

学生のソフトウェア開発実習における制約条件を利用した作業スケジュール作成

宮原 佑也[†] 伊藤 邦彦[†] 澤島 義人[†] 清水 誠[†] 小形 真平[‡] 栗原 紘樹[‡] 松浦 佐江子[†]
 芝浦工業大学 システム工学部 電子情報システム学科[†]
 芝浦工業大学大学院 工学研究科 電気電子情報工学専攻[‡]

1. はじめに

本学には学部3年生を対象に、半期をかけて、オブジェクト指向開発を用いたグループワークによるソフトウェア開発実習がある[1]。今年度は6人1グループの4グループである。本実習では、web上で掲示板やファイルアップローダを提供し、計画書の作成支援を行うグループワーク支援システム(以下、支援システム)を用いて、学生のソフトウェア開発実習を支援している。

実習では、表1のフェーズ期間毎にタスクの期間と担当者を決めて、計画書を作成する。計画立案には、いつ作業が完了するかを示すだけでなく、作業を管理可能な形に分割させる効果と、作業に担当者が記入されることで、メンバーに、心理的なプレッシャをかけて、責任を持って作業を進めようとする効果がある[2]。しかし、期日間に作業が集中したり、グループの集合日に行うべき作業や集合可能メンバーが分からないために、計画をうまく実行できないという問題が生じており、現在の計画書のみでは上記の効果を十分発揮していない。

本研究では、個人が担当作業に責任を持ち、管理できる効果を与えるために、フェーズの期間内に、自分が担当する作業の日程表を作業スケジュールと定義し、学生に明示することで問題解決を図る。

表1 計画書の内容

	説明	項目
フェーズ	・5つに分けられた開発工程 ・フェーズごとに成果物を提出する	開始日, 終了日, 納品日
タスク	・フェーズを完了にさせるために複数の必要な成果物を作成する作業 ・授業で内容は決められている	開始日, 予定終了日
作業項目	・タスクの詳細内容を示し, 作業の分担を行う	担当者

2. 現在の作業進行上の問題点と計画書の改善点

2.1 期日間に作業が集中する問題

タスクの実施期間はグループで決めているが、最初の見通しの甘さと、作業の先延ばしによって、提出期限間に今まで行わなかった分の重い作業をする問題がある。タスク毎の作業時間が分からないため、最初から作業を実行しないと時間に余裕がなく、最終的に作業が終わらないことに気がつかない。フェーズ開始時から期日までに自分がやるべき作業を明示して、個人が認識することで解決を図る。

2.2 集合日時と集まって行う作業の問題

メンバーは互いの集合可能時間が分からないことと、作業をグループと個人のどちらで進めるのかを示すタスクの作業形態を決めていないため、集合日と、その日にどの作業をするのか決められない。そのため、事前に作業の準備ができず、集合日に、作業を決めてから長時間議論しても、集中できず思ったように作業が進まない。集合日にどの作業をし、誰が集まれるか、その日までに個人がやるべき作業とその期間を明示することで作業の準備ができ、集

Generation of Personal Work Schedules for Software Development Experiments.

[†] Yuya Miyahara, Kunihiko Ito, Yoshito Sawahata,

Makoto Shimizu, Saeko Matsuura

[‡] Shinpei Ogata, Hiroki Kurihara

[†] Shibaura Institute of Technology

Department of Electronic Information Systems

[‡] Graduate School of Engineering, Shibaura Institute of Technology
 Department of Electrical Engineering and Computer Science

まれる機会は限られることを認識することで解決を図る。

3. 作業スケジュール

作業スケジュールでの作業にはグループ作業と個人作業がある。グループ作業では、議論や成果物の統合を行う集合日とその開始時間、終了時間(分単位)、その時間帯に誰が集まるかを示す。個人作業では、締切日や集合日までに担当作業を行う作業期間を示す。

作業スケジュールは図1のフェーズ単位と図2の日付単位で示す。図1はフェーズの期間をカレンダー形式で表示し、作業名とその作業日を表している。例えば、「クラス図の作成」は11月7日から11月13日まで個人でクラス図を作成し、11月12日と11月14日に統合するグループ作業日があることを示す。図2はその日の作業の集合時間、集合可能な担当者と個人でやるべき作業を表している。

学生が個人の予定を考慮して、フェーズ開始時にフェーズ期間の集合日時を決めることが難しいため、作業スケジュールを自動作成する機能を支援システムに導入する。

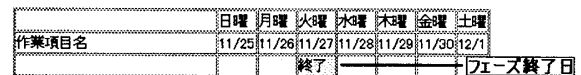
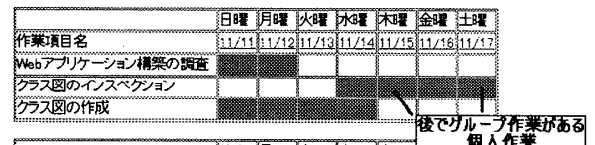


図1 フェーズの作業スケジュール

4. 作業スケジュールの自動作成

スケジュールを作成する方法として、数理計画法、メタヒューリスティクスがあるが、今回はフェーズ全体のスケジュールを即座に見せるために、短時間で作成できる優先規則を用いたシミュレーションを利用する[3]。これは、優先規則で並べた作業の順にスケジュールに割り振る手法で1つだけ解を求める。つまり、複数の作業スケジュールの中で最良のものを選択していない。そのため、優先規則をタスクの順序、タスクの予定終了日が早い順と、タスクの期間が短い順の3つの方法にし、この中で最も良いものを選択する。選択方法は作業毎に集まらない人数とその集合時間を掛けた値が最も少ない優先規則つまり、多くの担当者が集まれる時間帯が多いものとする。ここでタスクの順序は、作成と検証の手順と2006年度の主観的な経験により定義し、その順序に近づくように割り振る。

タスクを行うのに、必要な集まれる担当者と時間を与え、タスク毎に作業形態を考慮した作業スケジュールを作成

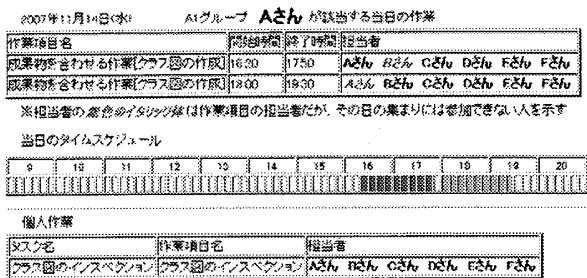


図2 日付の作業スケジュール

するために、計画書の項目に加えて、タスクの作業時間、週単位の集合可能な時間、タスクの作業形態、タスクが終了済みか、同じ時間帯に担当者は1つの作業しか行わないことを制約条件として、制約充足問題[4]を解くことで、作業スケジュールを作成する。タスクの作業時間は2006年度の報告書の作業時間から見積もりした結果を利用する。週単位の集合可能な時間は、月曜日から土曜日までの6曜日と9時から21時までの時限90分単位で全7時限42項目の時間帯がある。タスクの作業形態はグループでの議論によってタスクを完了するグループ話作業と、個人で成果物を作り、後にグループで統合作業をする個人分担当作業と、個人でタスクを完了する個人作業の3つがある。

作成方針はタスクの作業形態によって異なる。グループ話作業はタスクの開始日から予定終了日までの1時限毎に、すでに割り振られた他の作業と衝突せず、担当者で集まらない人数が規定人数であれば、その時間帯にグループ作業を割り振る。これをタスクの作業時間が0になるまで、タスクの期間内の時限毎に繰り返す。規定人数とは、初期値は全員集まれる0であり、タスクの期間内を終えて、タスクの作業時間が全て割り振られていなければ、規定人数を1増加し、再びタスクの期間に割り振れるかを繰り返す。つまり、2週目は1人集まなくても作業を割り振る。担当者が1人も集まれる時間がない場合は解を求めるために、割り振れなかったタスクとする。個人分担当作業は集合日をするためにタスクの予定終了日から開始日までをグループ話作業と同様のことを行い、割り振る。そして、タスクの開始日から集合日までの期間に個人作業を割り振る。個人作業はタスクの開始日から予定終了日までの期間に個人作業を割り振る。

5. 作業スケジュールの提示手順

- (1) 各メンバーが週単位の集合可能な時間を入力する。
- (2) グループが計画書の項目を入力する。
- (3) システムが制約条件に合うように、作業スケジュールを作成する。
- (4) メンバー(Aさん)が個人のフェーズの作業スケジュール(図1はAさんのもの)を表示させる。
- (5) メンバーが日付(11月14日)を選択すると、日付の作業スケジュール(図2)が表示される。

6. 現在のグループの作業状況における考察

図3は支援システムの会議内容を記録する議事録のグループ毎の週単位の集合時間を表している。ここで、横軸は週であり、縦軸は分単位の集合時間である。

2006年度は、週が進むにつれて、集合時間が増加していき、最も右の週つまり、期日間近に行う作業が最も時間が長い。2007年度の期日間近に行う作業は2006年度に比べて、緩和している。作業スケジュールの効果とは断定できないが、問題の解決に影響を与えている。

しかし、2007年度で1グループの作業時間が急激に増

えている。このグループはタスクの期間がフェーズの期間と同じであり、担当者を決めてない計画であるためである。

第2フェーズは第1と同様であるが、第3フェーズでは、議事録の記録が少なくなり、集合時間に差が出ている。

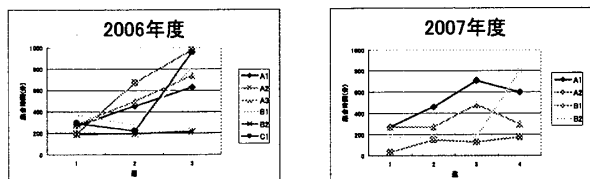


図3 2006年度、2007年度第1フェーズの週ごとの集合時間

表3 2グループの第1, 2フェーズの作業スケジュール作成内容

作成日	フェーズ番号	割り振られたタスク数	割り振れなかったタスク数	期限切れのタスク数
2007/9/19	1	1	0	0
2007/9/25	1	1	0	0
2007/9/26	1	8	7	0
2007/10/3	1	5	2	3
2007/10/10	1	0	0	14
2007/10/24	2	8	1	0
2007/10/29	2	4	0	5
2007/10/30	2	5	0	5
2007/10/31	2	10	1	2
2007/11/1	2	6	0	7
2007/11/3	2	4	0	9
2007/11/5	2	4	0	9

2007/9/26	1	1	0	0
2007/10/10	1	0	0	1
2007/10/24	1	0	0	15
2007/10/24	2	10	4	0

表3の項目を説明する。割り振られたタスク数と割り振れなかったタスク数とは、作業に必要な作業時間がスケジュールに割り振られた数とない数である。期限切れとは、終了にせず、作成日より前に予定終了日がある数である。

ここで、割り振れなかったタスク数は、そのタスクの期間内にどの時間帯も空いている時間がないことを示す。この数が多い表3の「7」の原因は、1日で多くの作業を行おうとした場合や、タスクの期間が集まらない曜日となっている場合が該当する。期限切れの数が多い各フェーズの終盤の表3の「14など」は、タスクが終わっていても、計画書で終了に設定することがされていないためである。

タスクの期間の長さ、順序、作業の担当者を十分に考慮して、計画立案すると、割り振られたタスク数が多く、役立つ作業スケジュールが作成される。

2つ目のグループの作成数が少ないのは、計画書にタスクの期間は定義しているが、担当者を決めていないなどの制約条件が不足して作成されなかったためである。

7. まとめ

作業スケジュールを学生に提示し、問題の解決を促した。しかし、計画を立てていないグループがあり、2つの問題の原因に、この要因も含んでいないと感じた。今後は、学生のファイルアップロード日から毎週の作業状況を分析することと、学生へのアンケートを実施し、問題の解決に作業スケジュールがどのような影響を与えたかを分析する。

8. 参考文献

- [1]松浦佐江子：実践的なソフトウェア開発実習によるソフトウェア工学教育、情報処理学会論文誌、Vol48, No.8, pp.2578-2595, 2007.
- [2]Scott Berkun : Art of Project Management, O'Reilly & Associates Inc,2005.
- [3]黒田充, 村松健児: 生産スケジュール, 朝倉書店, 2002.
- [4]淵一博 他: 制約論理プログラミング, 共立出版, 1989.