プロジェクト内に複数のワークフローが登録されている状況を想定した手法である。

エフォート率の自動探索手法では、吉村ら[46]の提案するヒューマンリソース配置アルゴリズムにおける遺伝的アルゴリズムと3.3.4のチーム作業シミュレーションを用いて、作業チーム内におけるすべての人員に対する適切なエフォート率 $eR_{m,w}$ を探索する。探索範囲を限定するために、式(3.39)と式(3.40)の2つの制約を導入する。式(3.39)に関しては、エフォート率 $eR_{m,w}$ を11等分して離散変数 $\{0.0, 0.1, 0.2, \dots, 1.0\}$ として扱うことを示している。また、式(3.40)では1人の人員が持つエフォート率 $eR_{m,w}$ の合計は1であることを示している。すなわち、人員配置案の探索アルゴリズムでは作業チームに属する人員は少なくとも1つのワークフローに携わる必要があるものとし、この時点で作業チームに加えるべき人員の検討はすでに終了しているものとする。

$$eR_{m,w} = \{0.0, 0.1, 0.2, \dots, 1.0\} \quad (3.39)$$

$$\sum_w eR_{m,w} = 1.0 \quad (3.40)$$

エフォート率の自動探索アルゴリズムでは遺伝的アルゴリズムによってエフォート率 $eR_{m,w}$ の探索を行う。遺伝的アルゴリズムにおける遺伝子のコード化手法を図3-15に示す。式(3.39)と式(3.40)の制約を遵守するために、遺伝子列の中の「0」はエフォート率0.1を、「1」はワークフローの区切りを示している。すなわち、作業情報に N_w 個のワークフローが存在する

場合、作業チーム内の 1 人の人員に関するエフォート率の情報は 10 個の「0」と、ワークフロー間の区切りを表す $N_w - 1$ 個の「1」の遺伝子列によって表現される。

また、遺伝的アルゴリズムの操作である交叉に関しては、交叉点をそれぞれの人員を表す遺伝子列の区切り部分でのみ起こるように設定し、突然変異が起きた場合には、別の遺伝子のコードも同時に変異させることで上記の整合性を保つようにする。

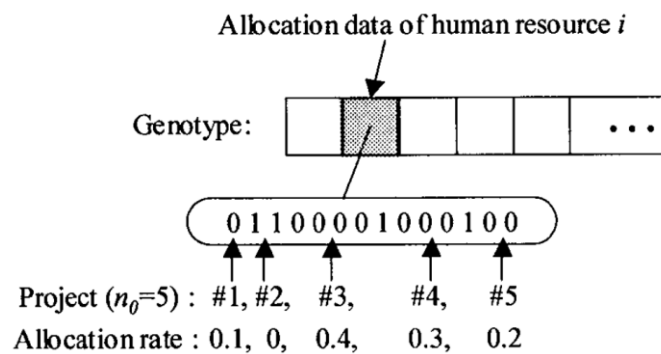


図 3-15 エフォート率の探索アルゴリズムにおける遺伝子のコード化[46]

上記でコード化された遺伝子を用いて、通常の遺伝的アルゴリズムと同様に交叉や突然変異による遺伝子群の操作とチーム作業シミュレーションを繰り返すことで、適切なエフォート率案の候補を算出する。

3.5.3.2 タスク割り振り規則の自動探索手法

タスク割り振り規則の自動探索手法では、タスク割り振り規則内の各ディスパッチングルールに対する重みを表す w の値を遺伝的アルゴリズムによって探索する。

タスク割り振り規則の自動探索手法では、ランダムキー型遺伝的アルゴリズム[47]とチーム作業シミュレーションを用いて、作業チーム内における適切なタスク割り振り規則の探索を行う。エフォート率と同様に探索範囲を限定するために、式(3.41)から式(3.49)で示すようにタスク割り振り規則 w の各パラメータを 0 から 1 の実数で探索を行い、探索結果の履歴には式(3.50)で示すようにタスク割り振り規則 w の二乗平均平方根が 1 になるよう正規化して出力を

行う.

$$0 \leq w_{EDD} \leq 1 \quad (3.41)$$

$$0 \leq w_{FIFO} \leq 1 \quad (3.42)$$

$$0 \leq w_{SPT} \leq 1 \quad (3.43)$$

$$0 \leq w_{EST} \leq 1 \quad (3.44)$$

$$0 \leq w_{TSLACK} \leq 1 \quad (3.45)$$

$$0 \leq w_{SPN} \leq 1 \quad (3.46)$$

$$0 \leq w_{SSP} \leq 1 \quad (3.47)$$

$$0 \leq w_{VC} \leq 1 \quad (3.48)$$

$$0 \leq w_{EC} \leq 1 \quad (3.49)$$

$$\sqrt{w_{EDD}^2 + w_{FIFO}^2 + w_{SPT}^2 + w_{EST}^2 + w_{TSLACK}^2 + w_{SPN}^2 + w_{SSP}^2 + w_{VC}^2 + w_{EC}^2} = 1 \quad (3.50)$$

遺伝的アルゴリズムを用いる際の遺伝子のコード化手法を図 3-16 に示す. タスク割り振りルール of 探索を行う際には, ランダムキー型遺伝的アルゴリズム[47]を用いて探索を行う. ランダムキー型遺伝的アルゴリズムは, 0 または 1 の羅列で 1 つの遺伝子が構成される通常の遺伝的アルゴリズムとは異なり, 0 から 1 の実数の羅列で 1 つの遺伝子を構成し, 交叉や突然変異を繰り返すことで探索を行う手法である. ランダムキー型遺伝的アルゴリズムを導入することで, タスク割り振りルール w の制約を遵守しながら遺伝的アルゴリズムと同様の探索を行うことが可能となる.

	EDD	FIFO	SPT	EST	TSLACK	SPN	SSP	VC	EC
遺伝子1	0.7894	0.9313	0.8734	0.3634	0.4073	0.7734	0.1535	0.7604	0.5374
遺伝子2	0.4011	0.3141	0.1452	0.1477	0.2604	0.7085	0.6818	0.5232	0.9781
遺伝子3	0.1965	0.6426	0.0999	0.4367	0.3907	0.8274	0.7515	0.8346	0.9659
遺伝子4	0.5209	0.0631	0.8900	0.5545	0.8657	0.9219	0.0296	0.8382	0.8655

図 3-16 タスク割り振りルールを表す遺伝子のコード化

3.5.4 目的関数の設定に応じた探索計算の詳細

提案手法では、3.3.5 の作業チームのパフォーマンス評価指標である「所要時間」と「合計コスト」から、対象のプロジェクトに応じて目的関数を 1 つ以上選択して探索を開始する。また、3.3.4.1 のチーム作業シミュレーションの設定でトラブルを考慮とした場合については、各タスクのトラブル率に従って確率的にトラブルが発生するためチーム作業シミュレーションの結果が一定ではない。その場合には、1 つの遺伝子につき複数回のチーム作業シミュレーションを行い、「所要時間」と「合計コスト」のそれぞれについての「平均」や「標準偏差」を目的関数として採用することも考えられる。

以降では、目的関数として「所要時間平均」と「合計コスト平均」の 2 つを採用した場合を例として説明する。提案手法による自動探索計算の詳細な流れを図 3-17 に示す。まず、所要時間平均最小を目的関数と設定して遺伝的アルゴリズムにより組織構造の探索を行う。次に、所要時間最少を目的関数と設定して同様に組織構造の探索を行う。どちらの探索でも、エリート戦略や交叉、突然変異により次世代の遺伝子群を作成してそれぞれの遺伝子をもとにチーム作業シミュレーションを行い、探索過程で発生する遺伝子とチーム作業シミュレーション結果の評価値である所要時間と合計コストの情報はすべて保存しておく。

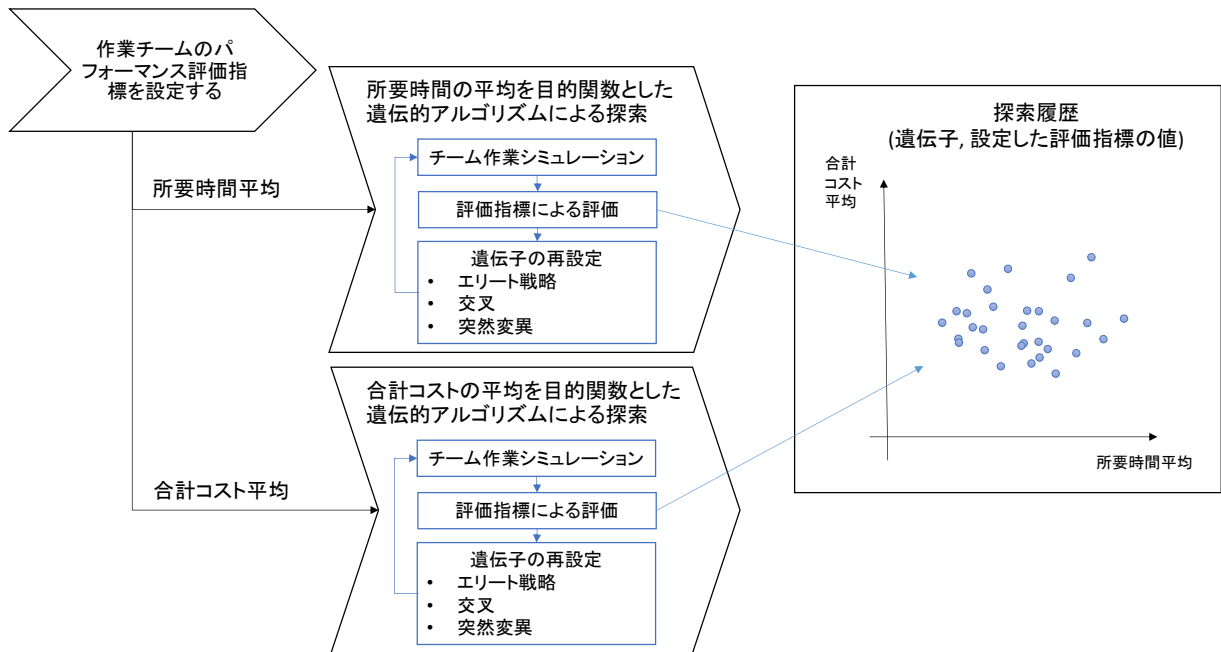


図 3-17 提案手法による自動探索計算の詳細な流れ

3.5.5 探索結果を利用した作業チームの決定と傾向分析

探索結果から最終的な作業チーム設計案を選択する流れを図 3-18 に示す。3.4 の手動探索による作業チーム設計手法と同様に、探索結果から設定した目的関数に対してパレートフロントを構成する作業チーム設計案を抽出し、その中から最終的な作業チームの設計案を選択する。

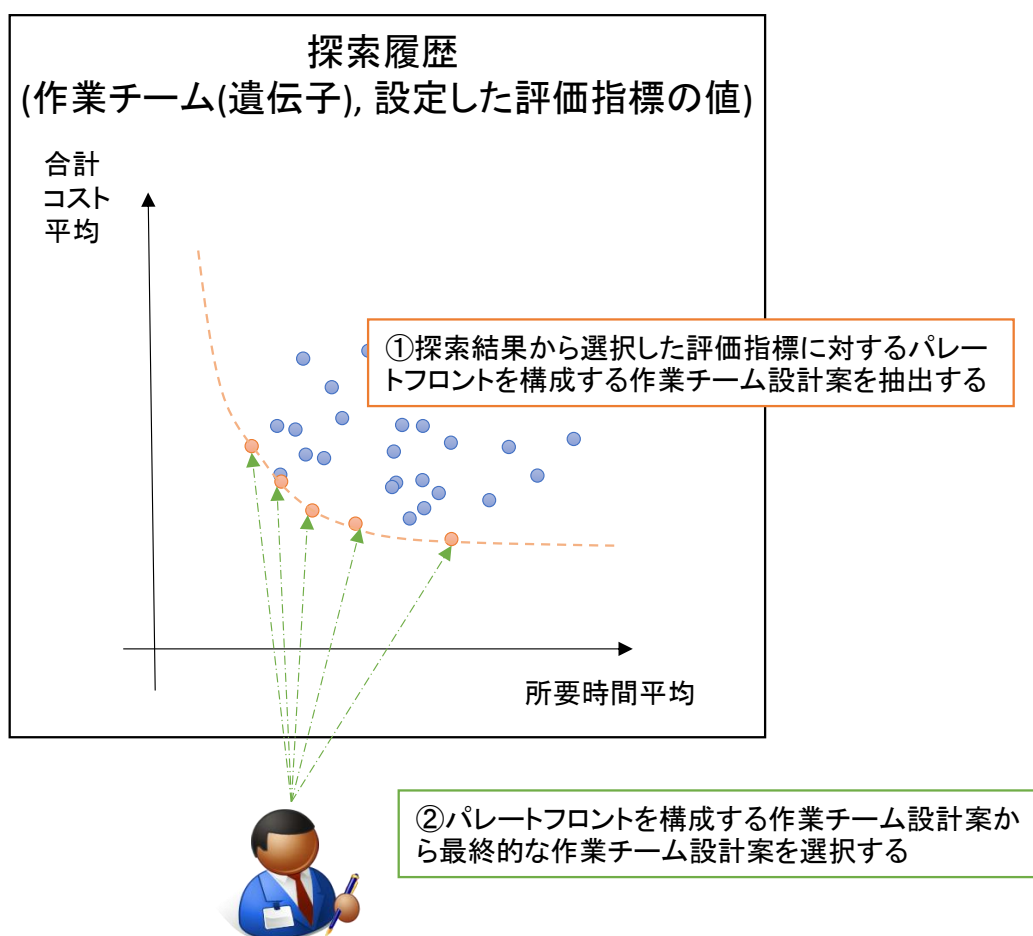


図 3-18 探索結果から最終的な作業チーム設計案を選択する流れ

また、3.4 の手動探索による作業チーム設計手法に比べて、遺伝的アルゴリズムによる作業チーム設計支援手法では自動で探索を行うため、パレトフロントを構成する作業チーム設計案を複数見つけやすいという特徴がある。そこで、単に作業チームの決定だけでなく作業チームの傾向分析を行うことも可能である。例えば、「所要時間平均」と「合計コスト平均」の2つを採用した場合においては、パレトフロントを構成する作業チーム設計案を分析することで、所要時間を短縮する作業チームと合計コストを短縮する作業チームはどのような違いがあるのかを分析することが可能である。

作業チームの傾向分析を行う場合、設計対象として組織構造やエフォート率を選択した場合には、パレトフロントを構成する作業チーム設計案を直接閲覧することで分析が可能で

あるが、タスク割り振りルールに関しては、各ディスパッチングルールの重みを正規化したベクトルでタスク割り振りルールを定義しているという特性上、結果を閲覧しても直感的に理解することは困難である。そのため、本節ではタスク割り振りルールの分析手法の一つとして、シナリオの変化に対するタスク割り振りルールの変化分析手法を提案する。シナリオの変化に対するタスク割り振りルールの変化分析手法の概要を図 3-19 に示す。予め設定した基本シナリオから設計されるタスク割り振りルールと、設備や法律の変更・突然のトラブルなどの基本シナリオから派生したシナリオをもとに設計されるタスク割り振りルールを比較することで、シナリオの変化に対するタスク割り振りルールの変化分析を行う。具体的には、タスク割り振りルールにおける各成分の log 値の差を比較することで、各ディスパッチングルールにおける評価値の桁数に関係なく、派生シナリオでは基本シナリオに比べて、タスク割り振りルール内のどのディスパッチングルールが以前よりも重視または軽視されたかを定量的に評価することが可能になる。

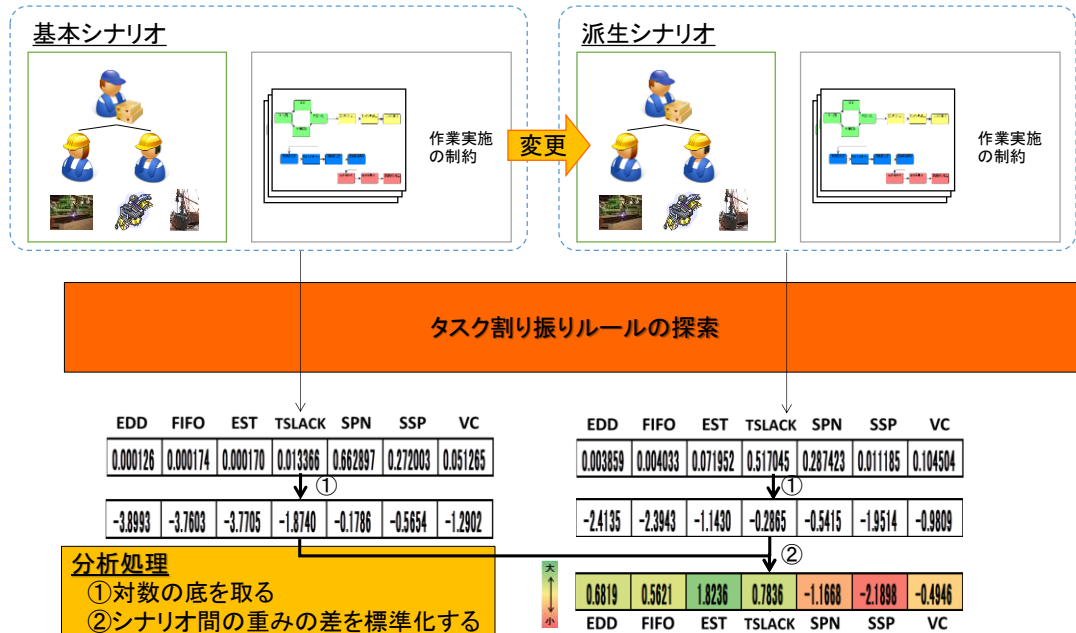


図 3-19 シナリオの変化に対するタスク割り振りルールの変化分析手法

第4章 開発した作業チーム設計支援システム

4.1 はじめに.....	66
4.2 作業チーム設計支援システムの基本機能.....	66
4.2.1 作業情報の入力.....	68
4.2.2 作業チームの入力.....	70
4.2.3 作業戦略の入力.....	75
4.2.4 チーム作業シミュレーション機能.....	77
4.3 手動探索による作業チーム設計支援機能.....	80
4.4 遺伝的アルゴリズムによる作業チーム設計支援機能.....	83
4.4.1 組織構造の設計支援機能.....	83
4.4.2 作業戦略の設計支援手法.....	85

4.1 はじめに

本章では、第 3 章で説明した作業チーム設計手法を実装したソフトウェアである作業チーム設計支援システムについて説明する。

作業チーム設計支援システムの対象に関する情報を表 4-1 に示す。作業チーム設計支援システムは、製造業やシステム開発業務などで日常的に行われている小規模から中規模程度のプロジェクト形式の業務を対象としており、マネージャやリーダーなどの作業チームを設計する立場の人員が作業チームを設計する際に用いられることを想定している。

表 4-1 作業チーム設計支援システムの対象に関する情報

	詳細
ユーザ (システムの使用者)	作業チームを設計する立場の人員 (例：マネージャやリーダーなど)
対象業務の種類	ワークフローなどによって業務の進め方が決まっているプロジェクト形式の業務 (例：製造業・システム開発業務など)
作業チームの人数	小規模～中規模 (2 人～50 人程度)

作業チーム設計支援システムは、IBM によって開発された統合開発環境(IDE)である Eclipse の GEF(Graphical Editing Framework)[48]を利用して開発し、eclipse RCP(Rich Client Platform)[49]の形で配布可能なシステムである。入力情報をもとにチーム作業シミュレーションを行う基本機能に加えて、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法をソフトウェア上で実行可能なシステムである。

4.2 作業チーム設計支援システムの基本機能

開発した作業チーム設計支援システムの基本機能の概要を図 4-1 に示す。作業チーム設計支援システムの基本機能は、ワークフロー作成画面・作業チーム作成画面・チーム作業シミュレーションの詳細設定画面の3つの画面で入力情報を定義し、チーム作業シミュレーショ

ン機能によってガントチャートを生成する機能である。生成されたガントチャートを分析することによって、入力情報として定義した作業チームのパフォーマンスを評価することが可能になる。

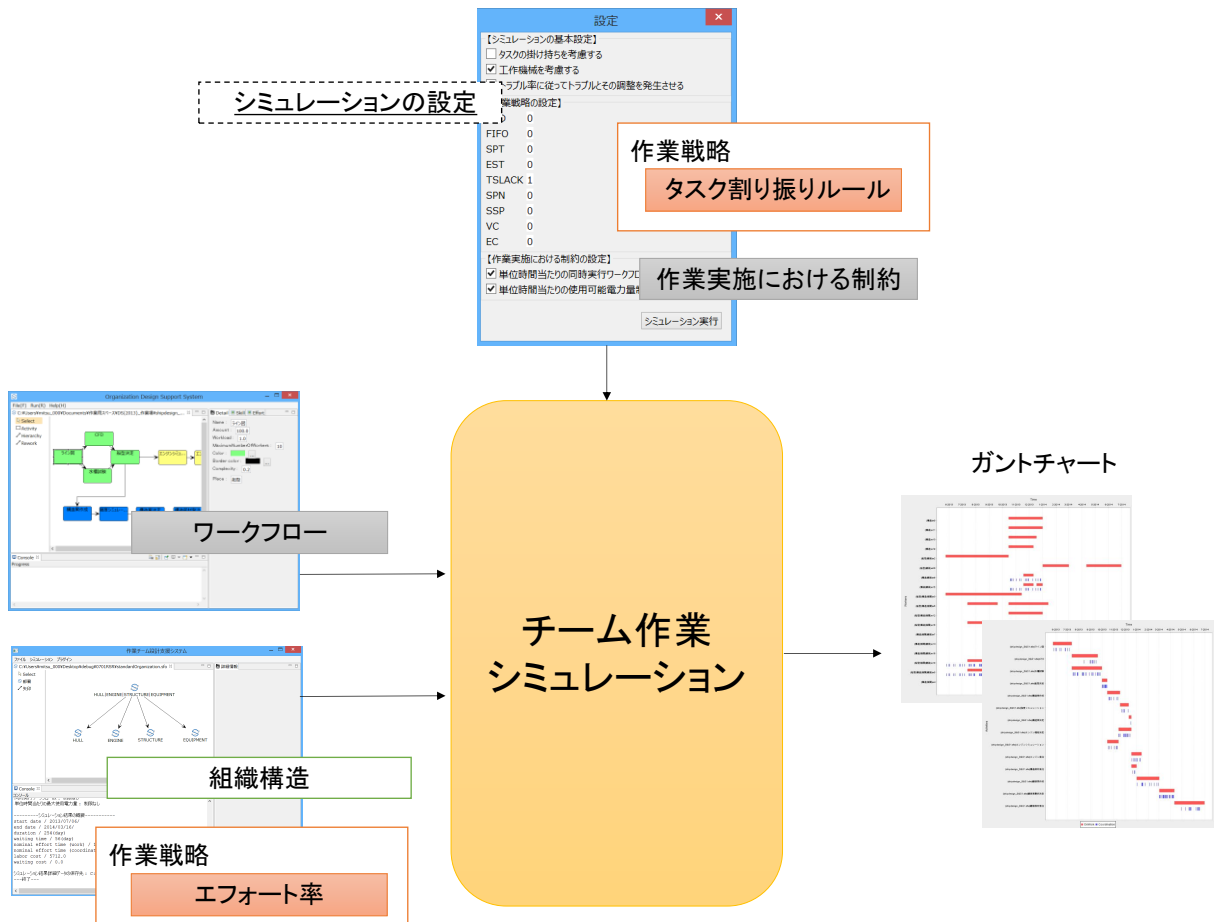


図 4-1 作業チーム設計支援システムの基本機能の概要

本節では、作業情報や作業チームをどのように定義するかを説明するとともに、チーム作業シミュレーション機能の詳細についても説明する。

4.2.1 作業情報の入力

4.2.1.1 ワークフローの入力

開発した作業チーム設計支援システムにおけるワークフローの入力画面を図 4-2 に示す。

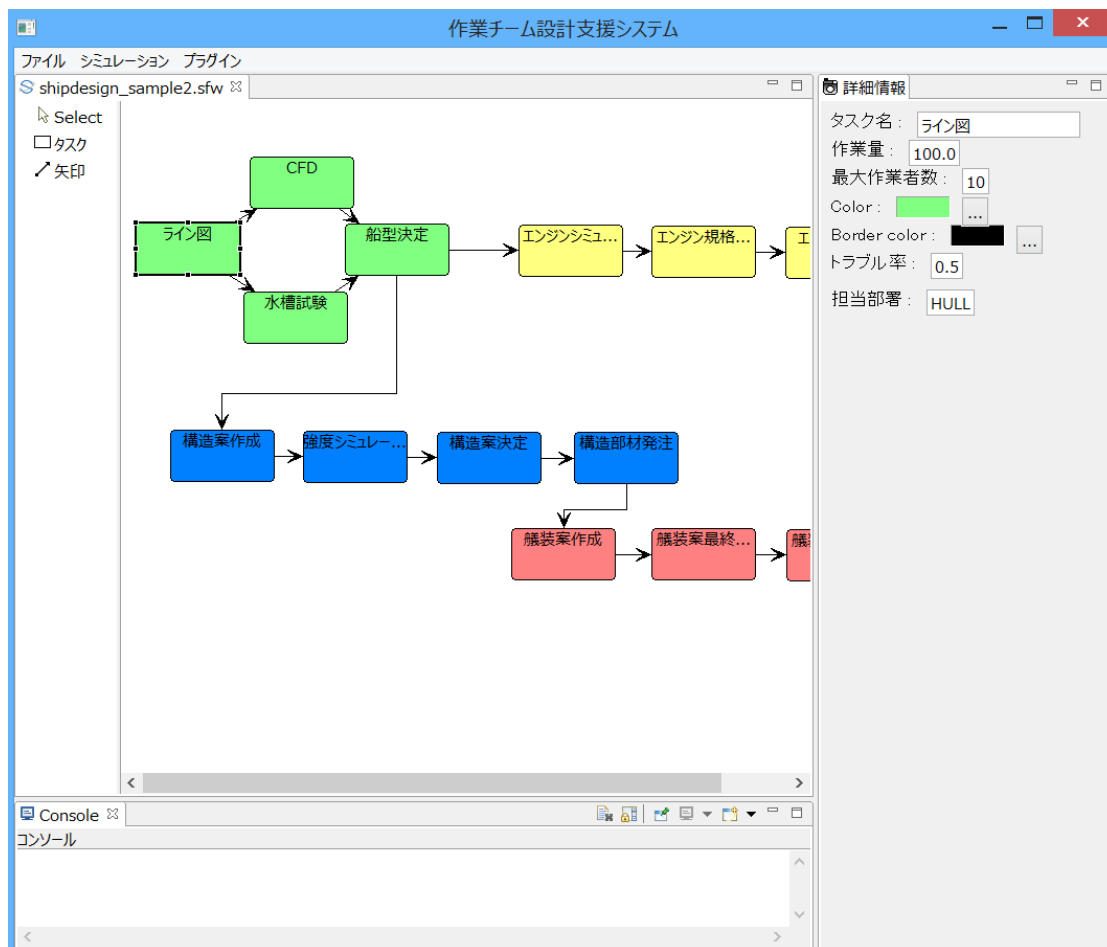


図 4-2 ワークフロー入力画面

ワークフローは BPMN(Business Process Modeling Notation)[50]形式で記述するために、文書管理システム ShareFast[51]のワークフロー記述手法を採用した。ユーザは、左側のパレットからタスクコマンドを選択し、ワークフローエディタ上でクリックすることでタスクを自由に配置することができ、配置したタスクを選択することで右側のビューで各タスクの属性を定義することが

できる。さらに、各タスクの依存関係を矢印で表現することで、ワークフローが完成する。

ワークフローのデータを表すファイルの一例を図 4-3 に示す。ワークフローファイルは ShareFast と同様に xml 形式で保存される。保存したファイルを再度開くことで、途中で中断したワークフローデータ作成を再開することも可能である。



図 4-3 ワークフロー保存ファイルの例

4.2.1.2 作業実施における制約の入力

作業実施において考慮すべき制約に関しては3.3.2.2で述べたとおり、対象とする業務や対象とする現場によって異なる。本研究では、次章以降のケーススタディで用いる「同時実行ワークフロー数」と「単位時間あたりに使用可能な電力使用量の最大値」のみを想定して、作業実施における制約の入力機能を実装した。入力画面を図4-4に示す。作業実施における制約に関しては、独立したファイルではなく開発したシステム上での設定として登録を行う。

【シミュレーションの基本設定】	
<input type="checkbox"/>	タスクの掛け持ちを考慮する
<input checked="" type="checkbox"/>	工作機械を考慮する
<input type="checkbox"/>	トラブル率に従ってトラブルとその調整を発生させる

【作業戦略の設定】	
EDD	0
FIFO	0
SPT	0
EST	0
TSLACK	1
SPN	0
SSP	0
VC	0
EC	0

【作業実施における制約の設定】	
<input checked="" type="checkbox"/>	単位時間当たりの同時実行ワークフロー数制限 2
<input checked="" type="checkbox"/>	単位時間当たりの使用可能電力量制限 255

シミュレーション実行

図 4-4 作業実施における制約の入力画面

4.2.2 作業チームの入力

4.2.2.1 組織構造の入力

組織構造情報を作成する前には、図4-5のように作業対象とするワークフローが保存されているフォルダを選択する。フォルダ内に保存されているワークフロー群に含まれるタスク名を抽出し、抽出したタスク名をもとにしてチーム人員のスキルを登録するためのスキルマップを作成する。

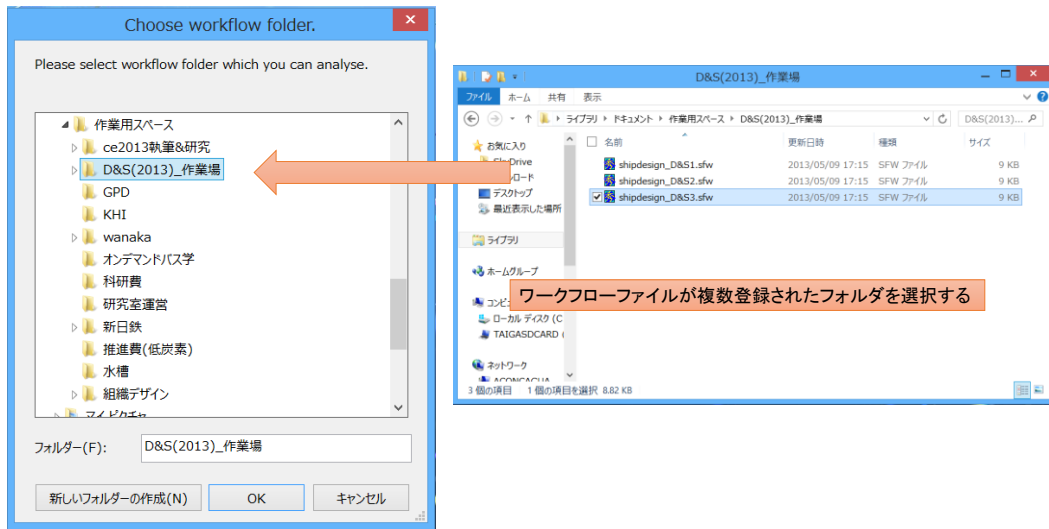


図 4-5 ワークフローフォルダの選択

まず、部署ごとに所属する人員と工作機械の情報を入力する。人員と工作機械の入力画面を図 4-6 に示す。まず、ワークフローの時と同様に部署をエディタ上に作成し、右側のビューでその部署に所属する人員と工作機械の登録と編集を行うことで、1 つの部署情報を作成する。チーム人員のスキルに関しては、右側のスキル表の部分で名前とコストに加えて図 4-5 で抽出したスキルマップに基づいたスキル値の登録を行う。この機能によって、ワークフロー内の各タスクと関連させたスキルの登録が容易になる。

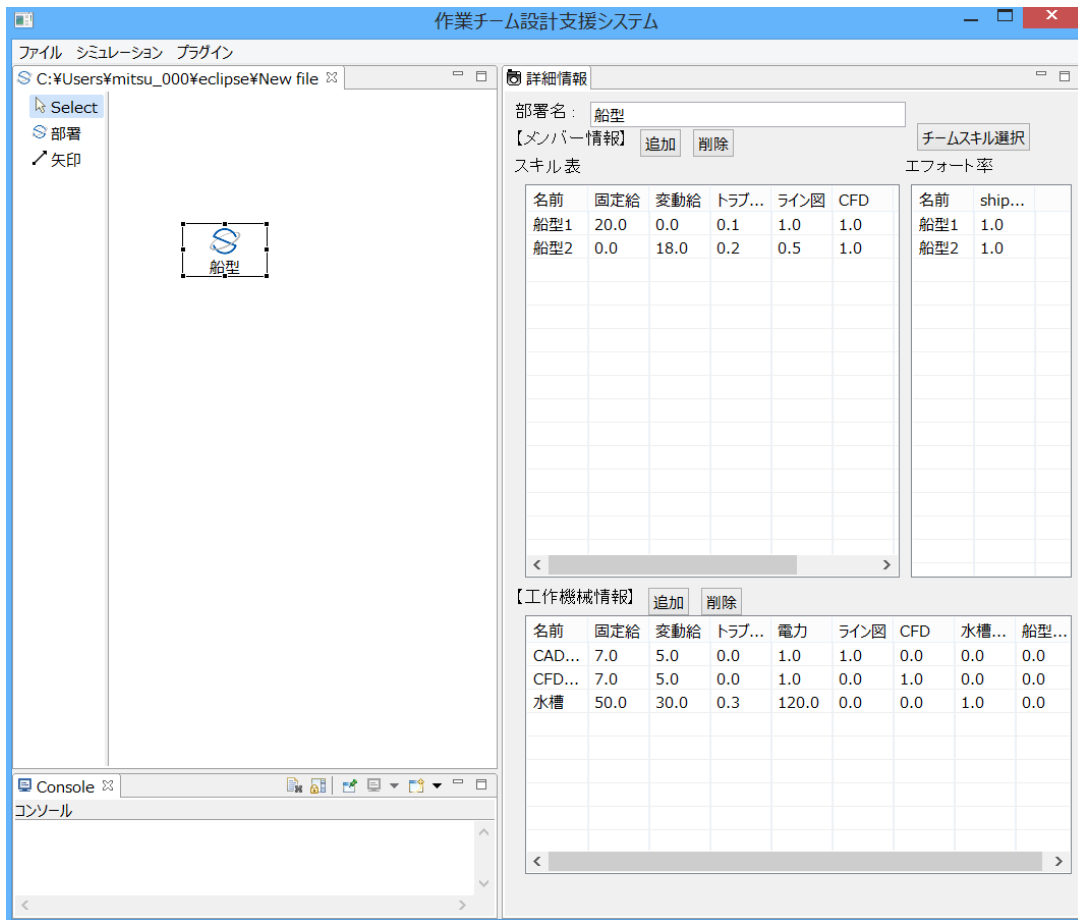


図 4-6 人員と工作機械情報の入力画面

次に、必要に応じて部署間の依存関係情報を入力する。部署間の依存関係情報の入力画面を図 4-7 に示す。ある部署内の人員が担当するタスクでトラブルが発生した場合に、どの部署内の人員と調整を行うかを矢印によって定義する。矢印は、部署間の依存関係情報を表しており、図 4-7 では部署「船型」に所属する人員の担当タスクでトラブルが発生した場合、部署「PM」に所属する人員との調整が行われることを示している。また、この矢印を引くことによって、同時に部署「船型」の各タスクに対する権限が部署「PM」に追加される。

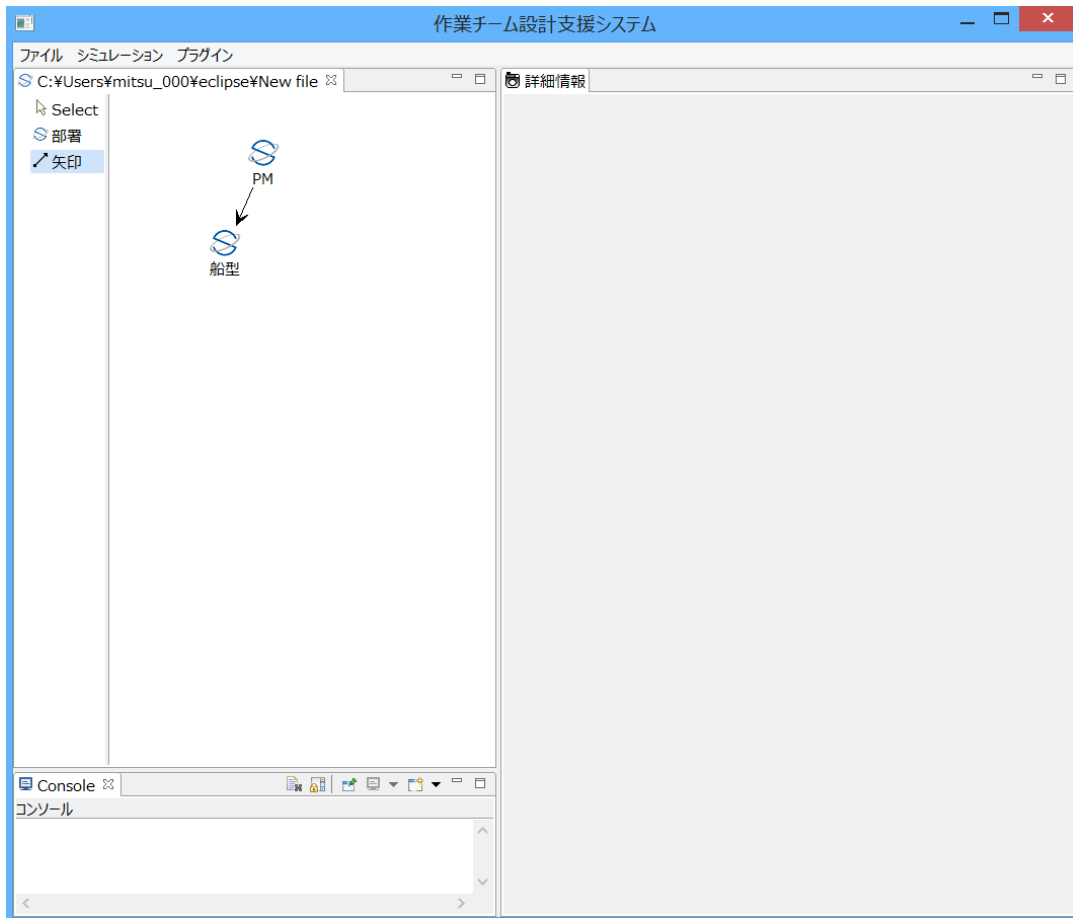


図 4-7 部署間の依存関係情報の入力画面

組織構造情報を保存したファイルの一例を図 4-8 に示す。組織構造情報はワークフローと同じ xml 形式で保存され、保存したファイルを再度開くことで、途中で中断した組織構造情報の作成を再開することも可能である。

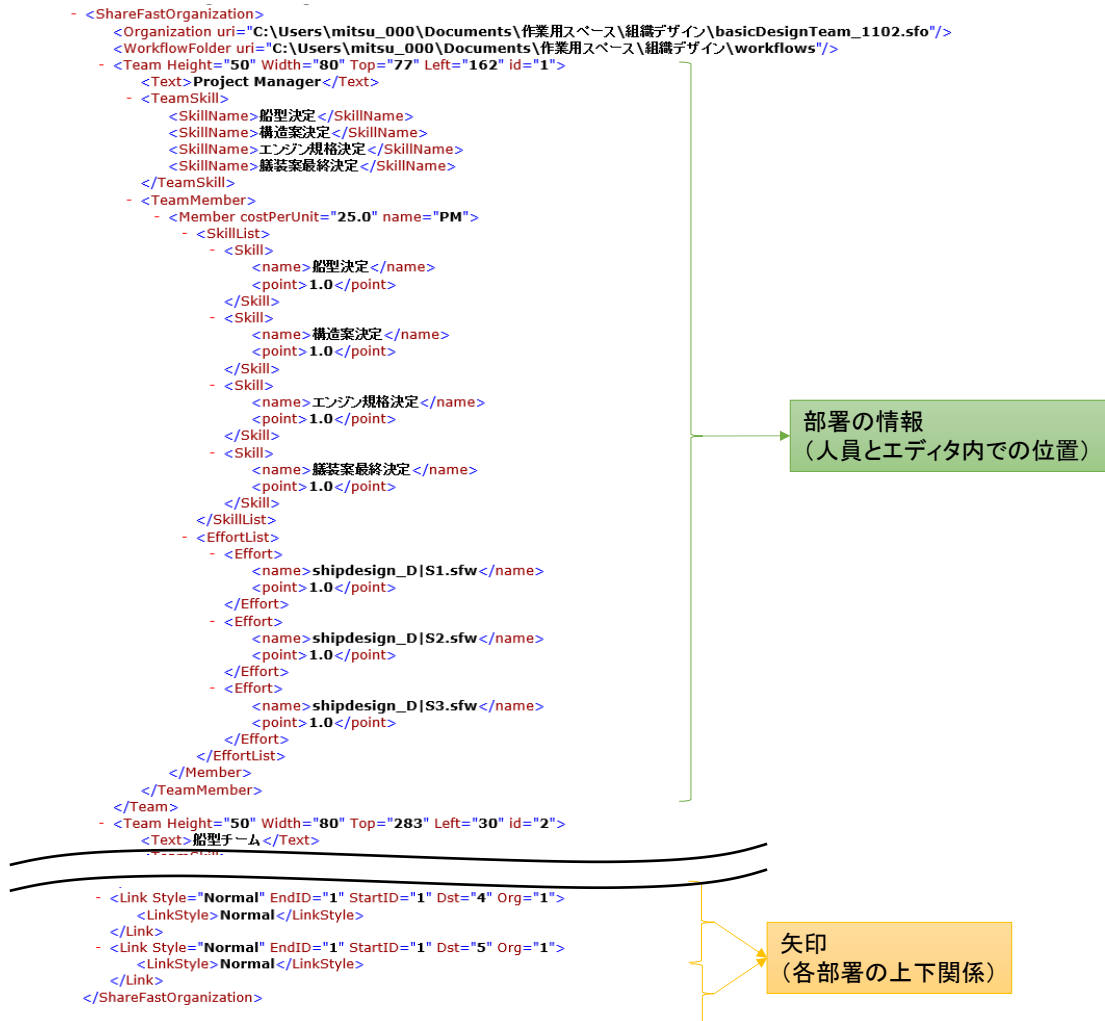


図 4-8 組織構造情報保存ファイルの例

4.2.3 作業戦略の入力

4.2.3.1 各ワークフローへのエフォート率の入力

各部署に所属するそれぞれの人員のエフォート率は、図 4-9 に示す通り右側のビュー内でのエフォート表で定義する。エフォート表の先頭行には図 4-5 で指定したフォルダ内に存在するワークフローファイルのファイル名が記載されており、それぞれの人員は個別に各ワークフローへのエフォート率を定義することが可能である。エフォート率の情報は、図 4-8 と同じ組織構造ファイルに保存される。

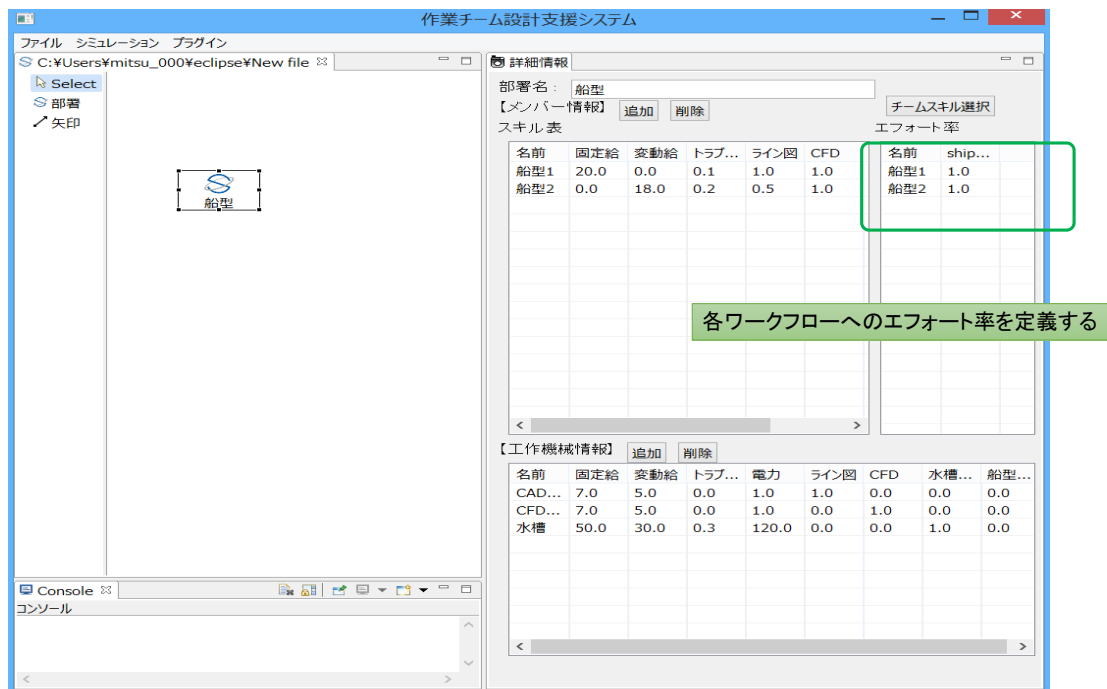


図 4-9 各ワークフローへのエフォート率の入力画面

4.2.3.2 タスク割り振りルールの入力

タスク割り振りルールは 3.3.3.3 で述べた通り、ディスパッチングルールの重みベクトルで表されるが、開発したシステム上では図 4-10 のように、タスク割り振りルール内の各ディスパッチン

グループに対して重みを表す数値を直接入力することでタスク割り振りルールを入力する。

設定

【シミュレーションの基本設定】

- タスクの掛け持ちを考慮する
- 工作機械を考慮する
- トラブル率に従ってトラブルとその調整を発生させる

【作業戦略の設定】

EDD	0
FIFO	0
SPT	0
EST	0
TSLACK	1
SPN	0
SSP	0
VC	0
EC	0

【作業実施における制約の設定】

- 単位時間当たりの同時実行ワークフロー数制限 2
- 単位時間当たりの使用可能電力量制限 255

シミュレーション実行

図 4-10 タスク割り振りルールの入力画面

4.2.4 チーム作業シミュレーション機能

4.2.4.1 チーム作業シミュレーションの設定

本研究におけるチーム作業シミュレーションの設定とは、3.3.4.1で述べた3つの項目に対して考慮するか否かを定めることである。開発したシステム上では、図4-11のようにチェックボックスの入力によってタスクの掛け持ち・工作機械の考慮・トラブルと調整の考慮の有無を決定する。

【シミュレーションの基本設定】	
<input type="checkbox"/>	タスクの掛け持ちを考慮する
<input checked="" type="checkbox"/>	工作機械を考慮する
<input type="checkbox"/>	トラブル率に従ってトラブルとその調整を発生させる

【作業戦略の設定】	
EDD	0
FIFO	0
SPT	0
EST	0
TSLACK	1
SPN	0
SSP	0
VC	0
EC	0

【作業実施における制約の設定】	
<input checked="" type="checkbox"/>	単位時間当たりの同時実行ワークフロー数制限 2
<input checked="" type="checkbox"/>	単位時間当たりの使用可能電力量制限 255

シミュレーション実行

図 4-11 チーム作業シミュレーション設定の入力画面

4.2.4.2 チーム作業シミュレーションの実行

チーム作業シミュレーションは開発したシステムの画面とは別スレッドで実行され、チーム作業シミュレーションの途中経過と終了の合図は図 4-12 で示すように、下のコンソールビューに表示される。チーム作業シミュレーションの終了時には、以下のファイルが 1 つの zip ファイルにまとめられた形で出力される。

- チーム作業シミュレーション結果の概要(所要時間やコスト情報など)
- 人員と工作機械に関する詳細なチーム作業シミュレーション結果ファイル
- 人員と工作機械に関するガントチャート
- 各タスクに関する詳細なチーム作業シミュレーション結果ファイル
- 各タスクに関するガントチャート

人員や工作機械と各タスクに関するガントチャートの例を図 4-13 に示す。人員や工作機械に関するガントチャートでは、各人員がどの時間にタスクを担当していたかを赤色で、どの時間に調整作業を行っていたかを青色で表している。また、各タスクに関するガントチャートでは各タスクの実作業が行われていた時間を赤色で、各タスクの調整作業が行われていた時間を青色で表している。また、それぞれのチーム作業シミュレーション結果ファイルにはガントチャートでは表現できない詳細な情報(ある時間に人員 A は実際にどのタスクを担当していたか等)が記載されている。なお、ガントチャートの出力機能に関しては、JFreeChart ライブラリ [52]を利用して実装した。

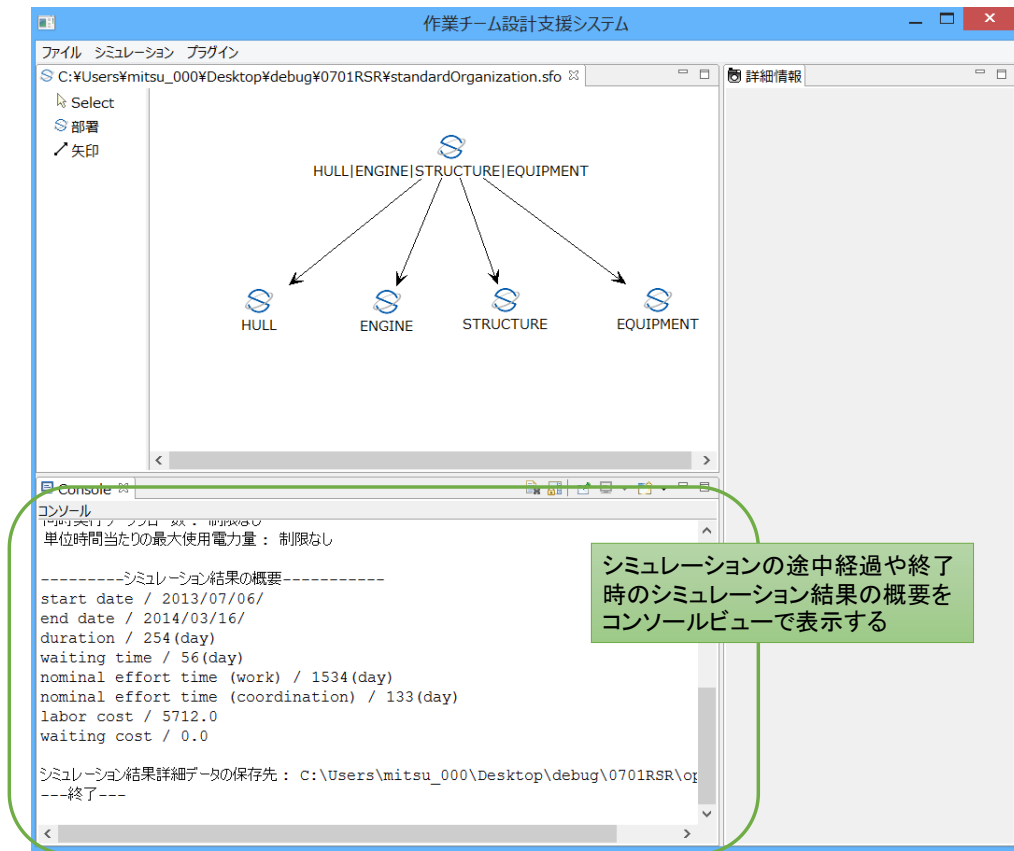


図 4-12 チーム作業シミュレーション実行中の画面



図 4-13 ガントチャートの例

4.3 手動探索による作業チーム設計支援機能

開発した作業チーム設計支援システムでは、3.4の手動探索による作業チーム設計手法を支援する機能をプラグインの形で実装した。手動探索による作業チーム設計支援機能の概要を図 4-14 に示す。

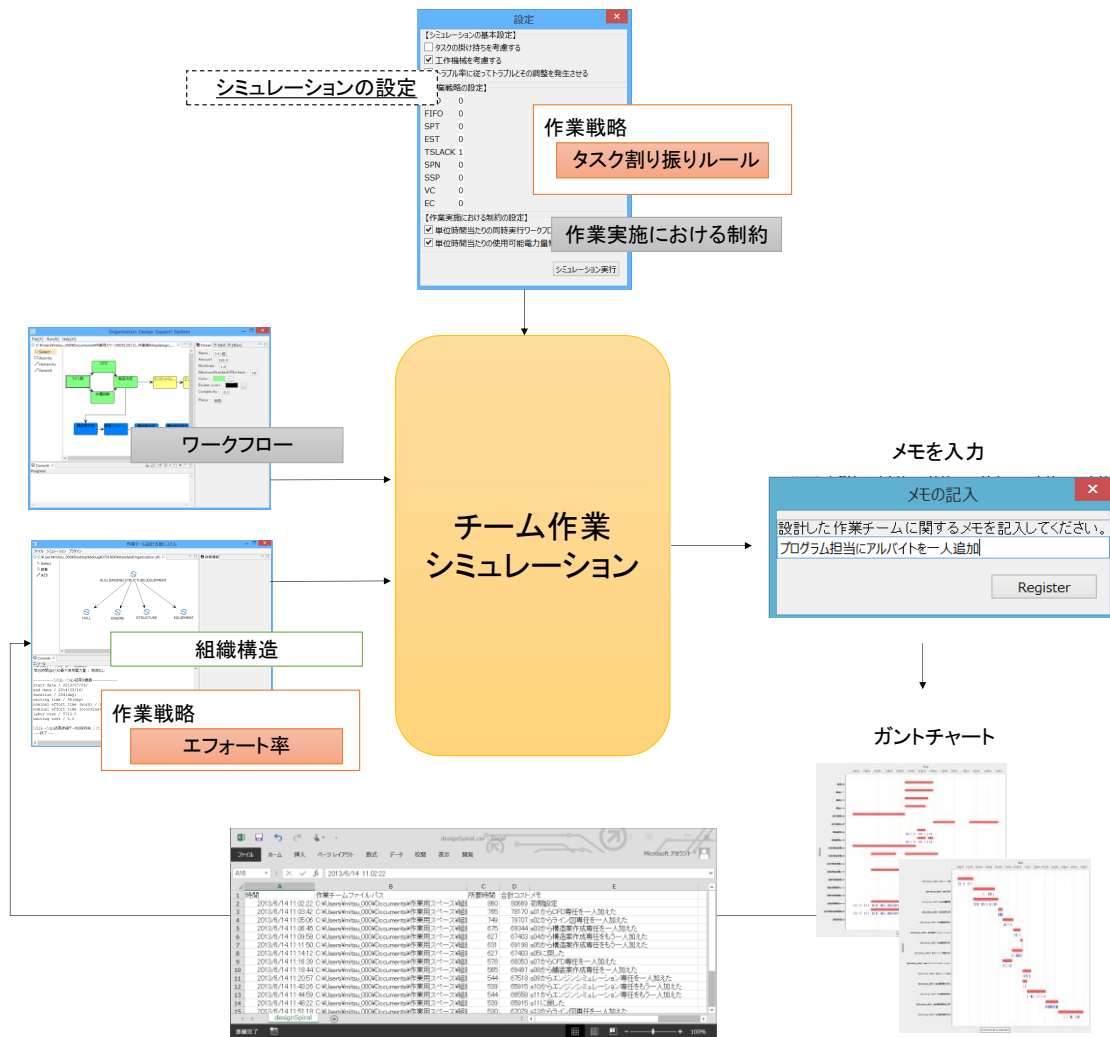


図 4-14 手動探索による作業チーム設計支援機能の概要

手動探索による作業チーム設計支援機能を実行すると、チーム作業シミュレーションを行うたびに、作業チームの変更に関するメモを記入するダイアログが図 4-15 のように表示されるよ

うになる。さらに、チーム作業シミュレーションを行うたびに、チーム作業シミュレーションの結果である所要時間・合計コストの情報と、作業チームの変更に関するメモ、入力データである作業チームのファイルパスなどの情報が図 4-16 のように CSV 形式で保存される。作成された CSV ファイルを Microsoft Excel などを開き、所要時間と合計コストを基準としてグラフを作成することで図 3-9 のようなグラフを作成できる。図 3-9 のように手動探索の履歴をグラフで表すことにより、作成・変更した作業チームが探索履歴の中でどの位置づけに当たるのかを確認することが可能である。

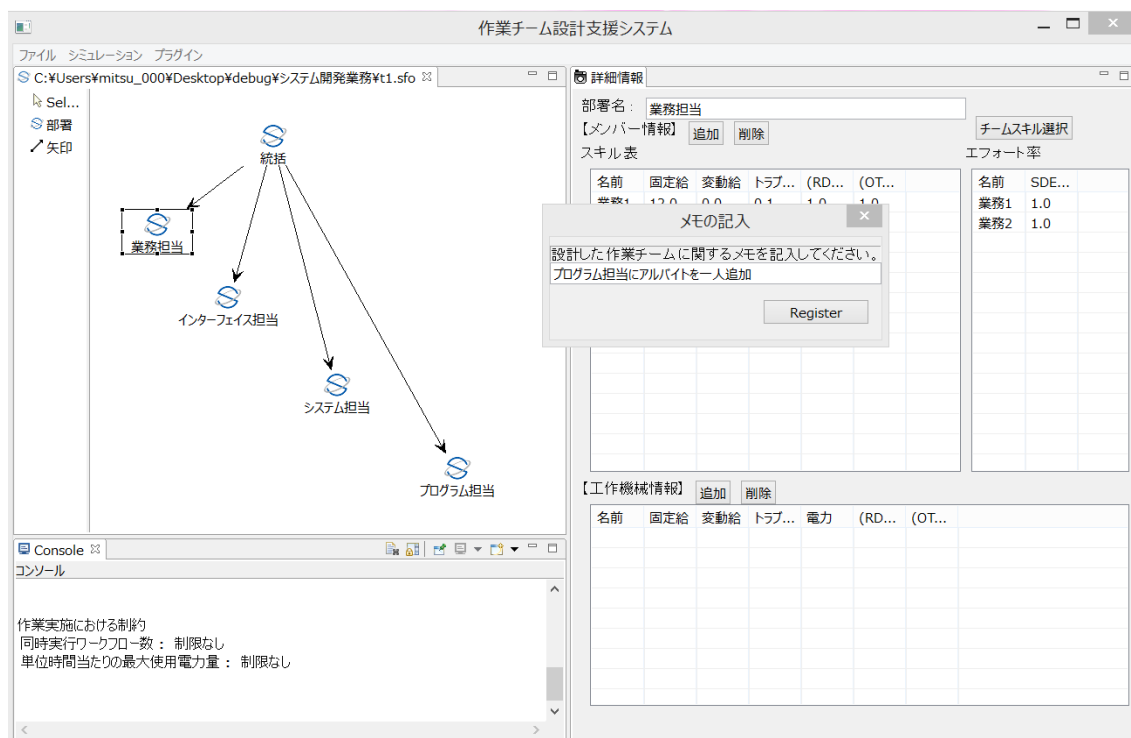


図 4-15 手動探索に関するメモの入力画面

開発した作業チーム設計支援システム

時間	ファイルパス	所要時間	合計コスト	メモ
2013/7/23 15:55	C:\Users\mitsu_000\	69.06	4807.87	初期状態
2013/7/23 15:56	C:\Users\mitsu_000\	67.58	4739.96	プログラム担当にアルバイトを一人追加
2013/7/23 15:58	C:\Users\mitsu_000\	66.74	4696.94	更にもう一人追加
2013/7/23 15:59	C:\Users\mitsu_000\	65.65	4640.83	更にもう一人追加
2013/7/23 16:00	C:\Users\mitsu_000\	64.97	4631.29	更にもう一人追加
2013/7/23 16:01	C:\Users\mitsu_000\	63.83	4523.85	更にもう一人追加
2013/7/23 16:03	C:\Users\mitsu_000\	63.29	4505.01	更にもう一人追加
2013/7/23 16:05	C:\Users\mitsu_000\	63.23	4526.99	システム担当にアルバイトを一人追加
2013/7/23 16:07	C:\Users\mitsu_000\	62.83	4544.79	インターフェイス担当にアルバイトを一人追加
2013/7/23 16:18	C:\Users\mitsu_000\	64.14	4716.64	プログラムアルバイト部署を作成した
2013/7/23 16:24	C:\Users\mitsu_000\	60.47	4271.99	システム担当にプログラム担当のスキルも追加した
2013/7/23 16:30	C:\Users\mitsu_000\	58.38	4092.26	インターフェイス担当にシステム担当のスキルを追加
2013/7/23 16:34	C:\Users\mitsu_000\	60.95	4290.97	インターフェイス担当にプログラム担当のスキルを追加&システム担当スキルを削除
2013/7/23 16:38	C:\Users\mitsu_000\	54.47	3861.91	一個前に戻る+業務担当にインターフェイス担当のスキルを追加
2013/7/23 16:39	C:\Users\mitsu_000\	54.91	3884.77	統括とインターフェイス担当の上下関係を削除
2013/7/23 16:41	C:\Users\mitsu_000\	54.8	3885.56	一個前に戻る+統括とシステム担当の上下関係を削除
2013/7/23 16:44	C:\Users\mitsu_000\	54.46	3852.7	結局3個前に戻る

図 4-16 手動探索の履歴が記録された CSV ファイル

4.4 遺伝的アルゴリズムによる作業チーム設計支援機能

開発した作業チーム設計支援システムでは、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法をプラグインの形で実装した。以降では、本研究で対象としている作業チーム設計変数である組織構造と作業戦略のそれぞれについて、開発した作業チーム設計支援システム上でどのような機能として追加したかについて説明する。

4.4.1 組織構造の設計支援機能

組織構造の設計支援機能の概要を図 4-17 に示す。

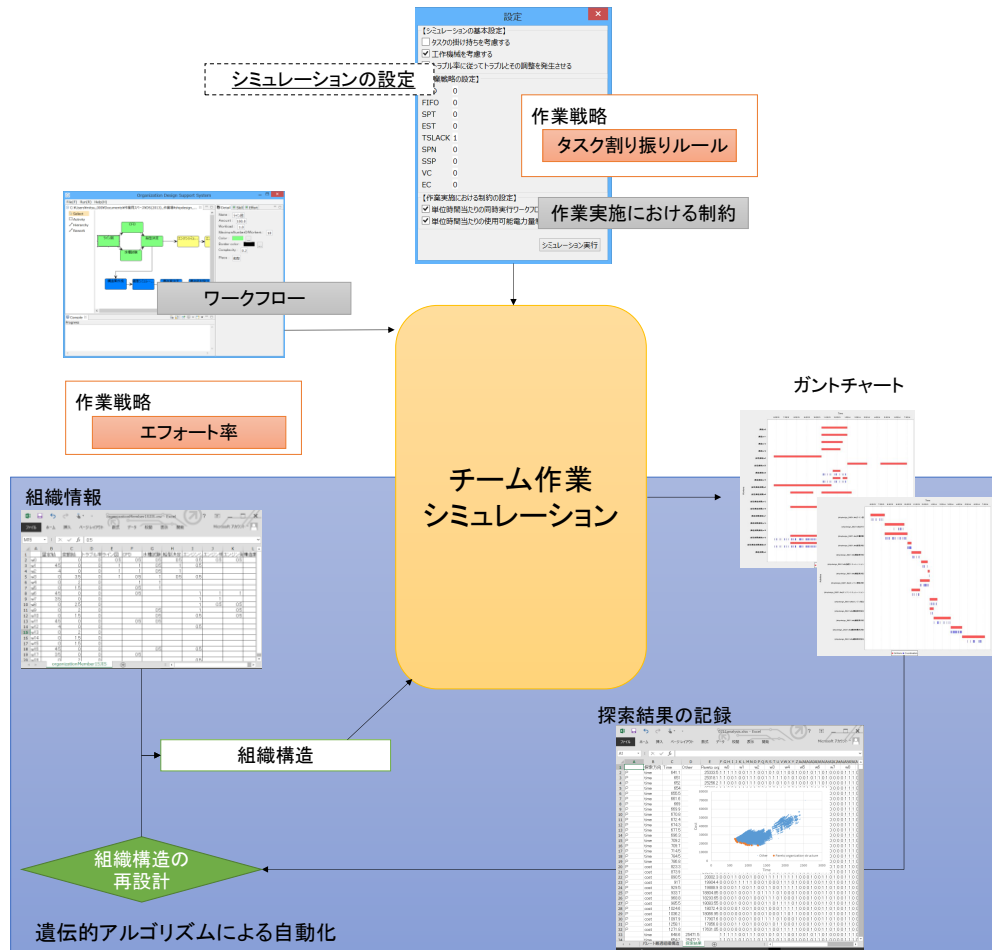


図 4-17 組織構造の設計支援機能の概要

組織構造の設計支援機能を実行する際には、プラグインとして埋め込んだコマンドを実行し、次に組織情報を表す CSV ファイルを入力する。すると、3.5.2 で説明した組織構造の自動探索により作業チームにおける組織構造の探索が開始される。探索処理は開発したシステムの画面とは別スレッドで実行され、探索の途中経過は下のコンソールビューに表示される。探索終了後は、図 4-18 の左図のように、探索過程で検討したすべての組織構造案を採用した場合における作業チームのパフォーマンスに関する情報や、複数の目的関数を採用した場合に関しては、採用した目的関数に対してパレートフロントを構成する組織構造とその他の組織構造を区別する記号などの情報を含んだ情報を、図 3-9 のグラフが作成しやすいように出力する。また、図 4-18 の右図のように、採用した目的関数に対してパレートフロントを構成する組織構造案を採用した場合における作業チームの詳細な情報は別ファイルとして出力され、4.2.4 のチーム作業シミュレーションを再度実行することで、詳細なチームパフォーマンスの評価や設計の検討を行うことが可能になる。

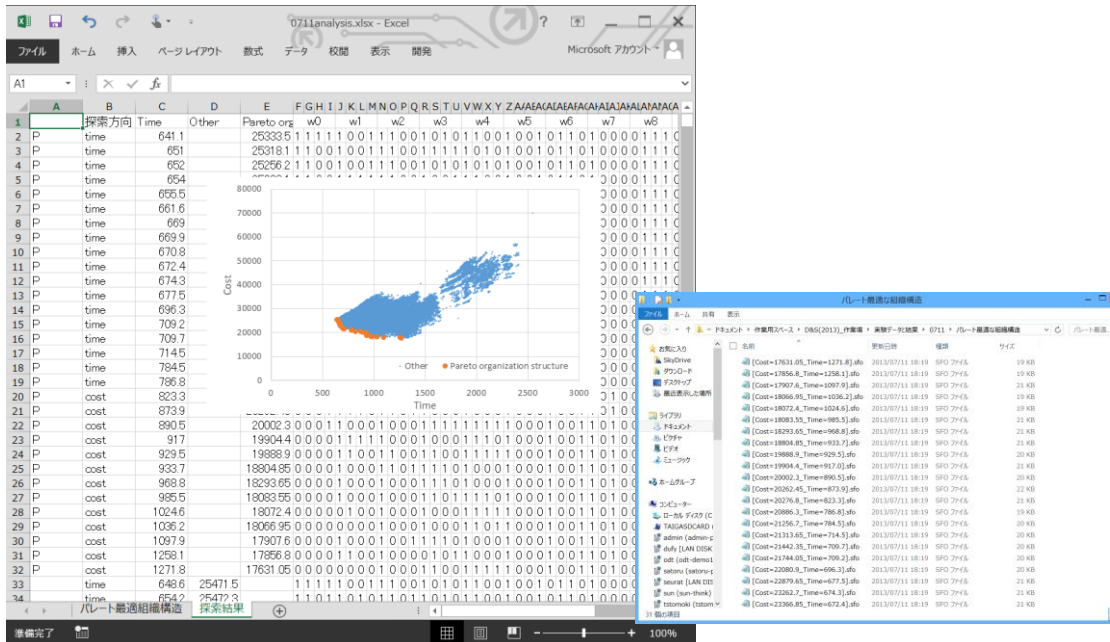


図 4-18 組織構造の設計支援機能実行に出力されるファイルなど

4.4.2 作業戦略の設計支援手法

4.4.2.1 エフォート率の設計支援機能

開発した作業チーム設計支援システムでは、3.5.3.1 のエフォート率自動探索手法をプラグインの形式で実装した。エフォート率の設計支援機能の概要を図 4-19 に示す。

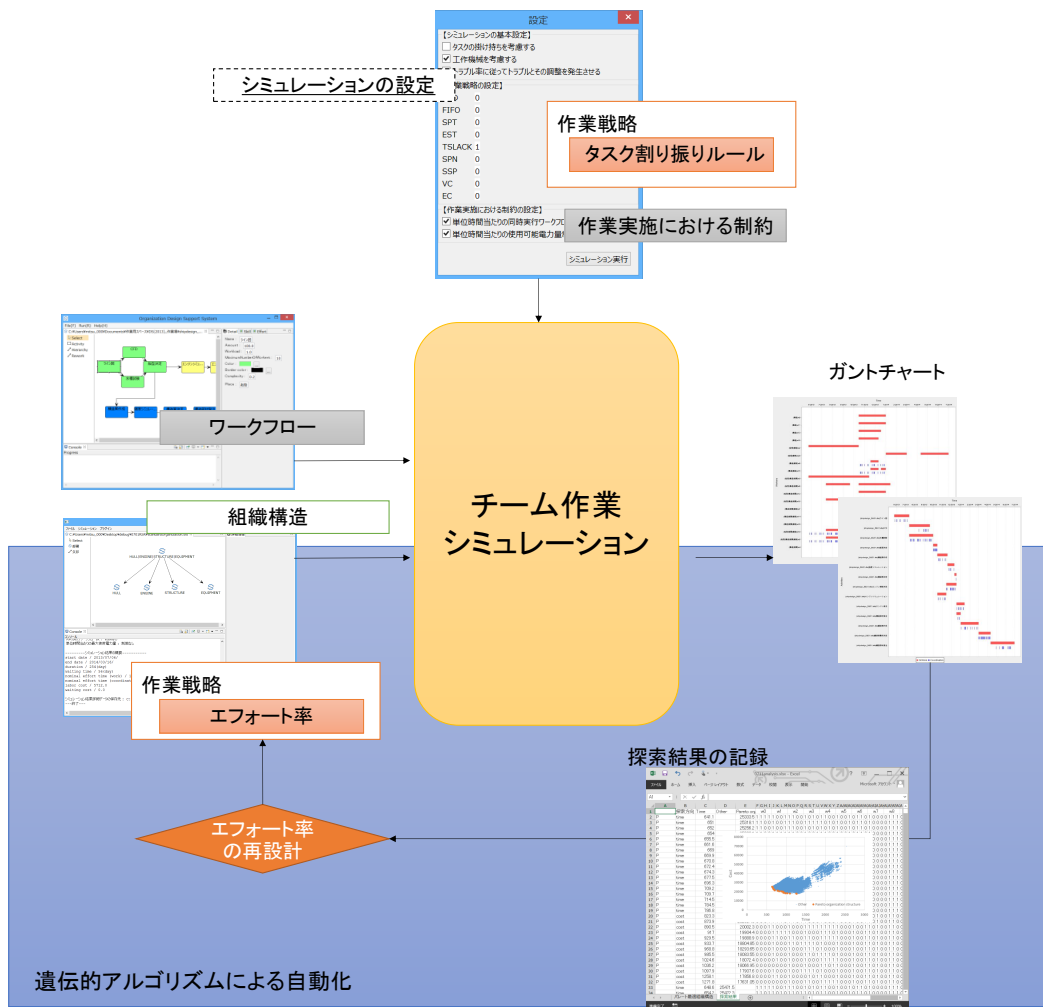


図 4-19 エフォート率の設計支援機能の概要

エフォート率の設計支援機能を実行すると、3.5.3.1 のエフォート率自動探索手法により、作業チーム内の各人員に対して適切なエフォート率を探索する。4.4 と同様に、探索計算は

開発したシステムの画面とは別スレッドで実行され、探索の途中経過は下のコンソールビューに表示される。探索過程で検討したすべてのエフォート率案を採用した際における作業チームのパフォーマンスに関する情報や、複数の目的関数を採用した場合に関しては、採用した目的関数に対してパレートフロントを構成するエフォート率案とその他のエフォート率案を区別する記号などの情報を含んだ情報を図 4-18と同様の形式で出力する。

なお、目的関数として 1 つのみを採用した場合、パレートフロントを構成するエフォート率案は一つしか存在しないため、結果として提案されたエフォート率案の情報は、図 4-20のようにシステム画面の右側のエフォート表に反映される。

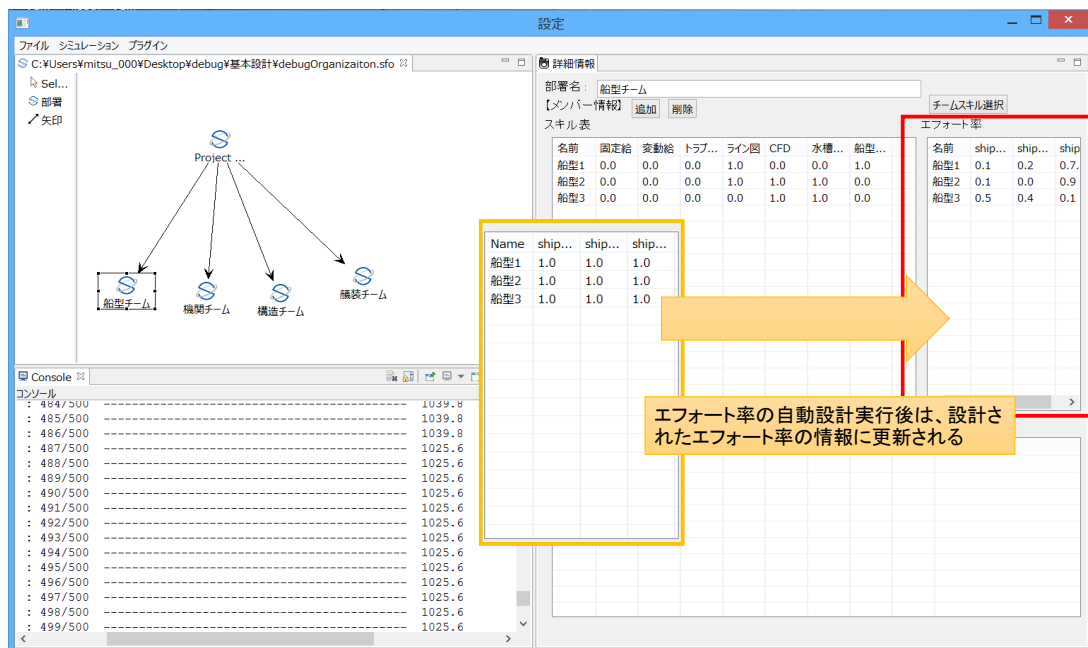


図 4-20 エフォート率の設計支援機能実行後の画面

4.4.2.2 タスク割り振りルールへの設計支援機能

開発した作業チーム設計支援システムでは、3.5.3.2 のタスク割り振りルールの自動探索手法をプラグインの形式で実装した。タスク割り振りルールの設計支援機能の概要を図 4-21 に示す。

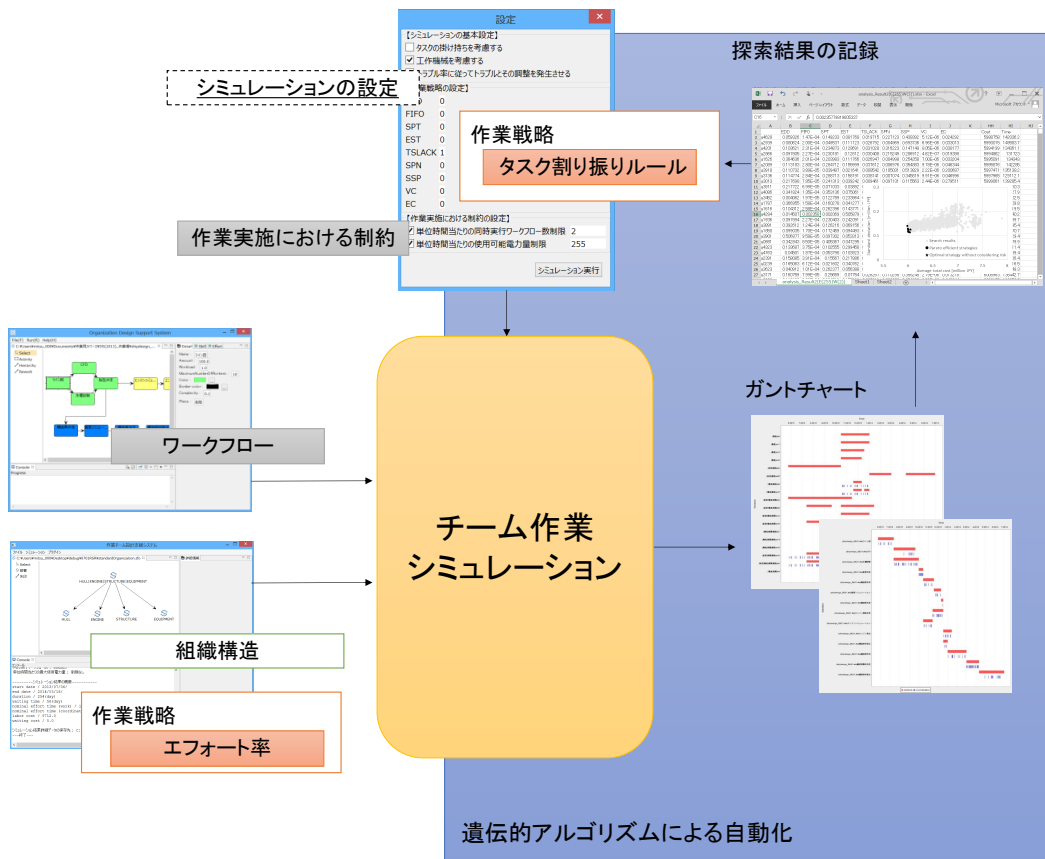


図 4-21 タスク割り振りルールの設計支援機能の概要

タスク割り振りルールの設計支援機能を実行すると、3.5.3.2 のタスク割り振りルールの自動探索手法が行われる。他のプラグイン機能と同様に、計算は開発したシステムの画面とは別スレッドで実行され、探索の途中経過は下のコンソールビューに表示される。計算が終了すると、出力ファイルとして設計されたタスク割り振りルールの情報に関するテキストデータに加えて、図 4-22 のように探索過程で検討したすべてのタスク割り振りルール案を採用した場合における作業チームのパフォーマンス情報や、複数の目的関数を採用した場合に関しては、

採用した目的関数に対してパレートフロントを構成するタスク割り振りルールとその他のタスク割り振りルールを区別する記号などの情報を含んだ情報を、図 3-9 のグラフが作成しやすいように出力する。

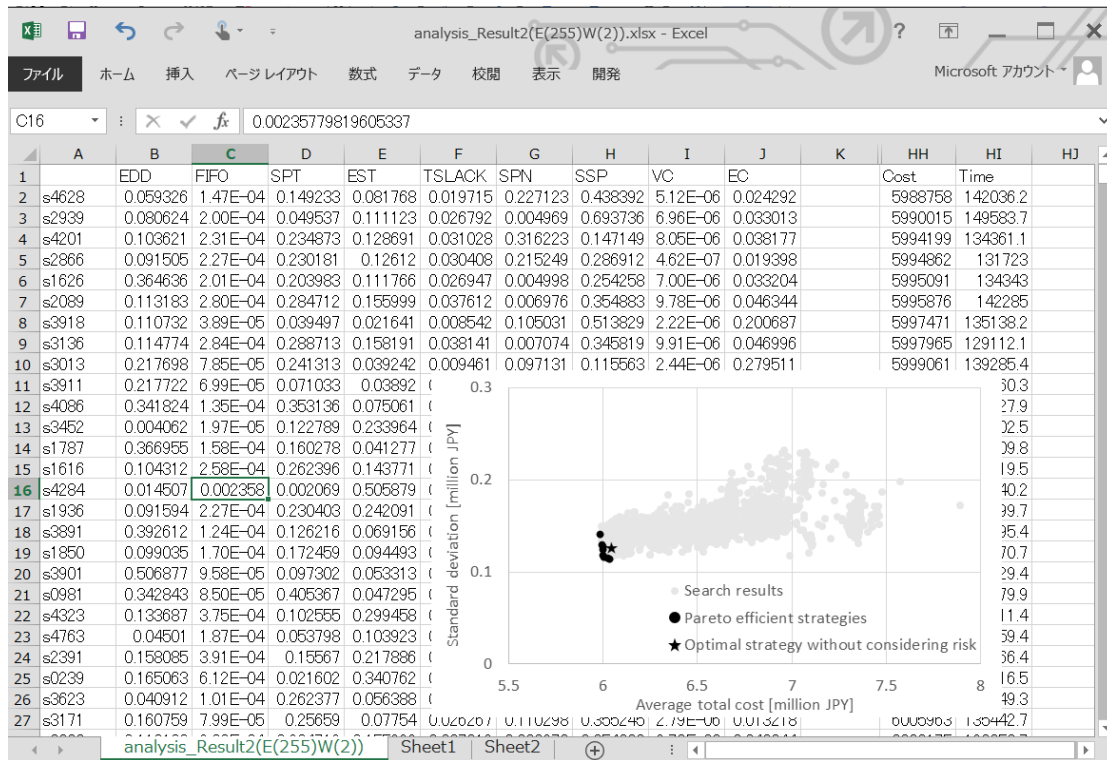


図 4-22 タスク割り振りルール的设计支援機能実行に出力されるファイル

第5章 システム開発チームの組織構造設計

5.1 はじめに	90
5.2 問題設定	90
5.2.1 ケーススタディで想定する状況	90
5.2.2 提案手法の設定	92
5.3 結果	94
5.3.1 初期作業チーム(T1)	94
5.3.2 各部署に加えるべき人員の検討(T2 - T9)	97
5.3.3 部署間の依存関係についての検討(T10 - T12)	100
5.3.4 手動探索の履歴と最終的に設計した作業チーム	106
5.4 まとめ	110

5.1 はじめに

本章では、IT ベンダーにおけるシステム開発業務を対象として、3.4 の手動探索による作業チーム設計手法に基づいて作業チームの設計を行う。なお、本章で使用した入力データに関しては、システム開発工程に関する文献[53]をもとに、仮想的なシステム開発環境を想定して作成したものである。

システム開発業務では、単にプログラム実装だけでなく、顧客のニーズに合わせたシステム設計や実運用に耐えうるテスト作業を行う必要があるため、複数の部署間で連携してプロジェクトを実施することが多い。一方で、一般に IT ベンダーは複数のプロジェクトを抱えているため、あるプロジェクトにおいてシステム開発チームを設計する際には、部署ごとにどの人員を作業チームに加えるかや短期で派遣社員やアルバイトを雇うべきか否かなどの人員選択と配置を含む組織構造を適切に設計する必要がある。

5.2 問題設定

5.2.1 ケーススタディで想定する状況

本章で対象とする IT システム開発業務は、図 5-1 のワークフローで表される。一般に、システム開発業務では、要件定義・設計段階とテスト段階を対応させ、テスト段階での検証や妥当性確認の内容を明確にすることで段階的に品質を確保するための V 字型のウォーターフォール型モデル(V 字モデル)を組み入れて工程を仕立てている。図 5-1 のワークフローは V 字モデルを組み込んだシステム開発工程のうち、「要件定義プロセス」、「開発プロセス」、「運用テスト・移行プロセス」の部分を切り出したものである[52]。なお、図 5-1 の数字は各タスクの作業量とトラブル率を表す。また、各タスクの最大同時担当者数は無制限であるとする。さらに、図 5-1 のワークフロー内の各タスクに定義されている担当部署は、「業務担当」(赤)・「インターフェイス担当」(黄)・「システム担当」(緑)・「プログラム担当」(灰)の 4 つであるとし、それぞれのタスクの担当部署を色で示している。また、作業実施における制約は今回のケー

スタディでは特に設けないものとする。

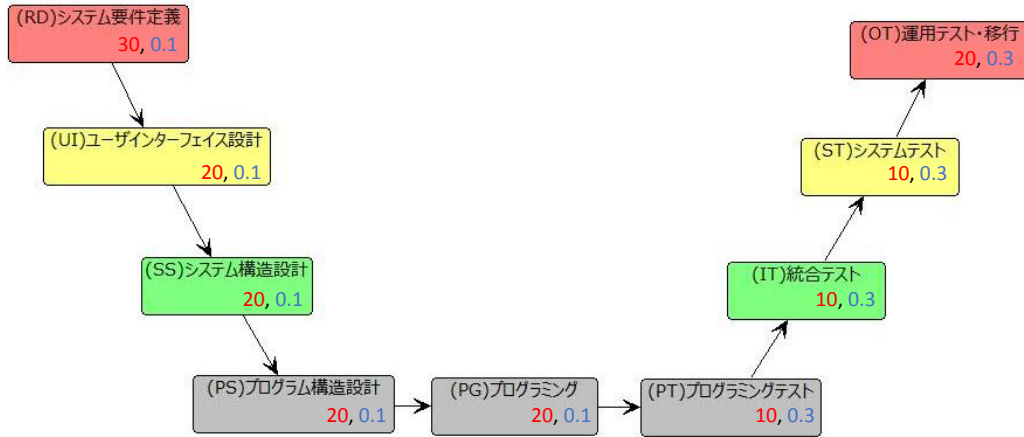


図 5-1 対象とするシステム開発業務を表すワークフロー

作業チーム設計の対象となる組織情報を表 5-1 に示す。組織情報には、1 名のマネージャと 8 名の正社員、そして 9 名のアルバイト人員が登録されている。

表 5-1 IT システム開発チームの候補人員に関する情報

名前	固定給	変動給	トラブル率	(RD)システム要件定義	(UI)ユーザーインターフェイス設計	(SS)システム構造設計	(PS)プログラム構造設計	(PG)プログラミング	(PT)プログラミングテスト	(IT)統合テスト	(ST)システムテスト	(OT)運用テスト・移行
マネージャ	15	0	0.1	1	1	1	1	1	1	1	1	1
業務1	12	0	0.1	1	1						0.5	1
業務2	10	0	0.2	0.5	0.5						0.5	1
インターフェイス1	12	0	0.1	0.5	1	0.5		0.5	0.5	0.5	1	0.5
インターフェイス2	10	0	0.2		0.5				0.5	0.5	1	
システム1	12	0	0.1			1	0.5	1	0.5	1		
システム2	10	0	0.2			0.5		0.5	0.5	1		
プログラム1	12	0	0.1			0.5	1	1	1	0.5		
プログラム2	10	0	0.2				0.5	1	1			
インターフェイスアルバイト1	0	8	0.3								0.5	
インターフェイスアルバイト2	0	8	0.3								0.5	
システムアルバイト1	0	8	0.3							0.5		
システムアルバイト2	0	8	0.3							0.5		
プログラムアルバイト1	0	8	0.3					0.5	0.5			
プログラムアルバイト2	0	8	0.3					0.5	0.5			
プログラムアルバイト3	0	8	0.3					0.5	0.5			
プログラムアルバイト4	0	8	0.3					0.5	0.5			
プログラムアルバイト5	0	8	0.3					0.5	0.5			

また、システム開発チームを設計するにあたって、今回のケーススタディではシステム開発業務プロジェクトを計画する際の初期段階であるものとし、表 5-1 の組織情報から作業チームとして誰を採用するかやどのような配置にするかについて検討する状況を想定している。なお、今回のケーススタディでは、単に作業チームの組織構造を決定するだけでなく、作業チームの組織構造に関する以下の2点について検討を行う。

- 部署間の依存関係は変更せず、どの部署にアルバイト人員を追加すべきか
- 作業チームにおける部署間の依存関係をどのように定義すべきか

5.2.2 提案手法の設定

今回のケーススタディにおける提案手法の設定を表 5-2 に示す。

今回のケーススタディでは作業チームの組織構造を設計の対象とし、3.4 の手動探索によ

る作業チーム設計手法を用いて組織構造の設計を行う。また、目的関数として「所要時間平均」と「合計コスト平均」を採用し、所要時間と合計コストの 2 つの指標を考慮しながら、組織構造の設計を手動探索により行う。

また、チーム作業シミュレーションの設定に関しては、「タスクの掛け持ちを考慮する」、「工作機械は考慮しない」、「トラブル率に従ってトラブルとその調整を発生させる」とし、チーム作業シミュレーションの繰り返し回数は 100 とした。

表 5-2 第 5 章における提案手法の設定

	設定項目	設定内容
探索の設定	探索手法	手動探索(3.4)
	設計変数	組織構造
	目的関数	<ul style="list-style-type: none"> ● 所要時間平均 ● 合計コスト平均
チーム作業シミュレーションの設定	各人員の同時刻におけるタスクの掛け持ち	あり
	タスク実行における工作機械の考慮	なし
	各タスクでランダムに発生するトラブルとその調整の考慮	あり
	チーム作業シミュレーションの繰り返し回数	100

今回のケーススタディにおける作業チームの設計変数と固定する変数の詳細に関する情報を表 5-3 に示す。設計変数として組織構造を採用し、作業戦略を固定した状態で組織構造の探索を行う。

作業戦略について、タスク割り振りルールに関しては、クリティカルパス上に近いタスクを優先して行うディスパッチングルールである TSLACK のみを採用する。すなわち、タスク割り振

りルールの重みベクトル成分のうち, **TSLACK** の成分を 1, そのほかの成分を 0 とする. また, 対象とするワークフローは 1 つであるため各人員のエフォート率は図 5-1 のワークフローに対して 1.0 と設定する.

表 5-3 第 5 章における作業チームの設計変数と固定する変数

	変数	詳細
組織構造	人員と工作機械のスキルセット	設計変数
	人員間の依存関係	設計変数
作業戦略	タスク割り振りルール	固定(TSLACK のみ)
	エフォート率	固定(すべての人員が図 5-1 のワークフローに対して 1.0)

5.3 結果

5.3.1 初期作業チーム(T1)

作業チームの設計にあたり, 最初に設計した作業チーム(T1)の組織構造を図 5-2 に, 作業チーム(T1)内の人員に関する情報を表 5-4 に示す. 図 5-2 で示す作業チーム(T1)の組織構造は最も単純な組織構造の一つであり, 表 5-4 で示すように各部署には正社員のみが登録されている. また, 表 5-4 の黒で塗りつぶした部分は, 各人員の所属する部署の権限によってスキルが制限された結果, チーム作業シミュレーション上ではスキル値が 0 と考慮される部分である.

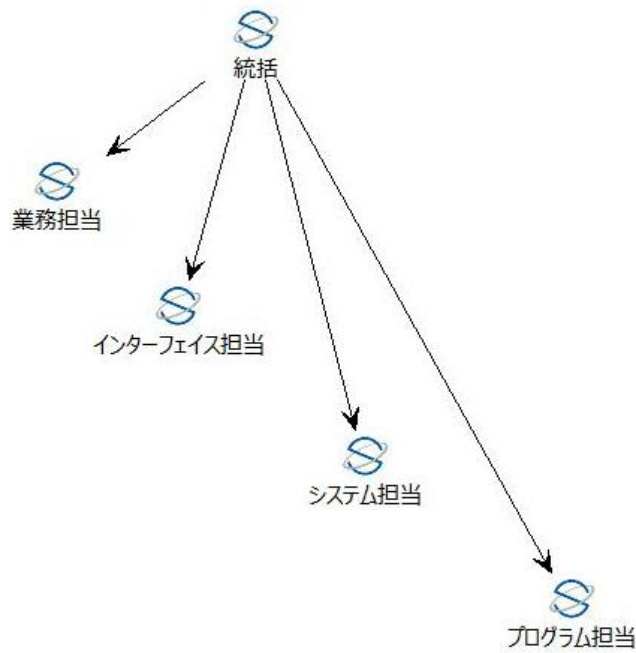


図 5-2 初期作業チーム(T1)の組織構造

表 5-4 初期作業チーム(T1)内の人員に関する情報

所属	名前	固定給	変動給	トラブル率	(RD)システム要件定義	(UI)ユーザーインターフェイス設計	(SS)システム構造設計	(PS)プログラム構造設計	(PG)プログラミング	(PT)プログラミングテスト	(IT)統合テスト	(ST)システムテスト	(OT)運用テスト・移行
統括	マネージャ	15	0	0.1	1	1	1	1	1	1	1	1	1
業務担当	業務1	12	0	0.1	1	1						0.5	1
	業務2	10	0	0.2	0.5	0.5						0.5	1
インターフェイス担当	インターフェイス1	12	0	0.1	0.5	1	0.5		0.5	0.5	0.5	1	0.5
	インターフェイス2	10	0	0.2		0.5				0.5	0.5	1	
システム担当	システム1	12	0	0.1			1	0.5	1	0.5	1		
	システム2	10	0	0.2			0.5	0	0.5	0.5	1		
プログラム担当	プログラム1	12	0	0.1			0.5	1	1	1	0.5		
	プログラム2	10	0	0.2				0.5	1	1			

100 回のチーム作業シミュレーションを繰り返した結果、所要時間の平均値が 69.05、合計コストの平均値が 4807.87 という結果になった。100 回のチーム作業シミュレーションのうち、1 回分のチーム作業シミュレーション結果として出力されたガントチャートを図 5-3 に示す。図 5-3 では、縦軸にそれぞれの人員の略称を、横軸に時間を取り、それぞれの人員が何らかの

タスクを担当している時間帯を赤色で、調整を担当している場合を青色で塗りつぶしたガントチャートを示している。初期作業チーム(T1)では、調整を行える上司はマネージャだけであるため、図 5-3 ではマネージャのみが調整を行っていることが分かる。

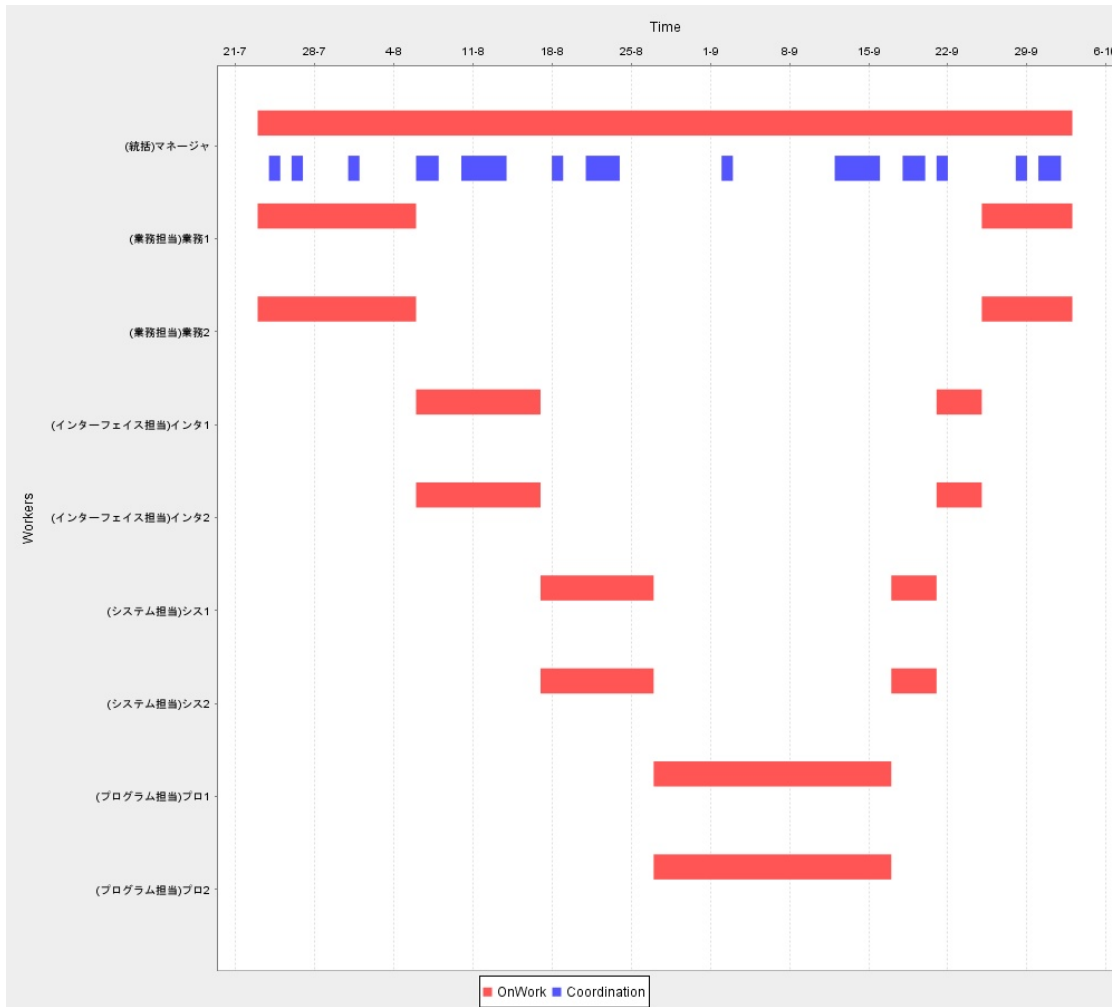


図 5-3 作業チーム(T1)のシミュレーション結果例

5.3.2 各部署に加えるべき人員の検討(T2 – T9)

次に、図 5-2 で示した初期作業チーム(T1)の組織構造を変更せずに、どの部署にアルバイト人員を何人追加すべきかについて検討することを考える。

まず、図 5-3 よりプログラム担当の部署に所属している人員の作業時間を減らすことが作業チームのパフォーマンス向上につながるのではと考え、プログラム担当の部署にアルバイト人員(プログラムアルバイト 1～プログラムアルバイト 6)を一人ずつ追加した作業チーム(T2～T7)を設計し、設計した作業チーム案のパフォーマンスを評価することを繰り返し行った。作業チーム T2 から T7 における 1 回分のチーム作業シミュレーションの結果として出力されたガントチャートを図 5-4 に、作業チーム T7 までのパフォーマンスの変化の様子を図 5-5 に示す。図 5-5 では、縦軸にチーム作業シミュレーション結果の合計コスト平均を、横軸にチーム作業シミュレーションの所要時間平均を取っている。また、グラフ上の点は各作業チームのパフォーマンス評価結果を表しており、右上の点が初期作業チーム T1 を、T1 と結ばれている点が作業チーム T2 を、という流れで順々に線をつないでいる。図 5-5 より、T1 から T7 のすべてに関して、一つ前の作業チームよりも所要時間平均と合計コスト平均の両方が減少していることが分かる。このことから、プログラム担当の部署にアルバイト人員を追加することは作業チームのパフォーマンス向上に効果があることが分かる。



図 5-4 作業チーム(T2 - T7)のチーム作業シミュレーション結果例
(上 : T2, T3, T4, 下 : T5, T6, T7)

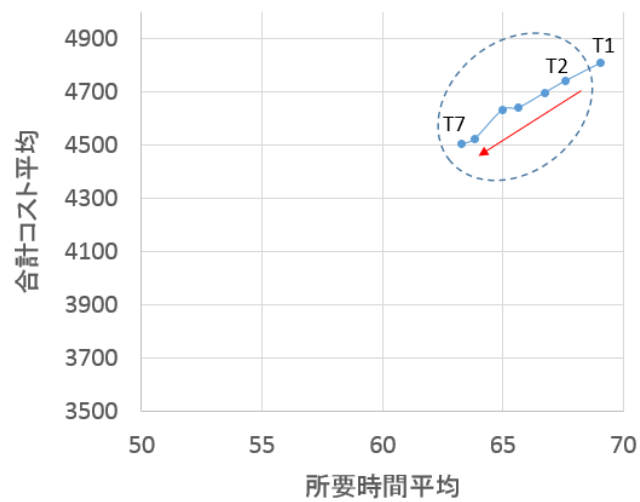


図 5-5 作業チーム(T7)までのパフォーマンス変化

続いて、作業チーム T7 にシステム担当の部署にアルバイト人員を 1 人追加した場合の作業チーム T8 と、作業チーム T8 にインターフェイス担当の部署にアルバイト人員を 1 人追加した場合の作業チーム T9 について検討を行った。作業チーム T8 と T9 における 1 回分のチーム作業シミュレーションの結果として出力されたガントチャートを図 5-6 に、作業チーム T9 までのパフォーマンスの変化の様子を図 5-7 に示す。図 5-7 では、作業チーム T8 と作業チーム T9 は一つ前の作業チームに比べて所要時間平均が少し減少している一方で、合計コスト平均が少し増加していることが分かる。このことから、システム担当の部署とインターフェイス担当の部署にアルバイト人員を加えることは、一概に作業チームのパフォーマンス向上につながるとは言えず、コストをかけて所要時間を短縮したいという場合のオプションと捉えることができる。

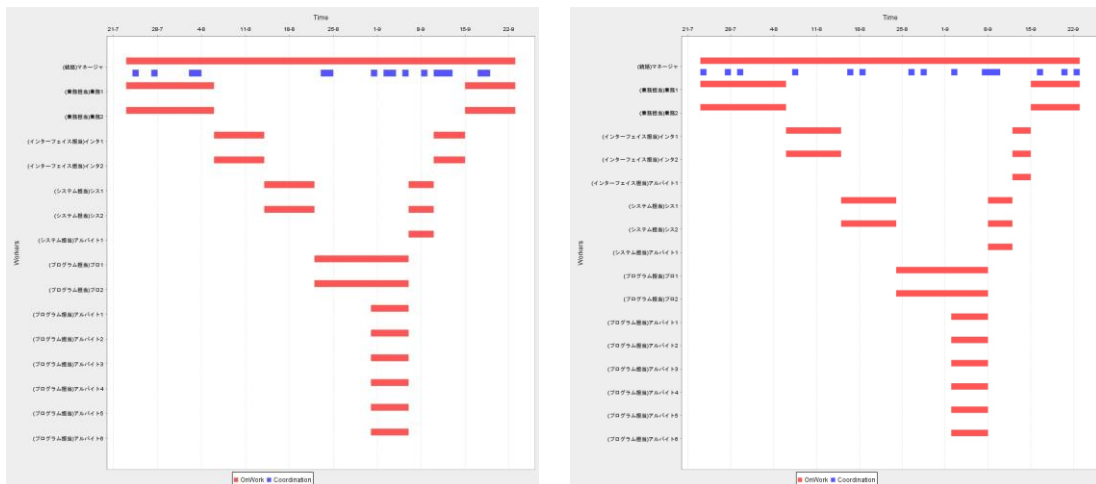


図 5-6 作業チーム(T8- T9)のチーム作業シミュレーション結果例

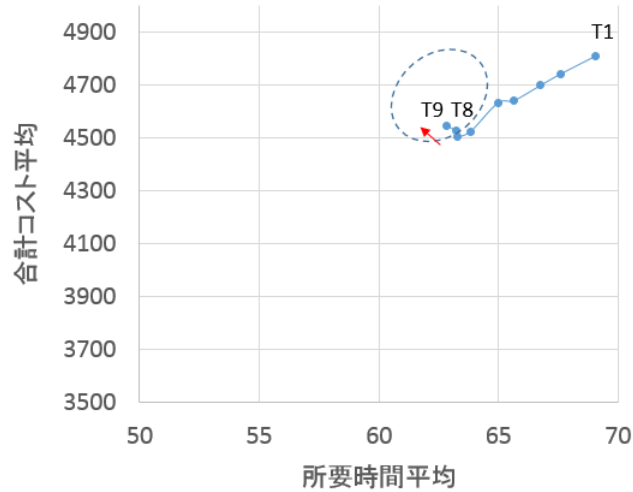


図 5-7 作業チーム(T9)までのパフォーマンス変化

5.3.3 部署間の依存関係についての検討(T10 - T12)

本節では、作業チーム T9 における各部署の人員は変更せず、各部署の依存関係を変更することを考える。

まず、プログラム担当の部署でのトラブルに対して調整を行う部署を、統括だけでなくシステム担当にも担当させるような組織構造を持つ作業チーム T10 を考える。図 5-8 に作業チーム T10 の組織構造を、表 5-5 に組織構造の変更に対応した作業チーム T10 内の人員に対する情報を示す。表 5-5 の黒で塗りつぶした部分と表 5-4 の黒で塗りつぶした部分には異なる部分が存在する。これは、システム担当の部署からプログラム担当の部署へ矢印が引かれたことで、システム担当の部署に所属する人員がプログラム担当の部署が担当するタスクも実行可能になったためである。

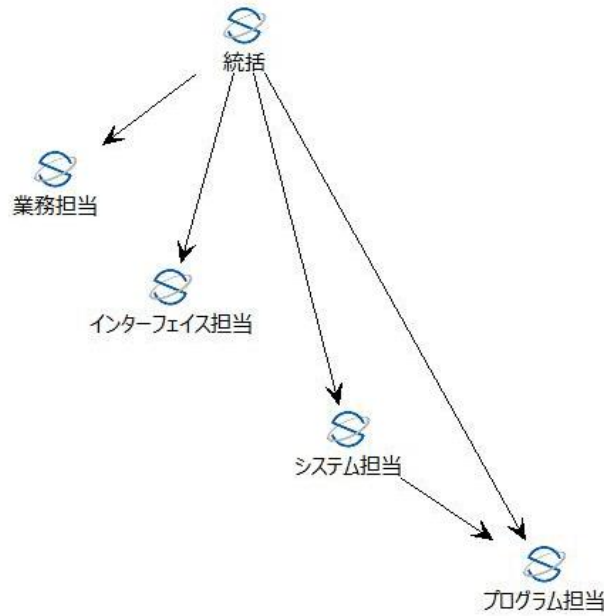


図 5-8 作業チーム(T10)の組織構造

表 5-5 作業チーム(T10)内の人員に関する情報

所属	名前	固定給	変動給	トラブル率	(RD)システム要件定義	(UI)ユーザーインターフェイス設計	(SS)システム構造設計	(PS)プログラム構造設計	(PG)プログラミング	(PT)プログラミングテスト	(IT)統合テスト	(ST)システムテスト	(OT)運用テスト・移行
統括	マネージャ	15	0	0.1	1	1	1	1	1	1	1	1	1
業務担当	業務1	12	0	0.1	1	1						0.5	1
	業務2	10	0	0.2	0.5	0.5						0.5	1
インターフェイス担当	インターフェイス1	12	0	0.1		1	0.5		0.5	0.5	0.5	1	0.5
	インターフェイス2	10	0	0.2		0.5				0.5	0.5	1	
	インターフェイスアルバイト1	0	8	0.3								0.5	
システム担当	システム1	12	0	0.1			1	0.5	1	0.5	1		
	システム2	10	0	0.2			0.5		0.5	0.5	1		
	システムアルバイト1	0	8	0.3							0.5		
プログラム担当	プログラム1	12	0	0.1			0.5	1	1	1	0.5		
	プログラム2	10	0	0.2				0.5	1	1			
	プログラムアルバイト1	0	8	0.3					0.5	0.5			
	プログラムアルバイト2	0	8	0.3					0.5	0.5			
	プログラムアルバイト3	0	8	0.3					0.5	0.5			
	プログラムアルバイト4	0	8	0.3					0.5	0.5			
	プログラムアルバイト5	0	8	0.3					0.5	0.5			

100 回のチーム作業シミュレーションのうち、1 回分のチーム作業シミュレーション結果として出力されたガントチャートを図 5-9 に示す。図 5-3 と比較すると、システム担当の部署に所属する正社員が、スキルを持つプログラム担当のタスクに対して作業や調整を行っていることが分かる。

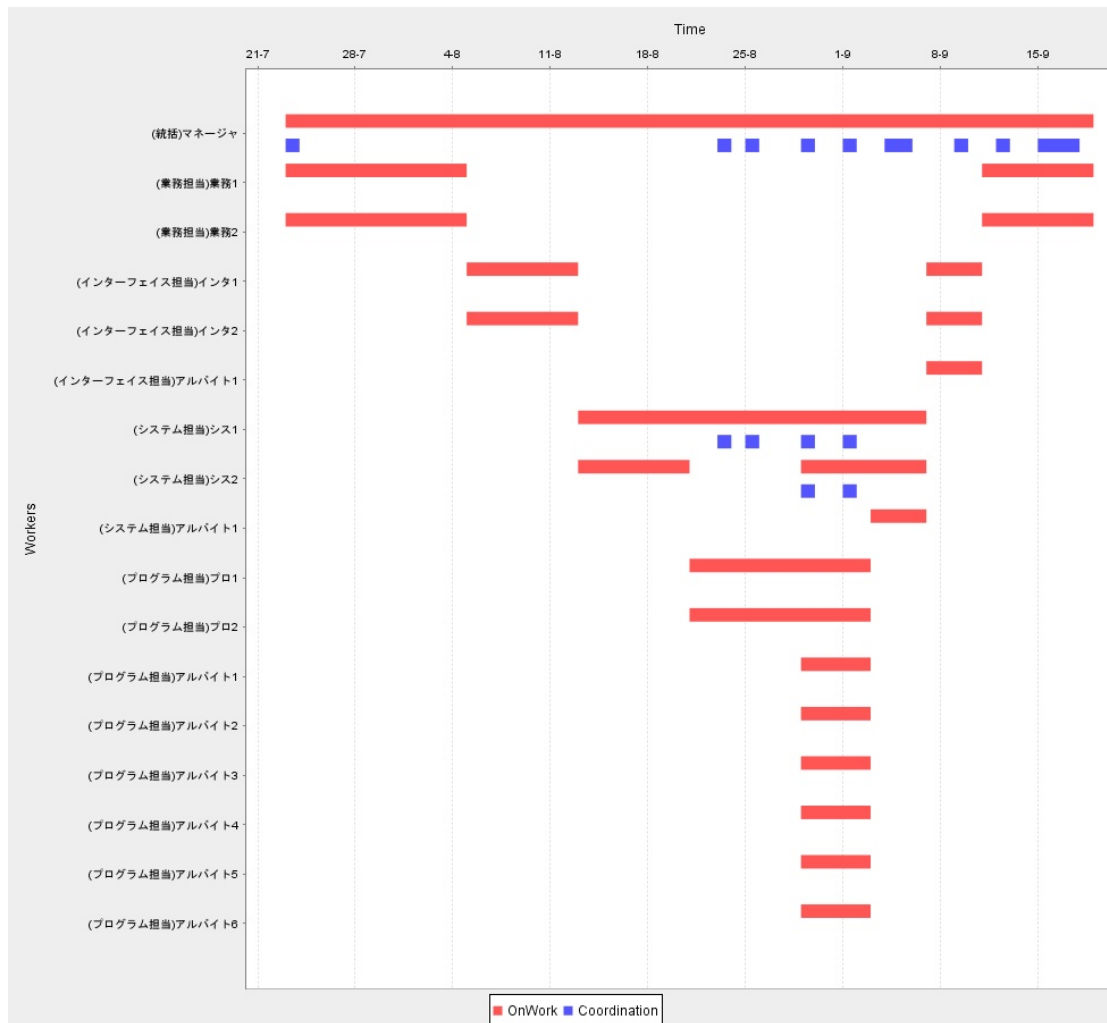


図 5-9 作業チーム(T10)のチーム作業シミュレーション結果例

作業チーム T10 までのパフォーマンスの変化の様子を図 5-10 に示す。図 5-10 では、5.3.2 の各ステップで行った人員の追加に比べて、所要時間平均と合計コスト平均の両方が大きく減少していることが分かる。このことから、プログラム担当の部署とシステム担当の部署

に依存関係を結ぶことは、作業チームのパフォーマンス向上に効果があることが分かる。

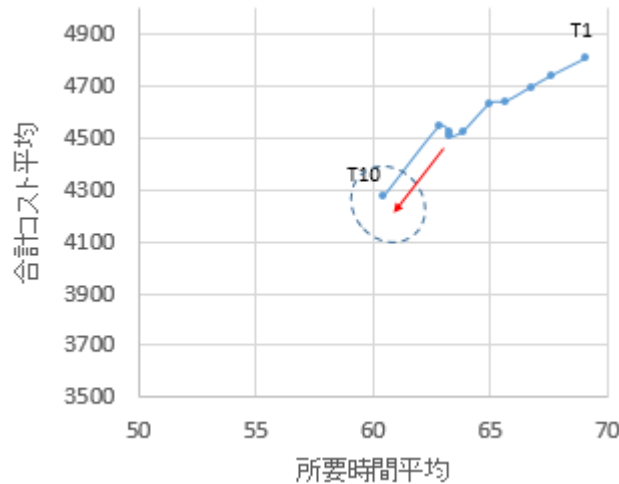


図 5-10 作業チーム(T10)までのパフォーマンス変化

次に、インターフェイス担当の部署に新たな依存関係を加えることを考える。具体的には、作業チーム T10 にインターフェイス担当の部署とシステム担当の部署間の依存関係を加えた作業チーム T11 と、作業チーム T10 にインターフェイス担当の部署とプログラム担当の部署間の依存関係を加えた作業チーム T12 について検討を行う。作業チーム T11 と T12 の組織構造を図 5-11 に、作業チーム T11 と T12 の人員に関する情報を表 5-6 と表 5-7 に示す。新しく依存関係を定義したことによって、表 5-5 の情報に加えて、作業チーム T11 ではインターフェイス担当の部署に所属する人員に関してシステム担当のスキルが使用可能になり、作業チーム T12 ではインターフェイス担当の部署に所属する人員に関してプログラム担当のスキルが使用可能になる。

作業チーム T11 と作業チーム T12 に関して、100 回のチーム作業シミュレーションのうち、1 回分のチーム作業シミュレーション結果として出力されたガントチャートを図 5-12 に示す。図 5-12 の左図が作業チーム T11 をもとに出力されたガントチャートを、右図が作業チーム T12 をもとに出力されたガントチャートを示している。左図では、インターフェイス担当の部署に所属している人員がシステム担当のタスクのスキルを持っている場合に、システム担当の部署に所属している人員と一緒に作業を行っていることが分かる。一方で、右図では、イン

ターフェイス担当の部署に所属している人員がプログラム担当のタスクのスキルを持っている場合に、プログラム担当の部署に所属している人員と一緒に作業を行っていることが分かる。

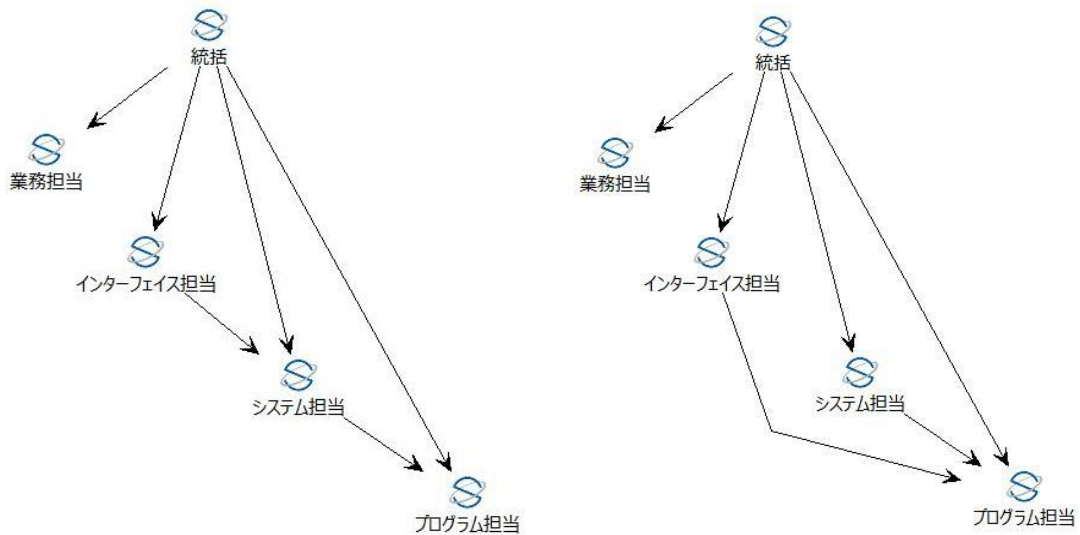


図 5-11 作業チーム(T11 と T12)の組織構造
(左 : T11, 右 : T12)

表 5-6 作業チーム(T11)内の人員に関する情報

所属	名前	固定給	変動給	トラブル率	(RD)システム要件定義	(UI)ユーザーインターフェイス設計	(SS)システム構造設計	(PS)プログラム構造設計	(PG)プログラミング	(PT)プログラムテスト	(IT)統合テスト	(ST)システムテスト	(OT)運用テスト・移行
統括	マネージャ	15	0	0.1	1	1	1	1	1	1	1	1	1
業務担当	業務1	12	0	0.1	1	1						0.5	1
	業務2	10	0	0.2	0.5	0.5						0.5	1
インターフェイス担当	インターフェイス1	12	0	0.1	0.5	1	0.5		0.5	0.5	0.5	1	0.5
	インターフェイス2	10	0	0.2		0.5				0.5	0.5	1	1
	インターフェイスアルバイト1	0	8	0.3								0.5	
システム担当	システム1	12	0	0.1			1	0.5	1	0.5	1		
	システム2	10	0	0.2			0.5		0.5	0.5	1		
	システムアルバイト1	0	8	0.3							0.5		
プログラム担当	プログラム1	12	0	0.1			0.5	1	1	1	0.5		
	プログラム2	10	0	0.2				0.5	1	1			
	プログラムアルバイト1	0	8	0.3					0.5	0.5			
	プログラムアルバイト2	0	8	0.3					0.5	0.5			
	プログラムアルバイト3	0	8	0.3					0.5	0.5			
	プログラムアルバイト4	0	8	0.3					0.5	0.5			
プログラムアルバイト5	0	8	0.3					0.5	0.5				

表 5-7 作業チーム(T12)内の人員に関する情報

所属	名前	固定給	変動給	トラブル率	(RD)システム要件定義	(UI)ユーザーフェイス設計	(SS)システム構造設計	(PS)プログラム構造設計	(PG)プログラミング	(PT)プログラミングテスト	(IT)統合テスト	(ST)システムテスト	(OT)運用テスト・移行
統括	マネージャ	15	0	0.1	1	1	1	1	1	1	1	1	1
業務担当	業務1	12	0	0.1	1	1						0.5	1
	業務2	10	0	0.2	0.5	0.5						0.5	1
インターフェイス担当	インターフェイス1	12	0	0.1	0.5	1	0.5		0.5	0.5	0.5	1	0.5
	インターフェイス2	10	0	0.2		0.5				0.5	0.5	1	
	インターフェイスアルバイト1	0	8	0.3								0.5	
システム担当	システム1	12	0	0.1			1	0.5	1	0.5	1		
	システム2	10	0	0.2			0.5		0.5	0.5	1		
	システムアルバイト1	0	8	0.3							0.5		
プログラム担当	プログラム1	12	0	0.1			0.5	1	1	1	0.5		
	プログラム2	10	0	0.2				0.5	1	1			
	プログラムアルバイト1	0	8	0.3					0.5	0.5			
	プログラムアルバイト2	0	8	0.3					0.5	0.5			
	プログラムアルバイト3	0	8	0.3					0.5	0.5			
	プログラムアルバイト4	0	8	0.3					0.5	0.5			
プログラムアルバイト5	0	8	0.3					0.5	0.5				

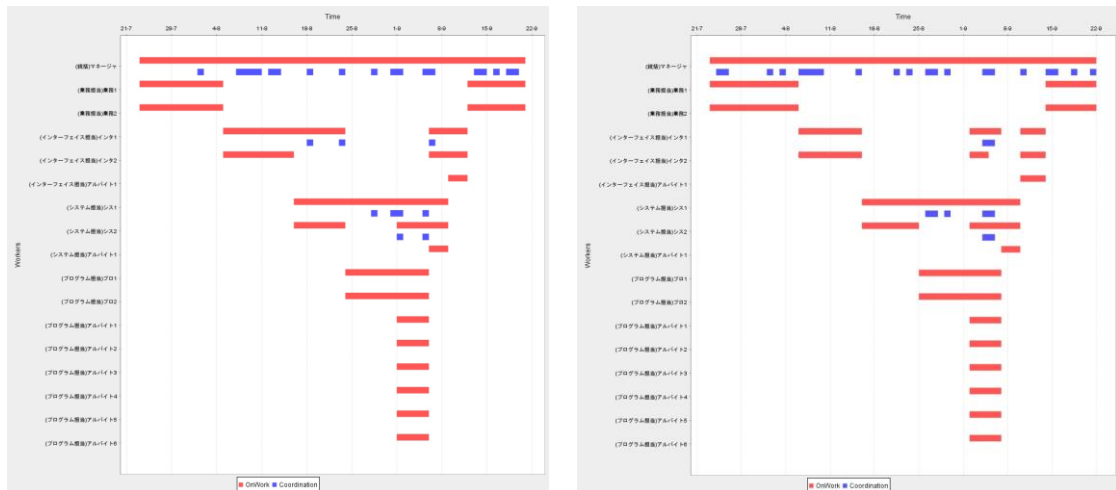


図 5-12 作業チーム(T11 と T12)のチーム作業シミュレーション結果例
(左 : T11, 右 : T12)

作業チーム T12 までのパフォーマンスの変化の様子を図 5-13 に示す。図 5-13 より、作

作業チーム T11 は作業チーム T10 に比べて所要時間平均と合計コスト平均の両方が減少している一方で、作業チーム T12 は作業チーム T10 に比べてわずかではあるが所要時間平均と合計コスト平均の両方が増加していることが分かる。このことから、インターフェイス担当の部署とシステム担当の部署に依存関係を結ぶことは作業チームのパフォーマンス向上に効果がある一方で、インターフェイス担当の部署とプログラム担当の部署に依存関係を結ぶことは作業チームのパフォーマンス向上には逆効果であることが分かる。

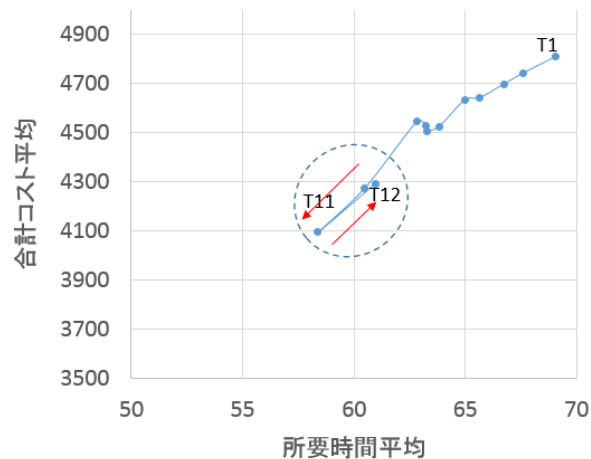


図 5-13 作業チーム(T12)までのパフォーマンス変化

5.3.4 手動探索の履歴と最終的に設計した作業チーム

作業チーム T12 をもとに、さらに組織構造を変更することで最終的な作業チームの設計を行った。今回のケーススタディでは、開発した作業チーム設計支援システムの機能の一つである 4.3 の手動探索による作業チーム設計支援機能を用いて作業チーム T1 から開始した探索の履歴を記録している。今回のケーススタディにおける組織構造の探索履歴を表 5-8 に示す。表 5-8 では、単に作業チームごとのパフォーマンス評価結果が記述されているだけでなく、メモによってどのように作業チームを変更したかに関連付けて記録されているため、どのような操作が作業チームのパフォーマンス向上に効果があったのかを容易に確認することが可能である。

表 5-8 組織構造の手動探索履歴

	日付	所要時間	合計コスト	メモ
T1	2013/7/23 15:55	69.05	4807.87	初期状態
T2	2013/7/23 15:56	67.58	4739.96	プログラム担当にアルバイトを一人追加
T3	2013/7/23 15:58	66.74	4696.94	更にもう一人追加
T4	2013/7/23 15:59	65.65	4640.83	更にもう一人追加
T5	2013/7/23 16:00	64.97	4631.29	更にもう一人追加
T6	2013/7/23 16:01	63.83	4523.85	更にもう一人追加
T7	2013/7/23 16:03	63.29	4505.01	更にもう一人追加
T8	2013/7/23 16:05	63.23	4526.99	システム担当にアルバイトを一人追加
T9	2013/7/23 16:07	62.83	4544.79	インターフェイス担当にアルバイトを一人追加
T10	2013/7/23 16:24	60.47	4271.99	システム担当にプログラム担当のスキルも追加した
T11	2013/7/23 16:30	58.38	4092.26	インターフェイス担当にシステム担当のスキルを追加
T12	2013/7/23 16:34	60.95	4290.97	インターフェイス担当にプログラム担当のスキルを追加&システム担当スキルを削除
T13	2013/7/23 16:38	54.47	3861.91	一個前に戻る+業務担当にインターフェイス担当のスキルを追加
T14	2013/7/23 16:39	54.91	3884.77	統括とインターフェイス担当の上下関係を削除
T15	2013/7/23 16:41	54.80	3885.56	一個前に戻る+統括とシステム担当の上下関係を削除
T16	2013/7/23 16:44	54.46	3852.70	結局3個前に戻る

最終的に設計された作業チーム T16 の組織構造を図 5-14 に、作業チーム T16 の人員に関する情報を表 5-9 に示す。また、作業チーム T2 から T7 における 1 回分のチーム作業シミュレーションの結果として出力されたガントチャートを図 5-15 に、作業チーム T1 から開始し、設計作業を繰り返しおけるパフォーマンスの変化を図 5-16 に示す。作業チーム T13 から作業チーム T16 において、統括の部署とその他の部署の依存関係をひとつずつ削除した場合のパフォーマンスの変化について検証した結果、いずれの場合もわずかではあるが所要時間平均と合計コスト平均のどちらも上昇するという結果になり、結果として図 5-14 の組織構造が最も良いと判断した。

今回のケーススタディでは、組織構造の違いによって作業チームのパフォーマンスが大きく変化する傾向が見られた。5.3.1 と 5.3.2 で検討した作業チーム(T1 – T9)に関しては統括部署に所属しているマネージャのみが調整作業を行える状態となっていた。一方で、5.3.3 で検討した組織構造(T10 – T11)では、依存関係を追加したことで作業チーム全体で使用可能なスキルの合計が増加しただけでなく、マネージャのみが行っていた調整作業を他の人員に分担しているために大きく作業チームのパフォーマンスが向上したと言える。ただし、作業チーム T12 に関しては、インターフェイス担当部署とプログラム担当部署に依存関係を結んだところ、わずかではあるが依存関係を結ぶ前の作業チーム T10 に比べてパフォーマンスが低下する結果になったが、これは既にプログラムに関するタスクを行う人員の数が十分であること、結果としてインターフェイス部署に所属する正社員のアイドル時間は変化していないことが原因として考えられる。

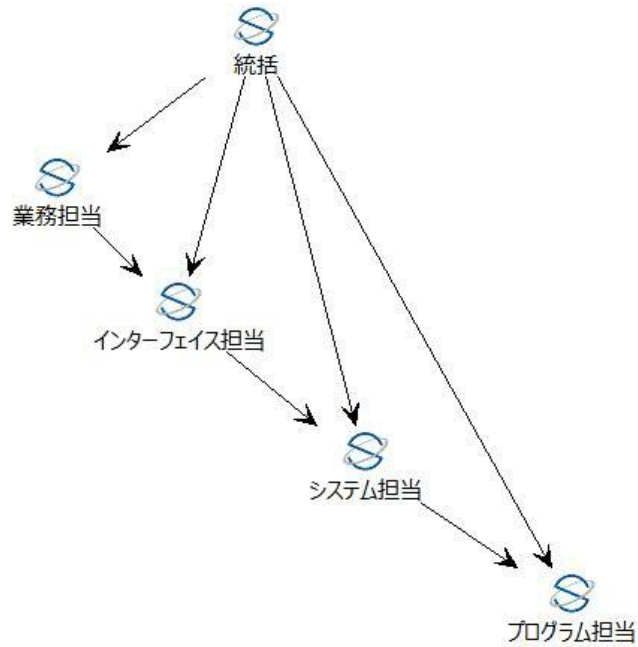


図 5-14 最終的に設計された作業チーム(T16)

表 5-9 作業チーム(T16)内の人員に関する情報

所属	名前	固定給	変動給	トラブル率	(RD)システム要件定義	(UI)ユーザーインターフェイス設計	(SS)システム構造設計	(PS)プログラム構造設計	(PG)プログラミング	(PT)プログラミングテスト	(IT)統合テスト	(ST)システムテスト	(OT)運用テスト・移行
統括	マネージャ	15	0	0.1	1	1	1	1	1	1	1	1	1
業務担当	業務1	12	0	0.1	1	1						0.5	1
	業務2	10	0	0.2	0.5	0.5						0.5	1
インターフェイス担当	インターフェイス1	12	0	0.1	0.5	1	0.5		0.5	0.5	0.5	1	0.5
	インターフェイス2	10	0	0.2		0.5				0.5	0.5	1	
	インターフェイスアルバイト1	0	8	0.3								0.5	
システム担当	システム1	12	0	0.1			1	0.5		1	0.5	1	
	システム2	10	0	0.2			0.5		0.5	0.5	1		
	システムアルバイト1	0	8	0.3							0.5		
プログラム担当	プログラム1	12	0	0.1			0.5	1	1	1	0.5		
	プログラム2	10	0	0.2				0.5	1	1			
	プログラムアルバイト1	0	8	0.3					0.5	0.5			
	プログラムアルバイト2	0	8	0.3					0.5	0.5			
	プログラムアルバイト3	0	8	0.3					0.5	0.5			
	プログラムアルバイト4	0	8	0.3					0.5	0.5			
	プログラムアルバイト5	0	8	0.3					0.5	0.5			

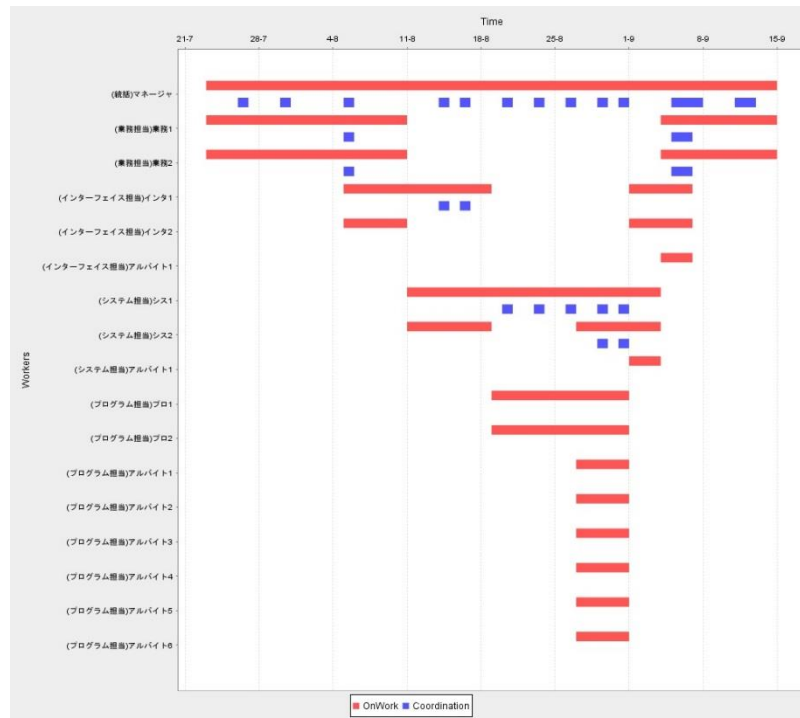


図 5-15 作業チーム(T16)のチーム作業シミュレーション結果例

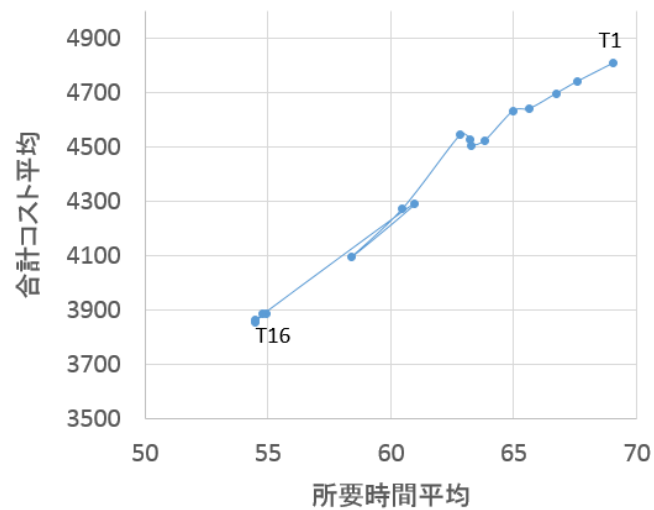


図 5-16 設計作業の繰り返しによるパフォーマンスの変化

5.4 まとめ

本章では3.4で提案した手動探索による作業チーム設計手法に関して、4.3の手動探索による作業チーム設計支援機能を利用してシステム開発チーム設計のケーススタディに適用することで、その評価を行った。本章のケーススタディを通して得られた知見を以下にまとめる。

- 提案手法によって、組織構造を固定した状態でどの部署にどの人員を加えるべきかを定量的に評価・決定することが可能である。
 - ▶ ケーススタディでは、プログラム担当の部署にアルバイト人員を加えると所要時間平均と合計コスト平均の両方が減少する一方で、その他の部署にアルバイト人員を加えると所要時間平均が減少する一方で合計コスト平均が増加する傾向が見られた。
- 作業チームの組織構造の違いはパフォーマンスに大きな影響を与える。特に、組織構造内での部署間の依存関係をやみくもに追加することがパフォーマンスに悪影響を及ぼすこともあることが確認できた。
 - ▶ ケーススタディでは、単一の分岐なしワークフローを対象としたため、タスクの前後関係に基づいて組織構造の依存関係を定義することがパフォーマンス向上につながる傾向が見られた。
- 設計変更のたびに記録するメモを蓄積することで、作業チーム設計支援システム上の手動探索履歴から容易に今回の分析を行うことが可能であることを確認できた。

第6章 造船設計チームの組織構造設計

6.1 はじめに	112
6.2 問題設定	112
6.2.1 ケーススタディで想定する状況	112
6.2.2 提案手法の設定	115
6.3 結果	119
6.3.1 2つのワークフローを対象とした場合	119
6.3.2 3つのワークフローを対象とした場合	129
6.4 まとめ	142

6.1 はじめに

本章では、造船所における仮想的な船舶基本設計プロセスを対象として、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法を用いて、船舶基本設計チームの組織構造設計のケーススタディを行う。なお、本章で使用した入力データに関しては、船舶設計に関する文献[54]や造船所でのインタビュー結果をもとに、仮想的な船舶基本設計プロセスを想定して作成したものである。

船舶基本設計プロセスは、計算や実験による船型やエンジンの決定、構造案や艤装案の決定などの、建造する船の骨格となる部分を設計するプロセスである。新しい船型を設計するときなどは、設計者間のすり合わせや船主の要望の変化などの様々なトラブルが起こることが予想される。そのため、船舶基本設計チームを設計する際には、単なる人員の選択だけでなく、様々なトラブルに対応するための上司や上流工程の担当者との調整を含めた作業チームの組織構造を考慮する必要がある。

6.2 問題設定

6.2.1 ケーススタディで想定する状況

本章で対象とする設計業務は、図 6-1 のワークフローで表される造船設計プロジェクトを対象とする。本章では、図 6-1 のワークフローで表される造船設計プロジェクトを 2 つあるいは 3 つ行うケーススタディを扱うが、それぞれの船舶基本設計プロジェクトは同一の案件ではなく、ある 1 つの船型を異なる 3 つの客先それぞれの要望に合うようにカスタマイズして製造する状況を想定している。図 6-1 の数字は各タスクの作業量とトラブル率を表す。また、各タスクの最大同時担当者数は無制限であるとする。さらに、今回対象とするワークフローでは、「HULL」(緑)、「ENGINE」(黄)、「STRUCTURE」(青)、「EQUIPMENT」(赤)の 4 つの担当部署が存在しているものとし、それぞれのタスクの担当部署を色で示している。また、作業実施における制約は今回のケーススタディでは特に設けないものとする。

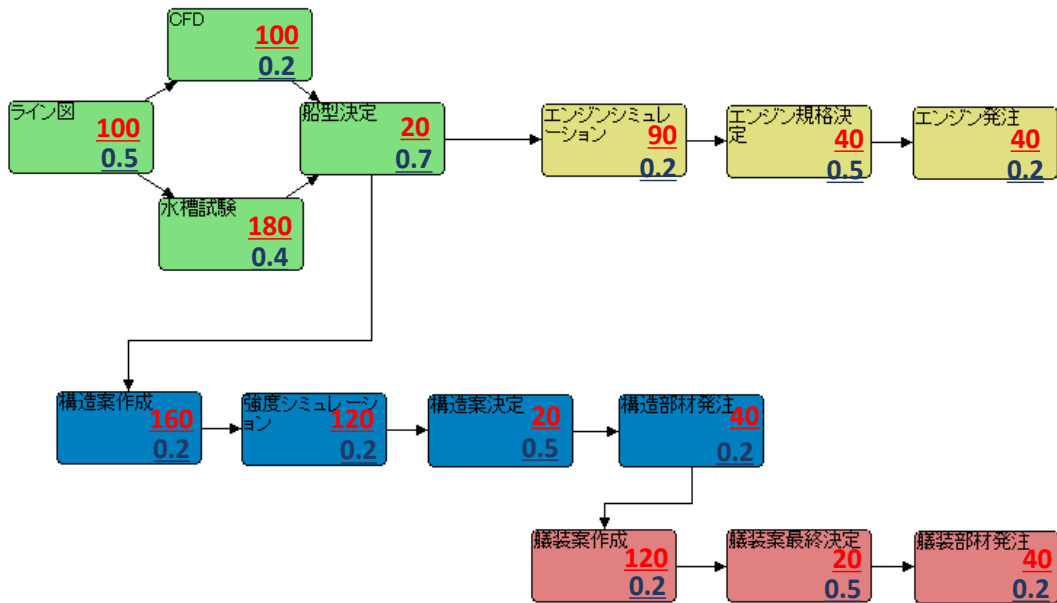


図 6-1 1つの造船設計プロジェクトを表すワークフロー

作業チーム設計の対象となる組織情報を表 6-1 に示す. 組織の人員表には 21 名の人員が登録されている.

表 6-1 造船設計組織の人員に関する情報

	固定給	変動給	ライン図	CFD	水槽実験	船型決定	エンジンシミュレーション	エンジン規格決定	エンジン発注	構造案作成	強度シミュレーション	構造案決定	構造部材発注	艤装案作成	艤装案最終決定	艤装部材発注
PM	7.0		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
Hu1	4.5		1.0	1.0	0.5	1.0	0.5							0.5		
Hu2	4.0		1.0	1.0	0.5	1.0					0.5					
Hu3		3.5	1.0	0.5	1.0	0.5	0.5									
Hu4		2.0		1.0	1.0											
Hu5		1.5		0.5	1.0											
En1	4.5			0.5			1.0	1.0	1.0	0.5	0.5					
En2	3.5						1.0	1.0		0.5	0.5			0.5		
En3		2.5					1.0	0.5	0.5		0.5					
En4		2.0			0.5		1.0		0.5							
En5		1.5			0.5		0.5		0.5							
St1	4.5			0.5	0.5					1.0	1.0	1.0	0.5			
St2	4.0						0.5			1.0	0.5	1.0	1.0			
St3		2.0								1.0	0.5	0.5				
St4		1.5								0.5	1.0					
St5		1.5									0.5		1.0			
Eq1	4.5				0.5		0.5			0.5				1.0	1.0	1.0
Eq2	3.5			0.5										1.0	1.0	1.0
Eq3		2.0					0.5							1.0	0.5	
Eq4		2.5								0.5	0.5			0.5		1.0
Eq5		1.5									0.5			0.5		0.5

また、船舶基本設計チームを設計するにあたって、今回のケーススタディでは船舶基本設計プロジェクトを計画する際の初期段階であるものとし、表 6-1 の組織情報から作業チームとして誰を採用するかやどのような配置にするかについて検討する状況を想定している。なお、今回のケーススタディでは、単に作業チームの組織構造を決定するだけでなく、以下の項目について分析することで、適切な組織構造設計のための知見を得ることを主な目的とする。

- 所要時間を短縮するための組織構造と合計コストを縮小するための組織構造の定性的な違いは何か
- プロジェクトで対象とするワークフロー数と、チーム作業シミュレーションの設定である同時実行可能ワークフロー数の違いによって、組織構造のパフォーマンスはどのように変化するか

6.2.2 提案手法の設定

本章では、表 6-2 で示す 5 つのパターンのシナリオに対するケーススタディを行う。図 6-1 のワークフローを 2 つ行う場合に、同時並行に 2 つのワークフローを実行せずに 1 つずつ行う場合を Case1-1 で、同時並行に 2 つのワークフローを実行する場合を Case1-2 で扱う。また、図 6-1 のワークフローを 3 つ行う場合に、同時並行に 3 つのワークフローを実行せずに 1 つずつ行う場合を Case2-1 で、2 つのワークフローまで同時に実行可能な制約下で行う場合を Case2-2 で、3 つのワークフローを同時並行で行う場合を Case2-3 で扱う。

表 6-2 第 6 章における作業情報の設定

シナリオ名	プロジェクトで対象とする ワークフロー(図 6-1)の数	同時実行ワークフロー数
Case 1-1	2	1
Case 1-2	2	2
Case 2-1	3	1
Case 2-2	3	2
Case 2-3	3	3

次に、提案手法の設定を表 6-3 に示す。今回のケーススタディでは作業チームの組織構造を設計の対象とし、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法を用いて組織構造の設計と分析を行う。また、目的関数として「所要時間平均」と「合計コスト平均」を採用し、所要時間と合計コストの 2 つの指標を考慮しながら、組織構造の設計を遺伝的アルゴリズムによって行う。遺伝的アルゴリズムによる探索に関しては、個体数 10、世代数 5000、交叉率 0.6、突然変異率 0.3 として探索を行う。

チーム作業シミュレーションの設定に関しては、「タスクの掛け持ちを考慮する」、「工作機械は考慮しない」、「トラブル率に従ってトラブルとその調整を発生させる」とし、チーム作業シミュレーションの繰り返し回数は 100 とした。また、今回のケーススタディでは表 6-1 における各人員のトラブル率は 0 とし、チーム作業シミュレーション上ではタスクに登録されているトラブル率のみによってトラブルが発生するものとする。

表 6-3 第 6 章における提案手法の設定

	設定項目	設定内容
探索の設定	探索手法	遺伝的アルゴリズムによる探索(3.5)
	設計変数	組織構造
	目的関数	<ul style="list-style-type: none"> ● 所要時間平均 ● 合計コスト平均
チーム作業シミュレーションの設定	各人員の同時刻におけるタスクの掛け持ち	あり
	タスク実行における工作機械の考慮	なし
	各タスクでランダムに発生するトラブルとその調整の考慮	あり
	チーム作業シミュレーションの繰り返し回数	100

今回のケーススタディにおける作業チームの設計変数と固定する変数の詳細に関する情報を表 6-4 に示す。設計変数として組織構造を採用し、作業戦略を固定した状態で組織構造の探索を行う。

作業戦略について、タスク割り振りルールに関しては、クリティカルパス上に近いタスクを優先して行うディスパッチングルールである TSLACK のみを採用する。すなわち、タスク割り振りルールの重みベクトル成分のうち、TSLACK の成分を 1、そのほかの成分を 0 とする。また、各人員のエフォート率は、今回対象とするワークフロー群(P1～P3)に対して同じ値を設定するものとする。

表 6-4 第 6 章における作業チームの設計変数と固定する変数

	変数	詳細
組織構造	人員と工作機械のスキルセット	設計変数
	人員間の依存関係	設計変数
作業戦略	タスク割り振りルール	固定(TSLACKのみ)
	エフォート率	固定(すべてのワークフローに対して均一)

6.3 結果

6.3.1 2つのワークフローを対象とした場合

6.3.1.1 同時実行ワークフロー数が1の場合(Case 1-1)

2つのワークフローをプロジェクトの対象とし、同時実行ワークフロー数を1とした場合に対して、提案手法によって組織構造の探索を行った履歴を図 6-2 に示す。提案手法では、所要時間平均の最小化を目的関数とした場合の探索と合計コスト平均の最小化を目的関数とした場合の探索を行っているため、結果として重複のある10万個の遺伝子で表される組織構造に対してチーム作業シミュレーションを行っているが、そのうち3244個の遺伝子で表される組織構造は、あるタスクを実行可能な人員がないなどの理由でチーム作業シミュレーションを完了することができなかった。そのため、図 6-2 では3244個の遺伝子で表される組織構造は含まれていない。また、図 6-2 の探索履歴から図 3-13 のスキル情報に基づく組織構造情報のフィルタリングによって重複をなくした場合の、所要時間と合計コストに対するパレートフロントを構成する組織構造が4個抽出された。

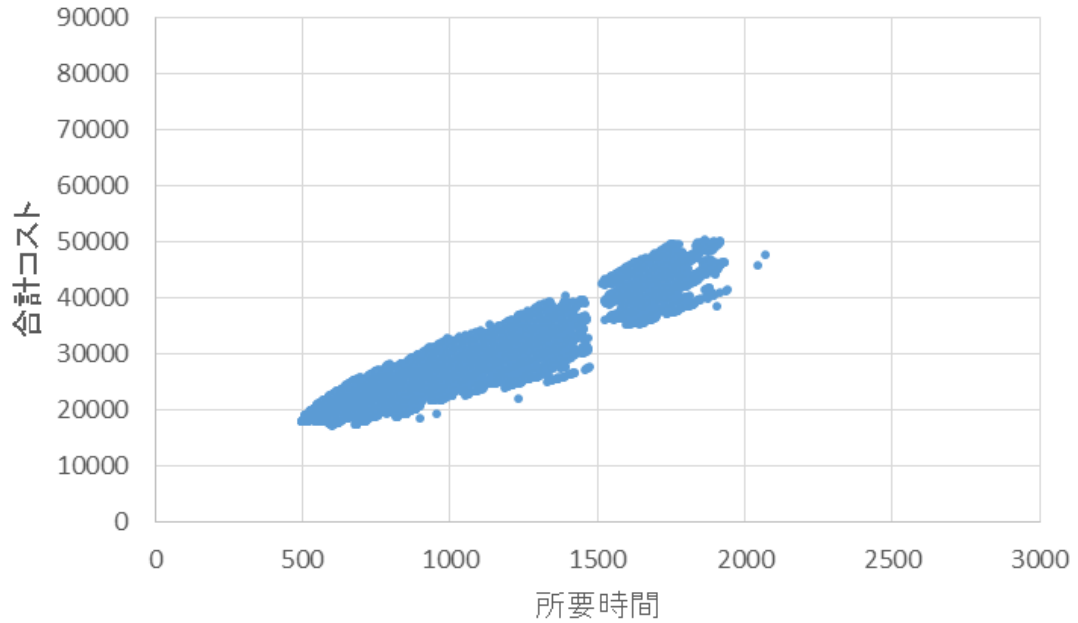


図 6-2 提案手法による探索結果(Case 1-1)

パレートフロントを構成する組織構造(PO11-1~PO11-4)のそれぞれに含まれる人員数・部門数と部門間の上下関係の線を表す依存関係の数を比べた結果を図 6-3 に示す。なお、図 6-2 におけるパレートフロントを構成する組織構造のうち、所要時間が最も短くコストが最も多い組織構造を PO11-1 とし、所要時間の短い順に PO11-2, PO11-3, PO11-4 として組織構造にラベル付けを行っている。図 6-3 では全体の傾向として、所要時間が短く合計コストが大きい組織構造ほど人員数、部署数、依存関係数のいずれも大きくなっていることが読み取れる。

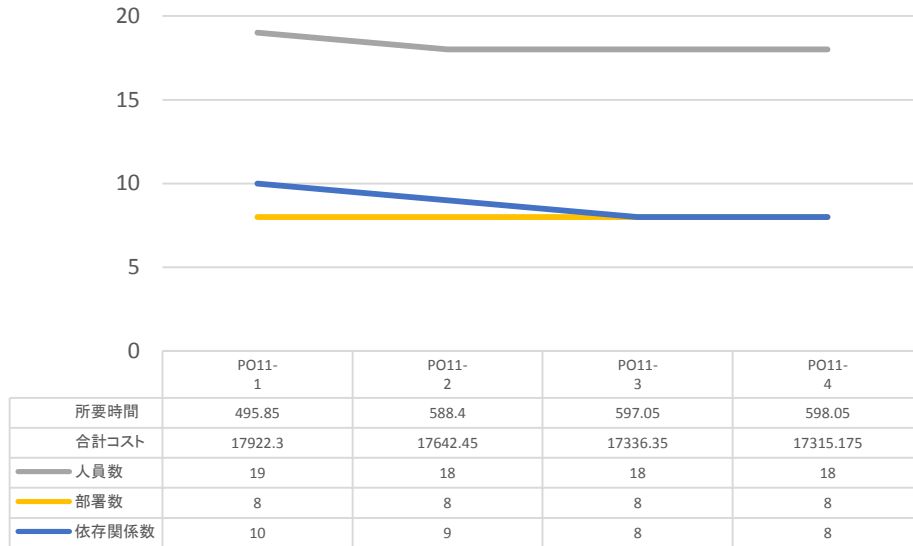


図 6-3 パレートフロントを構成する組織構造の比較(Case 1-1)

また、最も所要時間が短かった PO11-1 と最も合計コストが少なかった PO11-4 を図 6-4 と図 6-6 に、PO11-1 と PO11-4 を用いてチーム作業シミュレーションを行った場合の結果の一例を図 6-5 と図 6-7 に示す。図 6-4 と図 6-6 で共通する点として、「HULL」と「STRUCTURE」に多くの人員が所属している点が挙げられる。これは、図 6-1 のワークフローにおいてクリティカルパス上に存在するタスクを担当する部署が上記の 2 つの部署であるためであると言える。また、図 6-4 と図 6-6 で異なる点として所要時間が最も長く合計コストの最も少なかった組織構造である PO11-4 では、複数のスキルを持つ優秀な正規社員 PM が除外されている点が挙げられる。これは、表 6-1 の組織情報から分かる通り、優秀な正規社員は他の社員に比べて高い固定給が発生してしまうため、除外したり作業時間を短縮することでコストの削減を行っていることが分かる。また、図 6-5 と図 6-7 の左側に各タスクがどの時刻に実行されているかを表すガントチャートを示しているが、同時実行ワークフロー数 1 というチーム作業シミュレーションの設定に従って 1 つずつワークフローを実行していることが分かる。

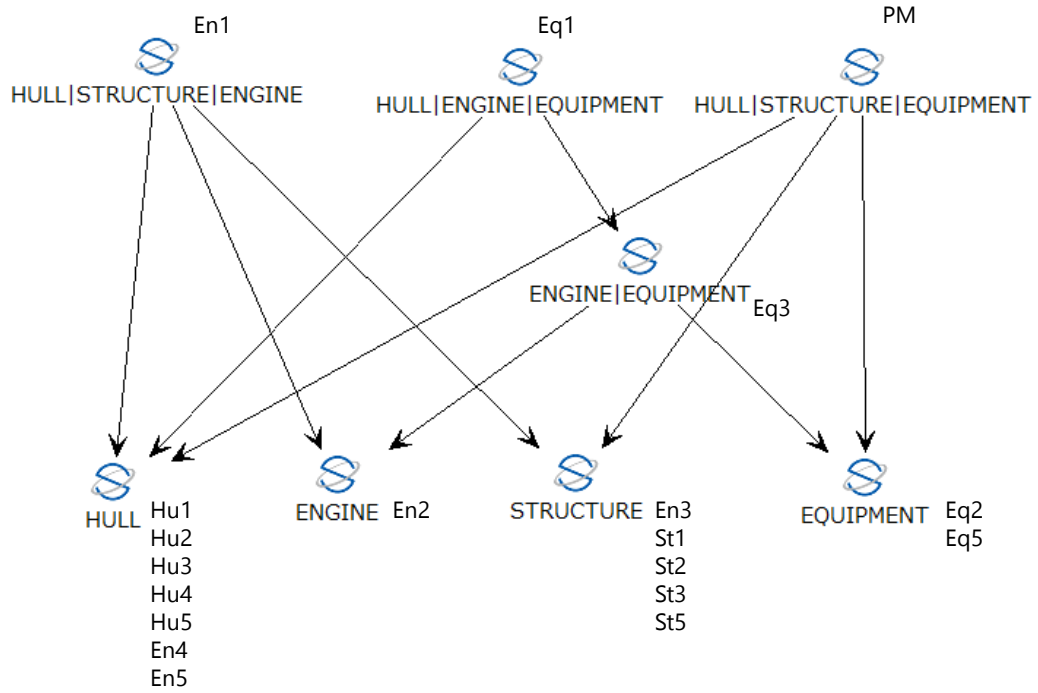


図 6-4 Case1-1 で所要時間の最も短かった組織構造(PO11-1)

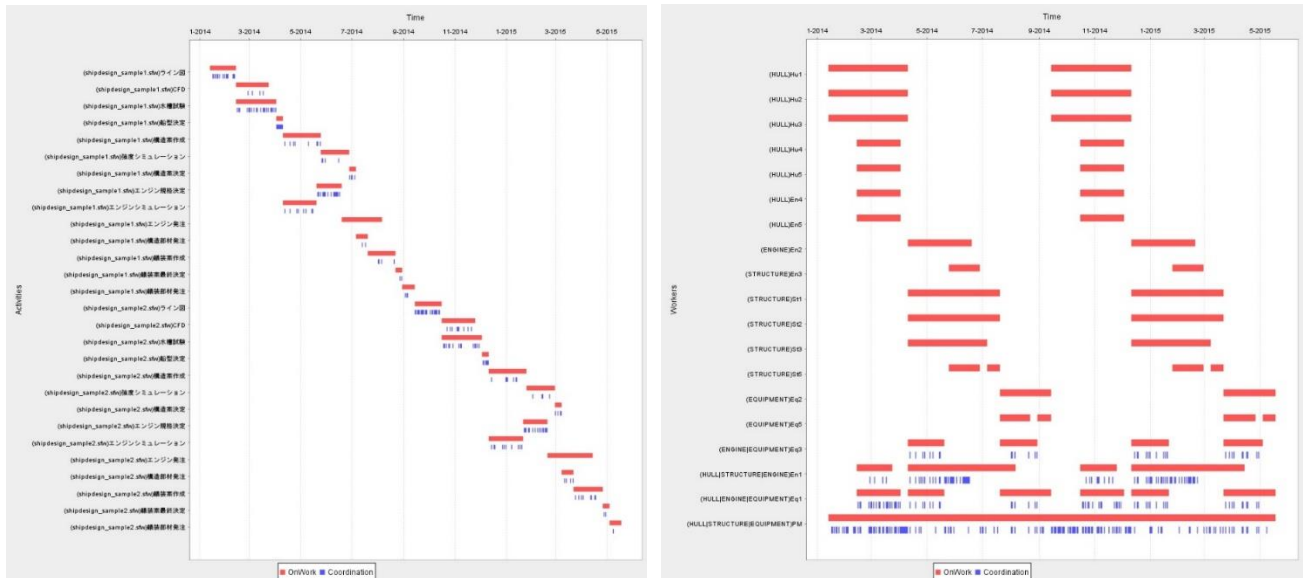


図 6-5 PO11-1 におけるチーム作業シミュレーション結果の一例

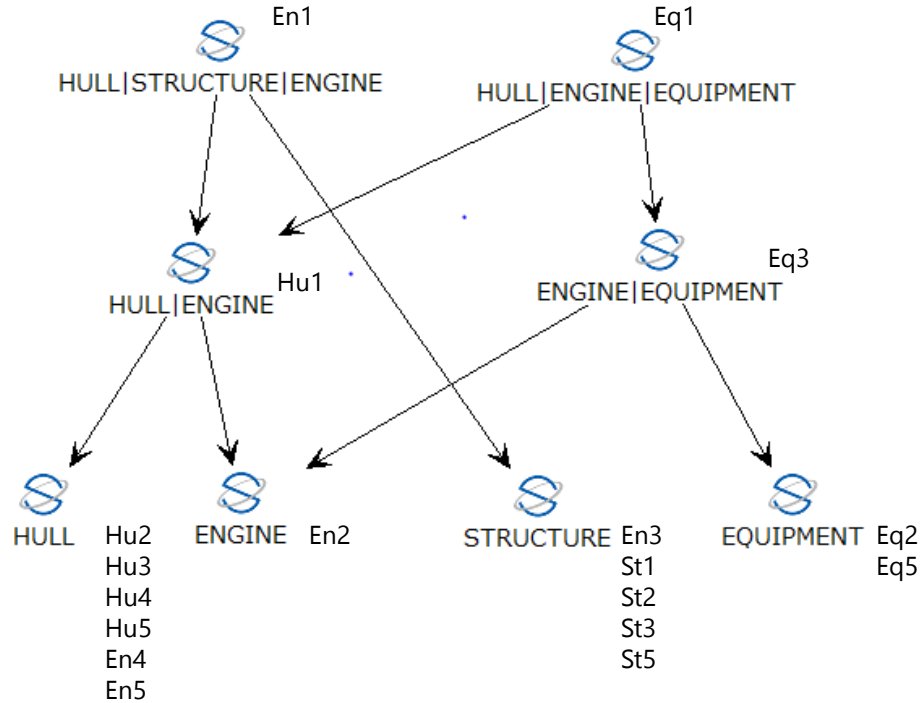


図 6-6 Case1-2 で合計コストの最も少なかった組織構造(PO11-4)

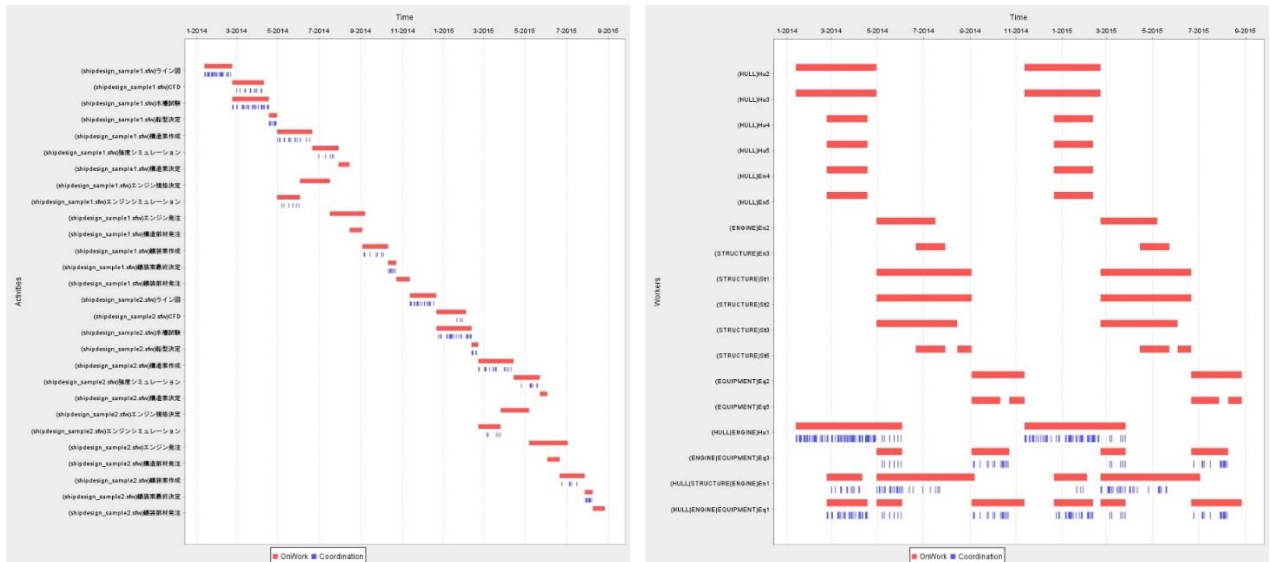


図 6-7 PO11-4 におけるチーム作業シミュレーション結果の一例

6.3.1.2 同時実行ワークフロー数が 2 の場合(Case 1-2)

2つのワークフローをプロジェクトの対象とし、同時実行ワークフロー数を2とした場合に対して、提案手法によって組織構造の探索を行った履歴を図 6-8 に示す。図 6-2 と同様に、図 6-8 では 3311 個の遺伝子で表される組織構造はチーム作業シミュレーションを完了することができなかつたため、図 6-8 では 3311 個の遺伝子で表される組織構造は含まれていない。また、図 6-8 の探索履歴から図 3-13 のスキル情報に基づく組織構造情報のフィルタリングによって重複をなくした場合の、所要時間と合計コストに対するパレートフロントを構成する組織構造が 15 個抽出された。

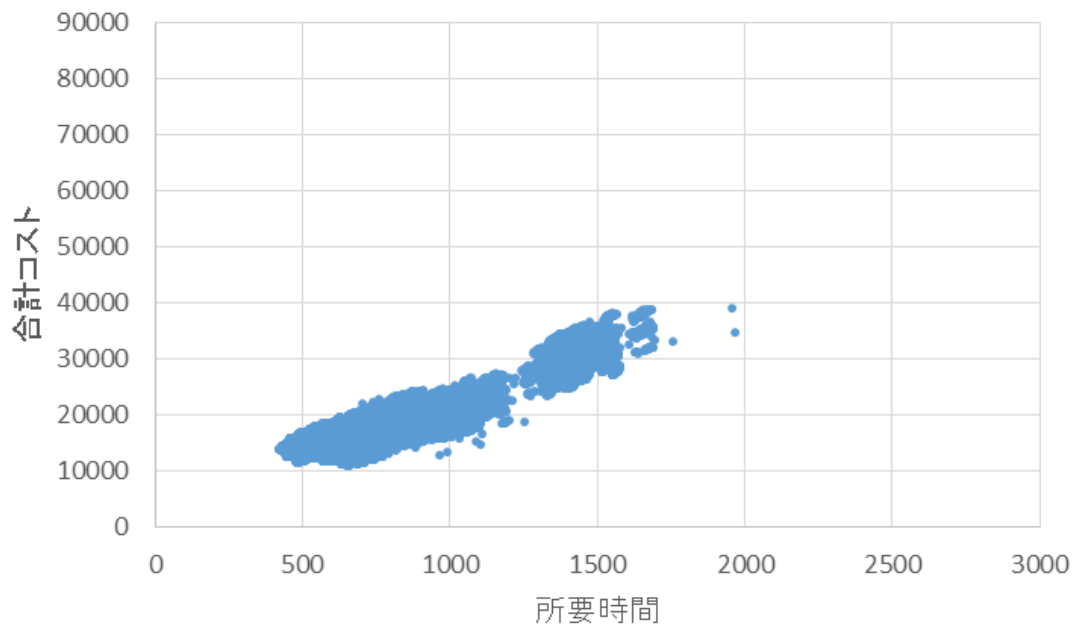


図 6-8 提案手法による探索結果(Case 1-2)

パレートフロントを構成する組織構造(PO12-1～PO12-15)のそれぞれに含まれる人員数・部署数と部署間の上下関係の線を表す依存関係の数を比べた結果を図 6-9 に示す。なお、図 6-8 におけるパレートフロントを構成する組織構造のうち、所要時間が最も短くコストが最も多い組織構造を PO12-1 とし、所要時間の短い順に PO12-2, PO12-3, …, PO12-15 として組織構造にラベル付けを行っている。図 6-9 では、所々に例外はあるものの、全体の傾

向としては図 6-3 と同様に, 所要時間が短く合計コストが大きい組織構造ほど人員数, 部署数, 依存関係数のいずれも大きくなっていることが読み取れる.

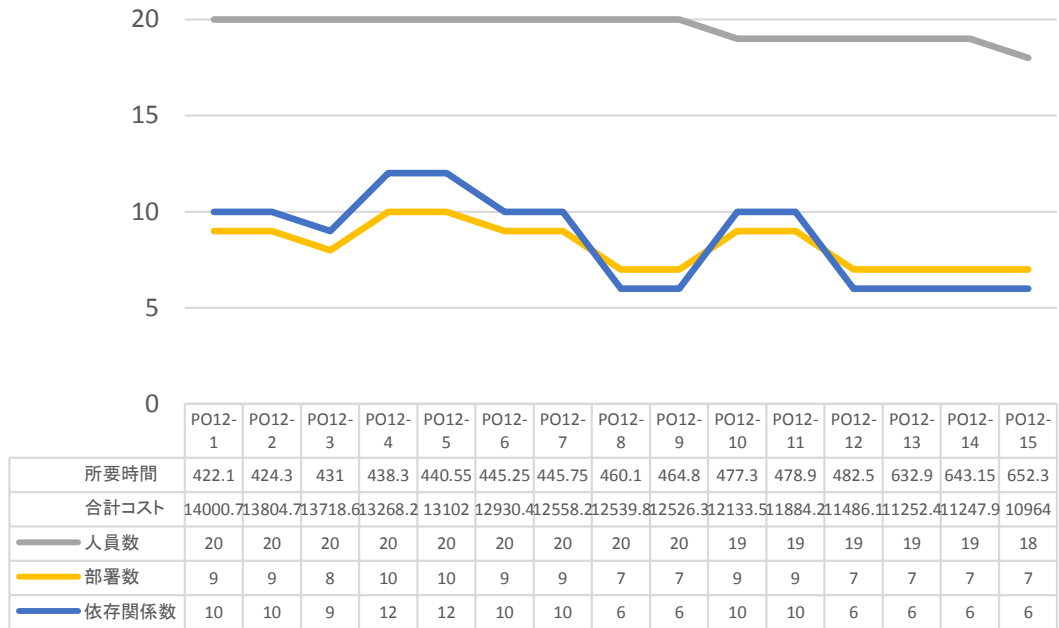


図 6-9 パレートフロントを構成する組織構造の比較(Case 1-2)

また, 最も所要時間が短かった PO12-1 と最も合計コストが少なかった PO12-15 を図 6-10 と図 6-12 に, PO12-1 と PO12-15 を用いてチーム作業シミュレーションを行った場合の結果の一例を図 6-11 と図 6-13 に示す. 図 6-10 と図 6-12 より, Case1-1 で見られた共通点と相違点が PO12-1 と PO12-15 でも見ることができる.

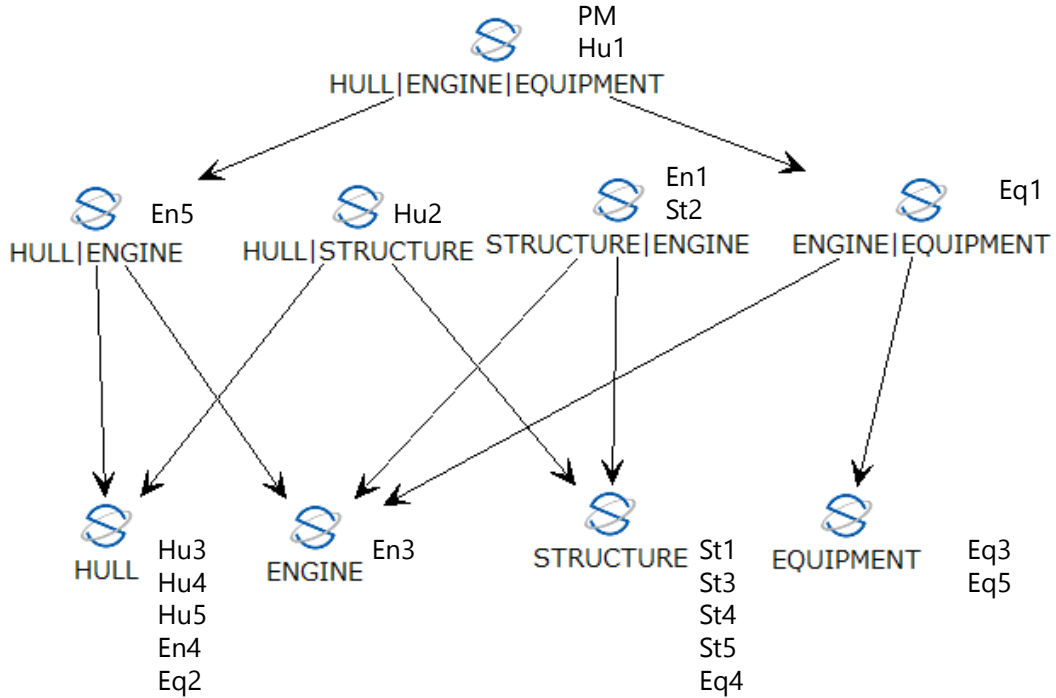


図 6-10 Case1-2 で所要時間の最も短かった組織構造(PO12-1)

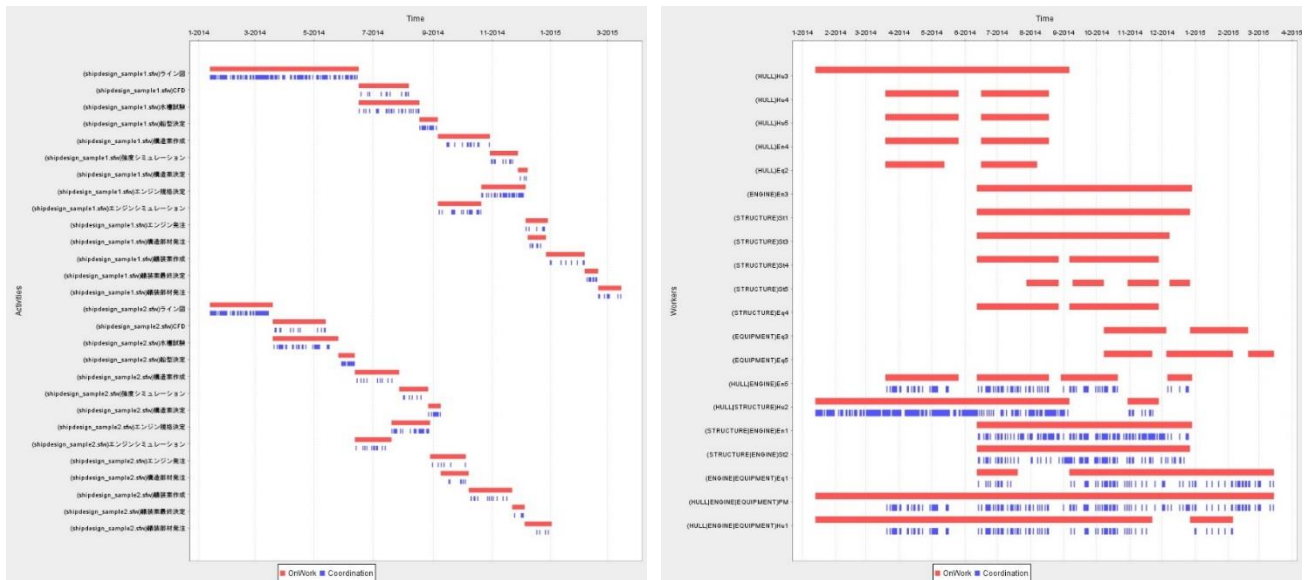


図 6-11 PO12-1 におけるチーム作業シミュレーション結果の一例

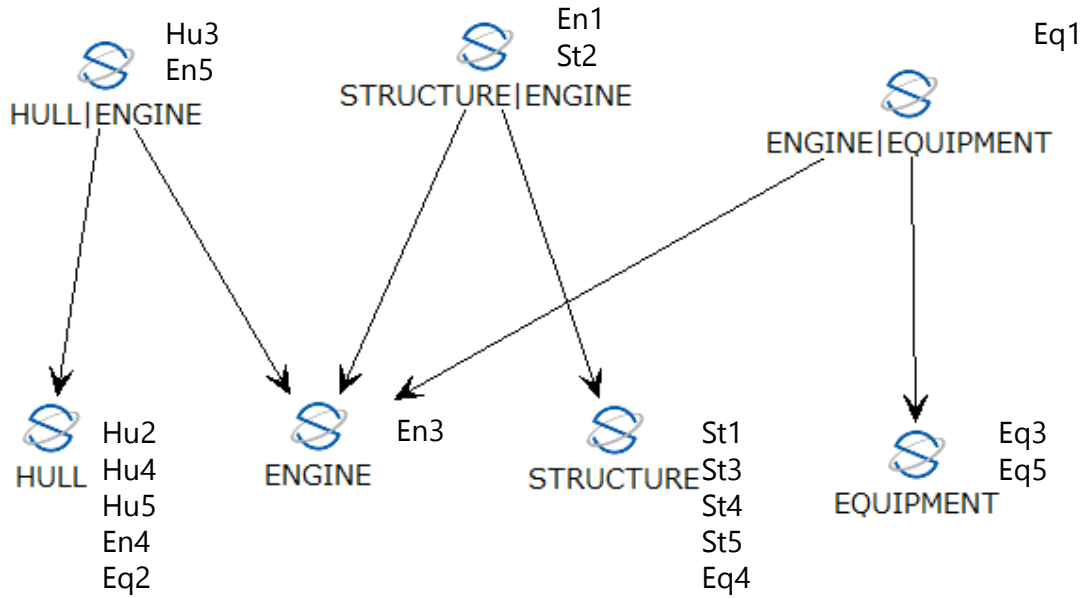


図 6-12 Case1-2 で合計コストの最も少なかった組織構造(PO12-15)

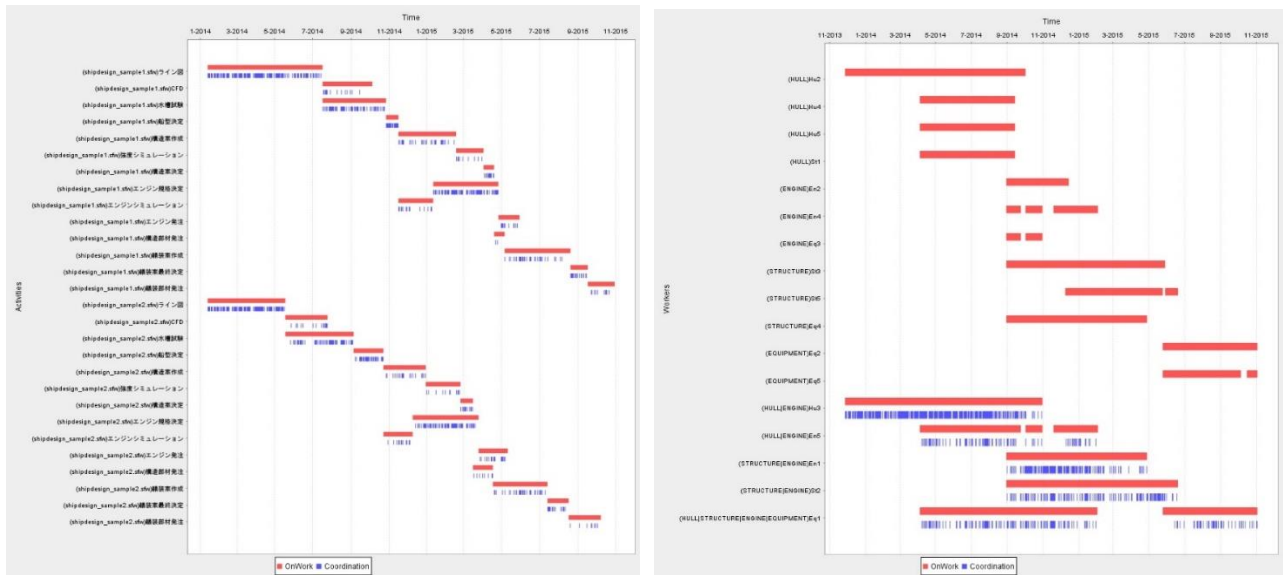


図 6-13 PO12-15 におけるチーム作業シミュレーション結果の一例

6.3.1.3 Case 1-1 と Case 1-2 で抽出された組織構造の比較

Case 1-1 と Case 1-2 で抽出された組織構造群を比較したところ、Case1-2 では Case1-1 に比べて、組織構造の上部に人員が配置されている傾向が見られた。本研究における組織構造のモデルでは、組織構造の上部に人員を配置することで、複数のスキルを持つ人員が複数のタスクを同時に実行することが可能になる。Case1-2 では Case1-1 に比べて同時実行ワークフロー数が増加しているため、単位時間あたりに実行可能なタスクの数が増加する。そのため、Case1-2 では単位時間あたりに増加したタスクの数に対応するために、組織構造の上部に人員を配置するように組織構造を変化させたと考えられる。

また、Case 1-1 と Case 1-2 で抽出された組織構造を持つ作業チームを入力としてチーム作業シミュレーションを行いパフォーマンスの比較を行った結果を図 6-14 に示す。図 6-14 では、1 つの組織構造を持つ作業チームを入力としてチーム作業シミュレーションを 100 回行い、目的関数である所要時間と合計コストのそれぞれに対する平均値をプロットしている。図 6-14 より、2 つのワークフローを並行して行うことで、1 つずつワークフローを行う場合に比べて所要時間が約 70~90、合計コストが約 4000 から 5000 削減可能なことが分かる。

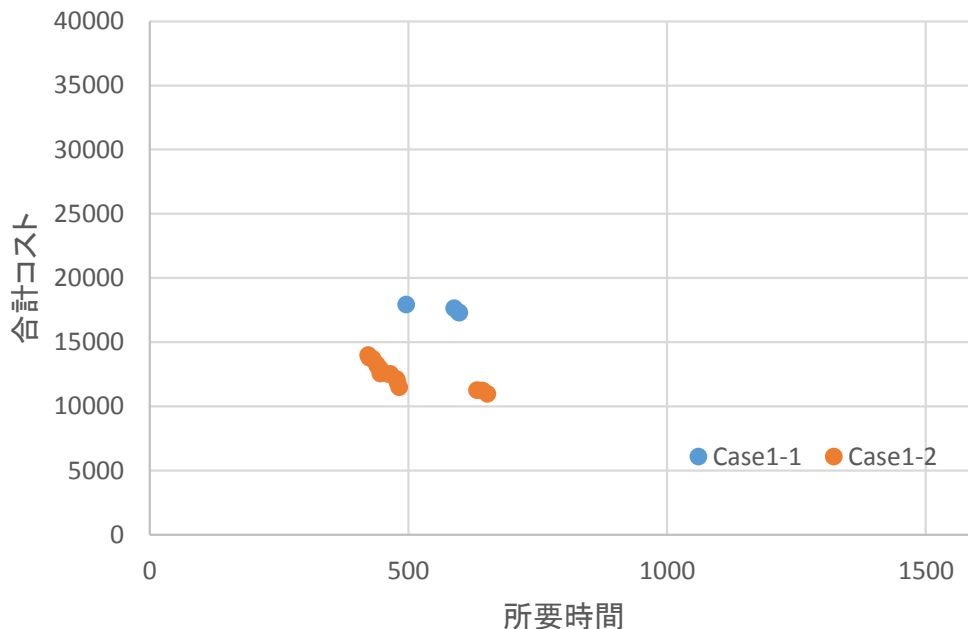


図 6-14 Case1-1 と Case1-2 で選択された組織構造の評価結果

6.3.2 3つのワークフローを対象とした場合

6.3.2.1 同時実行ワークフロー数が1の場合(Case 2-1)

3つのワークフローをプロジェクトの対象とし、同時実行ワークフロー数を1とした場合に対して、提案手法によって組織構造の探索を行った履歴を図 6-15 に示す。図 6-2 と同様に、図 6-15 では 3100 個の遺伝子で表される組織構造はチーム作業シミュレーションを完了することができなかつたため、図 6-15 では 3100 個の遺伝子で表される組織構造は含まれていない。また、図 6-15 の探索履歴から図 3-13 のスキル情報に基づく組織構造情報のフィルタリングによって重複をなくした場合は、所要時間と合計コストに対するパレートフロントを構成する組織構造が 7 個抽出された。

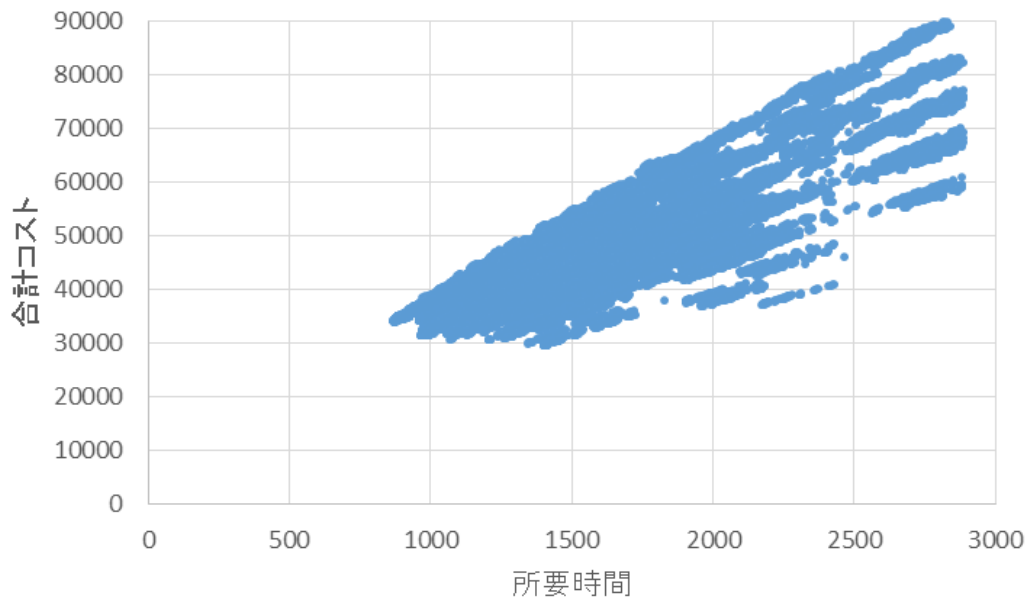


図 6-15 提案手法による探索結果(Case 2-1)

パレートフロントを構成する組織構造(PO21-1~PO21-7)のそれぞれに含まれる人員数・部門数と部門間の上下関係の線を表す依存関係の数を比べた結果を図 6-16 に示す。なお、図 6-15 におけるパレートフロントを構成する組織構造のうち、所要時間が最も短くコストが最も多い組織構造を PO21-1 とし、所要時間の短い順に PO21-2, PO21-3, …, PO21-7 として

組織構造にラベル付けを行っている。図 6-16 に関しても同様に、所要時間が短く合計コストが大きい組織構造ほど人員数、部署数、依存関係数のいずれも大きくなっている傾向が見られる。

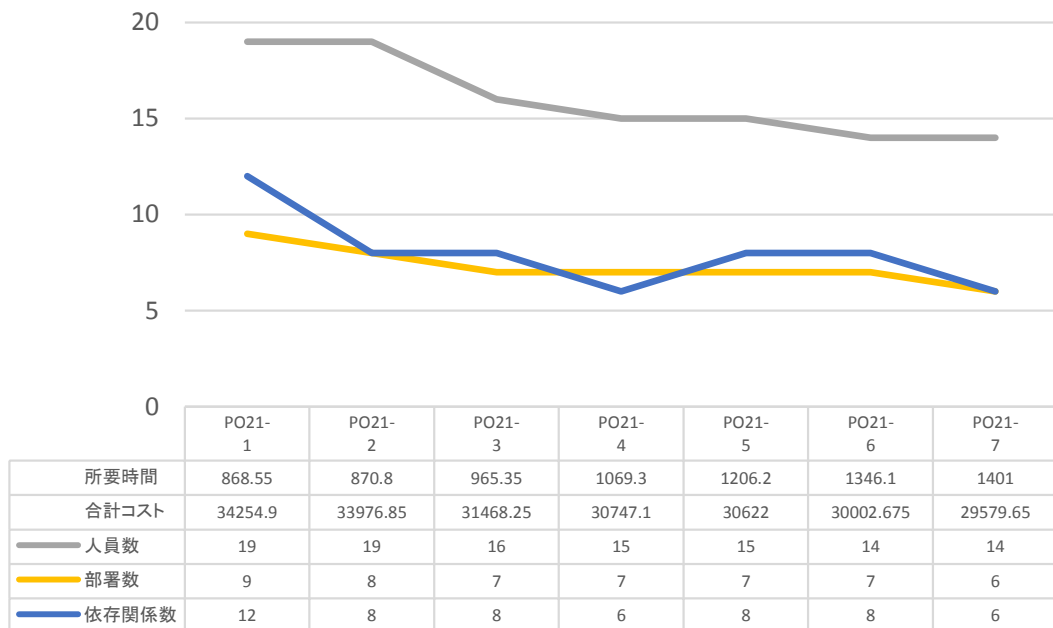


図 6-16 パレートフロントを構成する組織構造の比較(Case2-1)

また、最も所要時間が短かった PO21-1 と最も合計コストが少なかった PO21-7 を図 6-17 と図 6-19 に、PO21-1 と PO21-7 を用いてチーム作業シミュレーションを行った場合の結果の一例を図 6-18 と図 6-20 に示す。図 6-17 と図 6-19 で異なる点として所要時間が最も長く合計コストの最も少なかった組織構造である PO21-7 では、複数のスキルを持つ優秀な正規社員が除外されていることが挙げられる(PM, Hu1, St1)。また、PO21-1 では、PO11-1 と比較して「HULL」の部署に所属する人員数が減少し、その分「HULL」の上部の部署に人員が多く配属されていることが読み取れる。

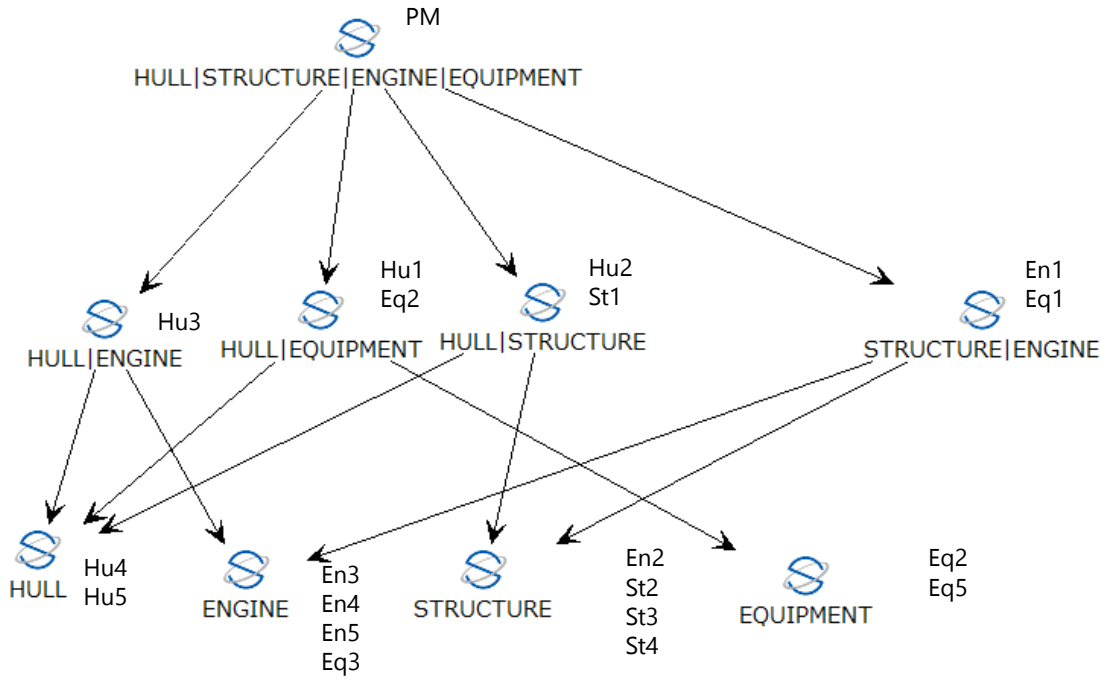


図 6-17 Case2-1 で所要時間の最も短かった組織構造(PO21-1)

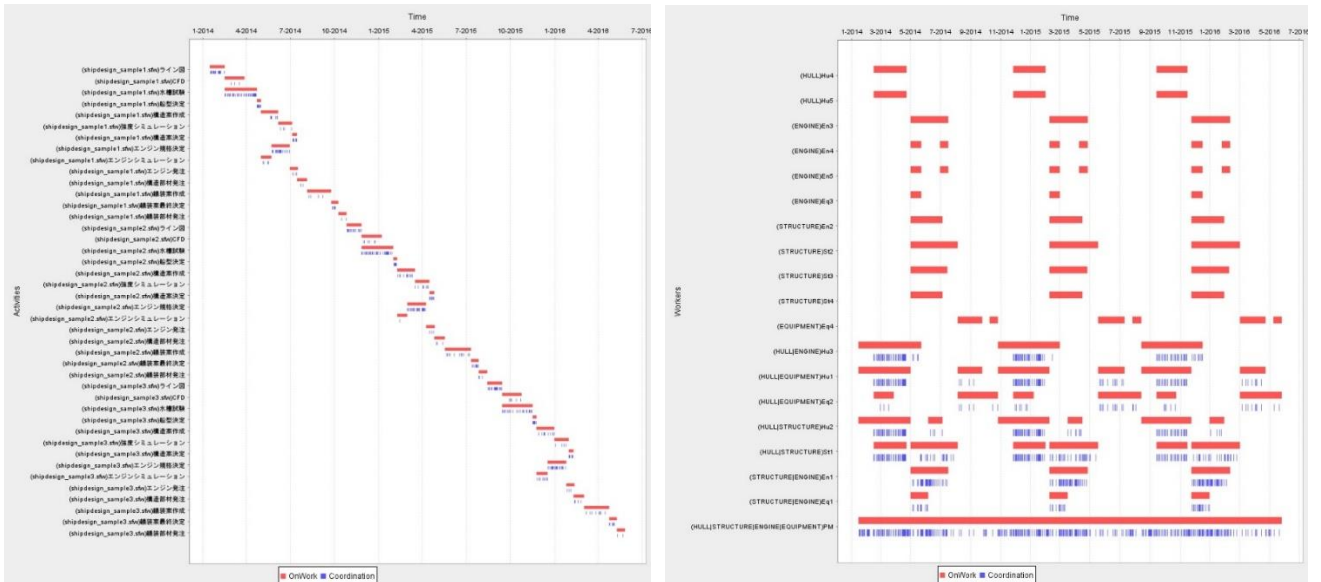


図 6-18 PO21-1 におけるチーム作業シミュレーション結果の一例

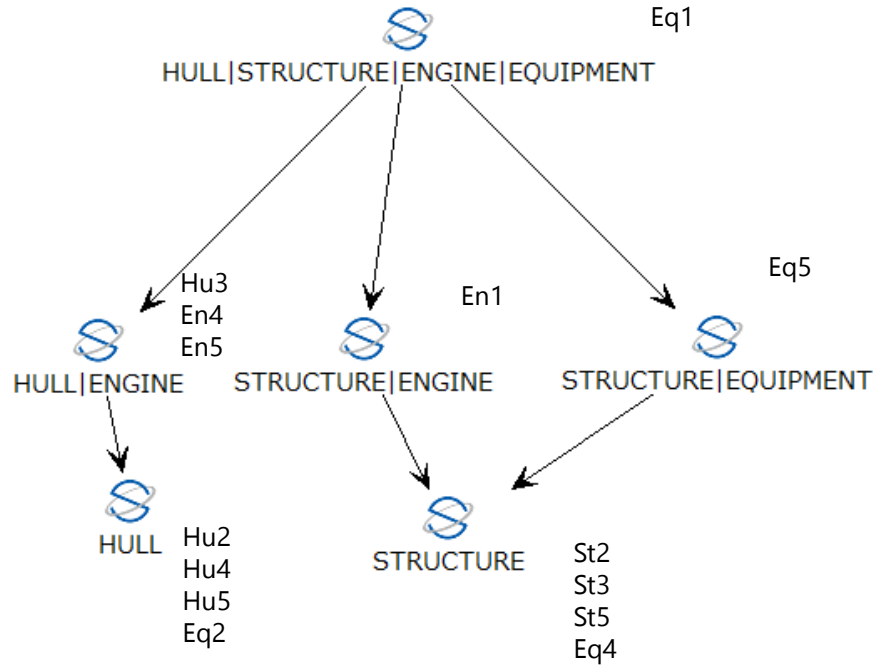


図 6-19 Case2-1 で合計コストの最も少なかった組織構造(PO21-7)

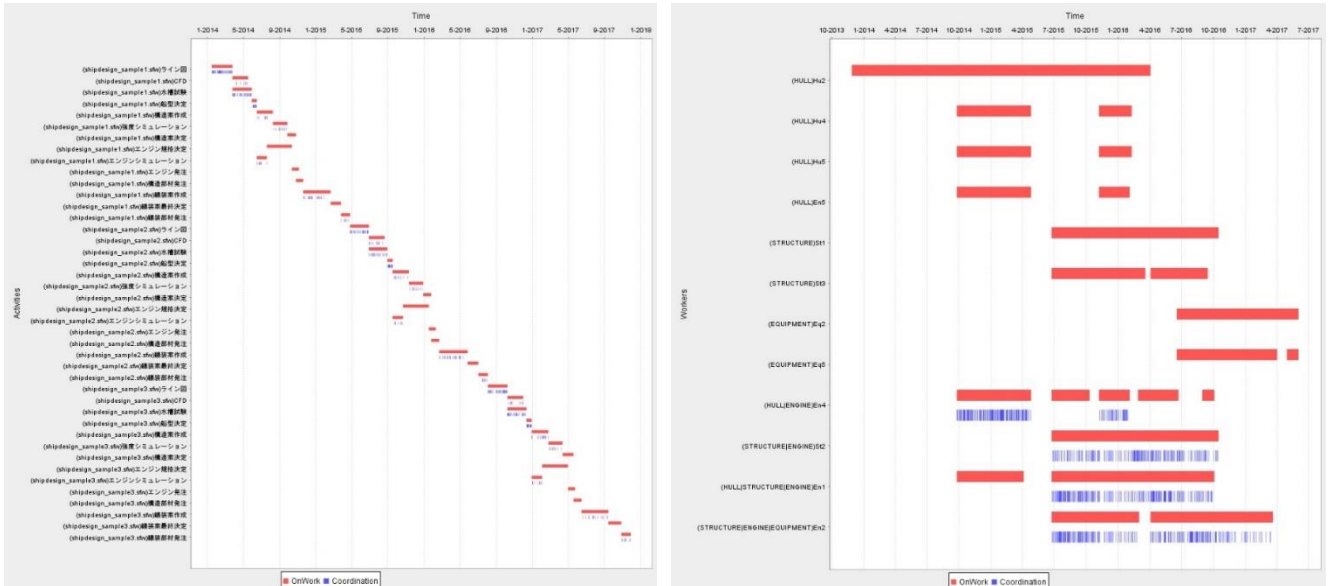


図 6-20 PO21-7 におけるチーム作業シミュレーション結果の一例

6.3.2.2 同時実行ワークフロー数が 2 の場合(Case 2-2)

3つのワークフローをプロジェクトの対象とし、同時実行ワークフロー数を2とした場合に対して、提案手法によって組織構造の探索を行った履歴を図 6-21 に示す。図 6-2と同様に、図 6-21 では 3576 個の遺伝子で表される組織構造はチーム作業シミュレーションを完了することができなかつたため、図 6-21 では 3576 個の遺伝子で表される組織構造は含まれていない。また、図 6-21 の探索履歴から図 3-13 のスキル情報に基づく組織構造情報のフィルタリングによって重複をなくした場合は、所要時間と合計コストに対するパレートフロントを構成する組織構造が 14 個抽出された。

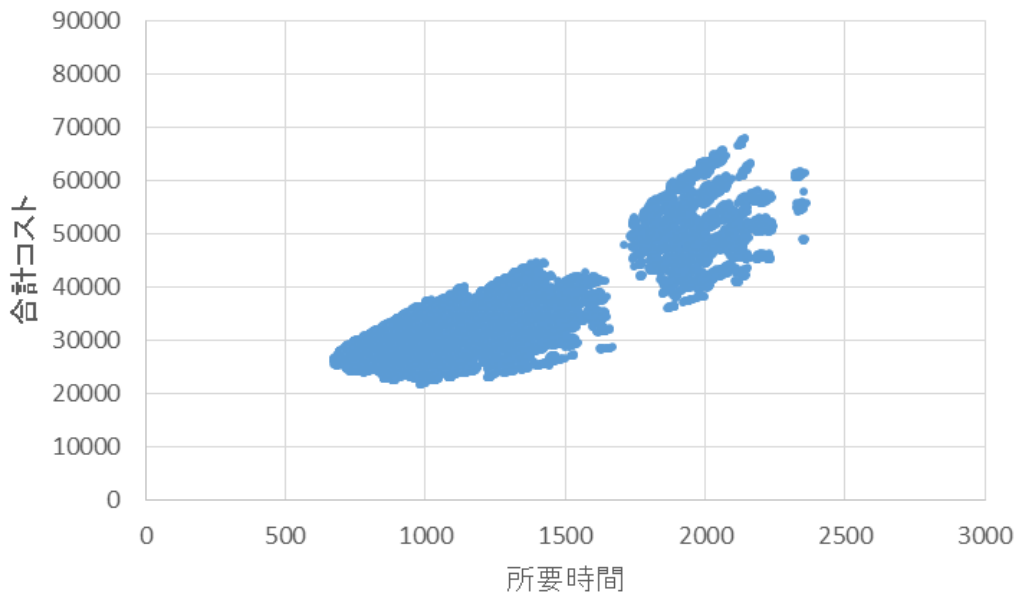


図 6-21 提案手法による探索結果(Case 2-2)

パレートフロントを構成する組織構造(PO22-1~PO22-14)のそれぞれに含まれる人員数・部門数と部門間の上下関係の線を表す依存関係の数を比べた結果を図 6-22 に示す。なお、図 6-21 におけるパレートフロントを構成する組織構造のうち、所要時間が最も短くコストが最も多い組織構造を PO22-1 とし、所要時間の短い順に PO22-2, PO22-3, …, PO22-14 として組織構造にラベル付けを行っている。図 6-22 では所要時間が短く合計コストが大きい

組織構造ほど人員数, 部署数, 依存関係数のいずれも大きくなっていったが, 部署数と依存関係数に関しては他のケースと比べるとその傾向は弱いものであった.

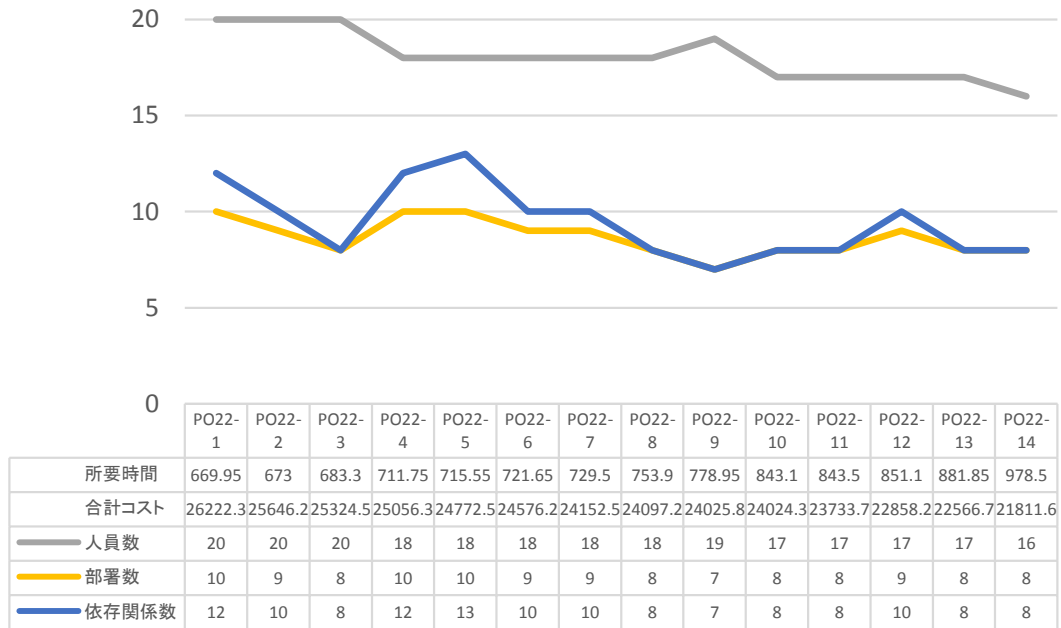


図 6-22 パレートフロントを構成する組織構造の比較(Case2-2)

また, 最も所要時間が短かった PO22-1 と最も合計コストが少なかった PO22-14 を図 6-23 と図 6-25 に, PO22-1 と PO22-14 を用いてチーム作業シミュレーションを行った場合の結果の一例を図 6-24 と図 6-26 に示す. 図 6-23 と図 6-25 で共通する点として, これまでのケーススタディと同様に「HULL」と「STRUCTURE」に関連する部署に多くの人員が所属していることや, 多くのスキルを持つ優秀な人員が PO22-14 では除外されている点が挙げられる.

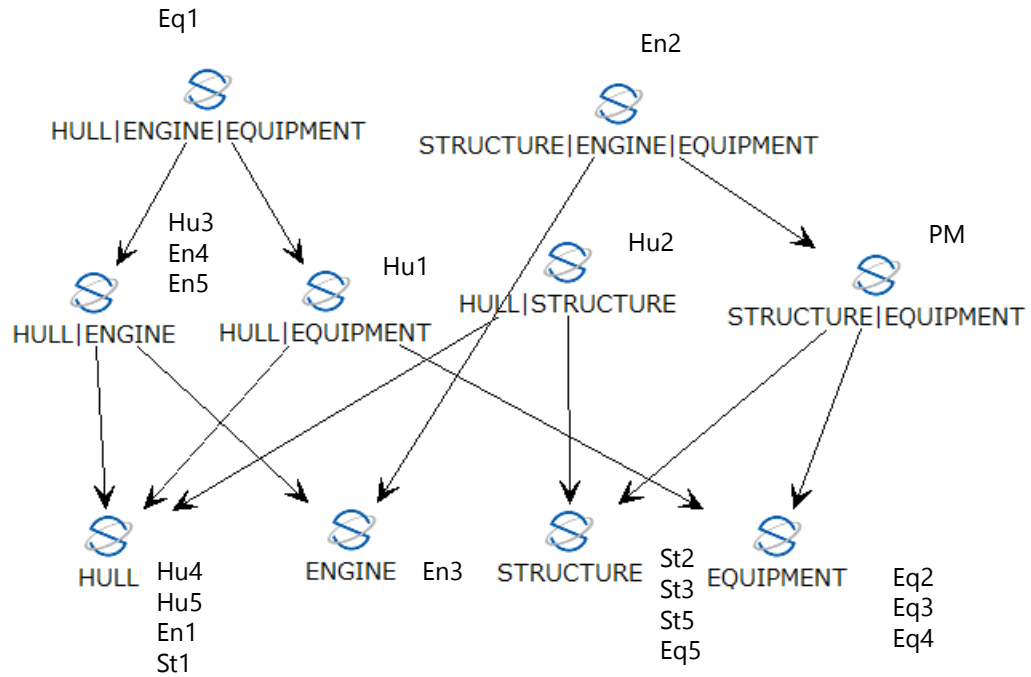


図 6-23 Case2-2 で所要時間の最も短かった組織構造(PO22-1)

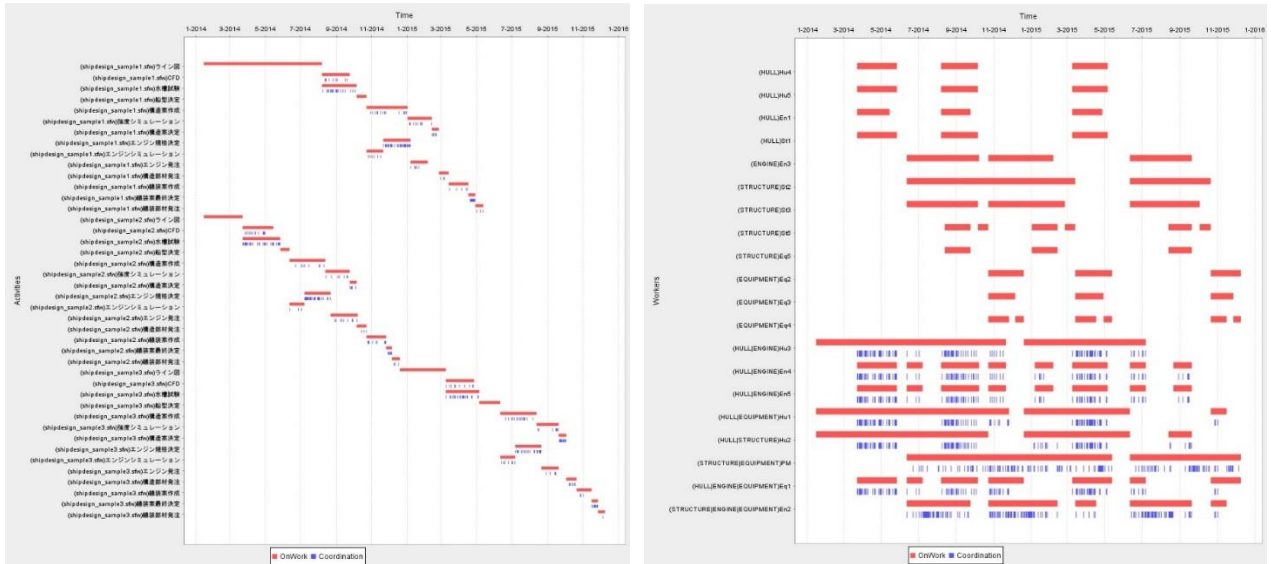


図 6-24 PO22-1 におけるチーム作業シミュレーション結果の一例

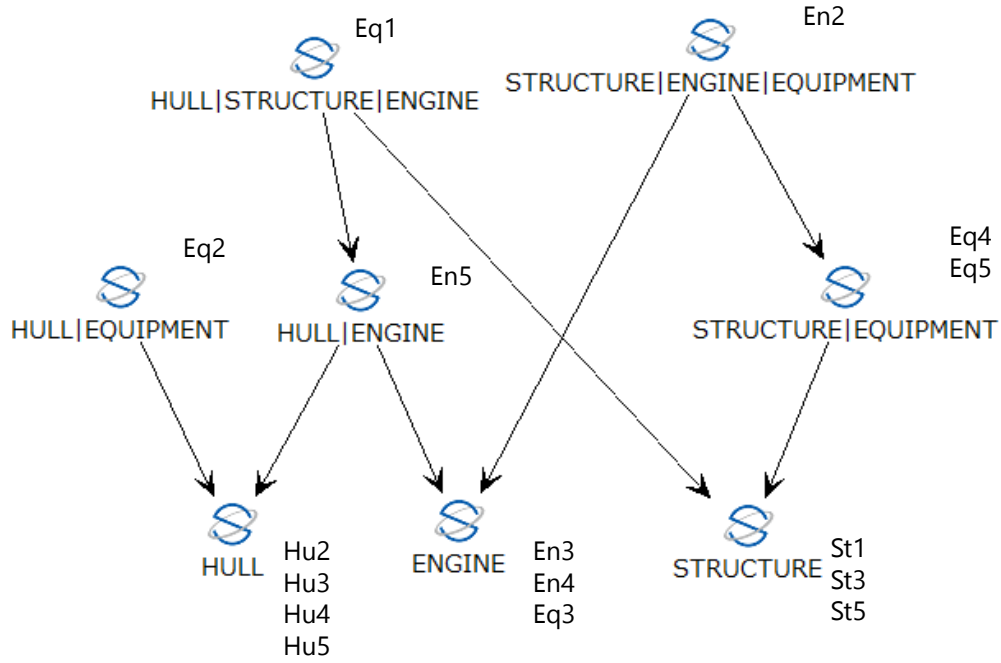


図 6-25 Case2-2 で合計コストの最も少なかった組織構造(PO22-14)

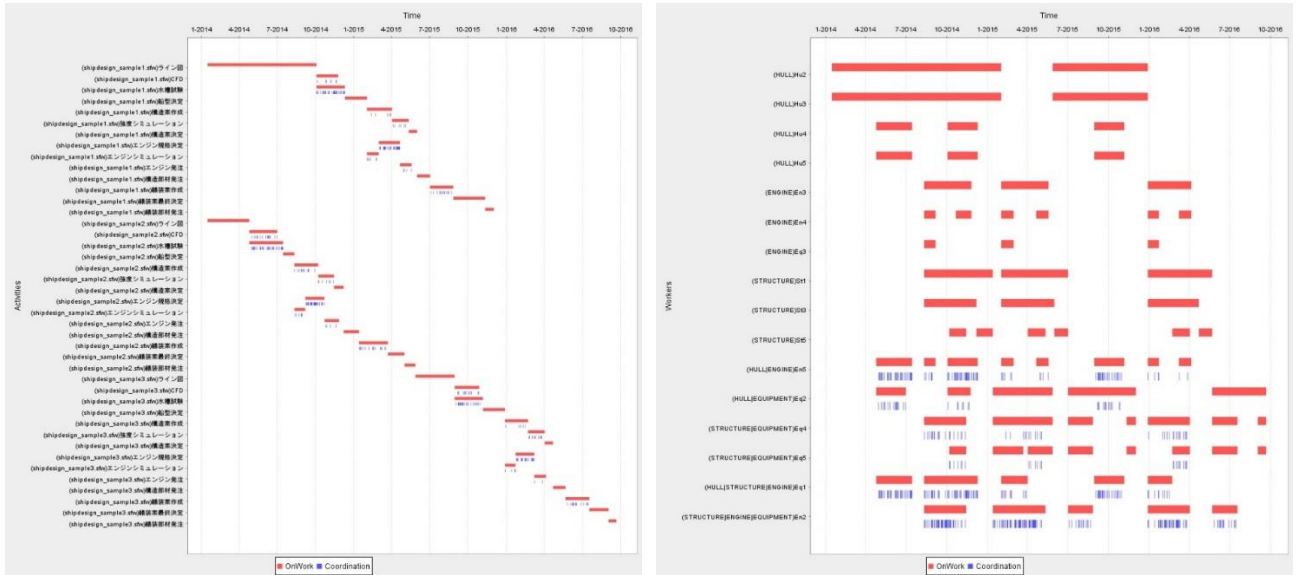


図 6-26 PO22-14 におけるチーム作業シミュレーション結果の一例

6.3.2.3 同時実行ワークフロー数が 3 の場合(Case 2-3)

3つのワークフローをプロジェクトの対象とし、同時実行ワークフロー数を3とした場合に対して、提案手法によって組織構造の探索を行った履歴を図 6-27 に示す。図 6-2 と同様に、図 6-27 では 3859 個の遺伝子で表される組織構造はチーム作業シミュレーションを完了することができなかつたため、図 6-27 では 3859 個の遺伝子で表される組織構造は含まれていない。また、図 6-27 の探索履歴から図 3-13 のスキル情報に基づく組織構造情報のフィルタリングによって重複をなくした場合の、所要時間と合計コストに対するパレートフロントを構成する組織構造が 13 個抽出された。

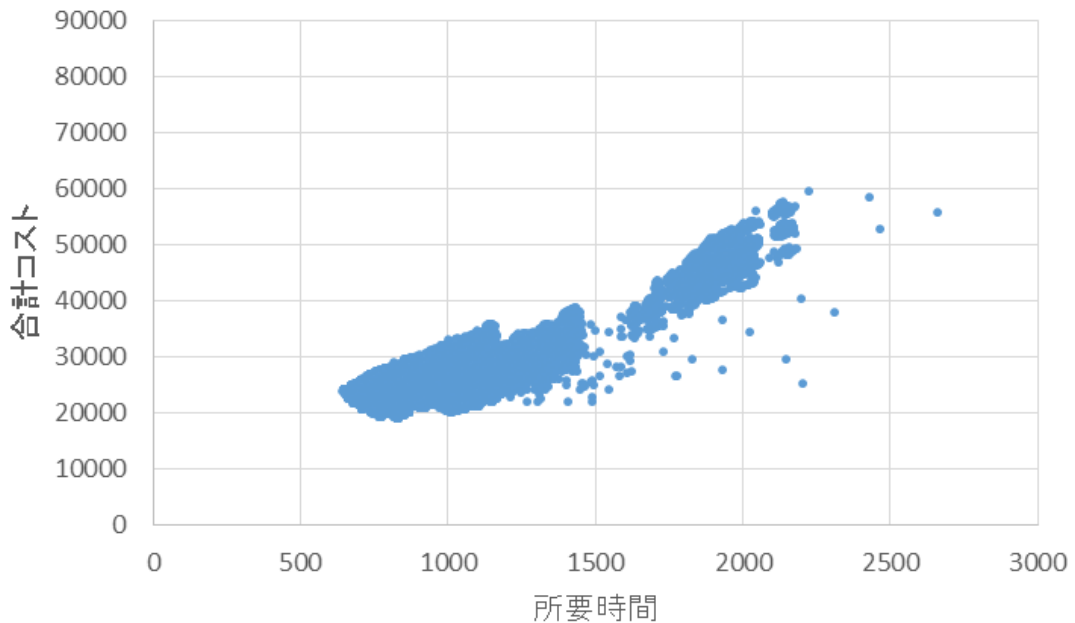


図 6-27 提案手法による探索結果(Case 2-3)

パレートフロントを構成する組織構造(PO23-1~PO23-13)のそれぞれに含まれる人員数・部門数と部門間の上下関係の線を表す依存関係の数を比べた結果を図 6-28 に示す。なお、図 6-27 におけるパレートフロントを構成する組織構造のうち、所要時間が最も短くコストが最も多い組織構造を PO23-1 とし、所要時間の短い順に PO23-2, PO23-3, …, PO23-13 として組織構造にラベル付けを行っている。図 6-28 でも同様に所要時間が短く合計コストが

大きい組織構造ほど部署数, 依存関係数のいずれも大きくなっていることが読み取れるが, 人員数に関しては他のケースに比べて変化が少ないという結果になった。

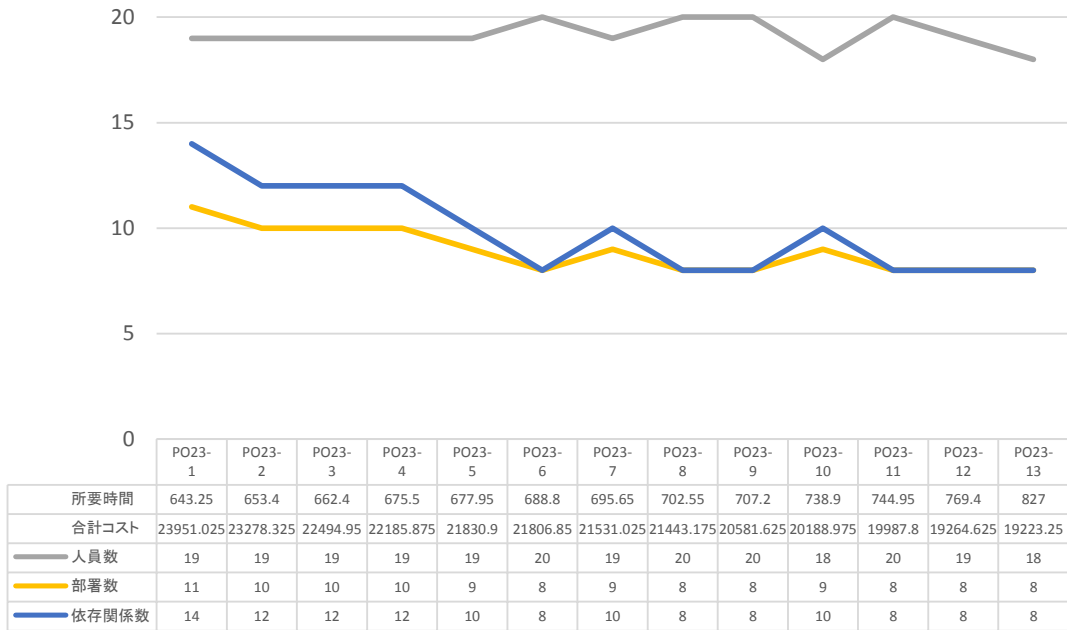


図 6-28 パレートフロントを構成する組織構造の比較(Case2-3)

また, 最も所要時間が短かった PO23-1 と最も合計コストが少なかった PO23-13 を図 6-29 と図 6-31 に, PO23-1 と PO23-13 を用いてチーム作業シミュレーションを行った場合の結果の一例を図 6-30 と図 6-32 に示す. 図 6-29 と図 6-31 で共通する点として, これまでのケースと同様に, 「HULL」と「STRUCTURE」に関連する部署に多くの人員が所属している点が挙げられる。

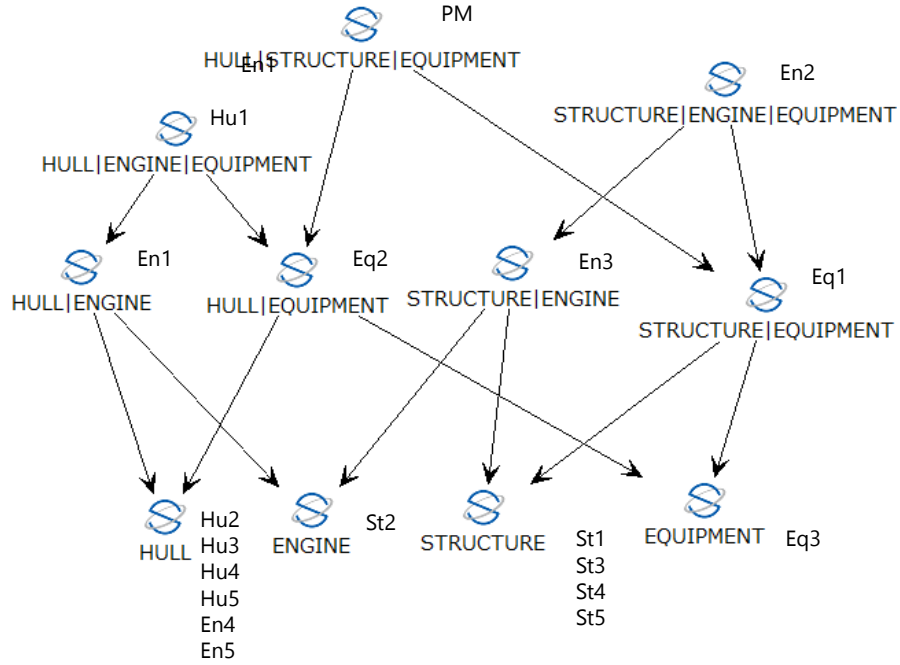


図 6-29 Case2-3 で所要時間の最も短かった組織構造(PO23-1)

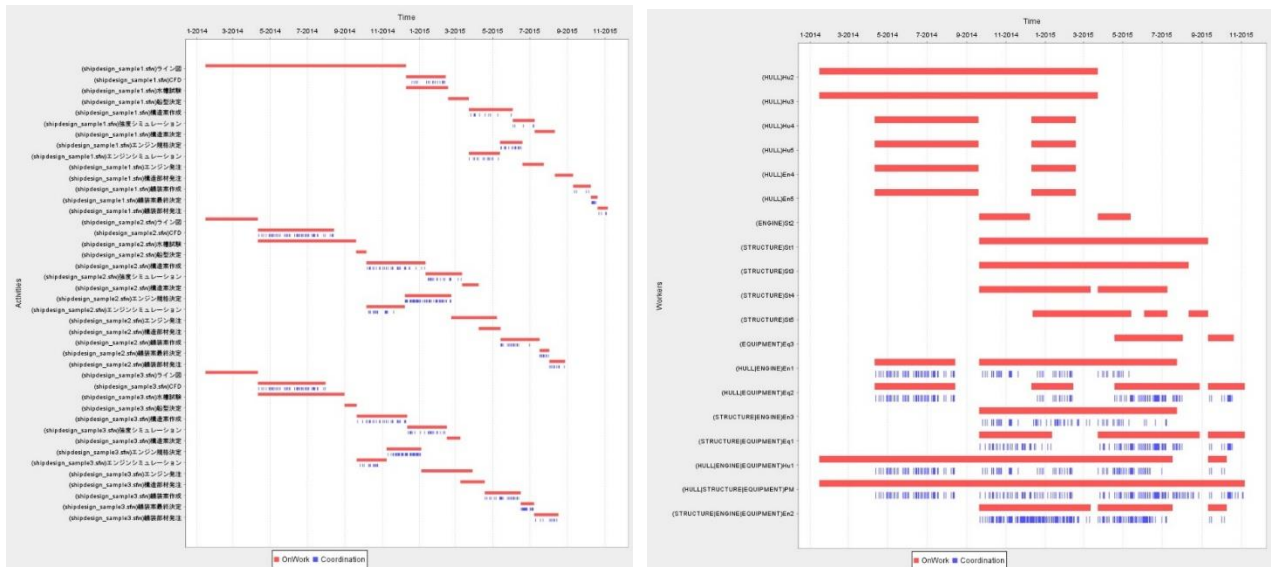


図 6-30 PO23-1 におけるチーム作業シミュレーション結果の一例

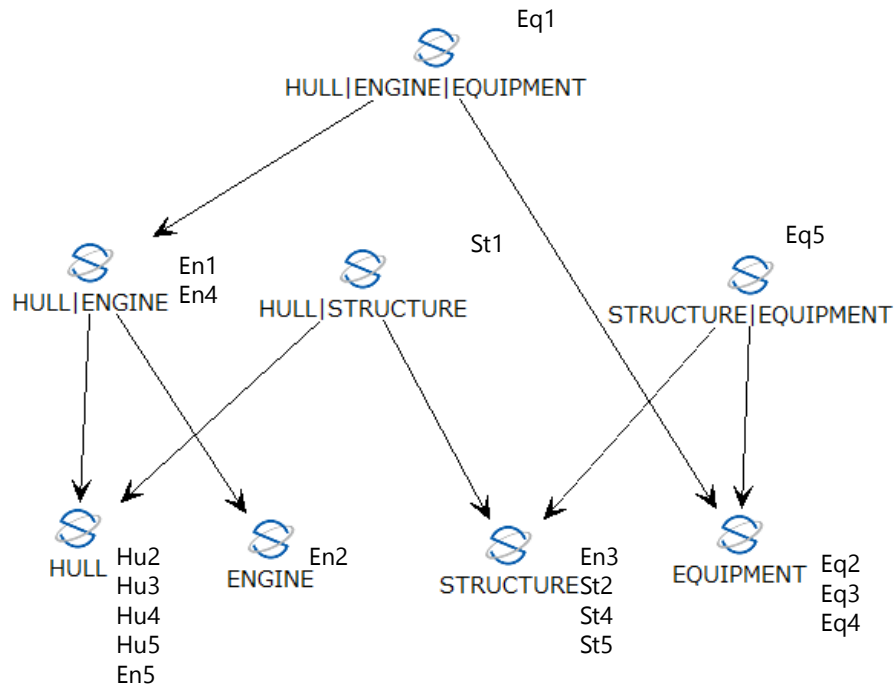


図 6-31 Case2-3 で合計コストの最も少なかった組織構造(PO23-13)

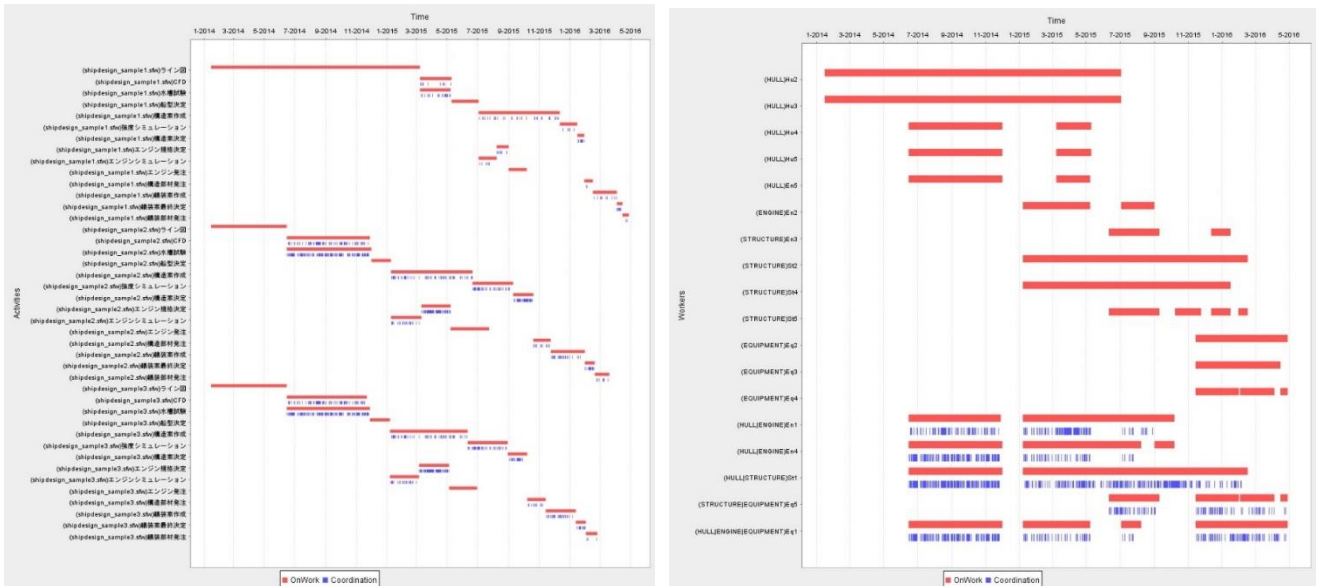


図 6-32 PO23-13 におけるチーム作業シミュレーション結果の一例

6.3.2.4 Case2-1 と Case2-2 と Case2-3 で抽出した組織構造の比較

Case 2-1 と Case2-2 と Case2-3 で得られた組織構造群を比較すると、Case1-1 と Case1-2 で比較した結果と同様に、同時実行ワークフロー数が増えるほど、組織構造の上部に人員が配置されている傾向が見られた。

また、Case 2-1 と Case2-2 と Case2-3 で抽出された組織構造を持つ作業チームを入力としてチーム作業シミュレーションを行いパフォーマンスの比較を行った結果を図 6-33 に示す。図 6-33 では、1 つの組織構造を持つ作業チームを入力としてチーム作業シミュレーションを 100 回行い、目的関数である所要時間と合計コストのそれぞれに対する平均値をプロットしている。図 6-33 では、同時実行ワークフロー数が 3 つの場合(Case2-3)と 2 つの場合(Case2-2)における組織構造のパフォーマンスの差は、所要時間が約 200~250、合計コストが約 9000~10000 であることが分かる。一方で、同時実行ワークフロー数が 2 つの場合(Case2-2)と 1 つの場合(Case2-1)における組織構造のパフォーマンスの差は、所要時間が約 20~30、合計コストは約 2000~500 となり、前者に比べてパフォーマンスの差が少なくなることが分かる。

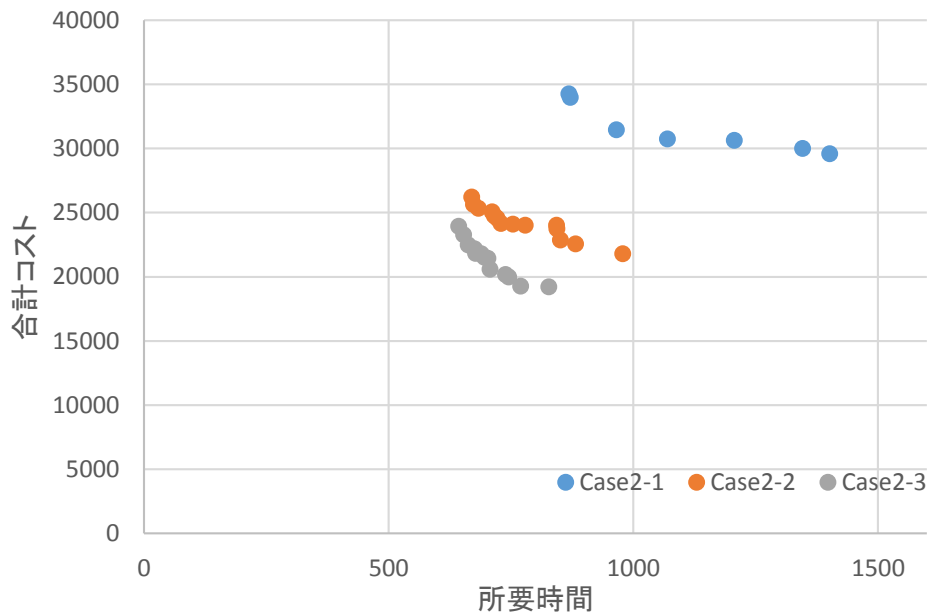


図 6-33 Case2-1 と Case2-2 と Case2-3 で選択された組織構造の評価結果

6.4 まとめ

本章では、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法を、組織構造を含めた船舶基本設計チームの設計のケーススタディに適用することで、その評価を行った。本章のケーススタディを通して得られた知見を以下にまとめる。

- 提案した組織構造を含めた作業チーム設計手法は、人手で適切な設計を行うことが困難な船舶基本設計プロセスに対する作業チーム設計支援に有効であることが確認できた。
- 本研究で提案した組織構造のモデル下において、所要時間を重視した組織構造と合計コストを重視した組織構造に共通する点として、ケーススタディではクリティカルパス上に存在するタスクを担当する部署に人員を多く所属させる傾向があること、同時実行ワークフローの最大数数が増加した場合、複数のスキルを持つ人員を組織構造の上部に所属させることが望ましいという傾向が見られた。
- 提案手法によって、同時実行ワークフロー数の違いによって取るべき組織構造の情報が得られると同時に、それに応じたパフォーマンスの変化を定量的に把握することが可能であることをケーススタディによって示すことができた。

第7章 複数の造船設計プロジェクトにおけるエ フォート率設計

7.1 はじめに	144
7.2 問題設定	144
7.2.1 ケーススタディで想定する状況	144
7.2.2 提案手法の設定	147
7.3 結果	150
7.3.1 エフォート率の探索結果	150
7.3.2 人手で考えうる単純な作業戦略案との比較	152
7.4 まとめ	156

7.1 はじめに

本章では、複数の造船設計プロジェクトを並行して行う際のエフォート率案を 3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法を用いて設計し、提案手法の有効性について議論する。なお、本章で使用した入力データに関しても第 6 章と同様に、船舶設計に関する文献[54]や造船所でのインタビュー結果から、仮想的な船舶基本設計プロセスを想定して作成したものである。

2013 年より IMO(International Maritime Organization)によって、造船設計における EEDI(Energy Efficiency Design Index)値の規制が始まった。そのため、船舶の設計段階から水槽試験や EEDI の算出などを行う必要があるように、造船のプロセスが以前とは大きく変わることが予想される。また、一般に、造船会社で造船設計を行う際には単一の船種の設計のみを行うのではなく、複数の船種の設計を並行して行っている。そのため、以前とは大きく変わることが予想される複数の造船設計プロジェクトに対して、単に組織構造だけではなく、会社内の人員を各プロジェクトへ適切に配置する必要がある。

7.2 問題設定

7.2.1 ケーススタディで想定する状況

本章で対象とする設計業務は、図 7-1 のワークフローで表される造船設計プロジェクト 3 つ(P1~P3)を同時並行で行う業務である。ただし、それぞれの船舶基本設計プロジェクトは同一の案件ではなく、ある 1 つの船型を異なる 3 つの客先それぞれの要望に合うようにカスタマイズして製造する状況を想定している。図 7-1 の数字は各タスクの工数を表す。また、各タスクの最大同時担当者数は無制限であるとする。さらに、今回対象とするワークフローでは、「船型」(緑)、「機関」(黄)、「構造」(青)、「艀装」(赤)の 4 つの担当部署が存在しているものとし、それぞれのタスクの担当部署を色で示している。また、作業実施における制約は今回のケーススタディでは特に設けないものとする。また、本章のケーススタディでは、チーム作業シミュレーション上でトラブルを考慮しないため、以降の説明ではトラブル率に関しては記載し

ていない。

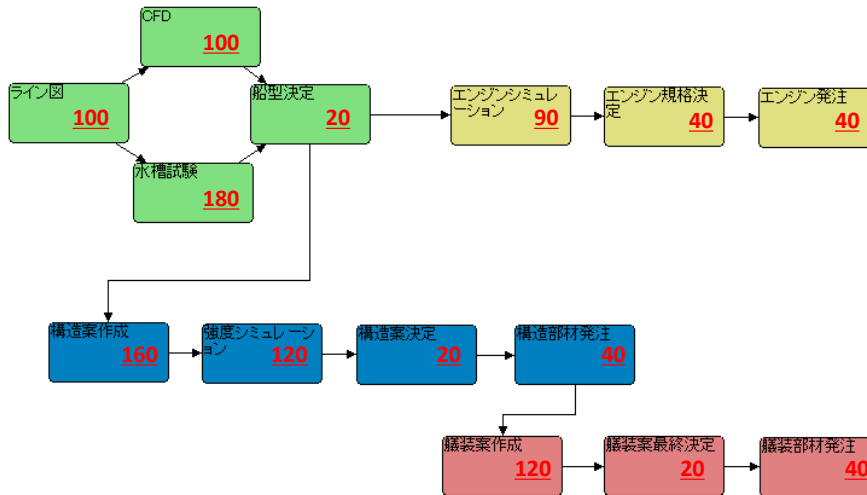


図 7-1 1つの造船設計プロジェクトを表すワークフロー

作業チームの組織構造に関して、人員に関するスキルセットを表 7-1に、人員間の依存関係を図 7-2に示す。今回の作業チームは「船型」・「構造」・「機関」・「艦装」の4つの部署とそれらを統括する「統括」部署で構成された組織構造を対象とする。

表 7-1 造船設計作業チームの人員に関するスキルセット情報

	ライン図	CFD	水槽実験	船型決定	構造案作成	強度シミュレーション	構造案決定	構造部材発注	エンジンシミュレーション	エンジン規格決定	エンジン発注	艤装案作成	艤装案最終決定	艤装部材発注
PM				1.0			1.0						1.0	
H1	1.0			1.0										
H2	1.0	1.0	1.0											
H3		1.0	1.0											
S1					1.0	1.0	1.0	1.0						
S2					1.0	1.0		1.0						
S3						1.0		0.5						
EN1									1.0	1.0	1.0			
EN2									0.5		1.0			
EQ1												1.0	1.0	1.0
EQ2												1.0		1.0

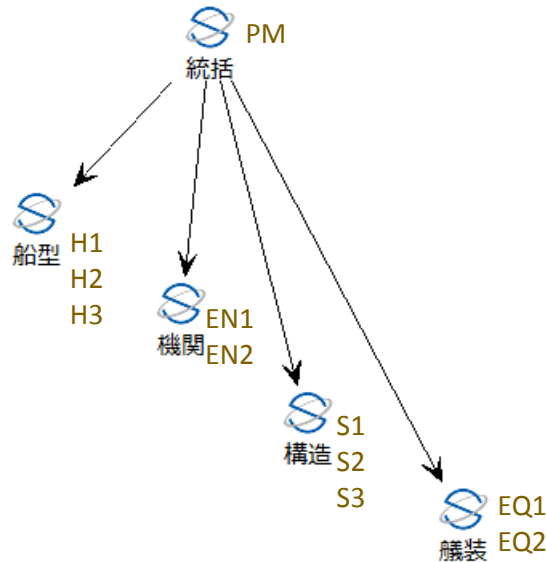


図 7-2 造船設計作業チームの人員に関する依存関係情報

また、船舶基本設計チームを設計するにあたって、今回のケーススタディでは船舶基本設計プロジェクトを計画する際には、すでにプロジェクトに関する詳細な情報が確定しており、あとは作業戦略のみを決定する段階であるものとする。なお、今回のケーススタディでは、単に作業チームのエフォート率案を決定するだけでなく、以下の項目について分析することで、適切なエフォート率案設計のための知見を得ることを主な目的とする。

- 所要時間を短縮するためにはどのようなエフォート率の配分にすべきか
- 人手で考える単純なエフォート率の設定と比較して、提案手法によって算出したエフォート率案がどの程度優れているか

7.2.2 提案手法の設定

今回のケーススタディにおける提案手法の設定を表 7-2 に示す。作業チームの作業戦略のうちエフォート率を設計の対象とし、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法を用いてエフォート率案の設計を行う。また、目的関数として、他の章とは異なり「所要時間」のみを採用し、所要時間のみを考慮してエフォート率案の設計を遺伝的アルゴリズムによ

複数の造船設計プロジェクトにおけるエフォート率設計

る探索により行う。また、提案手法における遺伝的アルゴリズムの設定を個体数 10, 世代数 5000, 交叉率 0.6, 突然変異率 0.3 に設定したうえで、提案手法として探索を行う。

チーム作業シミュレーションの設定に関しては、「タスクの掛け持ちを考慮する」、「工作機械は考慮しない」、「トラブル率に従ってトラブルとその調整を発生させない」とし、チーム作業シミュレーション上でトラブルの発生を考慮しないためチーム作業シミュレーションの繰り返し回数は 1 とした。

表 7-2 第 7 章における提案手法の設定

	設定項目	設定内容
探索の設定	探索手法	遺伝的アルゴリズムによる探索(3.5)
	設計変数	作業戦略 (エフォート率)
	目的関数	● 所要時間
チーム作業シミュレーションの設定	各人員の同時刻におけるタスクの掛け持ち	あり
	タスク実行における工作機械の考慮	なし
	各タスクでランダムに発生するトラブルとその調整の考慮	なし
	チーム作業シミュレーションの繰り返し回数	1

今回のケーススタディにおける作業チームの設計変数と固定する変数の詳細に関する情報を表 7-3 に示す。設計変数として作業戦略のエフォート率を採用し、タスク割り振りルールと組織構造を固定した状態でエフォート率案の探索を行う。

なお、タスク割り振りルールに関しては、クリティカルパス上に近いタスクを優先して行うディスパッチングルールである TSLACK のみを採用する。すなわち、タスク割り振りルールの重

みベクトル成分のうち, TSLACK の成分を 1, そのほかの成分を 0 とする.

表 7-3 第 7 章における作業チームの設計変数と固定する変数

	変数	詳細
組織構造	人員と工作機械のスキルセット	固定(表 7-1)
	人員間の依存関係	固定(図 7-2)
作業戦略	タスク割り振りルール	固定(TSLACK のみ)
	エフォート率	設計変数

7.3 結果

7.3.1 エフォート率の探索結果

提案手法の設定に従って算出されたエフォート率案を表 7-4 に、また表 7-4 のエフォート率案によってチーム作業シミュレーションを行った結果を図 7-3 に示す。横軸に時間を取り、上部に組織の人員がどの時間帯に作業を行っていたかを、下部には設計プロジェクト内の各タスクが実施されていた時間帯を示すガントチャートを表示している。図 7-3 では所属するプロジェクトの異なる同種のタスクの所要時間が異なっているが、これは表 7-4 によって定義されているエフォート率案と、時間ごとのそれぞれの人員の担当タスク状況により、各人員のそれぞれのタスクに取り組む割合が異なるためである。表 7-4 のエフォート率案によって実施した際のチーム作業シミュレーション結果では、所要時間が 680 日という結果になった。

表 7-4 算出されたエフォート率案

	ワークフローへのエフォート率		
	P1	P2	P3
PM	0.2	0.3	0.5
H1	0.9	0.1	0.0
H2	0.8	0.1	0.1
H3	0.7	0.1	0.2
S1	0.1	0.2	0.7
S2	0.1	0.4	0.5
S3	0.1	0.5	0.4
EN1	0.7	0.1	0.2
EN2	0.6	0.4	0.0
EQ1	0.1	0.8	0.1
EQ2	0.1	0.3	0.6

複数の造船設計プロジェクトにおけるエフォート率設計

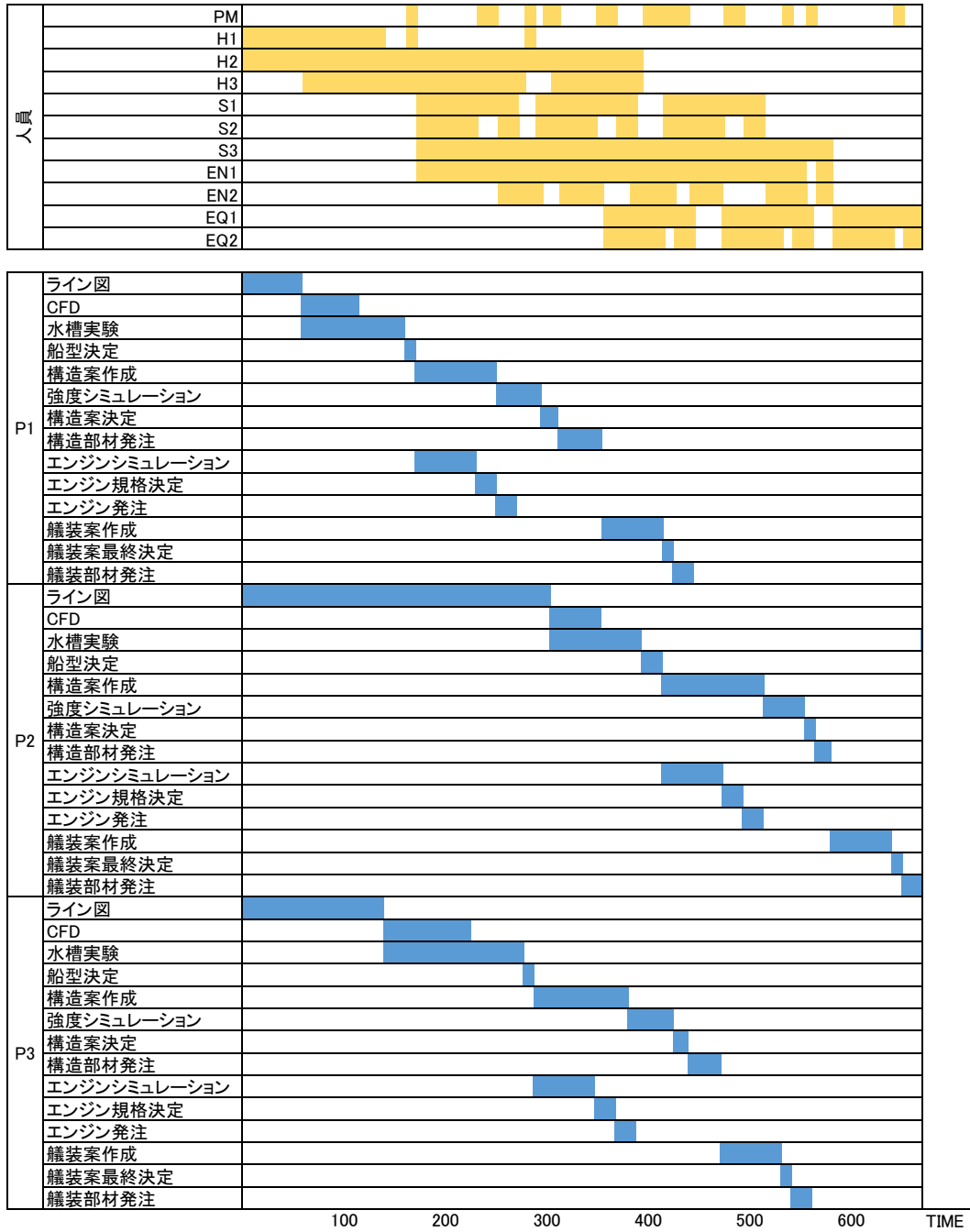


図 7-3 算出されたエフォート率案によるシミュレーション結果

また、チーム作業シミュレーション結果を詳細に分析したところ、図 7-3 のガントチャートにおいてエンジンチームの人員(EN1,EN2)と艀装チームの人員(EQ1,EQ2)については同時刻に 2 つ以上のタスクを担当している時間帯が存在しなかった。すなわち、表 7-4 におけるエンジンチームの人員(EN1,EN2)と艀装チームの人員(EQ1,EQ2)のエフォート率案に関しては、最終的なチーム作業シミュレーションの結果(図 7-3)には影響していないことを示している。また、他の人員(PM,H1,H2,H3,S1,S2,S3)に対しては、少なくとも 1 日は異なる設計プロジェクトのタスクを 2 つ以上担当している時間帯が存在した。すなわち、今回のケーススタディでは彼らの適切なエフォート率案を遺伝的アルゴリズムによって探索しているという結果になった。

7.3.2 人手で考えうる単純な作業戦略案との比較

さらに、本章で得られた結果の評価を行うために、人手で考えうる単純な作業戦略案との比較を行った。具体的には以下の 2 つの作業戦略案を採用した場合におけるチーム作業シミュレーションの結果と、提案手法によって得られたエフォート率案を採用した場合のチーム作業シミュレーションの結果を比較した。

- すべての人員のタスクの掛け持ちを禁止する(エフォート率に依存しない)
- すべての人員の各ワークフローへのエフォート率を均等に配分(1/3)にする

タスクの掛け持ちを禁止した場合のチーム作業シミュレーション結果を図 7-4 に、各ワークフローへのエフォート率を均等に配分した場合のチーム作業シミュレーション結果を図 7-5 に示す。

複数の造船設計プロジェクトにおけるエフォート率設計

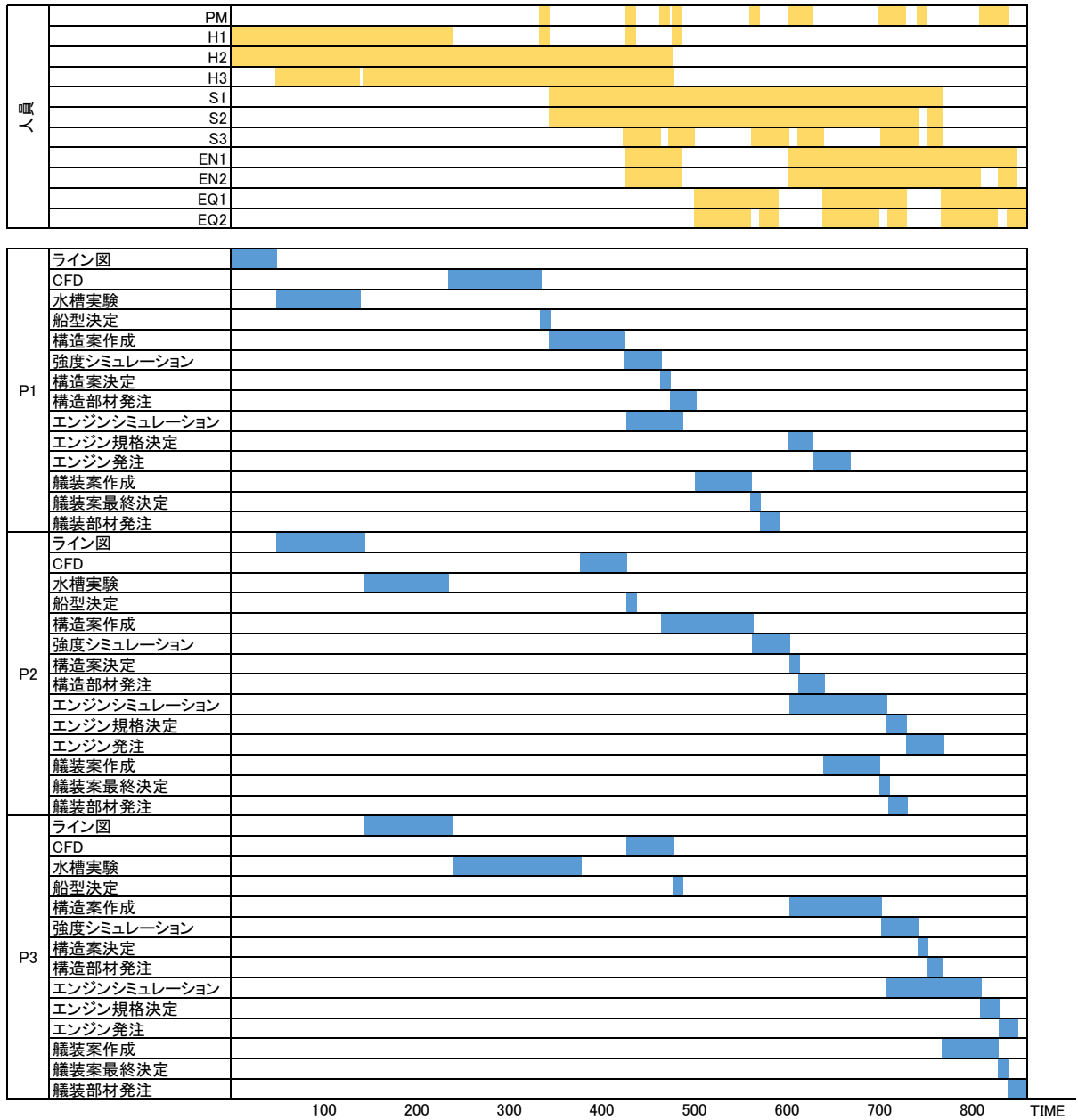


図 7-4 タスクの掛け持ちを禁止した場合のシミュレーション結果

複数の造船設計プロジェクトにおけるエフォート率設計

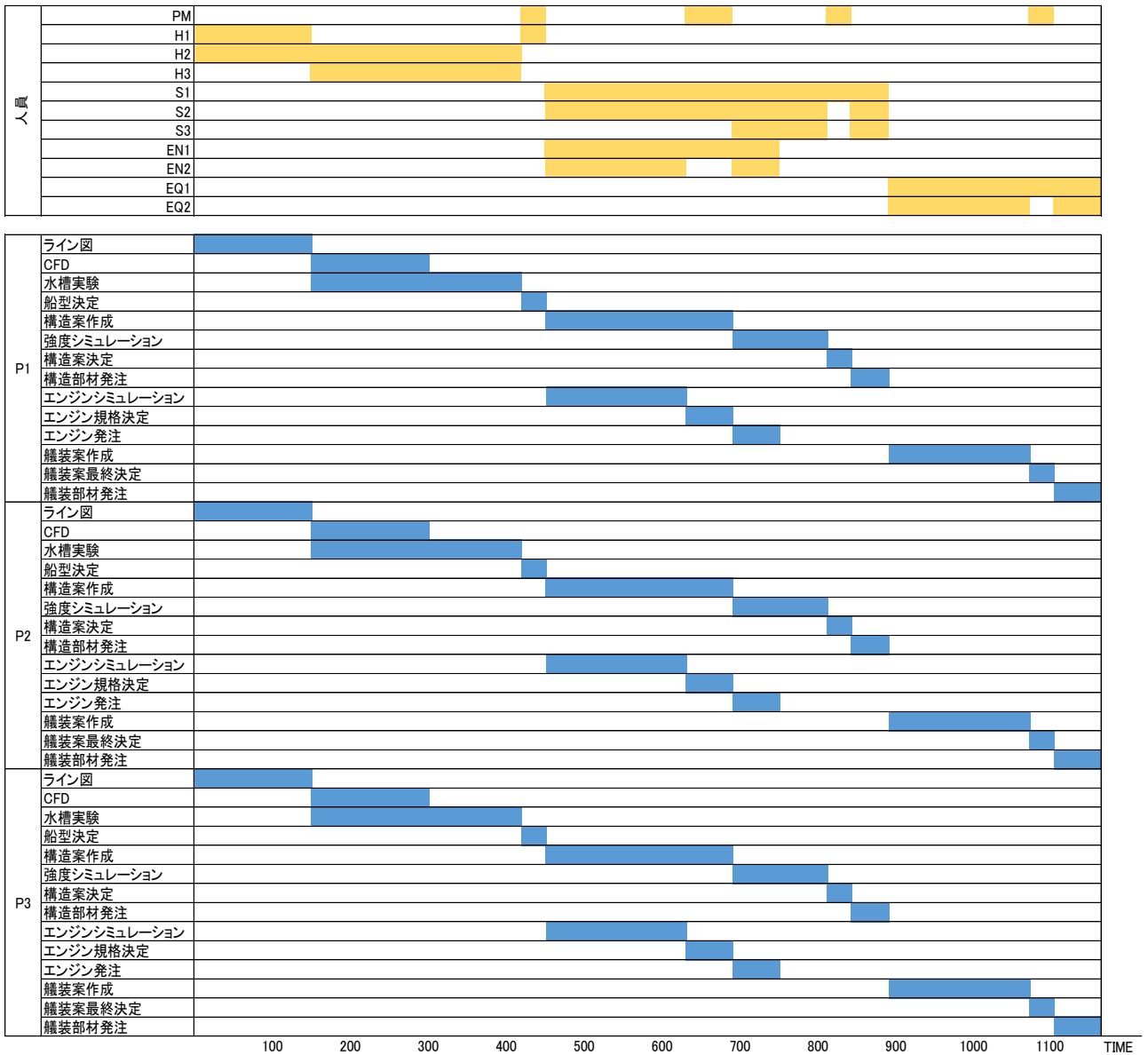


図 7-5 エフォート率を均等に配分した場合のシミュレーション結果

図 7-4 のチーム作業シミュレーション結果では所要時間が 860 日、図 7-5 のチーム作業シミュレーション結果では所要時間が 1164 日という結果になった。また、本章で行った 3 つのチーム作業シミュレーション結果において、時間ごとの作業人員数の推移を表したものを図 7-6 に示す。「タスクの掛け持ち禁止」と「各ワークフローへのエフォート率を同じに設定」という 2 つの作業戦略案は、人手で作業戦略案を考える際の有力な候補となりえるが、図 7-6

ではこれら 2 つの作業戦略案に比べて提案手法によって算出されたエフォート率案では前倒しで必要な作業を行えていることが分かる. 以上のことから, 今回のケーススタディでは所要時間の最小化を目指すという目的の中で, 提案手法は人手で考える単純な作業戦略案よりも良い案を見つけることができたと言える.

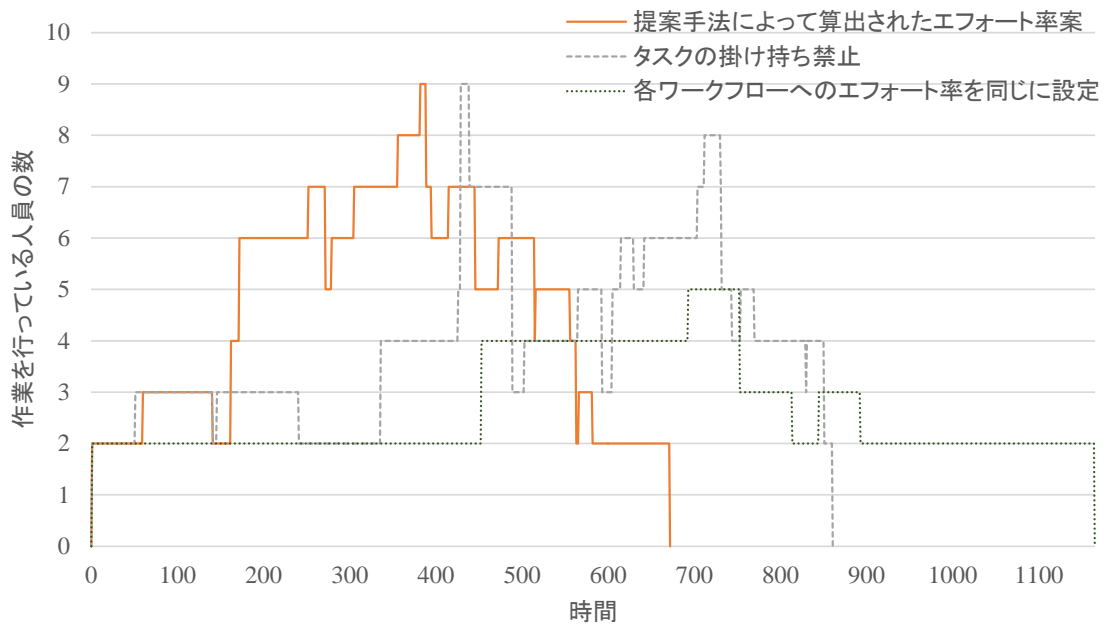


図 7-6 各ケーススタディにおける単位時間ごとの作業人員数の推移

7.4 まとめ

本章では 3.5 で提案した遺伝的アルゴリズムによる作業チーム設計支援手法を、複数の造船設計業務に対するエフォート率案作成のケーススタディに適用することで、その評価を行った。本章のケーススタディを通して得られた知見を以下にまとめる。

- 提案した遺伝的アルゴリズムによる作業チーム設計支援手法は、人手で考える作業戦略案に比べて所要時間を短縮することが可能なエフォート率案を算出できた。
- 複数のプロジェクトを 1 つの作業チーム内で同時に進行する必要がある場合に、エフォート率の適切な設定によって作業チームのパフォーマンスを向上させることが可能であることを、提案手法により定量的に示すことができた。

第8章 船殻ブロック組立におけるタスク割り振りルール設計

8.1 はじめに.....	158
8.2 問題設定.....	158
8.2.1 ケーススタディで想定する状況.....	158
8.2.2 提案手法の設定.....	161
8.3 結果.....	163
8.3.1 電力制限の有無によるタスク割り振りルールの比較.....	163
8.3.1.1 同時実行ワークフロー数が3の場合.....	163
8.3.1.2 同時実行ワークフロー数が2の場合.....	166
8.3.1.3 同時実行ワークフロー数の違いにおける考察.....	169
8.3.2 トラブル考慮の有無によるタスク割り振りルールの比較.....	170
8.3.2.1 電力制限がない場合.....	170
8.3.2.2 電力制限がある場合.....	172
8.3.2.3 電力制限の有無による比較.....	174
8.4 まとめ.....	176

8.1 はじめに

本章では、造船所における船殻ブロック組み立て工程を対象として、3.5の遺伝的アルゴリズムによる作業チーム設計支援手法を用いて、電力ピークカットの状況下におけるタスク割り振りルール設計のケーススタディを行う。なお、本章で使用した入力データに関しては、船殻ブロック組み立てに関する文献[55]や、造船所でのインタビュー結果をもとに、仮想的な船殻ブロック組み立て工程を想定して作成したものである。

2011年に発生した東日本大震災の影響で、経済産業省は夏期の電力需要が高い時期に前年比で使用最大電力の15%削減を工場に対して求めている。そのため、全国の工場では電力ピークカットを考慮した適切な作業計画と作業戦略の再設計が急務となっている。

8.2 問題設定

8.2.1 ケーススタディで想定する状況

本章で対象とする船殻ブロック組み立て業務は、図 8-1 のワークフローで1枚分の組み立て工程が表される船殻ブロックを、タクトタイムが25時間で10枚組み立てる業務である。今回のケーススタディでは、問題の簡略化のため各タスクの工数はすべて1時間であるものとし、どの船殻ブロックから組み立ててもよいものとする。チーム作業シミュレーション上でトラブルを考慮する場合には、各タスクでのトラブル率を0.10に設定する。担当部署に関しては、図 8-1 における「外付パネル」と「内部構造パネル」で括られているタスクは「中組」、「ブロック」で括られているタスクは「大組」が担当部署として登録されているものとする。また、作業実施における制約については本章の各ケーススタディによって変更するため後ほど説明する。

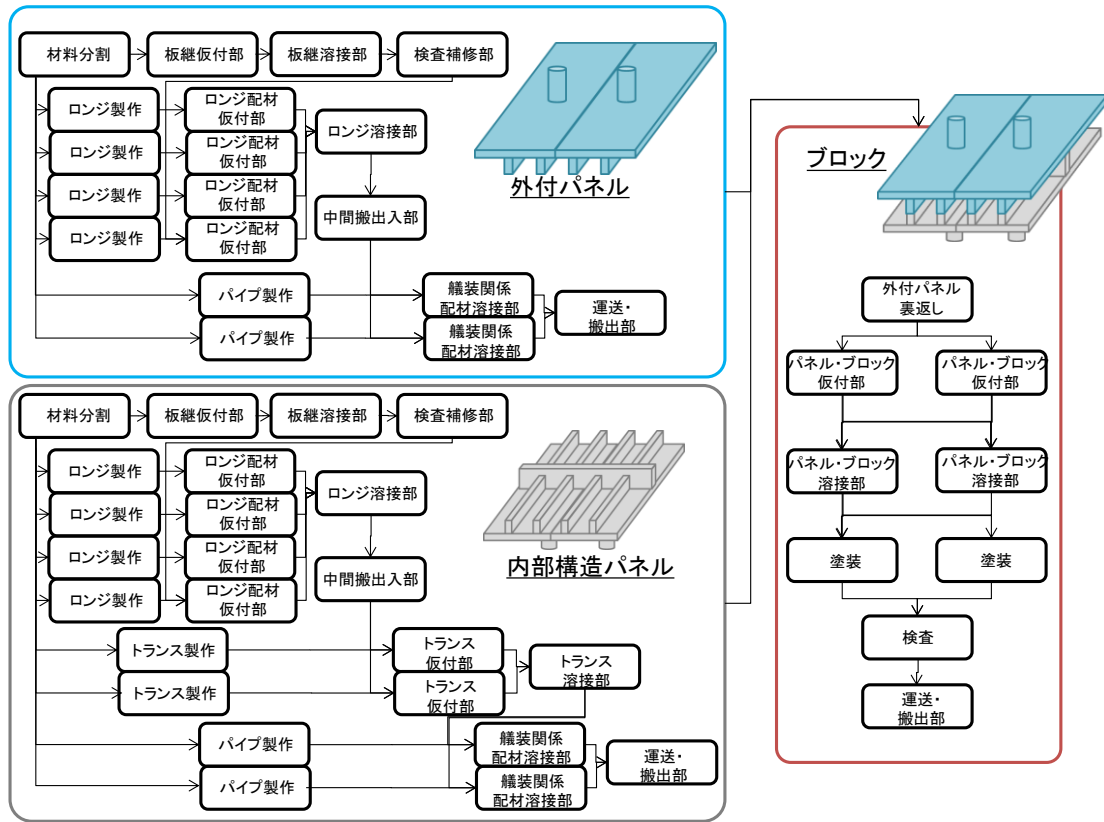


図 8-1 1 枚の船殻ブロック組み立て工程を示すワークフロー

対象とする作業チームの人員と工作機械に関するスキルセットを表 8-1 に、人員間の依存関係を図 8-2 に示す. 対象とする作業チームでは、4 人の人員から構成される「中組」部署と、3 人の人員で構成される「大組」部署が存在し、それぞれの部署で必要な工作機械が 16 個存在する. 今回のケーススタディでは 2 つの部署間の関係は独立であり、部署間の依存関係は存在しないものとする.

表 8-1 船殻ブロック組み立て作業チームの人員と工作機械に関する情報

	固定給 [円/時]	変動給 [円/時]	使用電力 [kW/時]	トラブ ル率	タスク										その他																									
					材料 分割	板継 仮付部	板継 溶接部	検査・ 補修部	ロン ジ製 作	ロン ジ配 材仮 付部	ロン ジ溶 接部	中 間搬 出部	ト ラ ン ス 製 作	ト ラ ン ス 配 材 仮 付 部	ト ラ ン ス 溶 接 部	パイ プ製 作	艀 装 係 配 材 溶 接 部	運 送・ 搬出 部	外 付 パ ネ ル 裏 返 し	パ ネ ル・ ブ ロ ッ ク 仮 付 部	パ ネ ル・ ブ ロ ッ ク 溶 接 部	塗 装	検 査																	
中組worker1	0	1300	0	0.10	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
中組worker2	0	1300	0	0.10	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
中組worker3	1800	0	0	0.05	1	1	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0		
中組worker4	1800	0	0	0.05	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
大組worker1	0	1300	0	0.10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
大組worker2	1800	0	0	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	
大組worker3	2500	0	0	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
手動溶接機1	700	500	10	0.10	0	1	0	1	0	1	0	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
手動溶接機2	700	500	10	0.10	0	1	0	1	0	1	0	0	0	0	1	1	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
手動溶接機5	700	500	10	0.10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
手動溶接機6	700	500	10	0.10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
パイプ製作機1	600	500	10	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
パイプ製作機2	600	500	10	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
塗装スプレー1	1500	700	20	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
塗装スプレー2	1500	700	20	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
ガス切断機	2000	500	90	0.15	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
プラズマ切断機	2200	1000	60	0.05	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ラインウェルダ	2500	800	80	0.05	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FCB片面自動溶接機	2500	1000	100	0.05	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
クレーン(中組)	3500	1000	100	0.05	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
クレーン(大組)	3500	1000	120	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
検査機器	0	0	0	0.00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
					中組										大組																									



中組

- 中組worker 1
- 中組worker 2
- 中組worker 3
- 中組worker 4



大組

- 大組worker 1
- 大組worker 2
- 大組worker 3

図 8-2 ブロック組立作業チームの人員に関する依存関係情報

また、船殻ブロック組み立てにおけるタスク割り振りルールを設計するにあたって、今回のケーススタディでは石斛ブロック組み立てプロジェクトを計画する際に、すでにプロジェクトに関する詳細な情報が確定しており、あとは作業戦略のみを決定する段階であるものとする。なお、今回のケーススタディでは、単にタスク割り振りルール案を決定するだけでなく、以下の項目について分析することで適切なタスク割り振りルール設計のための知見を得ることを主な目的とする。

- 電力ピークカット前後における適切なタスク割り振りルールの違いを比較(8.3.1)
- トラブルを考慮するか否かによる適切なタスク割り振りルールの違いを比較(8.3.2)

8.2.2 提案手法の設定

今回のケーススタディにおける提案手法の設定を表 8-2 と表 8-3 に示す。

今回のケーススタディでは作業チームの作業戦略のうちタスク割り振りルールを設計の対象とし、3.5 の遺伝的アルゴリズムによる作業チーム設計支援手法を用いて、タスク割り振りルールの設計を行う。また目的関数として、チーム作業シミュレーション上でトラブルを考慮しない場合は「合計コスト」を、トラブルを考慮する場合は「合計コスト平均」と「合計コスト標準偏差」を採用してタスク割り振りルールの探索を行う。また、提案手法における遺伝的アルゴリズムの設定はいずれの場合においても個体数 10、世代数 5000、交叉率 0.6、突然変異率 0.3 として探索を行う。

作業実施における制約とチーム作業シミュレーションの設定に関しては、表 8-3 のようにシナリオごとに変更して対応する。また、チーム作業シミュレーションの単位時間は 1 時間とし、トラブルを考慮する場合のチーム作業シミュレーションの繰り返し回数は 10 とした。

表 8-2 第 8 章における探索手法の設定

		設定項目	設定内容
探索の設定	探索手法		遺伝的アルゴリズムによる探索(3.5)
	設計変数		作業戦略 (タスク割り振りルール)
	目的関数		トラブルを考慮しない場合 ● 合計コスト トラブルを考慮する場合 ● 合計コスト平均 ● 合計コスト標準偏差

表 8-3 各シナリオの設定

シナリオ名	作業実施に関する制約		チーム作業シミュレーションの設定		
	電力制限	同時実行ワークフロー数	各人員の同時刻におけるタスク掛け持ち	タスク実行における工作機械の考慮	各タスクででランダムに発生するトラブルとその調整の考慮
Case 1	なし	3	なし	あり	なし
Case 2	Case 1 の 85%	3			なし
Case 3	なし	2			なし
Case 4	Case 1 の 85%	2			なし
Case 5	なし	2			あり
Case 6	Case 5 の 85%	2			あり

今回のケーススタディにおける作業チームの設計変数と固定する変数の詳細に関する情報を表 8-4 に示す. 設計変数として作業戦略のうちタスク割り振りルールを採用し, エフォート率と組織構造を固定した状態でタスク割り振りルールの探索を行う.

表 8-4 第 8 章における作業チームの設計変数と固定する変数

	変数	詳細
組織構造	人員と工作機械のスキルセット	固定(表 8-1)
	人員間の依存関係	固定(図 8-2)
作業戦略	タスク割り振りルール	設計変数
	エフォート率	固定(タスクの掛け持ちを考慮しないため影響しない)

8.3 結果

8.3.1 電力制限の有無によるタスク割り振りルールの比較

8.3.1.1 同時実行ワークフロー数が 3 の場合

Case1 のチーム作業シミュレーションを行った結果を表すガントチャートを図 8-3 に示す。図 8-3 では、同一船殻ブロック内の各タスクを担当した場合に同じ色で塗りつぶして表示している。所要時間は 142 時間、合計コストは 4,911,600 円となった。また単位時間あたりの最大電力使用量は 280kW であった。

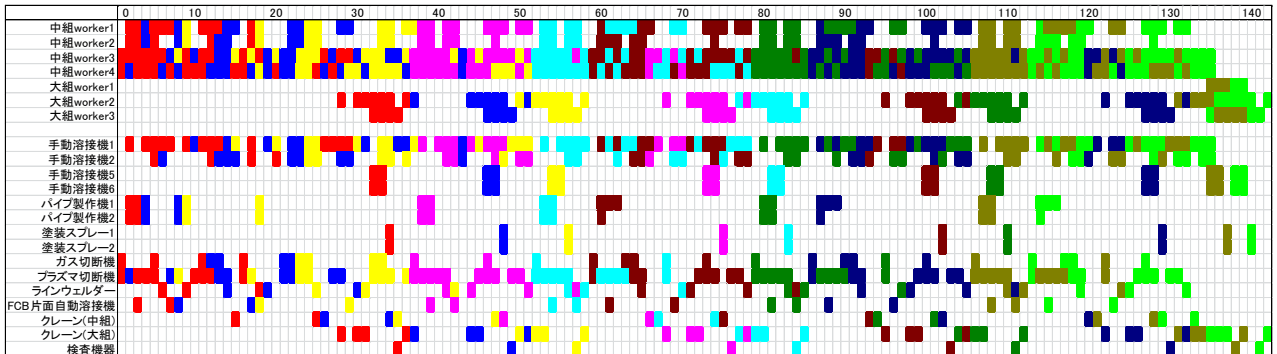


図 8-3 Case1 のチーム作業シミュレーション結果

次に、Case 2 ではその 85%に相当する 238kW を単位時間あたりに使用可能な電力使用量の最大値として設定した場合を考える。Case 2 のチーム作業シミュレーションを行った結果を表すガントチャートを図 8-4 に示す。所要時間は 142 時間、合計コストは 4,914,300 円となった。また単位時間あたりの最大電力使用量は 230kW であった。

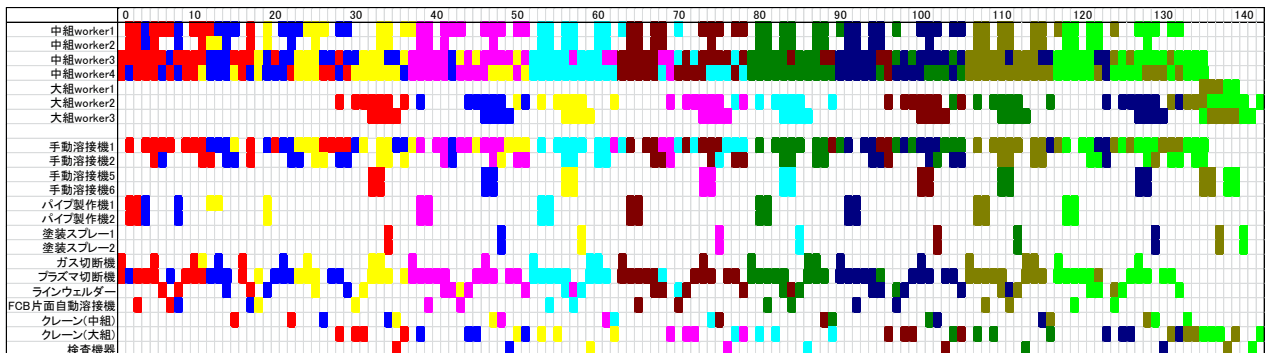


図 8-4 Case2 のチーム作業シミュレーション結果

Case 1 と Case 2 での各時刻における使用電力量の推移を図 8-5 に示す。図 8-5 より、Case 2 では Case 1 に比べて 2,700 円のコスト増を受け入れることで、所要時間を増加させることなく電力ピークカットを考慮したチーム作業の実施が可能であることが分かる。

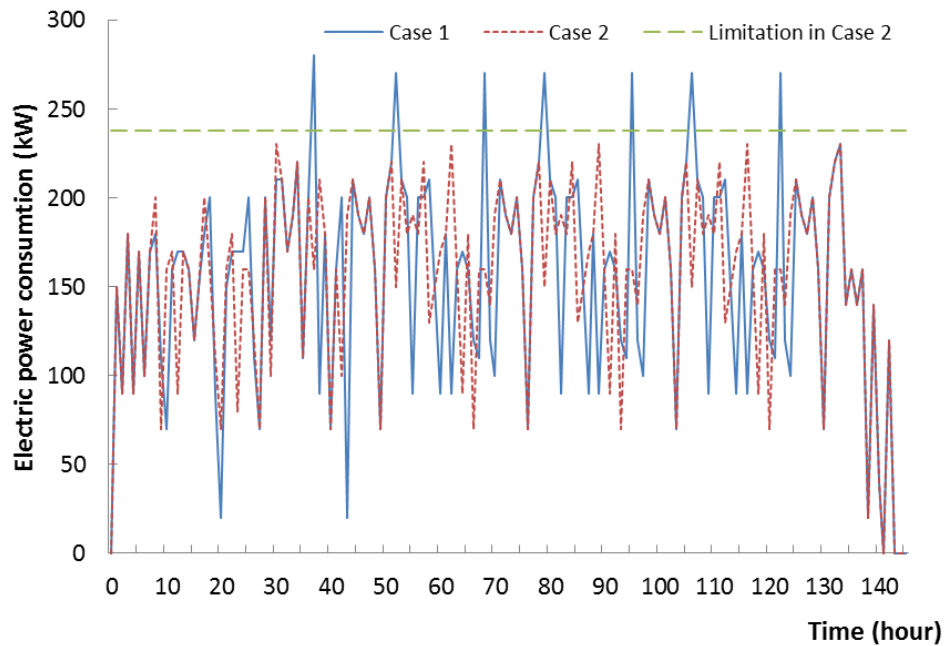


図 8-5 Case1 と Case2 での使用電力量の推移

また、3.5.5 の手法を用いてタスク割り振りルールの変化を分析した結果を表 8-5 に示す。Case 2 では Case 1 に比べて SPT・FIFO・EC が重視され、VC と SSP が軽視されていることが分かる。すなわち、電力制限が課せられたことによって、作業可能になったタスクに対して即座に電力使用の少ない設備を優先的に割り振るようにタスク割り振りルールが変化したことが分かる。また、一方で今回の問題ではすべてのタスクの工数を 1 としているため、SPT の値の変化には意味がないが、Case 2 では Case 1 に比べて SPT が重視されている結果になった。このことから、Case 1 と Case 2 では本質的にはタスク割り振りルールの変化には大きな違いがないと言える。

表 8-5 Case 1 と Case 2 のタスク割り振りルールの変化分析

ルール	log 値		
	Case 1	Case 2	差(正規化)
EDD	-2.862	-3.114	0.173
FIFO	-5.765	-5.316	0.658
SPT	-1.293	-0.207	1.099
EST	-0.629	-0.915	0.149
TSLACK	-1.038	-1.378	0.112
SPN	-1.130	-1.629	0.002
SSP	-0.330	-1.047	-0.148
VC	-6.879	-10.947	-2.468
EC	-1.101	-0.992	0.422

8.3.1.2 同時実行ワークフロー数が 2 の場合

次に、工場内に新規の施設を建設するなどの理由により Case1 と Case2 に比べて工場の作業面積が 2/3 に削減された場合において、電力制限の有無によるタスク割り振りルールの比較を行う。

Case1 から作業面積を削減した Case3 に関して、チーム作業シミュレーションを行った結果を表すガントチャートを図 8-6 に示す。所要時間は 143 時間、合計コストは 4,923,200 円となった。また単位時間あたりの最大電力使用量は 330kW であった。

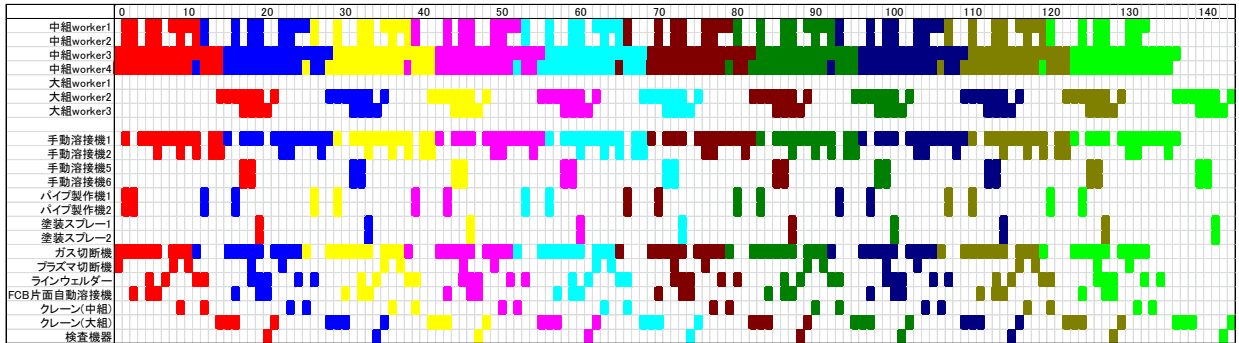


図 8-6 Case3 のチーム作業シミュレーション結果

次に、作業面積が削減される前の状況を想定して 15%の電力ピークカットを課された場合を考える。具体的には、同時実行ワークフロー数が 2 個かつ Case2 と同じ 238kW を単位時間あたりに使用可能な電力使用量の最大値として設定した場合を Case 4 として取り扱う。

Case 4 のチーム作業シミュレーション結果を図 8-7 に示す。所要時間は 144 時間、合計コストは 4,959,000 円となった。また単位時間あたりの最大電力使用量は 230kW であった。

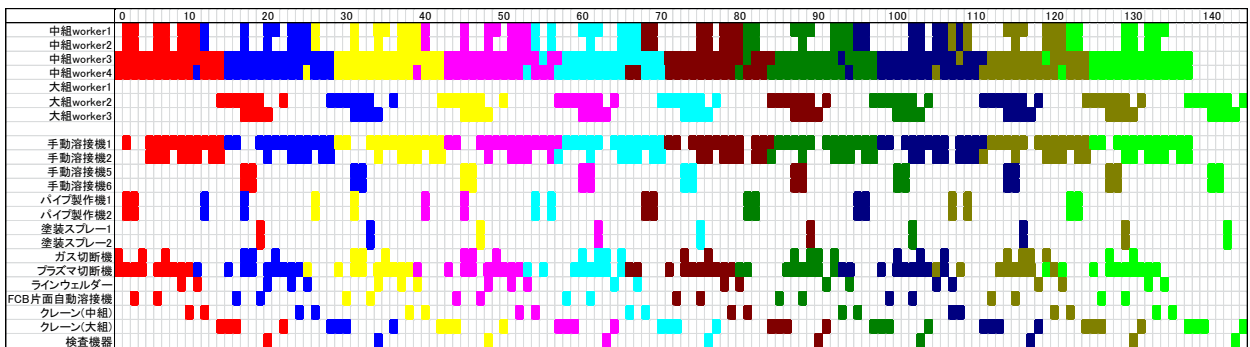


図 8-7 Case4 のチーム作業シミュレーション結果

Case 3 と Case 4 での各時刻における使用電力量の推移を図 8-8 に示す。図 8-8 より、Case 4 では Case 3 に比べて 35,800 円のコスト増と所要時間の 1 時間延長を受け入れることで、電力ピークカットを考慮したチーム作業の実施が可能であることが分かる。

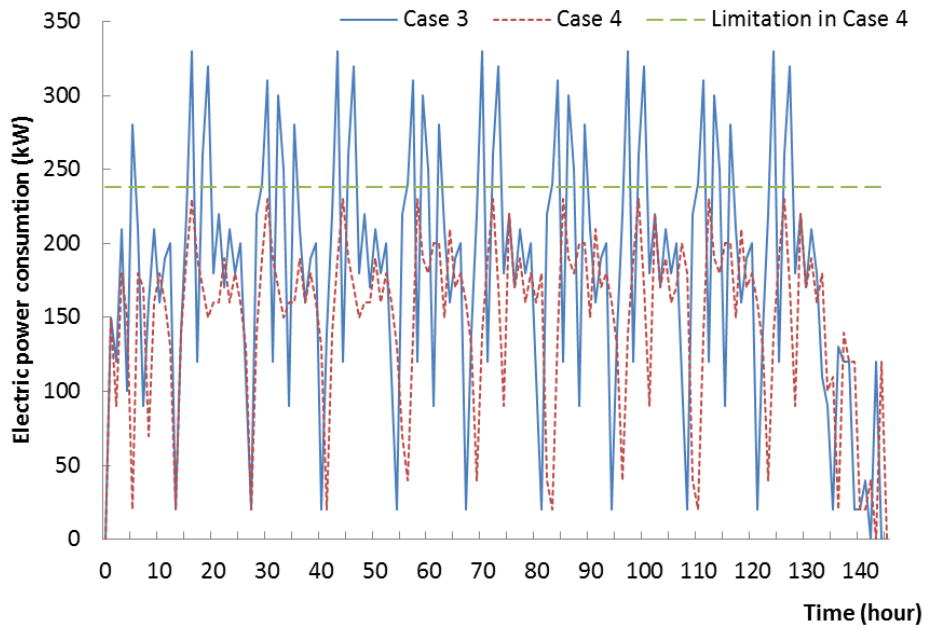


図 8-8 Case3 と Case4 での使用電力量の推移

また, 3.5.5 の手法を用いてタスク割り振りルールの変化を分析した結果を表 8-6 に示す. Case 4 では Case 3 に比べて EC が重視され, SPN・EDD・FIFO が軽視されていることが分かる. すなわち, Case 4 では Case 3 と比較して電力消費の小さい設備をより優先的に使用するようタスク割り振りルールが変化したことが分かる.

表 8-6 Case 3 と Case 4 のタスク割り振りルールの変化分析

ルール	log 値		
	Case 3	Case 4	差(正規化)
EDD	-1.406	-2.861	-0.550
FIFO	-2.856	-3.879	-0.407
SPT	-0.435	-0.742	-0.170
EST	-0.563	-1.165	-0.268
TSLACK	-0.926	-1.744	-0.340
SPN	-0.813	-3.297	-0.891
SSP	-1.337	-0.697	0.143
VC	-4.332	-4.318	-0.064
EC	-8.181	-0.276	2.549

8.3.1.3 同時実行ワークフロー数の違いにおける考察

本研究では、作業面積をチーム作業シミュレーション中の同時実行ワークフロー数によって表現しているが、作業面積の異なる 8.3.1.1 と 8.3.1.2 ではタスク割り振りルールの変化に関する傾向が異なる結果となった。本節ではこの点に関する考察を行う。

Case 1 から Case 4 までの各ケーススタディにおいて、電力消費量のピーク時に使用されていた工作機械についての情報を表 8-7 に示す。電力消費量のピーク時には、工作機械の中で最も使用電力量の大きい「クレーン(大組)」が使用されていることが分かる。また、電力ピークカット考慮時では考慮前と比較して、「ガス切断機」や「プラズマ切断機」を電力消費量のピーク時に使用しないように変化している傾向が読み取れる。

表 8-7 電力消費量ピーク時に使用されている工作機械

作業面積 電力ピークカット 考慮	同時実行ワークフロー数:3 (8.3.1.1)				同時実行ワークフロー数:2 (8.3.1.2)			
	なし(Case 1)		あり(Case 2)		なし(Case 3)		あり(Case 4)	
電力消費量ピーク時に使用されている工作機械	クレーン(大組)	120	クレーン(大組)	120	クレーン(大組)	120	クレーン(大組)	120
	ガス切断機	90	FCB片面自動溶接機	100	FCB片面自動溶接機	100	FCB片面自動溶接機	100
	プラズマ切断機	60	手動溶接機	10	ガス切断機	90	手動溶接機	10
	手動溶接機	10			パイプ製作機	10		
電力消費合計 (kW)		280		230		330		230

また、表 8-7 と表 8-1 における各工作機械のスキル情報やチーム作業シミュレーション結果を比較すると、Case 2 では Case 1 に比べて単に行うタスクの優先順位を少し変更するだけで電力ピークカットに対応可能であるが、Case 4 では Case 3 に比べてタスクの優先順位を変更するだけでは対応できず、リソース側である工作機械の優先順位を変更する必要があるという結果になった。特に、Case 1 に比べて Case 2 のチーム作業シミュレーション結果における同時実行ワークフロー数が増えていることを考慮すると、Case 2 では余分な作業面積を使用することで電力消費量のピークカットに対応しているが、Case 4 では Case 2 で存在した余分な作業面積が存在しないため、電力消費量の低い工作機械を優先的に使用することで対応していることが分かる。

8.3.2 トラブル考慮の有無によるタスク割り振りルールの比較

8.3.2.1 電力制限がない場合

Case3 において各タスクを実行する際にトラブル率に従ってトラブルが起きる状況を想定した場合(Case 5)と、トラブルを想定しない場合(Case 3)のタスク割り振りルールの比較を行う。

Case 5 において提案手法によりタスク割り振りルールの探索を行った結果を図 8-9 に示す。図 8-9 では、合計コストの平均と標準偏差に対するパレートフロントを構成するタスク割り振りルールが 4 つ含まれていた。また、トラブルを考慮しない場合である Case3 との結果を比較すると、Case5 の方が合計コストが約 90 万円ほど高くなるという結果になった。また、Case3 で設計したタスク割り振りルールを Case5 で適用した結果も図 8-9 に示しているが、図 8-9

から提案手法ではトラブルを考慮しない場合のタスク割り振りルールに比べて適切なタスク割り振りルールを設計することが可能なことが分かる。

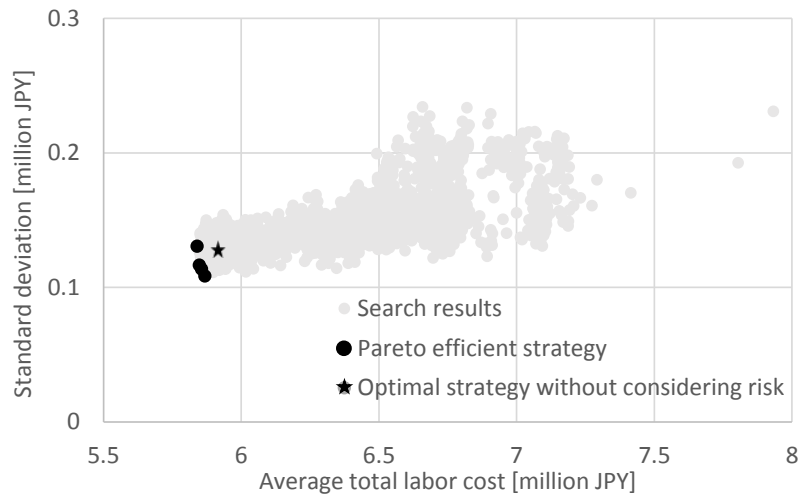


図 8-9 Case5 におけるタスク割り振りルールの探索結果

また、図 8-9 におけるパレートフロントを構成するタスク割り振りルール(P1-1～P1-4)の合計コストに関する情報と Case3 で求めたトラブルを考慮しない場合のタスク割り振りルールとの比較分析の結果を表 8-8 に示す。タスク割り振りルール P1-1 と P1-2 では、トラブルを考慮する場合のタスク割り振りルールではトラブルを考慮しない場合のタスク割り振りルールに比べて EDD と FIFO が重視され、SPT・EST・TSLACK が軽視されていることが分かる。言い換えると、トラブルが起きる場合を想定して先に作業を開始したワークフロー内の作業可能なタスクを優先して行うようにタスク割り振りルールが変化したという結果になった。また、タスク割り振りルール P1-3 と P1-4 では、トラブルを考慮しない場合のタスク割り振りルールに比べて VC が重視されていることが分かる。今回のケーススタディでは表 8-1 で示す通り、コストが高い人員や工作機械はトラブル率が低い値に設定されている。そのため、トラブルのリスクを軽減するために、コストの高い人員や工場設備を優先的に使うようにタスク割り振りルールが変化したという結果になった。

表 8-8 Case3 と比較した場合の Case 5 のタスク割り振りルールの変化分析

	合計コスト[百万円]		タスク割り振りルール(log値・正規化済み)								
	平均	標準偏差	EDD	FIFO	SPT	SST	TSLA CK	SPN	SSP	VC	EC
P1-1	5.840	0.131	0.90	1.82	-1.38	-0.82	-0.85	0.76	0.64	-0.67	-0.39
P1-2	5.848	0.116	0.64	1.80	-1.52	-0.83	-0.97	1.18	-0.11	-0.11	-0.09
P1-3	5.856	0.114	0.71	-1.08	-0.87	-0.67	-0.71	-0.04	0.20	2.36	0.12
P1-4	5.868	0.109	0.14	-0.04	-0.80	-0.78	-0.79	0.08	-0.20	2.64	-0.26

8.3.2.2 電力制限がある場合

Case4 において各タスクを実行する際にトラブル率に従ってトラブルが起きる状況を想定した場合(Case 6)と、トラブルを想定しない場合(Case 4)のタスク割り振りルールの比較を行う。これは、単に 8.3.2.1 と比べて、電力ピークカットを考慮した場合のケーススタディとなる。

Case 6 において提案手法によりタスク割り振りルールの探索を行った結果を図 8-10 に示す。図 8-10 では、合計コストの平均と標準偏差に対するパレートフロントを構成するタスク割り振りルールが 8 つ含まれていた。また、トラブルを考慮しない場合である Case4 との結果を比較すると、Case6 の方が合計コストが約 100 万円ほど高くなるという結果になった。また、Case4 で設計したタスク割り振りルールを Case6 で適用した結果も図 8-10 に示しているが、8.3.2.1 と同様に、提案手法ではトラブルを考慮しない場合のタスク割り振りルールに比べて適切なタスク割り振りルールを設計することが可能なことが確認できる。

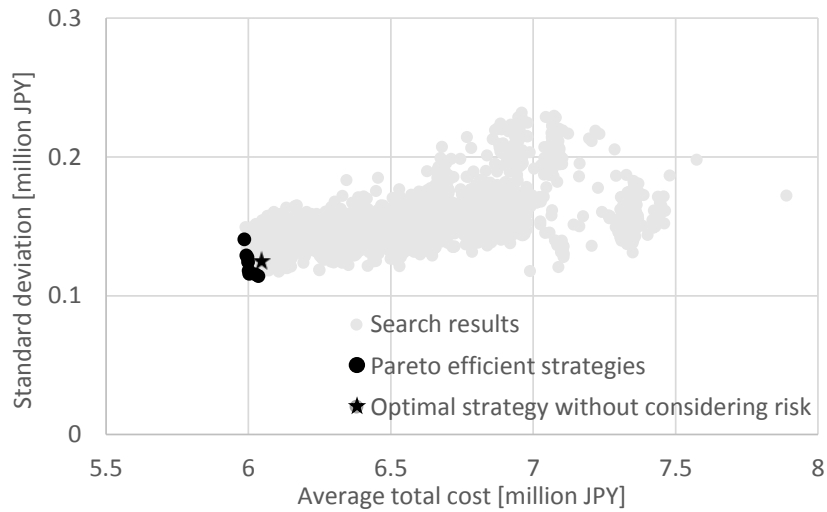


図 8-10 Case6 におけるタスク割り振りルールの探索結果

また、図 8-10 におけるパレートフロントを構成するタスク割り振りルール(P2-1～P2-8)の合計コストに関する情報と Case4 で求めたトラブルを考慮しない場合のタスク割り振りルールとの比較分析の結果を表 8-9 に示す。共通する傾向として、トラブルを考慮する場合のタスク割り振りルールではトラブルを考慮しない場合のタスク割り振りルールに比べて SPT と EDD が重視され、FIFO・TSLACK が軽視されていることが分かる。言い換えると、先に作業を開始したワークフロー内のタスクを優先的に行うようにタスク割り振りルールが変化したという結果になった。ただし、こちらも 8.3.1.1 と同様の理由で SPT の値には意味がないため、トラブルの考慮によるタスク割り振りルールの変化は少ないことが分かる。すなわち、Case4 ではすでに同時ワークフロー数が最大で 2 個という厳しい制約が課されているため、トラブルを考慮するための余裕があまりないことを示している。

表 8-9 Case4 と比較した場合の Case 6 のタスク割り振りルールの変化分析

	合計コスト[百万円]		タスク割り振りルール(log値・正規化済み)								
	平均	標準偏差	EDD	FIFO	SPT	SST	TSLA CK	SPN	SSP	VC	EC
P2-1	5.986	0.141	0.93	-1.45	1.55	-0.88	-1.15	-0.46	-0.12	0.64	0.94
P2-2	5.994	0.129	0.81	-1.16	1.74	-0.58	-0.62	-1.51	0.12	0.95	0.24
P2-3	5.997	0.128	1.42	-1.06	1.52	-0.55	-0.79	-1.37	-0.16	0.82	0.19
P2-4	5.999	0.124	0.80	-0.88	1.87	-0.39	-0.62	-0.20	-1.51	-0.17	1.10
P2-5	6.001	0.118	1.57	-1.32	1.68	-0.72	-0.77	-0.10	-0.71	0.56	-0.18
P2-6	6.003	0.116	0.81	-1.12	1.72	-0.55	-0.82	-1.46	0.23	0.95	0.25
P2-7	6.025	0.115	0.61	-0.80	1.72	-0.42	-0.60	0.34	-0.02	-1.83	1.01
P2-8	6.036	0.114	1.15	-1.39	1.88	-0.61	-1.09	-0.18	-0.41	0.67	-0.02

8.3.2.3 電力制限の有無による比較

電力制限を考慮しない場合(Case 5)と考慮した場合(Case 6)において、パレートフロントを構成するタスク割り振りルール群のパフォーマンスがどのように変化したかを可視化したものを図 8-11 と図 8-12 に示す。図 8-11 は電力制限が新たに付加されたことによってパレートフロントがどのように移動したかを示しているが、トラブルが与える影響はあまり変化なく単に合計コスト平均が 20 万円ほど大きくなることが分かる。また、図 8-12 では Case 5 と Case 6 で得られたタスク割り振りルールによってチーム作業シミュレーションを行い、合計コストの平均から、標準偏差の 2 倍の値を追加削減した値によってチーム作業シミュレーション結果の有効範囲を定義したものである。図 8-12 より、各ケース内で得られたタスク割り振りルールから得られたチーム作業シミュレーション結果には大きな差がないこと、電力制限を考慮した場合(Case 6)で得られたタスク割り振りルールから得られるチーム作業シミュレーション結果のうちトラブル発生回数が低かった場合の結果は、電力制限を考慮しない場合(Case 5)で得られたタスク割り振りルールから得られるチーム作業シミュレーション結果のうちトラブル発生回数が多かった場合に比べて合計コストが低いという結果になった。よって、今回のケーススタディでは電力制限はプロジェクトにそれほど大きな影響を与えるものではないと判断可能である。

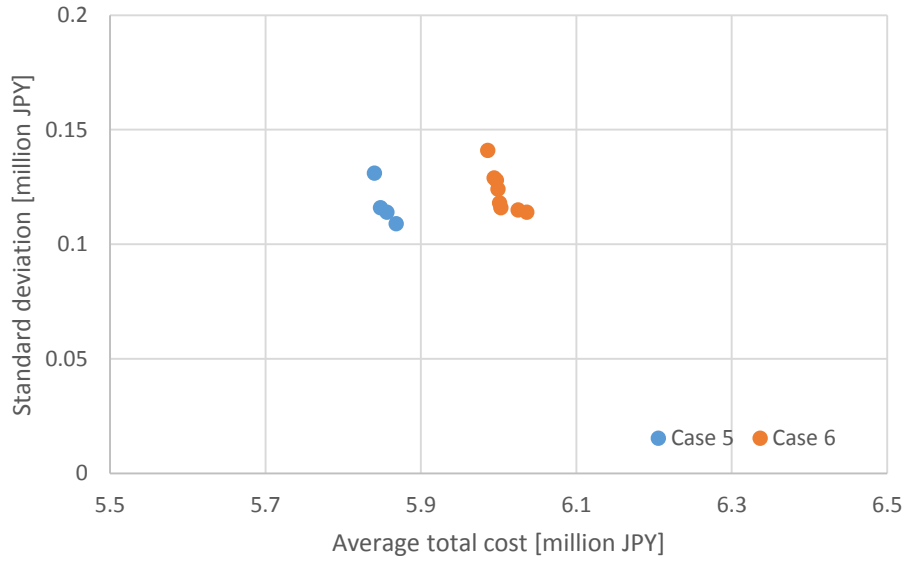


図 8-11 電力制限の有無によるパレートフロントの比較

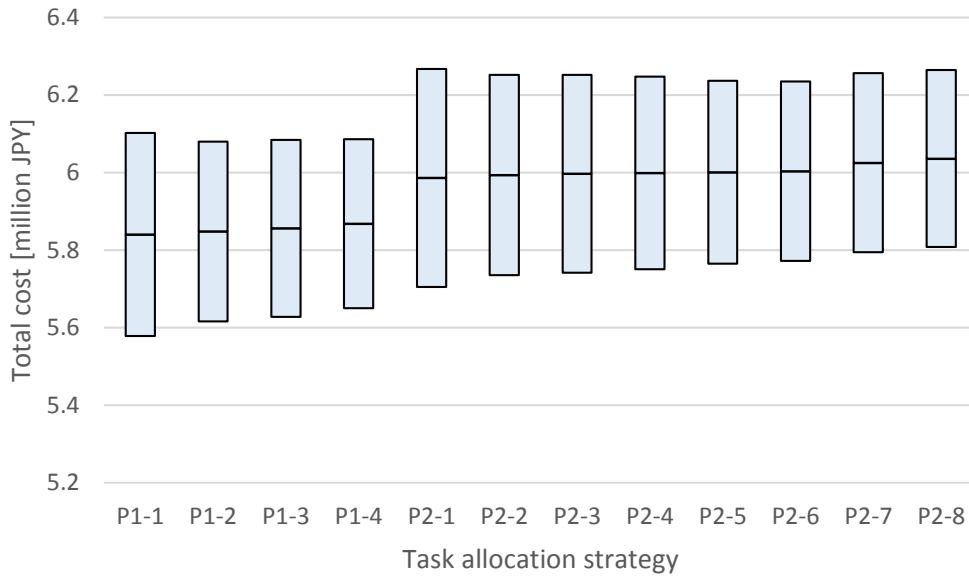


図 8-12 得られたタスク割り振りルールによるシミュレーション結果の範囲 ($\pm 2\sigma$)

8.4 まとめ

本章では, 3.5 で提案した遺伝的アルゴリズムによる作業チーム設計支援手法を船殻ブロック組み立てのケーススタディに適用することで, その評価を行った. 本章のケーススタディを通して得られた知見を以下にまとめる.

- 提案した遺伝的アルゴリズムによる作業チーム設計支援手法は, 電力ピークカットや作業面積制限などの制約に応じて適切なタスク割り振りルールを設計可能である.
- 提案したタスク割り振りルールの分析手法は, シナリオの変更によって生じたタスク割り振りルールの変化を定量的に評価可能であり, またその評価結果を定性的に言い換えることも可能である.
- すでに作業実施における制約の厳しい工場では, トラブルに対応する余裕が存在しないため, 新たに制約を導入してもタスク割り振りルールに大きな変化は見られないという結果を今回のケーススタディの中で定量的に示すことができた.

第9章 考察

9.1 はじめに.....	178
9.2 提案する作業チーム設計手法の比較.....	178
9.3 提案する作業チーム設計手法の有効性.....	180
9.4 提案する作業チーム設計手法の現場への導入.....	181
9.4.1 提案手法の適用範囲について.....	181
9.4.2 現場で求められる困難な作業チーム設計問題への対応.....	182

9.1 はじめに

本研究では、組織構造と作業戦略を考慮した作業チーム設計手法を提案した。また、開発した作業チーム設計支援システムを用いて、システム開発業務(第5章)、造船所の設計業務(第6章、第7章)と生産業務(第8章)に対して提案手法を適用した。

本章では、本研究を以下の視点でまとめ、考察を行う。

- 提案する作業チーム設計手法の比較
- 提案する作業チーム設計手法の有効性
- 提案する作業チーム設計手法の現場への導入

9.2 提案する作業チーム設計手法の比較

本研究で提案した作業チーム設計手法は、手動探索を行うことによる作業チーム設計手法(3.4)と、遺伝的アルゴリズムを用いた作業チーム設計支援手法(3.5)の2つに大別可能である。本論文で行った各ケーススタディを通して得られた知見により、これら2つの手法の特徴を表9-1のように整理した。

表 9-1 提案手法内におけるそれぞれの作業チーム設計手法の比較

	手動探索	遺伝的アルゴリズムによる支援
得られる作業チームの最適性	△ 可能性はあるが、非常に困難	× 遺伝的アルゴリズムの性質上、最適性は失われる
適切な作業チーム設計の容易さ	× 解空間が膨大であるため非常に負荷がかかる	○ 候補となる作業チーム案から選択する
作業チームの傾向分析	ミクロな視点での分析 作業チームの設計変数の変更による影響評価が可能	マクロな視点での分析 各作業チームの特性や傾向を把握することが可能

手動探索を行うことによる作業チーム設計手法に関しては、作業チームの解空間は膨大であるが、勘と経験による探索や全探索を行うことにより、与えられた制約条件下での最適な作業チームを行うことができる可能性は存在する。一方で、最適とは言えないまでも適切な作業チームを設計することを考えた場合、作業チームの解空間が膨大であるため非常に負担がかかる作業となる。開発した作業チーム設計支援システム上における手動探索による作業チーム設計支援機能(4.3)により、作業チーム設計案の評価と比較に関しては設計者の負担を軽減することが可能であるが、本質的には作業チームの解空間を狭めることはできない。

遺伝的アルゴリズムを用いた作業チーム設計支援手法では、遺伝的アルゴリズムの性質上、得られる解の最適性は保証されない。最適とは言えないまでも適切な作業チームを設計することを考えた場合、遺伝的アルゴリズムの探索によって抽出されたパレートフロントを構成する作業チームは、その他の作業チームと比較して与えられた制約下では良い作業チームであると言える。本論文で行ったケーススタディ(第6章, 第7章, 第8章)でも、遺伝的アルゴリズムを用いた作業チーム設計支援手法によって、人手で決定することの難しい作業チームの組織構造と作業戦略を適切に設計できることを示した。

また、本論文で行ったすべてのケーススタディから、2つの作業チーム設計手法のそれぞれを利用して行うことが可能な作業チームの分析にも違いがあることが分かる。手動探索による作業チーム設計手法では、ある作業チームを対象として人員追加などの変更を行った際の影響を定量的に評価するという、ミクロな視点での分析が可能である。これは、2.5や2.6で紹介したすべての研究事例でも同様のことが言える。一方で、遺伝的アルゴリズムによる作業チーム設計支援手法では、遺伝的アルゴリズムによる探索の結果を分析開始時に可視化して表示することが可能であるため、グラフ内の各点を表す作業チームの情報を比べることで、所要時間を短縮することが可能な作業チーム設計変更案の傾向把握などの、マクロな視点での分析が可能である。2.5や2.6で紹介した既存研究ではマクロな視点での分析を行うことが難しいことから、遺伝的アルゴリズムによる作業チーム設計支援手法を利用して作業チームの傾向を把握するためのマクロな分析を行うことが可能な点は、本研究の強みであると言える。

9.3 提案する作業チーム設計手法の有効性

本研究では、組織構造と作業戦略を考慮した作業チーム設計手法を提案した。一方で、チーム作業シミュレーションを用いた作業チーム設計の既存研究[36][37][38][39][40][41]と本研究で提案する作業チーム設計手法を比較すると、2.7 で述べたように本研究の新規性として以下の2点が挙げられる。

- 現実的な作業チーム設計を行うための、作業戦略(タスク割り振りルールや各プロジェクトへのエフォート率)を含めた作業チーム設計手法の提案
- 設計者の負担を軽減するための、遺伝的アルゴリズムを利用した適切な作業チーム設計支援手法の提案

作業戦略として、本研究では各プロジェクトへのエフォート率とタスク割り振りルールという2つの項目を導入し、それぞれの項目に対する設計手法提案を行った。また、第7章と第8章で状況に応じて取るべき作業戦略が異なることをケーススタディを通して定量的に示すことができた。一方で、既存研究では、単一のプロジェクトのみを対象としているため、複数のプロジェクトに対するエフォート率を設計することができない、シミュレーションを行う際にタスク割り振りルールを設計の対象とすることができないという問題があった。従来行うことが難しかった作業戦略の設計を提案手法と開発した作業チーム設計支援システムによって行えるようになったことは、一つの重要な進歩と言える。

また、作業チーム設計のプロセスについて、既存研究[39][41]でも3.4の手動探索と同様のプロセスを踏むことによって作業チームや組織のパフォーマンスを向上させた事例を紹介している。一方で、本研究で提案した作業チーム設計手法の特徴として、3.5の遺伝的アルゴリズムを利用して選択した目的関数に対するパレートフロントを構成する作業チーム群を複数抽出し、設計者の判断で抽出した作業チーム群から最終的な作業チームを選択することが挙げられる。3.2.1で述べた通り、現実のプロジェクトでは時間が経過すると、プロジェクトの取り巻く環境が変化することが想定されるが、複数の適切な組織構造案や作業戦略案を保持しておくことで、状況の変化に応じて柔軟に組織構造と作業戦略の適切な組み合わせを選択可能になり、結果として状況の変化に柔軟に対応することが容易な手法となっている。すなわち、遺伝的アルゴリズムを利用した自動探索を行うことで、設計者の負荷を軽減するだけでなく、柔軟な作業チーム設計が可能になったと言える。

9.4 提案する作業チーム設計手法の現場への導入

本論文では4つのケーススタディによって提案する作業チーム設計手法の有効性を評価した。本節では、提案手法によって実際に様々な現場で作業チーム設計を行うことに関する考察を行う。

9.4.1 提案手法の適用範囲について

提案手法によって実際の現場で適切な作業チーム設計を行うためには、入力データであるワークフローと組織情報に関して適切なデータを用意することが前提となる。現場の実情に合わせた適切な入力データを用いることで、提案手法によるチーム作業シミュレーション結果に妥当性が生まれ、結果として設計した作業チームに信頼性が生まれるためである。

提案手法を適用可能な分野や範囲については、現場の実情に適切な入力データ用意できるか否かで判断することが可能である。具体的には、提案手法を適用可能な分野や業務に関しては、以下の2点を満たす必要がある。

- プロジェクトの作業内容をワークフローによって明確に定義可能であること
- 人員のスキルなどに関する組織情報を明確に定義可能であること

製造業やプラント業務、システム開発業務などの工学分野に属する業務に関しては、プロジェクトの作業内容をワークフローなどの形式で標準化する取り組みが多く行われている。また、上記のような分野では組織の人員のスキルの標準化に関する取り組みに関しても多く行われているため、提案手法における入力データ(ワークフロー、組織情報)を現場の情報やデータに基づいて合理的に定義することが可能である。一方で、デザインや芸術に代表される、ある種のひらめきや偶然性を必要な分野に関してはワークフローやスキルを現場の情報やデータに基づいて合理的に定義することができないため、提案手法によって適切な作業チームに設計することが難しい。

9.4.2 現場で求められる困難な作業チーム設計問題への対応

実際の現場で作業チームの設計を行う場合には、本研究で行ったモデル化以外にも様々な事象を考慮しなければならない。例えば、作業チームに関しては、現場での固定化された組織構造の部分が存在し、組織構造の内面に存在する人間関係などの作業チーム固有の風土や文化などを考慮する必要がある[10]。さらに、吉村ら[46]が言及しているように、実際に組織の中で各プロジェクトに対して人員を配置する際には、人員のモチベーションやキャリアパスなどを考慮して人員配置を決定することが望ましい。また、人員ごとに雇用形態やプロジェクトへのかかわり方も異なってくる。提案手法では、上記の項目を完全に考慮することは難しいため、開発した作業チーム設計支援システムを用いて複数の作業チーム候補案を作成し、最終的に設計者が上記の項目を考慮して作業チームを設計する必要がある。

また、人員配置の困難さだけでなく、対象とするプロジェクトの状況による困難さについても考慮しなければならない。作業チーム設計問題に関して、現実のプロジェクトで求められている状況は様々なものが考えられるが、その中でも代表的な状況として以下の4点について、提案手法による作業チーム設計を行う際の困難な点と対応方法について述べる。

- 今まで経験のない、新規のプロジェクトへ対応する必要がある場合

企業や組織が、今までに経験のないプロジェクトに対して作業チームの設計を行う場合に、提案手法を適用するのが困難な点として、人員の各タスクへのスキル値とトラブル率を適切に決定することが難しい点が挙げられる。このような状況に対応するためには、企業や組織に蓄積されている組織情報を利用して、類似するタスクに関するスキル値に基づいて各タスクのスキル値を決定する、人員ごとの過去の実績データを参照してトラブル率を決定するなどの対応が必要になる。

- 工場の海外移転など、プロジェクトの作業内容に関してはそれほど変化がないが、組織の人員や環境が大きく変化する場合

最近では製造業に代表されるように、企業や工場の海外移転の事例が増加している。海外移転を行う際には、日本で行われていた作業内容を現地の人員で行う必要が出てくるため、トラブルが増加することが予想される。本研究で提案した作業チームのモデル化とチーム作業シミュレーションでは、ワークフロー内の各タスクのトラブル率と人員のトラブル率を増加させると、所要時間と合計コストが増加するとともに、チーム作業シミュレーションの結果のブレも大きくなる。そのため、作業チームのパフォーマンスを正確に見積もることが難しくなってしまう。このような状況に対応するためには、Moser[41]が指摘しているように、常にプロジェクトの進捗状況をワークフローに入力した状態でチーム作業シミュレーションを行うことにより、徐々にチーム作業シミュレーションの精度を上げながら作業チームの設計を検討する必要がある。

また、海外移転において考慮しなければならない事象として、その国独自の制度や規制が挙げられる。本研究では、3.3.2.2 で作業面積と電力使用量に関する制約を導入しているが、必要に応じて新たな制約の導入とそれに応じた作業チームのモデル化の変更を行う必要がある。

- プロジェクト内で大きなトラブルが発生した際に、可能な限り早くプロジェクト全体の再設計や作業チームの再設計が必要な場合

本研究では、組織構造内でのトラブルは遅延が発生する程度の小さなトラブルのみを取り扱っており、プロジェクト全体が停止してしまったり、プロジェクトの方針を変更しなければならない程度の大きなトラブルは想定していない。このような状況に対応するためには、大きなトラブルが発生した段階で、プロジェクトの方針や作業内容を再度検討し、開発した作業チーム設計支援システム上で検討した作業内容を表すワークフローを定義した上で作業チームの設計を行う必要がある。

また、予めこのような状況に対応した作業チームを設計する必要がある場合には、プロジェクトに存在するリスクの洗い出しと、それぞれのリスクが起きた場合の影響と対策をワークフローで定義した上でチーム作業シミュレーションを行う必要がある。すなわち、リスク発

生に対応したシナリオを複数用意し、評価をすることを繰り返し行うことで、大きなトラブルが発生した際に、設計した作業チームのパフォーマンスがどの程度低下するかをあらかじめ把握した状態で、作業チームの設計を行わなければならない。

- 作業途中のプロジェクトが複数存在する中で、どのように新規プロジェクトの作業を割り込ませる必要のある場合

本研究では、複数のプロジェクトを同時に考慮するケーススタディに関しても検討を行った。ケーススタディでは複数のプロジェクトはすべて新規のプロジェクトである場合を想定していたが、実際には作業途中のプロジェクトが複数存在する中で、どのように新規プロジェクトの作業を割り込ませるかを検討しなければならない状況が常に存在する。このような状況に対応するためには、現在進行中のプロジェクトの状況をワークフローに反映させた状態でチーム作業シミュレーションを行うことで対応することが可能である。

第10章 結論

10.1 結論.....	186
10.2 今後の展望.....	187

10.1 結論

本研究では、組織構造と作業戦略を考慮した作業チームの設計手法を提案し、提案手法をソフトウェア上で実現可能な作業チーム設計支援システムを実装した。また、遺伝的アルゴリズムを利用して作業チームの組織構造と作業戦略の設計を支援する手法を提案し、製造業やシステム開発業務の設計・生産プロセスを対象に提案手法の有効性の検証を行った。これらの結果として以下の結論を得た。

(1) 作業チームの詳細なモデル化に基づいたチーム作業シミュレーション機能と作業チーム設計支援システムの実装

対象のプロジェクトの内容を表すワークフローと作業実施における制約、人員と工作機械のスキルセットとその依存関係によって表される組織構造、各人員のエフォート率とタスク割り振りルール(ディスパッチングルールの重みベクトル)によって表される作業戦略などの詳細なモデルを入力として、タスク割り振りルールに基づいたタスクの割り振りと組織構造内でのトラブル対応を計算機上で表現することで、作業チームのパフォーマンスを評価可能なチーム作業シミュレーション機能を実装した。また、チーム作業シミュレーションを利用して、作業チームの設計と評価を繰り返すことが可能な作業チーム設計支援システムを構築した。

本論文でのケーススタディを通して、作業戦略の違いや作業チームの組織構造の違いが与える作業チームのパフォーマンスの変化を定量的に評価しながら設計案を変更することを繰り返すことで、より良い作業チーム設計が可能になることを確認できた。

(2) 遺伝的アルゴリズムを用いた作業チーム設計支援手法の提案

作業チームの組織構造作業戦略などの人手で適切に設計することが困難な設計項目に対して、遺伝的アルゴリズムを用いて設計を支援する手法を提案した。遺伝的アルゴリズムによってランダムに生成される設計案を入力としたチーム作業シミュレーションの結果

をコストと所要時間の観点から評価することを繰り返し、探索結果の中で採用した目的関数に対するパレートフロントを構成する作業チームの設計案を抽出することで、設計者はあらかじめ妥当であると定量的に評価された設計案の中から取捨選択し、必要に応じて変更を加えることで適切な作業チーム設計を支援可能なことが確認できた。

(3) 設計過程における作業チームの特性の理解

設計者の手動探索や遺伝的アルゴリズムを利用した適切な作業チームの探索過程を分析することで、対象とするプロジェクトに応じて作業チームの特性を理解するための分析が可能であることがケーススタディから確認できた。

具体的には、第6章では、コストを重視するか時間を重視するかによって作業チームの取るべき組織構造がどのように変化することが分析することができた。また、第7章では、人手で考える単純なエフォート率の設定に比べて各人員のエフォート率を適切に設定することで作業チームのパフォーマンス向上の可能性のあることを分析により明らかにすることができた。さらに、第8章では、作業実施制約の変化に応じたタスク割り振りルールの変化を定量的に評価することで、対象とする作業チームがどの程度余裕があるかまで分析することができた。

10.2 今後の展望

本研究では、作業チームの組織構造と作業戦略を考慮したチーム作業シミュレーションを利用した作業チームの設計支援について述べた。定義した作業チームの組織構造モデルの中で、提案手法と開発したシステムにより作業チーム設計が支援できることを確認できた。一方で、トラブル発生時の組織構造内での調整に関しては、本研究では先行研究[37][41]のモデル化を援用した単純なモデルを採用したが、現場での運用を考慮するといくつか課題が存在すると考える。例えば、トラブル対応における残業や休日出勤、先行タスクへの手

戻りが発生するレベルのトラブルなどは、本研究で採用した組織構造モデルでは考慮できていない。また、この人員は他のプロジェクトとの兼ね合いで対象とするプロジェクトでは限定的にこの期間のみ参加することができるなどの、プロジェクト参加に関する制約も現在の組織構造モデルでは考慮できていないが、現場ではこのような状況が頻繁に発生するものと考えられる。また、提案手法では固定給と変動給、エフォート率によって人員の雇用形態を定義したが、プロジェクトの一時期のみ参画する人員や、直接の作業には参加せず調整作業のみを行う人員の考慮など、これらの項目だけでは表せない雇用形態も考慮する必要がある。今後、様々な現場を対象に提案手法を適用し、チーム作業シミュレーション結果と現実との乖離を分析することで、作業チームの精緻なモデル化が可能になると考えられる。

また、Moser が言及しているように、実際の現場では設計者は1人ではなく、それぞれの部署ごとの代表者間での話し合いによって作業チームの設計が行われていることが多い

[41]. 開発した作業チーム設計支援システムをこのような現場で活用するためには、設計した作業チームのパフォーマンスを示すチーム作業シミュレーションの出力について、より分かりやすく可視化する必要がある。特に、作業チームのどの部分にリスクがあり、どの部分に関して再検討すべきかなどの決断の材料となる情報をチーム作業シミュレーション結果で示唆できるような方法の開発が考えられる。

謝辞

本研究を進めるにあたって、多くの方々にご助力を頂きました。ここに感謝の意を述べさせていただきます。

東京大学大学院新領域創成科学研究科人間環境学専攻 教授 大和裕幸先生には大変ご多忙の中貴重な時間を割いていただき、研究の方針はもとより、一般教養、学生としての心構え等、様々なご指導をいただきました。厳しくも真に学生のことを考えてのアドバイスは、どれも深く心に残る言葉ばかりでありました。心から深く感謝いたします。

東京大学大学院新領域創成科学研究科人間環境学専攻 教授 奥田洋司先生、東京大学大学院新領域創成科学研究科海洋技術環境学専攻 教授 尾崎雅彦先生、東京大学大学院工学系研究科システム創成学専攻 教授 青山和浩先生、東京大学大学院工学系研究科システム創成学専攻 特任准教授 田中謙司先生には、副査の申し出を引き受けていただきありがとうございます。ここに厚く御礼を申し上げます。

東京大学大学院新領域創成科学研究科人間環境学専攻 准教授 稗方和夫先生には、特に技術面においてもっとも多くの具体的で有意義なアドバイスをいただきました。加えて、研究に対する姿勢、プログラミング作法、論文の書き方、仕事に対する心構えなど様々な面でご指導、ご教示いただきました。心より深く感謝いたします。

電気通信大学大学院情報理工学研究科 教授 増田宏先生、東京大学大学院工学系研究科システム創成学専攻 准教授 白山晋先生には設計工学研究室大輪講などにおいて、的確で具体的なアドバイスをいただきました。大変感謝いたします。

Global Project Design.inc の Bryan R Moser 様には、同じ研究分野ということもあり様々なトピックで有意義な議論させていただきました。本論文は、Global Project Design.inc でのインターンシップ経験によって得られた知見を土台として執筆させていただきました。心より深く感謝いたします。

海洋技術安全研究所 松倉洋史様, 横浜国立大学工学研究院 准教授 杉本千佳先生, 東京大学大学院新領域創成科学研究科人間環境学専攻 特任研究員 久保登様には, 学部時代の頃に, 研究に関する様々な面でサポートをいただきました. ありがとうございます.

Yahoo! Japan 研究所の坪内孝太様には, 研究室の教員として, また頼れる先輩として, 私ともに大変お世話になりました. 時には厳しく時には優しく私に指導していただき, 多くのことを学ばせて頂きました. 感謝してもきれません.

東京大学大学院工学系研究科システム創成学専攻 技術専門員 榎本昌一様には, ネットワーク, サーバをはじめとした様々な情報技術に関するサポートをいただきました. 深く御礼申し上げます.

水野淳子様, 山本和子様, 谷村育子様, 上村美江子様, 山田祐子様には, 研究に専念できるように膨大な数の煩雑な事務手続きを行っていただきました. また, 生活面においても, 常に優しく接していただきました. 本当にありがとうございました.

共同研究員である, 富士通株式会社の岡田伊策様, 齋藤稔様とは, 研究室の勉強会や研究のディスカッションなどで大変お世話になりました. ビジネス視点でのお話や情報技術に関する最先端のお話などは大変に興味深く, 積極的にディスカッションさせていただきました. 本当にありがとうございました.

研究室の先輩である, A.A.Dinariyana 様, 辻本翔様, Michelle Buen Tumilba 様, 角田浩平様, 研究生の Francois Poizat 様, Pierre Belin 様には研究とプライベートの両面で大変お世話になりました. 様々な思い出を共有できたことは忘れることのできない思い出です. 本当にありがとうございました.

研究室の同期である, Kim Jae Youl 様, 新木仁士様, 飯坂祐司様, 寺西慶祐様とは, 日

頃からお互いの研究に関してディスカッションを行い、切磋琢磨することができました。また、卒業後も何でも話せる友人を得られたことは私にとってかけがえのない財産になっています。心より感謝します。

研究室の後輩である、于佰鑫様、笈田佳彰様、高橋孝輔様、柳澤龍様、鹿渡俊介様、木村彰吾様、長谷川嵩様、曹樺楠様、孫晶鈺様、中村覚様、三浦慎也様、砂川辰徳様、舩島一樹様、石黒慧様、河野裕様、廣井貴彬様、深田直人様、黒川達也様、角野為耶様、津村和輝様、宮崎邦洋様、安藤早紀様、齋藤智輝様、松原洸也様、和中真之介様、Angel Yang様には、日頃の研究室生活において、多大なるサポートを受けました。ありがとうございます。皆様のような優秀な方が研究室の後輩として入ってくるたびに、私の研究室生活に緊張と楽しさが生まれ、充実した研究室生活を送ることができました。本当にありがとうございました。

東京大学工学院工学系研究科システム創成学専攻に所属していた設計工学研究室の皆様とは、設計研共通のイベントを通して研究に関する活発な議論をすることができました。

最後になりましたが、研究生生活を支えてくれた家族に感謝いたします。本当にありがとうございました。

参考文献

- [1]. Katzenbach, J.R. and Smith, D.K., The discipline of teams, Harvard Business Review, Vol. 71 (1993), pp.111-120.
- [2]. Katzenbach, J.R. and Smith, D.K., 「高業績チーム」の知恵—企業を革新する自己実現型組織, (1994), ダイヤモンド社.
- [3]. 大月博司, 藤田誠, 奥村哲史, 組織のイメージと理論, (2001), 創成社.
- [4]. チャールズ・J・ペレリン, NASA のチームビルディング, (2010), アチーブメント出版.
- [5]. PMI Standards Committee, A guide to the project management body of knowledge (PMBOK guide), (2000), Project Management Institute.
- [6]. 桑田耕太郎, 田尾雅夫, 組織論, (1998), 有斐閣.
- [7]. Weber, M., The theory of social and economic organization, (1947), New York: Oxford University Press, pp.341-342.
- [8]. Bernard, C.I, 山本安次郎, 田杉競, 飯野春樹, 経営者の役割:その役割と機能, (1956), ダイヤモンド社.
- [9]. Simon, H.A., The architecture of complexity, Proceedings of the American Philosophical Society, Vol.106, No.6 (1962) pp.467-482.
- [10]. 野中郁次郎, 加護野忠雄, 小松陽一, 奥村昭博, 坂下昭宣. 新装版 組織現象の理論と測定, (2013), 千倉書房.
- [11]. Burton, R.M., Obel, B.R., Strategic organizational diagnosis and design: developing theory for application, (1995), Kluwer Academic Publishers, Boston.
- [12]. 藤田彰久, IE の基礎, (1997), 建帛社.
- [13]. 千住鎮雄, 川瀬武志, 佐久間章行, 矢田博, 中村善太郎, 作業研究 (経営工学シリーズ), (1987), 日本規格協会.
- [14]. 泉英明, 生産工学 (生産管理 理論と実践), (1994), 日刊工業新聞社.
- [15]. テーラーF.W., 上野陽一. 科学的管理法, (1969), 産能大学出版部.
- [16]. Fernandez-Sanz, L., Escribano, J.J., Hilera, J.R., "Software operation time evaluation based on MTM, Advances in Engineering Software, Vol. 40, No.8 (2009) pp. 583-592.

-
- [17]. 佐相邦英, 長坂彰彦, 行待武生, チーム行動の分析・評価方法の開発--チーム行動特性評価手法の開発, 人間工学, Vol.29, No.2 (1993) pp.111-120.
- [18]. 佐相邦英, 長坂彰彦, 行待武生, チーム行動の分析・評価方法の開発--チーム行動特性に関する研究, 人間工学, Vol.27, No.3 (1991) pp.159-168.
- [19]. 安藤英幸, 大和裕幸, 角田領, ブリッジにおける協調作業の対話分析, 日本造船学会論文集, No.195 (2004) pp. 63-70.
- [20]. Sanderson, P.M., Fisher, C., Exploratory sequential data analysis: foundations, Human-Computer Interaction, Vol. 9, No.3-4 (1994) pp. 251-317.
- [21]. Sanderson, P., Scott, J., Johnston, T., Watanabe, L., James, J., MacSHAPA and the enterprise of exploratory sequential data analysis (ESDA), International Journal of Human-Computer Studies, Vol. 41, No.5 (1994) pp. 633-681.
- [22]. 満行泰河, 大和裕幸, 稗方和夫, 小林敬和, 黒川哲明, 作業者の位置情報を用いたチーム作業工程分析手法に関する研究, 日本機械学会論文集 C 編, Vol.79, No.799 (2013) pp.837-844.
- [23]. 三沢良, 佐相邦英, 山口裕幸, 看護師チームのチームワーク測定尺度の作成, 社会心理学研究, Vol.24, No.3 (2009) pp. 219-232.
- [24]. Garrett, C.J., Guarro, S.B., Apostolakis, G.E., The dynamic flowgraph methodology for assessing the dependability of embedded software systems, IEEE Transactions on Systems Man and Cybernetics, Vol.25, No.5 (1995) pp. 824-840.
- [25]. Laughery, R., Using discrete-event simulation to model human performance in complex systems, Proceedings of the 31st conference on Winter simulation, Vol.1 (1999) pp.815-820.
- [26]. Yow, A., Walters, B., Plott, B., Laughery, R., Persensky J., Predicting nuclear power-plant operator performance using discrete event simulation, Cognition, Technology & Work, Vol.7, No.1 (2005) pp.29-35.
- [27]. Kontogiannis, T., Integration of task networks and cognitive user models using coloured Petri nets and its application to job design for safety and productivity, Cognition, Technology & Work, Vol.7, No.4 (2005) pp. 241-
-

261.

- [28]. 角田領, 大和裕幸, 安藤英幸, 小山健夫, 中村紳也, シミュレーションによるBRMシミュレータトレーニングシナリオの分析, 日本船舶海洋工学会論文集, Vol.6 (2007) pp. 99-107.
- [29]. 角田領, 大和裕幸, 安藤英幸. シミュレーションによるブリッジチームのパフォーマンス評価手法, 日本船舶海洋工学会論文集, Vol.5 (2007) pp. 47-55.
- [30]. 関根智明, PERT・CPM, (1973), 日科技連.
- [31]. 黒田充, 村松健児, 生産スケジューリング, (2004), 朝倉書店.
- [32]. Hartmann, S., Project scheduling under limited Resources, Lecture Notes in Economics and Mathematical Systems, Vol.478 (1999) Springer.
- [33]. Hartmann, S., A competitive genetic algorithm for resource constrained project scheduling, Naval Research Logistics, Vol.45, No.7 (1998) pp.733-750.
- [34]. Sprecher, M.G., Vercellis, C., Hierarchical models for multi project planning and scheduling, European Journal of Operational Research, Vol.64, No.2 (1993) pp.780-790.
- [35]. Yamato, H., Hiekata, K., Mitsuyuki, T., A study on evaluation and improvement of organizational performance in fabrication shops,” International Conference on Computer Applications in Shipbuilding, Vol.1 (2009) pp.157-163.
- [36]. Levitt, R.E., Thomsen, J., Christiansen, T.R., Kunz, J.C., Jin, Y., Nass, C., Simulating project work processes and organizations: toward a micro-contingency theory of organizational design, management science, Vol.45, No.11 (1999) pp.1479-1495.
- [37]. Jin, Y., Levitt, R.E., The virtual design team: a computational model of project organizations, Computational and Mathematical Organizational Theory, Vol.2, No.3 (1996) pp.171-196.
- [38]. Kunz, J.C., Christiansen, T.R., Cohen, G.P., Jin, Y., Levitt, R.E., The virtual design team: a computational simulation model of project organizations, Communications of the Association for Computing
-

-
- Machinery(CACM), Vol.41, No.11 (1998) pp.84-91.
- [39]. Horii, T., Jin, Y., Levitt, R.E., Modeling and analyzing cultural influences on project team performance, Computational & Mathematical Organization Theory, Vol.10, No.4 (2005) pp.305-321.
- [40]. Suzuki, Y., Yahyaei, M., Jin, Y., Koyama, H., Kanga, G., Simulation based process design: modeling and applications, Advanced Engineering Informatics, Vol.26, No.4, (2012) pp.763-781.
- [41]. Moser, B.R., The design of global work: simulation of performance including unexpected impacts of coordination across project architecture, 東京大学新領域創成科学研究科学学位論文, 2012.
- [42]. Global Project Design, Inc., “TeamPort Software”. Global Project Design. (online), available from <<http://gpdesign.com/teamportsoftware.html>>, (accessed 2013-12-10).
- [43]. 黄東明, 圓川隆夫, 秋庭雅夫, 成功例に基づく平均納期遅れ基準のスケジューリング問題におけるディスパッチングルール生成に関する研究, 日本経営工学会誌, Vol.41, No.6 (1991) pp.383-389.
- [44]. Zitzler, E., Thiele, L., Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach, IEEE Transaction on Evolutionary Computation, Vol.3, No.4 (1999) pp.257-271,
- [45]. Murata, T., Ishibuchi, H., Tanaka, H.. Multi-objective genetic algorithm and its applications to flowshop scheduling, Computers & Industrial Engineering, Vol.30, No.4 (1996) pp.957-968
- [46]. 吉村允孝, 藤見善久, 泉井一浩, 西脇眞二, 複数開発プロジェクトへのヒューマンリソース最適配置, 日本機械学会論文集 C 編, Vol.71, No.702 (2005) pp.677-684.
- [47]. Bean, J.C., Genetics and random keys for sequencing and optimization, ORSA Journal on Computing, Vol.6, No.2 (1994) pp.154-160.
- [48]. The Eclipse Foundation, “GEF”. The Eclipse Foundation open source community website. (online) available from <<http://www.eclipse.org/gef/>>, (accessed 2013-12-10).
- [49]. The Eclipse Foundation, “Rich Client Platform”. The Eclipse Foundation
-

-
- open source community website. (online) available from <<http://www.eclipse.org/home/categories/rcp.php>>, (accessed 2013-12-10).
- [50]. Object Management Group, Inc., “BPMN ver2.0”. Business Process Model And Notation (BPMN). (online) available from <<http://www.omg.org/spec/BPMN/2.0/>>, (accessed 2013-12-10).
- [51]. Hiekata, K., Yamato, H., Oishi, W., Nakazawa, T., Ship design workflow management by ShareFast, Journal of Ship Production, Vol.23, No.1 (2007) pp. 23-29.
- [52]. Object Refinery Limited, “JFreeChart”. JFree. (online) available from <<http://www.jfree.org/jfreechart/>>, (accessed 2013-12-10).
- [53]. 室中健二, 原直朗, システム構築の標準プロセス体系:SDEM, Fujitsu, Vol.63, No.2 (2012) pp.193-199.
- [54]. Lamb, T., Ship Design and Construction, (2003), The Society of Naval Architects and Marine Engineers.
- [55]. 奥本泰久, 造船技術と生産システム, (2009), 成山堂書店.

研究業績

(1) 業績の概要

受賞	1 件
論文誌	3 件
本文査読付き国際会議	3 件
本文査読無し国際会議	2 件
国内会議	15 件

(2) 受賞

- [1]. 満行泰河, 日本船舶海洋工学会奨学褒賞, 社団法人日本船舶海洋工学会, 2011.

(3) 論文誌

- [1]. Taiga Mitsuyuki, Kazuo Hiekata, Hiroyuki Yamato, Design of Production Strategy considering the Cutting Peak Demand of Electricity in the Ship Building Industry, *Journal of Marine Science and Technology*, Vol.xx, No.xx, (2014). (投稿中)
- [2]. 満行泰河, 大和裕幸, 稗方和夫, 複数設計業務のシミュレーションと組織内人員の最適配置, *日本機械学会論文集 C 編*, Vol.79, No.806 (2013) pp.3930-3938.
- [3]. 満行泰河, 大和裕幸, 稗方和夫, 小林敬和, 黒川哲明, 作業者の位置情報を用いたチーム作業工程分析手法に関する研究, *日本機械学会論文集 C 編*, Vol.79, No.799 (2013) pp.837-844.

(4) 国際会議(本文査読付き)

- [1]. Taiga Mitsuyuki, Hiroyuki Yamato, Kazuo Hiekata, Bryan Moser, The Design of Production Strategy based on Risk Analysis using Process Simulation, *Proc. of the 20th ISPE International Conference on Concurrent Engineering*, (2013) pp.244-253.
- [2]. Taiga Mitsuyuki, Kazuo Hiekata, Hiroyuki Yamato, Kazuki Haijima, A Study on Evaluation of Organizational Performance considering the Workers and Facilities, *Concurrent Engineering Approaches for Sustainable Product Development in a Multi-*
-

-
- Disciplinary Environment, Proc. of the 19th ISPE International Conference on Concurrent Engineering, Vol.1 (2012) pp.533-544, Springer.
- [3]. Hiroyuki Yamato, Kazuo Hiekata, Taiga Mitsuyuki, Satoru Nakamura, Historical Design Review based on the Digital Archive with the Semantic Web Approach, Proc. of the 11th International Marine Design Conference, Vol.1, (2012) pp.457-464.
- (5) 国際会議(本文査読無し)
- [1]. Hiroyuki Yamato, Kazuo Hiekata, Taiga Mitsuyuki, Yu Kawano, Masakazu Enomoto, Yoshihiro Tsuchiya, 'Robot tank system' and development of integrated management system for various measured data in tank tests, Proc. of 16th International Conference on Computer Applications in Shipbuilding, Vol.3 (2013) pp.21-26.
- [2]. Hiroyuki Yamato, Kazuo Hiekata, Taiga Mitsuyuki, A Study on Evaluation and Improvement of Organizational Performance in Fabrication Shops, Proc. of 14th International Conference on Computer Applications in Shipbuilding, Vol.1 (2009) pp.157-163.
- (6) 国内会議
- [1]. 満行泰河, 大和裕幸, 稗方和夫, 組織構造を考慮した作業チーム設計支援システムの開発, 日本船舶海洋工学会講演会論文集, 第 17 号 (2013) pp.473-476.
- [2]. 満行泰河, 大和裕幸, 稗方和夫, Bryan Moser, 遺伝的アルゴリズムを用いた組織構造の最適化に関する研究, 日本機械学会第 23 回設計工学・システム部門講演会論文集 (2013) OS1-1-1404
- [3]. 大和裕幸, 稗方和夫, 榎本昌一, 土屋好寛, 満行泰河, 長谷川嵩, 木村彰吾, 河野裕, ロボット水槽構想と異種計測データの統合管理, 日本船舶海洋工学会講演会論文集, 第 16 号 (2012) pp.95-98.
- [4]. 満行泰河, 稗方和夫, 大和裕幸, 電力ピークカットを考慮した作業計画立案手法の研究, 日本船舶海洋工学会講演会論文集, 第 15 号 (2012) pp.123-126.
- [5]. 大和裕幸, 稗方和夫, 榎本昌一, 土屋好寛, 満行泰河, 長谷川嵩, 木村彰吾, 河
-

-
- 野裕, 水槽試験に関する異種計測データ統合管理システムの開発, 日本船舶海洋工学会講演会論文集, 第 15 号 (2012) pp.57-60.
- [6]. 稗方和夫, 大和裕幸, 満行泰河, 舩島一樹, 組織の人員と工場設備を考慮した作業計画立案手法に関する研究, 日本船舶海洋工学会講演会論文集, 第 14 号 (2012) pp.403-406.
- [7]. 佐藤知正, 進木裕大, 満行泰河, 下坂正倫, 大和裕幸, 高齢者向け社会参加サービスのデザイン～サービスと人のアロケーションツールの構築～, 第 30 回日本ロボット学会学術講演会予稿集 (2012) 3H1-6.
- [8]. 満行泰河, 坪内孝太, 大和裕幸, 高齢者と社会サービスのマッチングシステムに関する検討, 第 29 回日本ロボット学会学術講演会予稿集 (2012) 2N2-1.
- [9]. 大和裕幸, 稗方和夫, 満行泰河, 中村覚. 平賀譲デジタルアーカイブを用いた歴史研究支援システムの構築, 日本船舶海洋工学会秋季講演会ポスターセッション (2011).
- [10]. 井上美喜子, 北村光司, 西田佳史, 満行泰河, 大和裕幸, うれしさネットワーク分析に基づいた生活機能統合による社会参加支援, 第 12 回計測自動制御学会システムインテグレーション部門講演会予稿集 (2011) pp.2352-2355.
- [11]. 坂口憲一, 平松祥史, 稗方和夫, 大和裕幸, 満行泰河, 業務プロセスの改善による現場力の向上 -ケーススタディの紹介-, 人工知能学会第2種研究会(SIG-KST), (2010) SIG-KST-2010-02-03.
- [12]. 稗方和夫, 大和裕幸, 満行泰河, CAD 操作ナビゲーションシステムの開発と実証, 人工知能学会第2種研究会(SIG-KST) (2009) SIG-KST-2009-02-04.
- [13]. 大和裕幸, 稗方和夫, 満行泰河, 組織構造を考慮したコラボレーション支援システムの研究, 日本船舶海洋工学会講演会論文集, 第 8 号 (2009) pp. 543-546.
- [14]. 大和裕幸, 稗方和夫, 満行泰河, 生産プロセスを考慮した組織パフォーマンス評価に関する研究, 日本機械学会第 19 回設計工学・システム部門講演会論文集 (2009) pp. 600-603.
- [15]. 大和裕幸, 稗方和夫, 満行泰河, 生産プロセスを考慮した組織パフォーマンス評価手法に関する研究, 人工知能学会第2種研究会(SIG-KST) (2009) SIG-KST-
-

2009-02-04.