

制約に基づくソフトウェア開発計画自動立案システム -GA 利用のための要員割り当て方法の改良-

木下 大輔^{*1} 根本 恵美子^{*2} 古澤 三奈^{*2} 八重樫 理人^{*3} 古宮 誠一^{*1}

^{*1} 芝浦工業大学大学院 ^{*2} 芝浦工業大学情報工学科 ^{*3} 埼玉大学大学院

1. まえがき

ソフトウェアの開発はプロジェクトを組んで行われるのが一般的である。また、ソフトウェア開発プロジェクトには開発計画というものが必ず存在する。そして、プロジェクトを成功へ導くためには開発計画を基に管理目標を設定し、進捗をフォローする方法が効果的である。しかし、これらの作業はそれほど簡単なことではない。何故なら、作業量の見積りやリスクの予測が難しいため、開発計画の立案が困難だからである。また、ソフトウェアは目に見えないため、設計作業の検討の深さや作業の実質的な進捗把握が困難だからである。このため、管理作業の自動化によりプロジェクト管理者の負担を軽減し、担当者との対話時間を増やすことにより、実質的な進捗把握を可能にするとともに、高度なプロジェクト管理を可能にするようなシステムの開発が必要不可欠である。そこで、このような要件を満足するシステムとして、ソフトウェア開発計画の自動立案システムを研究開発する。

要員割り当てを含むソフトウェア開発計画案を遺伝的アルゴリズムを用いて自動生成する方法を既に文献[1]で示した。しかし、文献[1]のシステムでは各工程に対して1工程に1人しか割り当てられないという制限があった。本稿では、その問題を解決し、複数の要員割り当てを可能にする方法について述べる。

後述の説明のためにひとつのソフトウェア開発計画例題を以下に示す。

[例題プロジェクトの作業概要]

例題プロジェクトとして、8月1日から開催される展示会のデモシステムを開発するプロジェクトを考える。このプロジェクトは7月2日に組織され、7月31日までにシステムを完成させなければならない。システム開発のために行わなくてはならない作業は、基本設計、概要設計、概要設計レビュー、詳細設計、コーディング、単体テスト、統合テストである。

プロジェクト計画立案の際に考慮すべき制約は以下のとおりである。デモシステムは高性能を要求されたシステムであるため、詳細設計の一部に性能見積もりを行い、コーディングに反映させる。展示会で使用するハードウェアは出展のためのマシンであり、7月26日に使用可能となり、31日には展示会場に搬出する必要がある。マシンの使用可能期間が短いため、テスト効率化のための支援ツールを開発することとなった。

表 1: 要員の属性情報の例

Name	Skill	Available time	Cost/h
A	OOA, C	Any	2800
B	SA, C	Any	2300
C	JSD, Performance Analysis	7/15-7/18 8/1-8/5	3000
D	SA, C	7/2-7/19 7/28	3500

2. ソフトウェア開発計画立案問題が持つ制約とその内容

本研究では、ソフトウェア開発計画立案問題を、制約を満たすリソースの最適割り当て問題であると捉える。制約の内容については以下で説明する。

(1) 作業順序に関する制約

ソフトウェア開発の各工程には、中間成果物を介して、それらの実施順序が決定する。例題プロジェクトの性能見積もりの工程を例に挙げて説明する(図1参照)。PE(性能見積もり)を実施するにはDD1(詳細設計の一部)の工程が行われ、その成果物(中間成果物)である詳細設計書が作成されなければならない。故に、これを工程PEの**事前条件**と呼ぶ。工程PEが終了すると、その成果物として性能評価報告書が生成されなければならない。故に、これをPEの**事後条件**と呼ぶ。性能評価報告書は、これを基にCT1(コーディング&テストの一部)が行われるので、同時にCT1の事前条件でもある。このような制約を**作業順序に関する制約**と呼ぶ。

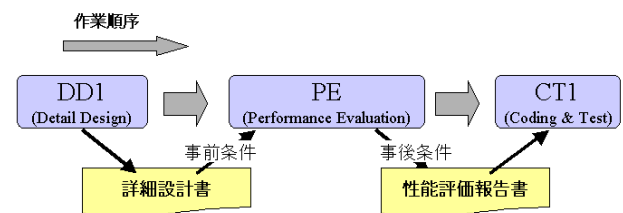


図 1: 中間成果物による作業順序制約

(2) リソースの割り当て条件に関する制約

ソフトウェア開発の各作業には、その作業を行うのに必要なスキル、資格、機能などを持つ要員(人的リソース)しか割り当てることができない(表1参照)。これを**リソースの割り当て条件に関する制約**と呼ぶ。例えば、コーディング&テストの作業では、使用されているプログラミング言語の能力を持つことが制約となる。

(3) リソース割り当て可能期間に関する制約

ソフトウェア開発の各作業には、リソースの割り当て条件を満たすリソースでもその期間にスケジュールが空いてなければ割り当てることができない(表1参照)。これを**リソース割り当て可能期間に関する制約**と呼ぶ。非人的リソースでも同様のことが言える。それ故、ソフト

^{*}A Constraints Based Schedule Planning System for Software Development: Improvement of the Assign Method of Software Workers

for Applying GA," written by Daisuke Kinoshita^{*1}, Emiko Nemoto^{*2}, Mina Furusawa^{*2}, Rihito Yaegashi^{*3}, and Seiichi Komiya^{*1}.

^{*1}Graduate School of Shibaura Institute of Technology

^{*2}Shibaura Institute of Technology

^{*3}Graduate School of Saitama University

ウェア開発計画は人的リソースと非人的リソースの割り当て可能期間に依存する。

(4) リソースの能力的限界に関する制約

リソースが1日で実施可能な作業量には限界がある。例えば、要員が1日にAという作業とBという作業をそれぞれ4時間実施するという作業計画は、この要員の作業時間が合計で8時間になってしまう。1日に割り当て可能な時間を最大で8時間と考え、複数の作業8時間分を8時間でこなすのは不可能なので、capacity(能力的限界値)という概念を導入し、どんなに効率よく働いても、例えば8時間の80%しか有効に働けないと考える。このような制約を「リソースの能力的限界に関する制約」と呼ぶ。上記の例題はこの制約には該当しない。

3. 各工程に対する複数要員の割り当て方法

ほとんどの場合、各工程に割り当てられる要員の数は複数である。文献[1]のように1工程に対して1人しか割り当てることができないというのは非現実的である。そこで1工程に対して複数要員を割り当て可能にする方法を考える。

計画案の立案にはGA(遺伝的アルゴリズム)を用いており、すでに文献[1]でその有効性が示されている。ところが、GA(遺伝的アルゴリズム)では解(要員の割り当て案)の表現が一樣な長さでなければならない。しかし、各工程に対して複数要員を割り当てると、解の長さが一樣でなくなるという問題が出てくる。そこで、各工程に対する複数要員の割り当てを割り当てパターンという形で用意する。割り当てパターンについては以下で説明する。

表2 要員割り当てパターンの例

	Pattern1	Pattern2	Pattern3
A(よくできる)	1	1	
B(できる)			2
C(不得意)		1	
D(遂行不可能)			
所要日数	3	2	2

表3: 工程情報一部の例

工程名	必要なスキル	事前条件	事後条件	Pattern
DD1	SA 法、SD 法	レビュー報告書	詳細設計書	1,2,3
PE	性能評価技術	詳細設計書	性能評価報告書	1,2,3
CT1	C	性能評価報告書	プログラムチェックリスト etc	1,2

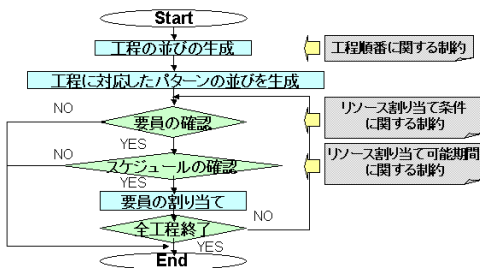


図2: ソフトウェア開発計画立案の流れ

例を挙げて説明する。表2は、この工程の要員割り当て人数のパターンが3つあることを示している。本研究では、作業者の能力を4段階{よくできる、できる、不得

手、遂行不可能}で評価し、それに基づいて割り当てパターンを表現する。パターン1は(この工程に求められる技術が) <よくできる>の水準にある要員がこの工程を一人で担当すると作業に3日かかり、パターン2は<よくできる>の水準にある要員1名と<不得手>の水準にある要員1名の計2名で担当すると作業に2日かかり、パターン3は<できる>の水準にある要員が2名で担当すれば作業に2日かかることを示している。このような要員割り当てパターンを基に工程の要員の数を決定する。

4. ソフトウェア開発計画の立案

ソフトウェア開発計画立案の流れは図2のようになる。はじめに工程の並びを生成する。工程の並びは作業順序の制約に基づいて生成する(図3上段)。工程の並びが生成されたらそれに対応したパターンの並びを生成する(図3中段)。パターンの並びは工程情報で与えられているパターンの中からランダムに選択し、生成される。

パターンの並びが生成されたら、左の工程から要員の確認、スケジュールの確認、要員の割り当てを行っていく。まず、その工程を行うのに必要なスキルを持った要員をデータベースから探索する。必要な要員が見つかったら要員のスケジュールを確認し、割り当て可能ならば要員を割り当てていく。もし、要員が存在しない、スケジュールに合う要員がない場合、この割り当てパターンは条件を満たしていない解となり破棄される。この作業を全工程に対して行い、すべてにおいて割り当て可能であった場合、その解を最適解の候補とする(図3下段)。

SA	SD1	SD2	DR	DD1	PE	CT1	DD2	DT	CT2	IT
Pattern2	Pattern1	Pattern3	Pattern1	Pattern1	Pattern1	Pattern3	Pattern3	Pattern2	Pattern1	Pattern1
A B	A	B	A B C	B	C	B	B	D	D	E F

図3: 工程要員人数決定方法

生成された最適解の候補の中から、プロジェクトマネージャーがもっとも適していると思われる計画案を選択できる。例えば、開発コストがもっとも低い案を選択したり、あるいは、開発期間がもっとも短い計画案を選択したりできる。

5. おわりに

本稿では、ソフトウェア開発計画立案問題を解決するために、ソフトウェア開発計画立案問題を行う際に考慮すべき制約を明らかにするとともに、制約の内容について述べた。そして、複数要員割り当て問題における問題を挙げ、その問題を解決するための方法を示した。

[参考文献]

[1] 八重樫理人, 原田将来, 野田寛之, 齋藤匠, 古谷慶太, 古宮誠一, "GAによるソフトウェア開発計画自動立案", 信学技報 TECHNICAL REPORT OF IEICE KBSE2002-48(2003-3), pp.27-32,2003
 [2] 古宮誠一, 樫山淳雄, "ソフトウェア開発における工程管理のためのプロセス・モデル," 知能ソフトウェア工学の動向と展望シンポジウム論文集, Pp.35-44, March 1992.
 [3] 古宮誠一, 澤部直太, 樫山淳雄, "制約に基づくソフトウェア開発計画の立案", 電子情報通信学会論文誌 D-I Vol.J79-D-I No.9, pp.544-557, 1996.