

ソフトウェア・マネージメント技法

7G-5

—プロジェクト管理シミュレーションモデルの構築—

上村松男¹ 山田 茂² 阿部勝徳¹ 森川佳子¹ 尾崎俊治² 藤野喜一³

1 日本電気マイコンテクノロジー(株) 2 広島大学工学部 3 日本電気(株)

1 はじめに

情報システムの高度化・複雑化にともない、プロジェクト管理(企画・計画・事業)には多大な努力が払われている。

ソフトウェアマネージメント活動における生産管理は、予算・費用管理、スケジュール・工数管理、事業戦略/製品計画、外注折衝、負荷管理など数多くあるものの、これらは管理者の管理事項の一部にすぎない。管理者は先の管理事項以外にも生産技術に関連した技術管理事項が山積みしている。これらの計画や管理は、プロジェクト管理が進むにつれてソフトウェア製品化過程に生ずる種々の要因と問題が複雑に絡み合い、時間経過に伴って実態を見えにくくしている。このことが、見える管理が重要になっている所以でもあり、これらを体系的にまとめた、一貫性のあるマネージメント技法[1]が望まれる。

本論文では、ソフトウェア生産管理のマネージメント活動を支援するプロジェクト管理モデルの構築法とシミュレーション実行結果の有効性を評価したので報告する。

2 プロジェクト管理技術

プロジェクト管理におけるスケジューリングと負荷計画は、納期遅れ、設備稼働率、工程負荷平準化、在庫適正化など重要な問題である。例えば、この基本となる工程の終了の判断となると、期日になった、予定工数に達した、出荷品質[2]を満たしたなど、その基準は様々である。また、ソフトウェアは個別受注生産方式であるため業務分析や機能分析、いわゆる科学的なマネージメントシステムWBS(Work Breakdown Structure)技法による活動の詳細化における、非定型作業の計測をはじめ、定式化が難しい。さらには、製品仕様は顧客仕様の傾向が強くなり(付加価値)、しかも製品のライフサイクルが極端に短縮される。このような傾向は、ソフトウェア事業においてますます強まるであろう。このようなことから、新製品の導入や、先の諸問題に素早く対応できるシミュレーションシステムの構築が望まれている。

次章では、プロジェクト管理の進捗度に関する具体的評価を試みたので概説する。なお、解析には汎用シミュレーション言語としてSLAMII(Simulation Language for Alternative Modeling)[3]を用いた。

3 プロジェクト管理シミュレーション

プロジェクト形態には、①個別受注生産プロジェクト、②既存システム維持プロジェクト、③研究開発プロジェクトなどがある。

プロジェクトとは、“明確な達成目標をもち要求機能品質を実現するための作業体制・組織”である。このことから、プロジェクトは技術・製品化ノウハウの蓄積や知的財産権(特許・著作権・商標)などを確保する上で一般的に体系化されることが多い。

表1は、先のマネージメント活動をサブ・プロジェクト

ト単位に労力を配分したもので、プロジェクト構成ネットワークに割り当てられる(3.2節参照)。このPERTネットワークフローにおいて、各アクティビティは対応するプロジェクトの作業を表す。ここで一つのアクティビティの作業を行う要員の集合をプロジェクト単位と呼ぶ。各プロジェクト単位への要員数の最適配分は、線形計画法を応用することによって決定できる。

記号	内容	期間(三角分布)		
		最短	最尤	最長
A	要求仕様定義	0.8	1.0	1.2
B	開発計画	0.5	0.5	0.7
C	基本設計	0.4	0.5	0.6
D	機能設計	0.7	0.8	1.3
E	詳細設計	1.3	1.5	1.7
F	コーディング/単体テスト	2.0	2.3	3.0
G	結合テスト	0.8	1.0	1.5
H	既存プログラムの活用	2.5	3.5	3.8
I	結合テスト	1.6	2.5	2.8
J	DB設計	2.5	3.0	3.3
K	DB構築	2.5	2.8	3.3
L	統合テスト	0.8	1.0	1.3
M	機能設計	0.7	1.0	1.1
N	詳細設計	1.3	1.4	1.6
O	コーディング/単体テスト	2.2	2.5	3.0
P	基本設計(通信系)	0.7	1.0	1.2
Q	機能設計	0.8	0.9	1.2
R	詳細設計	1.3	1.5	1.7
S	コーディング/単体テスト	1.8	2.0	2.4
T	結合テスト	1.8	2.1	2.5
U	開発計画	0.5	0.5	0.7
V	テストケースの設計	1.3	1.4	1.6
W	評価パッケージの開発	1.3	1.5	1.6
X	評価プログラムのテスト	2.2	2.4	3.0
Y	結合テスト	1.5	1.9	2.5
Z	製品テスト	1.0	1.2	1.5

表1 費用見積りモデルに基づく期間配分

3.1 マネージメントモデル

ソフトウェア生産管理は一般に、①要求仕様定義をもとにシステム分析を行う、②労力投入配分と最小プロジェクト単位まで掘り下げたスケジュールを立てる、③ネットワークで示した開発計画が計画通りに進行するかどうかを診断する、④③で得た情報を②にフィードバックして、施策や改善策をとる、などの手順で進められる。

本モデルの特徴は、個々の業務(アクティビティ)で費やされた活動を確率分布として取り扱った点にある。通常のPERT/CPMを用いる手法では前述のソフトウェア開発における問題点である不確定要因による作業の遅れや、その影響を考慮できない。そこで、我々は以下の前提条件を設定してモデル化を試みた。

- (1) アクティビティの相互関係はPERT図で表現する。
- (2) 複数のアクティビティが一つのノードに集まる場合は、すべてが揃わなければ次工程に移行できないものとする。
- (3) 各アクティビティの完了期間は、過去の経験と分析結果から推測した分布に従う。

Software Management Technology - A Construction of Software Project Management Simulation Model
Matsuo UEMURA¹, Shigeru YAMADA², Katsunori ABE¹, Keiko MORIKAWA¹, Shunji OSAKI², Kiichi FUJINO³

1 NEC Microcomputer Technology, Ltd.

2 Department of Industrial and Systems Engineering, Hiroshima University.

3 NEC, Ltd.

3.2 ネットワークフローの構築

モジュールの流れはSLAMネットワーク記述言語で表現する。これは、PERTネットワークモデルの各アクティビティに、その作業の完了期間の分布を付記したものである。各ノードでは多様な情報を保持でき、結果の出力時に利用できる。これをソース(始点)からシンク(終点)まで100回程度繰り返しシミュレートすることで、進捗度のばらつきの様子を推測する(図2参照)。

そこでまず、各プロジェクトの構成として図1は、管理業務と作業手順を示している。例えば、A-B-C-D-E-F-G-L-Zは業務アプリケーション、A-B-C-H-I-L-Zは既存プログラムの流用、A-B-C-J-K-L-Zはデータベース、A-B-C-M-N-O-T-Zは通信関係業務、A-B-P-Q-R-S-T-Zは通信系、およびA-U-V-W-X-Y-Zはテスト関係のパスを表している。このネットワークフローをプログラム化すると、シミュレーションプログラムが生成される。

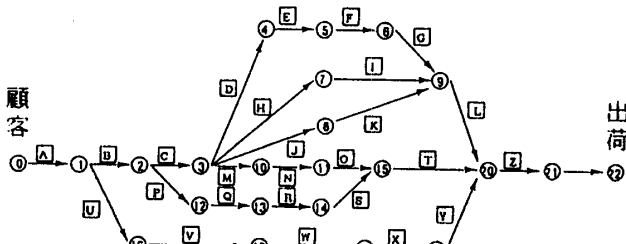


図1 マネージメント・ネットワークフロー

将来的には、このアクティビティにマネージメント活動のWBSを作成する必要がある。WBSは、プロジェクト全体からそれを構成するシステムレベルへ、さらにサブシステムレベルへと順次トップダウンで展開されるものであり、容易にしかも漏れなく全作業を把握・識別することが可能である。WBSを最末端までブレイクダウンすれば、それぞれのワーク(作業)は単一のリソースに一致するところまでブレイクダウンされることにより、リソースの見積りが可能になる。また、これを上位にサマライズすれば、各コントロールに必要なリソースを配分できる。そのためには、まず①生産物を規格化、共通化することによりノウハウの蓄積を可能にする、②作業手順や作業指示方法を統一することにより、作業効率や管理レベルを向上させる、③開発対象、環境、組織やプロジェクトごとに適した開発手法を採用することにより、個人差による品質のばらつきを最小にとどめるなどの整備が必要となる。

4 シミュレーション結果

シミュレーション評価では、完了時間の分布を過去のプロジェクトの費用見積りモデル[1]から設定した(表1参照)。図2はSLAMIIでシミュレートした結果であり、各ノードの到着時間経緯の平均、標準偏差、変動係数、最小値、および最大値を算出した。また、ヒストグラムを用いてプロジェクトの完了時間の回数の累積値を出力した。これは100回のシミュレーション結果なので、このデータは中心極限定理を適用することができ、平均値の99.7%の信頼区間は、

$$\text{信頼区間} = \text{平均} \pm 3 \sqrt{\frac{\text{標準偏差}^2}{\text{シミュレーション回数}}} \\ = 11.0 \pm 0.1 (\text{ヶ月})$$

となる。ここで、入力データの三角分布の最尤値に対してクリティカルパスを結ぶと、約10ヶ月程度でプロジェクトは完了する。マネージメント・ネットワークフローを設計する上で、ノード(WBS)のコスト(重み)を決めるためには、他のパスとの日程調整が必要不可欠であるこ

とはいうまでもない。なぜなら、並列作業の最も遅れた作業が進捗度を決定することが大きな理由の一つである。表1の三角分布に従えば平均的に約11ヶ月で完了することになる。

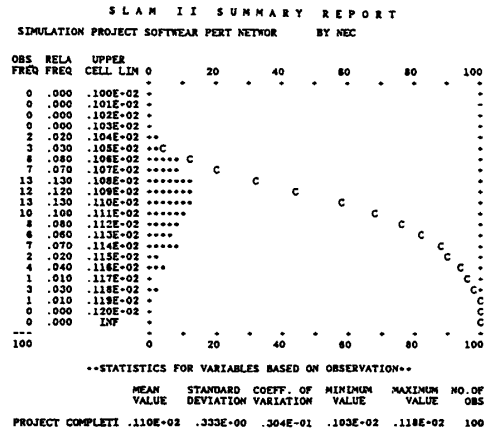


図2 プロジェクト管理シミュレーション結果

以上の分析結果から完成期間の確率は、10.5ヶ月では5%、11.0ヶ月では58%、11.5ヶ月では95%、11.9ヶ月で100%と推定できることが分かった。今後より信頼(納得)できるかどうかは管理者の判断)できる値を得るためには、シミュレーション回数を増やすことが必要である。また一方では、より複雑なプロジェクトに対応させるためには、スケジュールの遅れに対して①工数を増やす(要員の再投入、残業/休出)、②納期を遅伸する(段階的リリース)、③線表上でやりくりする(並列)、④工数を削減(機能を削減、やり方の改善)などシミュレーションの段階での基礎評価が、構築の際考慮すべき重要な点である。

5 おわりに

本論では、費用見積りモデル[1]で得られた期間から、マネージメント活動ネットワークチャートを構築して、実際の業務アプリケーションの最適化を試みた。しかし実際のプロジェクト管理の場面では、納期遅れ、予算の大幅な超過、品質不良・性能不足など重要性を認識しながらも抜本的対策がないというのが実状である[4]。このような背景から、プロジェクト管理シミュレーションを遂行するためには、今後かなり多くのプロセスを踏む必要があり、かつ多大な時間を費やさなければならない。シミュレーションプロセスとしては、マネージメント活動データの計測、モデリング、シミュレーション実行代替案の作成、出力結果の分析、妥当性評価および経済性、さらには結果のプレゼンテーションなど、結果をいかによく表現し説得力をもたせ、評価・決断に結びつけていくかがプロジェクトへの適用の鍵である。

最後に、日頃ご指導頂いている日本電気マイコンテクノロジー(株)の松津取締役様に謝意を表します。

参考文献

[1] 上村松男, 山田茂, 広中正彦, 尾崎俊治: "プロジェクト管理シミュレーションモデルの構築・評価", 情報処理学会ソフトウェア工学研究会報告, 71-13(1990).
 [2] 上村松男, 寺本雅則共訳: "ソフトウェアの信頼性(M. L. Shooman著)", 丸の内出版, 東京(1989).
 [3] A. Alan, B. Pritsker: "SLAM III Network Models for Decision Support", Prentice-Hall, New Jersey(1989).
 [4] 山田茂, 大寺浩志: "ソフトウェアの信頼性 - 理論とその実践的応用 -", ソフト・リサーチ・センター, 東京(1990).