

5S-9 ソフトウェア開発期間推定における作業時間分布の考察

渡辺淳志† 島和之† 松本健一† 鳥居宏次†‡

†大阪大学 ‡奈良先端科学技術大学院大学

1. まえがき

ソフトウェア開発計画の作成において開発期間の予測は重要な作業の一つである⁽¹⁾。ソフトウェア開発の各作業時間を確率分布で表し、プロジェクト全体の開発期間を確率分布として予測する方法がいくつか提案されている⁽²⁾。しかし、それらの方法において仮定されている作業時間の独立性や作業時間分布の形状の妥当性は明らかでない。

本稿では、実際のソフトウェア開発過程から収集したデータに基づいて、作業時間の独立性を確認する。更に、実データとの適合度検定の結果より、作業時間を表す分布関数としてポアソン分布が適していることを示す。

2. 開発期間の推定

2.1 PERT/CPM

PERT/CPM(Program Evaluation and Review Technique / Critical Path Method)は開発期間の推定法として広く用いられている⁽²⁾。PERT/CPMでは以下のような手順で開発期間を推定する。

1. 開発を設計、コーディング、デバッグといった作業に分割する。
2. 作業を矢印(アロー)で表し、作業間の順序関係に従って結合点(ノード)で作業を接続したアローダイアグラムを作成する(図1参照)。
3. 各作業に要する時間(一定値)を決定し、矢印に付加する。
4. 開発の開始から終了までの道(パス)の中で、パス上の作業時間の合計が最大のパス(クリティカルパス)を求める。
5. クリティカルパス上の作業時間を合計して、開発期間全体の作業時間を求める。図1の場合、開発期間は13日である。

2.2 作業時間分布

PERT/CPMでは、通常、各作業に要する時間を一定と仮定している。しかし現実には、作業時間は必ずしも一定であるとはいえない⁽²⁾⁽³⁾。そこで、PERT上の各作業に要する時間は確率分布に従うと仮定して、開発期間全体を求める方法が提案されている。代表的な方法として、「三点見積り法」と山田が提案する「数値シミュレーション法」がある⁽³⁾。

三点見積りでは、各作業時間を β 分布に従う確率変数と仮定し、各作業に要する時間の平均、分散を推定する⁽³⁾。クリティカルパス上の各作業時間の期待値と分散のそれぞれの合計が、開発期間の期待値と分散になる。

数値シミュレーション法では各作業が個別の時間分布に従って進行するとして、モンテカルロ法を用いて開発期間の分布を算出する。作業時間分布としては三角分布が用いられている⁽³⁾。

2.3 開発期間分布の算出

アローダイアグラムで示された作業間の順序関係と各作業の時間分布より、開発期間全体の分布を次のように算出することが出来る。

アローダイアグラム上の直列な2つの作業について考える(図2(a)参照)。作業1、作業2に要する時間の確率密度関数をそれぞれ f_1 、 f_2 とする。2つの作業に要する時間を t とすると、その確率密度関数 f は次式(1)で表すことが出来る。

$$f(t) = \int_0^t f_1(t_1)f_2(t-t_1)dt_1 \quad (1)$$

従って、直列な2つの作業は、作業時間の確率密度関数が式(1)となるような1つの作業で置き換えることが出来る(図2(b)参照)。

次に、アローダイアグラムの並列な作業について考える(図3(a)参照)。式(1)を用いることにより、並列な作業は、並列な2つの作業に置き換えることが出来る(図3(b)参照)。図3(b)の並列な2つの作業、作業1、作業2に要する時間の確率密度関数それぞれ f_1 、 f_2 とする。2つの作業が共に終了するまでの時間を t とすると、その確率密度関数 f は、次式(2)で表すことが出来る。

$$f(t) = f_1(t) \int_0^t f_2(x)dx + f_2(t) \int_0^t f_1(x)dx \quad (2)$$

従って、並列する2つの作業は、作業時間の確率密度関数が式(2)となるような1つの作業で置き換えることが出来る。

以上に示したような作業の置換を繰り返すことにより、アローダイアグラム上の作業を1つの作業に帰着することが出来る。帰着された作業の時間の確率密度関数が開発期間の分布を表す。

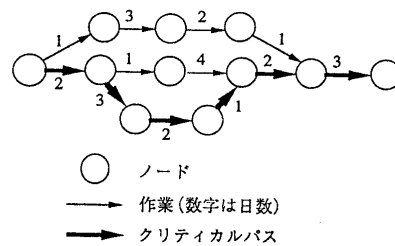


図1 アローダイアグラムの例

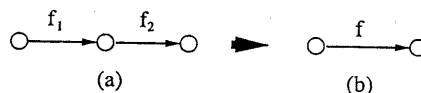


図2 直列な作業の置換

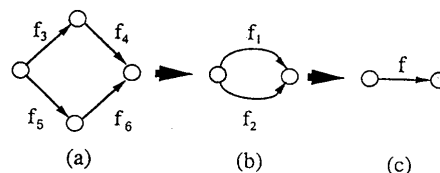


図3 並列な作業の置換

3. 実データに基づく分析

3.1 収集データ

作業時間の分布を求めるため、大阪大学基礎工学部情報工学科2年生に対して行われた学生実験からデータを収集した。収集データは各学生が実行したコマンドの履歴データである。実験では、同一の仕様に基づいて、各学生が一種の在庫管理プログラムを作成する。使用言語はC言語、プログラムサイズは平均300行、学生数は55名である。

A Study on Work Time Distribution for Estimating the Period of Software Development.

Atsushi WATANABE†, Kazuyuki SHIMA†, Ken-ichi MATSUMOTO† and Koji TORII†‡

†Osaka University, ‡Advanced Institute of Science and Technology, Nara

ここでは、開発作業を次の3つに大別した。これら3つの作業は直列な関係にある。

- 作業 1: 設計及びコーディング
- 作業 2: シンタックスエラーの除去
- 作業 3: テスト及びデバッグ

作業時間はそれぞれの作業に対応するコマンドの実行結果より学生毎に算出した。それぞれの作業時間の分布とそれらを合計して求めた開発期間の分布を図4に示す。

各作業時間及び開発期間の統計量を表1にまとめる。表1中の積率歪度は歪みを表し、時間分布が正(右方向)に歪んでいる時、正の値となる。また、積率尖度は尖りを表し、正規分布では値が3になる。

表1 作業時間の統計値

作業	平均	標準偏差	積率歪度	積率尖度
作業 1	8.44	3.26	0.11	2.55
作業 2	0.89	1.30	1.98	7.12
作業 3	3.25	1.75	0.77	3.23
開発期間	12.58	4.12	-0.11	2.20 (単位: 日)

表1より各作業時間の積率歪度はいずれも正であることがわかる。従って、作業時間分布を表す分布関数としてポアソン分布が適当ではないかと考えられる。また、表1より作業3の積率尖度が3に近い。従って、作業時間分布を表す分布関数として正規分布も考慮する必要があると思われる。

3.2 作業時間の独立性

作業時間分布から開発期間を推定するためには、各作業時間が独立であると仮定する必要がある。この作業時間の独立性を確認するために、作業時間の相関を調べた。作業1と作業3の作業時間の散布図を図5に示す。相関係数は作業1と作業2の間で0.17、作業2と作業3の間で0.04、作業1と作業3の間で0.02である。

従って、本実験における3つの作業時間は互いに独立であると考えられる。

3.3 4つの分布関数との適合度

作業時間分布として三点見積り法では β 分布が、数値シミュレーション法では三角分布がそれぞれ用いられている。これに対し、3.1で示したように、実データの形状からは、ポアソン分布や正規分布が作業時間分布を表す分布関数として適しているように思われる。そこで実データに対するこれら4つの分布関数の適合度を比較する。 χ^2 検定による検定値と検定結果を表2にまとめる。検定の有意水準は10%である。

表2 作業時間分布の検定値と検定結果

分布関数	作業 1	作業 2	作業 3
	χ^2 値 (自由度)	χ^2 値 (自由度)	χ^2 値 (自由度)
三角分布	3.98 (8) 採択	15.88 (5) 棄却	2.95 (5) 採択
β 分布	2.90 (8) 採択	6.76 (3) 棄却	4.85 (5) 採択
ポアソン分布	1.89 (7) 採択	3.70 (3) 採択	1.06 (6) 採択
正規分布	2.20 (7) 採択	24.90 (4) 棄却	8.13 (6) 採択

表2より、 β 分布、三角分布、及び、正規分布は作業2で棄却され、ポアソン分布は全ての作業で採択されたことがわかる。また、ポアソン分布では全ての作業で χ^2 値が最小となっていることがわかる。

以上の結果より、プログラム開発の作業時間分布としては、従来用いられていた β 分布や三角分布よりもポアソン分布を使用するのが適していると考えられる。

3.4 ポアソン分布を用いた開発期間の推定

2.3で述べた方法を用い、各作業時間分布にポアソン分布を使用して開発期間に要する時間の分布を求めた。求めた分布と実際の開発期間との間で χ^2 検定を有意水準10%で行なった結果、 χ^2 値7.04、

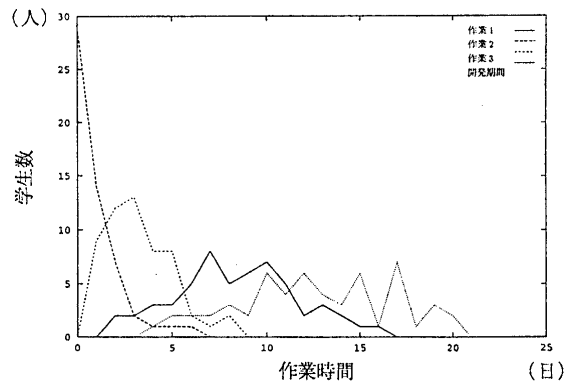


図4 作業時間分布

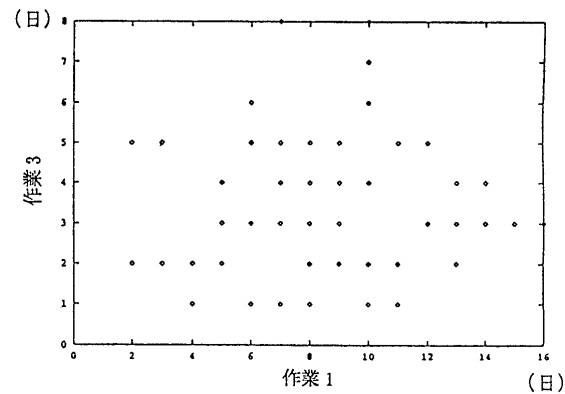


図5 作業時間の散布図

自由度8となり、採択された。

4. あとがき

本稿では大学の学生実験から収集したデータに基づいて、ソフトウェア開発の各作業時間の独立性、及び、作業時間分布と代表的な4つの確率分布との間の適合度を分析した。その結果、各作業時間が独立であることを確認した。更に、作業時間分布を表す分布関数としてポアソン分布を使用することが有効であることを確認した。

現在、作業間の関係がより複雑な開発過程を対象としたデータの収集、分析を検討中である⁽⁴⁾。

文 献

- (1) A. Ralston and E. D. Reilly, Jr. : "Encyclopedia of Computer Science and Engineering", Van Nostrand Reinhold Company(1983), 棟上昭男 監訳, "コンピュータ大辞典", pp. 1006-1009, 朝倉書店(1987).
- (2) 情報システムハンドブック編集委員会: "情報システムハンドブック", 培風館(1989).
- (3) 山田茂: "ソフトウェアマネジメントモデル入門", bit, 23, 3, pp. 300-307(1991).
- (4) 渡辺淳志: "作業時間の確率分布を用いたソフトウェア開発期間推定の試み", 大阪大学基礎工学部特別研究報告(1992).