

自主研究

# COCOMO IIによる工数見積り

「経済調査会ソフトウェア開発データリポジトリ」を用いた検証

# COCOMO IIによる工数見積り

## 「経済調査会ソフトウェア開発データリポジトリ」を用いた検証

松本 健一 奈良先端科学技術大学院大学 情報科学研究科  
大岩佐和子 押野 智樹 一般財団法人 経済調査会 調査研究部 第二調査研究室

### 1 はじめに

ソフトウェア開発プロジェクトにおいて、開発工数の見積りは、工期やスケジュールを検討し、必要な開発体制や人員を確保する上で重要な作業の一つである。情報処理推進機構が2012年度に行った調査でも、エンタプライズ系ソフトウェア開発企業の77.4%、組込みシステム開発企業・組込み系ソフトウェア開発企業の65.1%が、社内基準、もしくは、過去の実績データに基づいて、開発工数（同調査では、「工程」とされている）の見積りを行っている [IPA]。

代表的な見積り手法の一つは、開発対象ソフトウェアや開発プロジェクトの特性値をパラメータとする数理モデル、いわゆる「見積りモデル」を用いる手法である。古くは、Walstonらが、米国IBM社におけるソフトウェア開発プロジェクト60件の実績データに基づき1977年に提案したモデル [Walston] が知られているが、より広く用いられているものとしては、Boehmが、米国TRW社で収集したソフトウェア開発データに基づき1981年に提案したCOCOMO (COConstructive COst MOdel) がある [Boehm]。

COCOMOは、多くの工数見積りモデルと同様に、開発対象ソフトウェアの規模（行数、KLOC）を主要なパラメータとした、比較的単純な構造のモデルである。ただし、モデルの概念や適用方法に関する文書が整備され、適用事例が多くの企業から報告されている。ソフトウェア開発管理の新しい概念や技術の登場および開発プロセスの変化に対応するため、モデルの改良が行われ、いくつかの派生モデルも提案されている。現時点では、2000年に提案されたCOCOMO II 2000（以下では、単にCOCOMO IIと表す）が最新となる [CII] [Kikuchi]。

本稿では、最初にモデルが提案されてから30年余

り、最新モデルが提案されてからでも10年余りが経過している米国生まれのCOCOMOが、今日の日本のソフトウェア開発プロジェクトにも適用可能かどうかを、実際のソフトウェア開発プロジェクトのデータを用いて検証する。具体的には、「経済調査会 ソフトウェア開発データリポジトリ」が提供するデータを対象として、COCOMO II適用の具体的な方法を示すと共に、COCOMO IIによる見積り工数と実績工数の比較を通じて、その有用性を検討する。以降、2章では、COCOMO IIの概要を紹介し、3章では、同リポジトリが提供するソフトウェア開発データと見積りモデルパラメータの対応関係や見積り値の算出方法などを具体的に示し、4章において、いくつかのモデル適用結果を列挙し、5章では、今後の課題について考察し、6章で全体を簡単にまとめる。

### 2 COCOMO II

COCOMO IIにおける工数見積りモデル（見積り式）は次のとおりである [CII]。

$$PM = 2.94 \times Size^E \times \Pi EM_i + PM_{AUTO} \quad (式1)$$

$$Size = (1 + REVL/100) \times (KNSLOC + KASLOC)$$

$$E = 0.91 + 0.01 \times \sum SF_j$$

PM：工数見積り値（人月）。対象とする工程は、「基本設計」から「総合テスト（ベンダ確認）」まで。

PM<sub>AUTO</sub>：ソースコードの自動生成・変換に要する工数  
REVL：要件の変動率。「要件変更により破棄されるソースコードの割合 (%)」

KNSLOC：新規作成されるソースコード規模 (KLOC)

KASLOC：再利用や流用により得られるソースコード規模 (KLOC)。自動生成・変換されるソースコー

ドの割合、および、再利用や流用が設計、コーディング、テスト等の工数に与える影響に基づいて補正済み。

$SF_j$  : 規模要因 (Scale Factor) 値

$EM_i$  : コスト要因 (Cost Driver) 値

COCOMO IIによる工数見積値 $PM$ は、いわゆる、基本設計から総合テスト（ベンダ確認）までの工数であり、ラショナル統一プロセス (Rational Unified Process ; RUP) が定義するところの推敲 (Elaboration) と作成 (Construction) の工数である (図-1参照)。計画や要件定義、ユーザ環境におけるテスト等の工数は対象外である。また、計画工程や要件定義工程など、プロジェクトの初期に用いるための「Early Designモデル」と、基本設計工程以降で用いる「Post-Architectureモデル」の2つが定義されている。両モデルの主な違いは、コスト要因の違いである。Post-Architectureモデルでは17個のコスト要因を用いるが、Early Designモデルでは、それら17個のコスト要因を縮退した (統合した) 5個のコスト要因を用いる。本稿では、Post-Architectureモデルを用いるものとする。

ソースコード規模の単位は、KLOC (Kilo source Line of Code) であり、Software Engineering Institute (SEI) が定義する「論理行数」を用いることとされている [Park]。また、Early Designモデルでは、未調整ファンクションポイントUFP) をKLOCに換算して用いてもよいとされており、主要な開発言語については換算

表が用意されている。

規模要因とコスト要因にはそれぞれ6つの等級 (Early Designモデルのコスト要因のみ7つの等級) が設定されており、見積り対象ソフトウェアやその開発プロジェクトがそれぞれの等級に該当するかを評価するための基準、および、見積り式で用いる値が定義されている。規模要因とコスト要因の等級と見積り式で用いる値の一覧を付録に示す。評価基準については、文献 [Kikuchi] [CII] を参照されたい。

なお、ソースコードの自動生成・変換に要する工数 $PM_{AUTO}$ は、自動生成・変換されるコードの割合や自動生成・変換における生産性などに基づいて算出されるとされているが、詳細については割愛する。また、再利用や流用により得られるソースコード規模 $KASLOC$ の算出や補正の詳細についても割愛する。更に、COCOMO IIでは、見積り工数 $PM$ 等に基づいて開発期間を見積もるための式も定義されているが、ここでは割愛する。

### 3 モデル適用

#### 3-1 利用データ

モデル適用に用いるのは、「経済調査会 ソフトウェア開発データリポジトリ (以降では、「経済調査会リポジトリ」と略す。)」に蓄積されているソフトウェア開発データである [ERR]。同リポジトリは、経済調査会が平成19年度から24年度に実施した「ソフトウェ

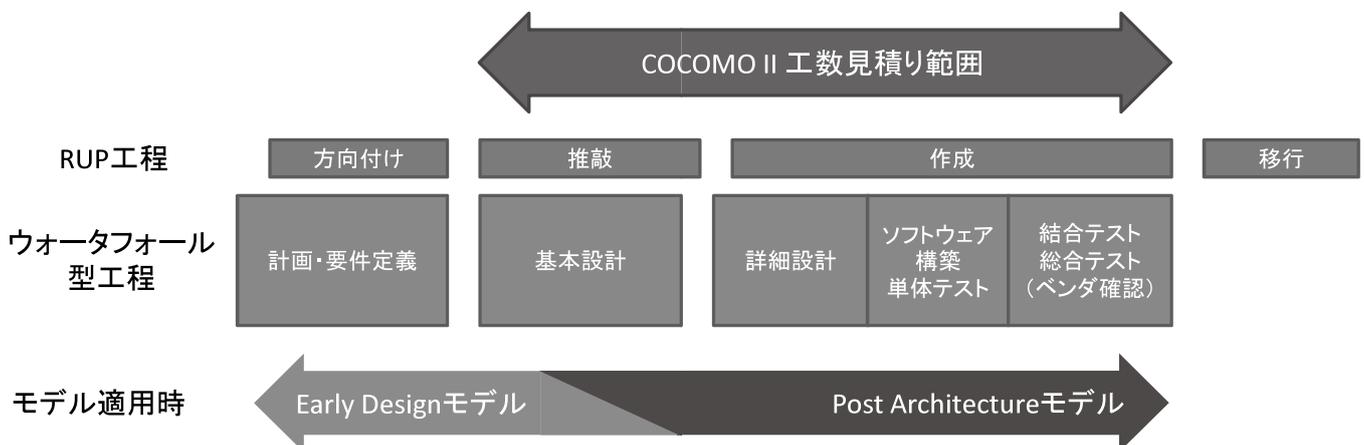


図1 COCOMO IIにおける工数見積り範囲とモデル適用時期

「ソフトウェア開発に関する調査」で収集された181社666プロジェクトのデータで構成されている。本稿で用いるのは、そのうち、次の条件を満たすプロジェクト388件のデータ(プロジェクト特性値)である。

- 基本5工程(「基本設計」、「詳細設計」、「ソフトウェア構築」、「結合テスト」、「総合テスト(ベンダ確認)」)が実施されている。
- ソフトウェア規模として、KLOCまたはファンクションポイント(FP)の実績値が記録されている。
- 工数(人月)の実績値が記録されている。

### 3-2 モデルパラメータとプロジェクト特性の対応付けと既定値

本稿では、(式1)で定義されているパラメータのうち、経済調査会リポジトリに蓄積されているプロジェクト特性値と直接的に、あるいは、近似的に対応付けることのできるパラメータのみを工数見積りモデルに用いることとする。具体的には次のとおり。

- ソースコードの自動生成・変換に要する工数( $PM_{AUTO}$ )を別途見積もることはしない。
- 要件の変動率、あるいは、要件変更により破棄されるソースコードの割合として定義される $REVL$ は用いない。
- 再利用や流用により得られるソースコード規模 $KASLOC$ を別途算出することはしない。(新規作成されるソースコードの規模が工数に与える影響と再利用や流用で得られるソースコードの規模が工数に与える影響との間には差がないものとする。)
- 5個の規模要因のうち、 $PREC$ 、 $RESL$ 、 $PMAT$ の3個のみを用いる。残る $FLEX$ と $TEAM$ については、等級 $N$ (中位)をとるものとする。
- 17個のコスト要因のうち、 $RELY$ 、 $PVOL$ 、 $ACAP$ 、 $PCAP$ 、 $SCED$ の5個のみを用いる。残る12個については、等級 $N$ (中位)をとるものとする。(等級 $N$ の値は1と定義されている。コスト要因は、(式1)においてソースコード規模への乗数とされているので、これら12個のコスト要因は、見積りには反映されないことになる。)

さらに、対象プロジェクトの状況やニーズにより即した見積りとすべく、工数見積りモデルに次のような

修正を加えるものとする。

- 工数見積り値を、対象プロジェクトにおける「1か月の所定労働時間 $MM$ (時間)」で補正する。具体的には、COCOMO IIでは1人月が152時間と規定されていることから、見積り値に $152/MM$ を乗ずるものとする。
- ソフトウェア規模としてKLOCだけでなくファンクションポイント(FP)も用いることができるものとする。COCOMO IIでは、Early Designモデルにおいて、未調整ファンクションポイント(UFP)をKLOCに換算して用いてもよいとされている。この考え方をPost-Architectureモデルに拡張することとし、ソフトウェア規模換算係数 $S_{conv}$ をモデルパラメータとして追加する。これにより、ソフトウェア規模の実績値としてFPのみが記録されているプロジェクトにもモデル適用が可能になる。

以上の結果、以降で用いる工数見積りモデルは次のとおりとなる。

$$PM = 152/MM \times 2.94 \times Size^E \quad (式2)$$

$$\times (EM_{RELY} \times EM_{PVOL} \times EM_{ACAP} \times EM_{PCAP} \times EM_{SCED})$$

$$Size = S_{conv} \times S$$

$$E = 0.91 + 0.01 \times (SF_{PREC} + 3.04 + SF_{RESL} + 3.29 + SF_{PMAT})$$

$PM$ : 工数見積り値(人月)

$SF_{xxxx}$ : 規模要因値

$EM_{yyyy}$ : コスト要因値

$S$ : ソフトウェア規模(KLOC、もしくは、FP)

$S_{conv}$ : ソフトウェア規模換算係数。「1FPあたりのKLOC」、ただし、 $S$ がKLOCの場合は1。

$MM$ : 1か月の所定労働時間(時間)

3.04: 規模要因 $SF_{FLEX}$ の等級 $N$ (中位)での値

3.29: 規模要因 $SF_{TEAM}$ の等級 $N$ (中位)での値

モデルパラメータとプロジェクト特性の対応関係を表-1にまとめる。プロジェクト特性の詳細については、文献[ERR]を参照されたい。表-1にはプロジェクト特性がどのモデルパラメータに対応付けられるか

表1 モデルパラメータとプロジェクト特性の対応関係

| 規模要因<br>(スケールファクタ) | 等級                | 非常に低い<br>VL                           | 低い<br>L  | 中位<br>N                                      | 高い<br>H                                 | 非常に高い<br>VH                               | 極めて高い<br>XH |
|--------------------|-------------------|---------------------------------------|--|--|---|---|-------------|
| モデルパラメータ           | $SF_{PREC}$       | 6.20                                  | 4.96   | 3.72   | 2.48                                    | 1.24                                      | 0.00        |
| プロジェクト特性           | 先行モデルの流用と標準モデルの採用 | 1 先行モデルは存在しなかった、または、適合する標準モデルは存在しなかった | 2 先行モデルは存在したがほとんど流用出来なかった、または、適合する標準モデルは存在したが採用しなかった | 3 既定 先行モデルは部分的に流用出来た、または、適合する標準モデルは部分的に採用出来た | 4 先行モデルはかなり流用出来た、または、適合する標準モデルはかなり採用出来た | 5 先行モデルは全面的に流用出来た、または、適合する標準モデルは全面的に採用出来た | -           |
| モデルパラメータ           | $SF_{RESL}$       | 7.07                                  | 5.65   | 4.24   | 2.83                                    | 1.41                                      | 0.00        |
| プロジェクト特性           | プロジェクト管理者の経験と能力   | 1 経験無かった                              | 2 少数の小中規模プロジェクトの管理を経験していた                            | 3 既定 多数の小中規模プロジェクトの管理を経験していた                 | 4 少数の中大規模プロジェクトの管理を経験していた               | 5 多数の中大規模プロジェクトの管理を経験していた                 | -           |
| モデルパラメータ           | $SF_{PMAT}$       | 7.80                                  | 6.24   | 4.68   | 3.12                                    | 1.56                                      | 0.00        |
| プロジェクト特性           | CMM達成レベル          | -                                     | 1 レベル1   | 2 レベル2                                       | 3 レベル3 既定                               | 4 レベル4                                    | 5 レベル5      |

| コスト要因<br>(コストドライバ) | 等級                 | 非常に低い<br>VL                | 低い<br>L                         | 中位<br>N                         | 高い<br>H                        | 非常に高い<br>VH                  | 極めて高い<br>XH |
|--------------------|--------------------|----------------------------|---------------------------------|---------------------------------|--------------------------------|------------------------------|-------------|
| モデルパラメータ           | $EM_{RELY}$        | 0.82                       | 0.92                            | 1.00                            | 1.10                           | 1.26                         | -           |
| プロジェクト特性           | 信頼性                | 5 ソフトウェア障害は軽微な損失が想定された     | 4 ソフトウェア障害は簡単に復旧可能な小規模な損失が想定された | 3 既定 ソフトウェア障害は復旧可能な中規模の損失が想定された | 2 ソフトウェア障害は財政上の大規模な損失が想定された    | 1 ソフトウェア障害は人命にかかわる損失が想定された   | -           |
| モデルパラメータ           | $EM_{PVOL}$        | -                          | 0.87                            | 1.00                            | 1.15                           | 1.30                         | -           |
| プロジェクト特性           | プラットフォーム<br>(の適合性) | -                          | 5, 4 非常に、あるいは、かなり適合性があった        | 3, 2 既定 適合性があった                 | 1 全体的に、あるいは、部分的に適合性に欠けた        | -                            | -           |
| モデルパラメータ           | $EM_{ACAP}$        | 1.42                       | 1.19                            | 1.00                            | 0.85                           | 0.71                         | -           |
| プロジェクト特性           | アナリストの経験と能力        | 1 経験無かった                   | 2 少数の小中規模プロジェクトのアナリストを経験していた    | 3 既定 多数の小中規模プロジェクトのアナリストを経験していた | 4 少数の中大規模プロジェクトのアナリストを経験していた   | 5 多数の中大規模プロジェクトのアナリストを経験していた | -           |
| モデルパラメータ           | $EM_{PCAP}$        | 1.34                       | 1.15                            | 1.00                            | 0.88                           | 0.76                         | -           |
| プロジェクト特性           | SE・プログラマの経験と能力     | 1 要員の過半数は経験無かった            | 2 要員の半数はある程度経験していた              | 3 既定 要員の過半数はある程度経験していた          | 4 要員の半数は十分な経験、残り半数はある程度経験していた  | 5 要員の過半数は十分な経験をしていた          | -           |
| モデルパラメータ           | $EM_{SCED}$        | 1.43                       | 1.14                            | 1.00                            | 1.00                           | 1.00                         | -           |
| プロジェクト特性           | 開発スケジュール要求         | 1 スケジュールは当初の想定より25%以上短縮された | 2 スケジュールは当初の想定より10%~25%前後短縮された  | 3 既定 スケジュールは当初の想定どおりであった        | 4 スケジュールは当初の想定より10%~50%前後許容された | 5 スケジュールは当初の想定より50%以上許容された   | -           |

と共に、その値によって、どの等級とみなされるかを示した。表中で、「既定」とされているのは、当該プロジェクト特性値が記録されていない(欠損している)場合に用いる既定値である。プロジェクト388件における各プロジェクト特性の最頻値を既定値としている。参考までに、各プロジェクト特性の度数分布を図-2に示す。結果として、 $SF_{PMAT}$ を除くパラメータでは、既定値に対応する等級はすべてN(中位)となる。また、1か月の所定労働時間MMが記録されていない場合には、適用対象プロジェクト388件における中央値155

(時間)を既定値として用いる。

ソフトウェア規模換算係数 $S_{conv}$ には、モデル適用対象プロジェクト388件のうち、KLOC実績値とFP実績値が共に記録されているプロジェクト123件における「1FPあたりのKLOC」の中央値0.076を用いる。開発言語の違いによる差は考慮していない。参考までに、モデル適用対象プロジェクト、COCOMO II [CII]、および、関連資料 [SPR] [QSM] における「1FPあたりのKLOC」を表-2にまとめる。

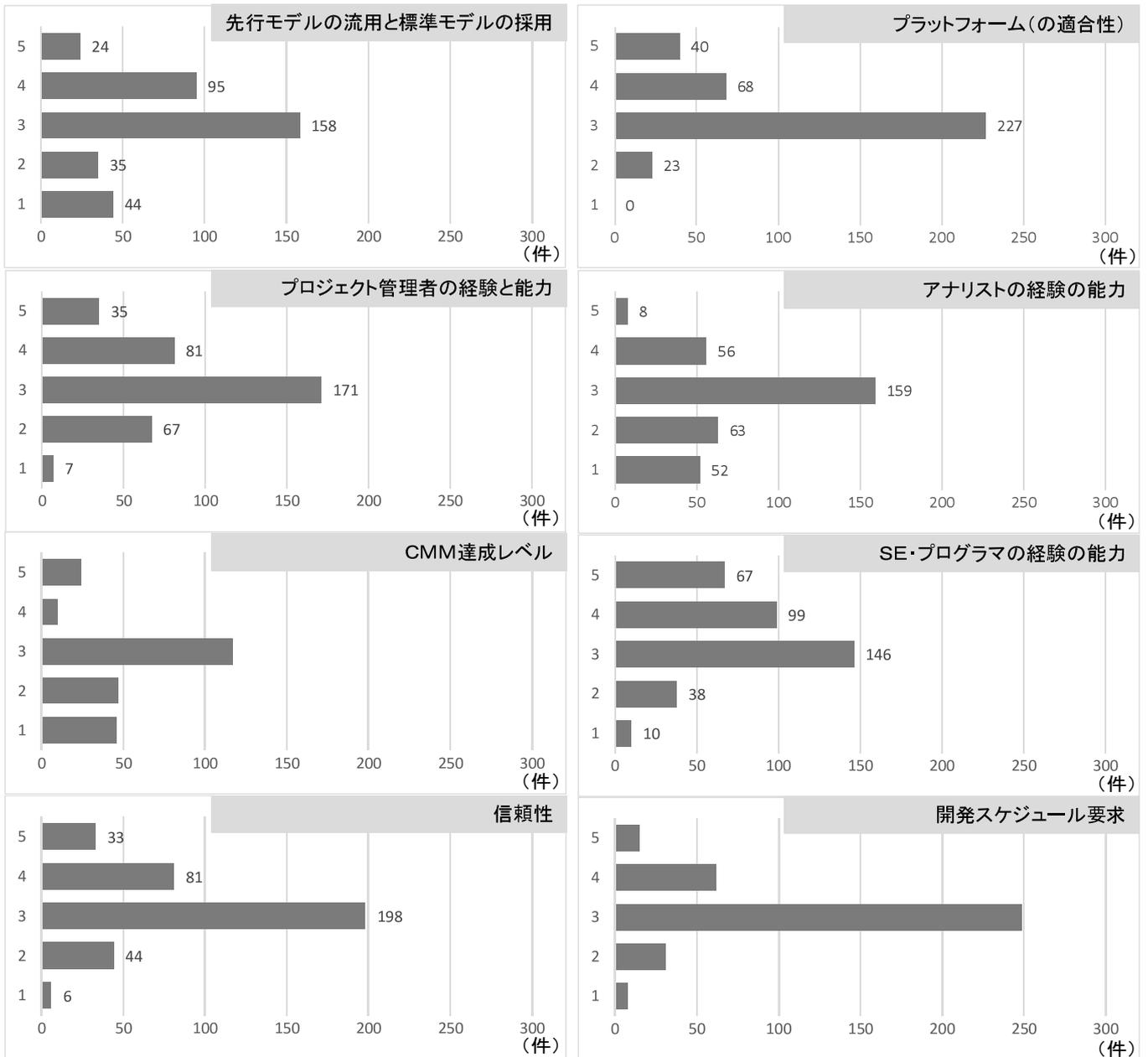


図2 プロジェクト特性値の度数分布

表2 1FPあたりのKLOC (規模換算係数)

|                        | Java  | COBOL | VB.NET |
|------------------------|-------|-------|--------|
| 本稿における<br>モデル適用対象      | 0.068 | 0,088 | 0.079  |
| COCOMOII 2000          | 0,053 | 0,091 | —      |
| SPR社資料 <sup>1)</sup>   | 0.046 | 0.107 | 0.025  |
| QSM社公開資料 <sup>2)</sup> | 0.053 | 0.055 | 0.060  |

1) Software Productivity Research, SPR Programming Languages Table, Ver. PLT2007c, Dec. 2007.

2) <http://www.qsm.com/resources/function-point-languages-table>

### 3-3 外れプロジェクト除去

対象プロジェクト388件のうち、生産性の実績値(ソフトウェア規模の実績値/工数の実績値)が極端に大きなプロジェクト、および、小さなプロジェクトは、特異なプロジェクト(外れプロジェクト)として除去した[Monden]。具体的には、KLOCが記録されているプロジェクトについて、KLOCに基づく生産性(KLOC/工数)の対数値の平均( $\bar{x}$ )と標準偏差( $\sigma$ )を求め、生産性が $\bar{x} \pm 3\sigma$ の範囲から外れるプロジェクト5件のデータは、KLOCをソフトウェア規模とするモデル適用では用いないこととした。同様に、FPに基づく生産性(FP/工数)の対数値が $\bar{x} \pm 3\sigma$ の範囲から外れるプロジェクト3件のデータは、FPをソフトウェア規模とするモデル適用では用いないこととした。参考までに、生産性の対数値の分布を図-3に示す。

## 4 適用結果

3章で示した方法で対象データにモデルを適用した結果を図-4~8に示す。適用結果は、x軸を「(式2)による工数見積り」、y軸を「対象プロジェクトにおける工数実績値」とする散布図(両対数)と、見積りを説明変数、実績値を被説明変数とした場合の回帰式

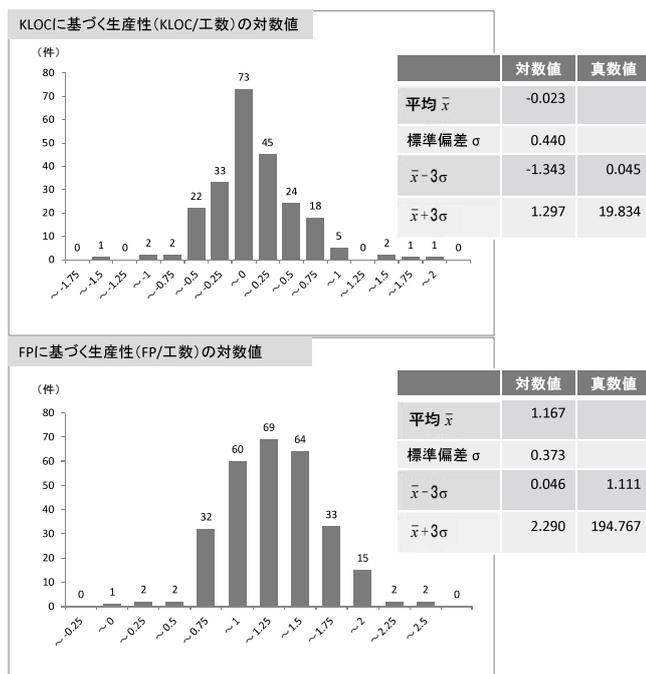


図3 生産性の分布と外れ値除去

$y = ax^b$ 、および、決定係数( $R^2$ )で構成されている。

図-4は、KLOC実績値が記録されているプロジェクト224件に対して、ソフトウェア規模単位をKLOCとしモデルを適用した結果である。表-1に示した「モデルパラメータとプロジェクト特性の対応関係」に基づく見積りの補正が有効であったかどうかを確認するため、次の4つの散布図で構成されている。

左上:規模要因とコスト要因の両方で補正

左下:規模要因でのみ補正

右上:コスト要因でのみ補正

右下:補正なし

これら4つの散布図を比較すると、データの散らばり具合や決定係数の値に大きな差はなく、規模要因とコスト要因、いずれによる補正も、見積り結果にはほとんど影響していないことが分かる。決定係数の値はいずれも0.67前後となっており、見積りと実績値の間に、ある程度の関連性があると考えられる。なお、回帰式の係数の値が $a \approx 1$ 、 $b < 1$ となっていることから、対象プロジェクトが小規模で工数も比較的小さい範囲では、工数の見積りと実績値に大きな差はないが、工数が大きくなるにつれ、実際に必要となる工数よりも大きく見積もってしまう傾向にあることが分かる。散布図をみると、(実績)工数が10人月程度であれば見積りも10人月程度であるが、1,000人月では、見積りはおよそその10倍となっている。

同様に、FP実績値が記録されているプロジェクト279件に対して、ソフトウェア規模単位をFPとしモデルを適用した結果を4つの散布図として図-5に示す。4つの散布図を比較すると、やはりこの場合も、規模要因とコスト要因いずれによる補正も、見積り結果にはほとんど影響していないことが分かる。決定係数の値は図-4で示した場合よりも大きく、この結果を見る限り、工数見積りは、KLOCよりもFPに基づいて行うべきと考えられる。なお、回帰式の係数の値が $a < 1$ 、 $b \approx 1$ であることから、プロジェクトの規模や工数が大きくなるにつれて工数をより大きく、あるいは、より小さく見積もってしまうことはないが、常に工数を大きく(2~3倍に)見積もってしまう傾向にあることになる。

開発言語が見積りに与える影響を確かめるため、主

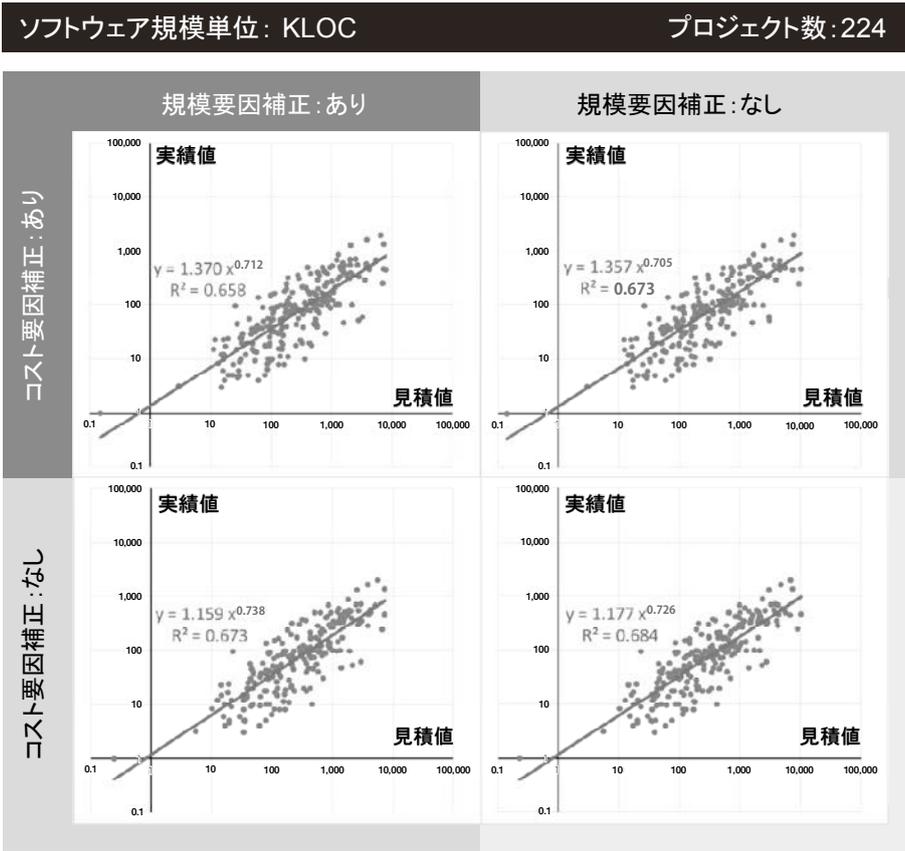


図4 モデル適用結果 (ソフトウェア規模単位: KLOC)

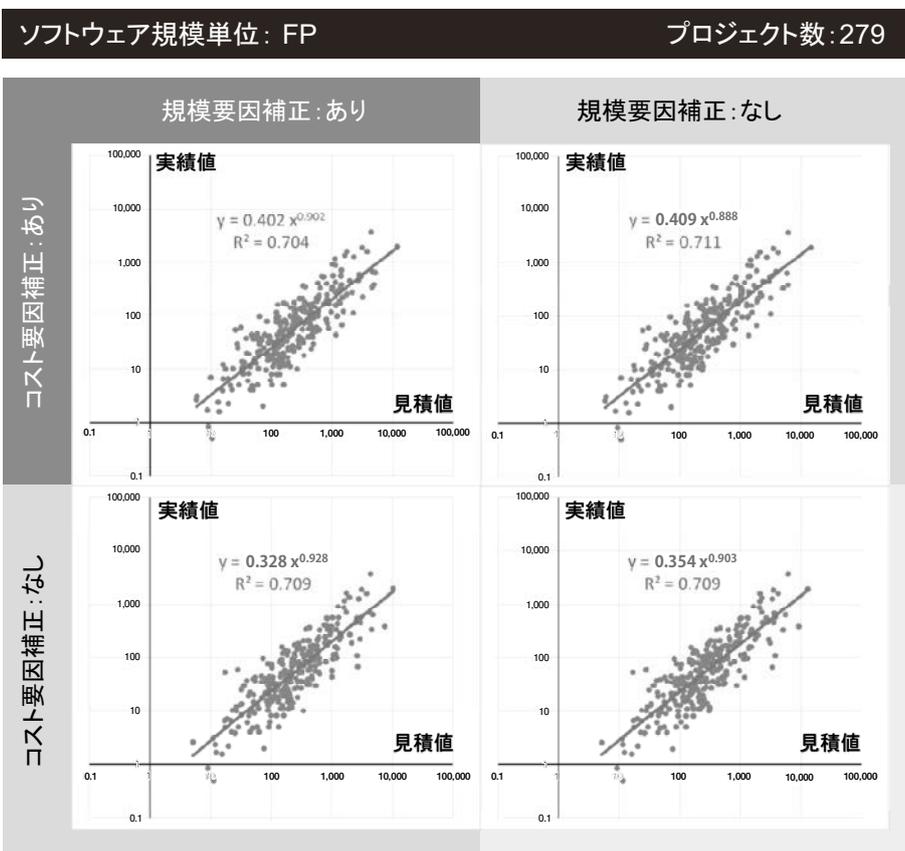


図5 モデル適用結果 (ソフトウェア規模単位: FP)

ソフトウェア規模単位: FP 主開発言語: Java プロジェクト数: 78

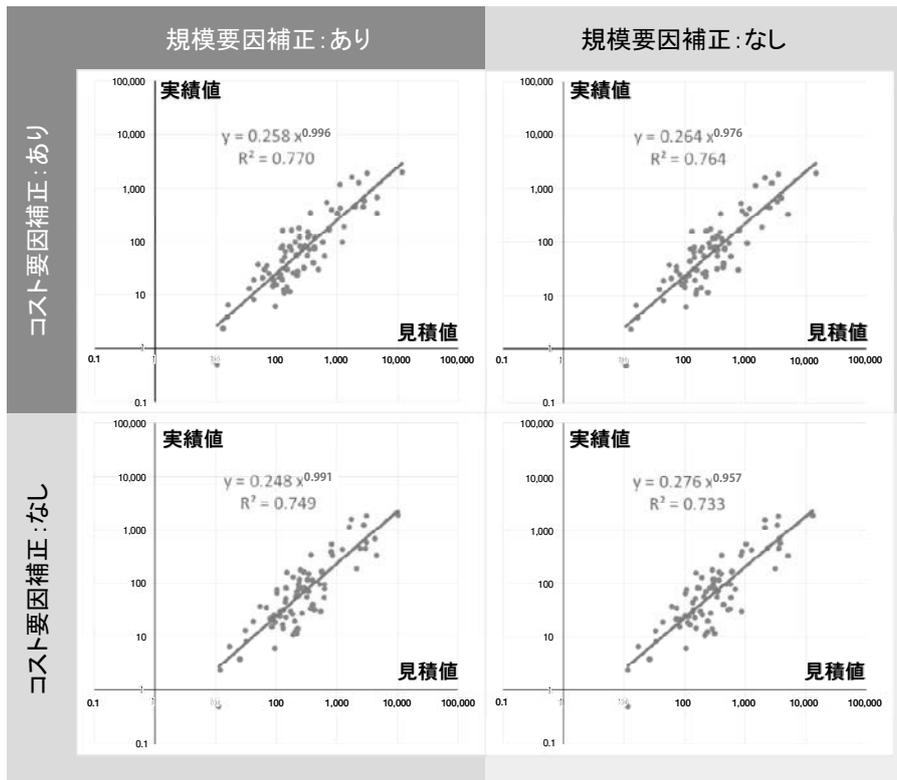


図6 モデル適用結果 (ソフトウェア規模単位: FP、主開発言語Java)

ソフトウェア規模単位: FP 主開発言語: VB.NET プロジェクト数: 23

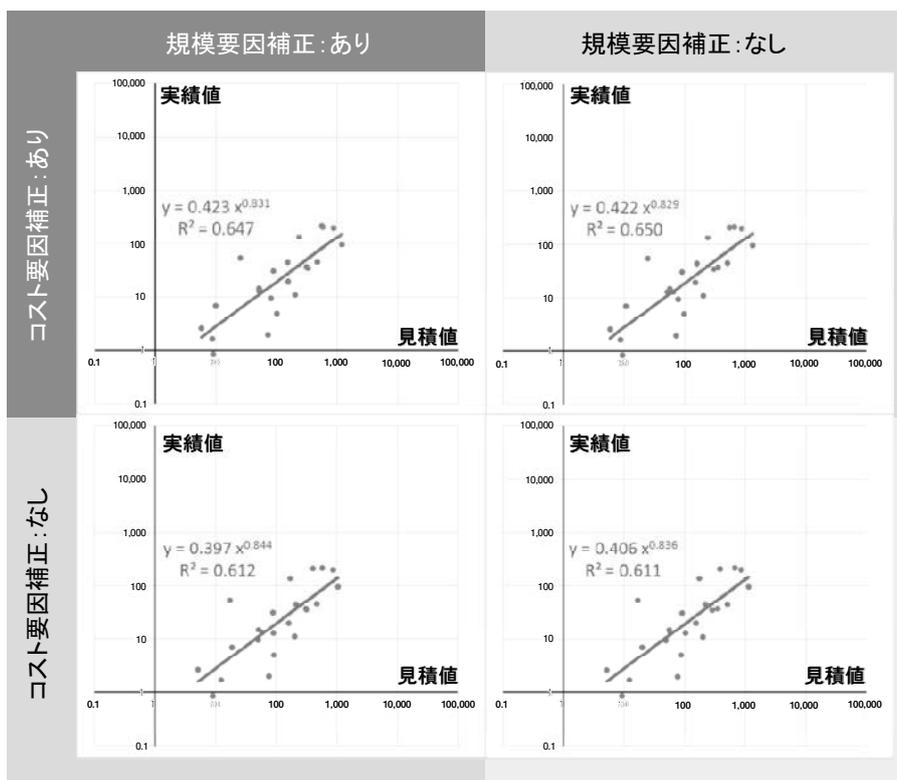


図7 モデル適用結果 (ソフトウェア規模単位: FP、主開発言語VB.NET)

ソフトウェア規模単位: FP 主開発言語: COBOL プロジェクト数: 17

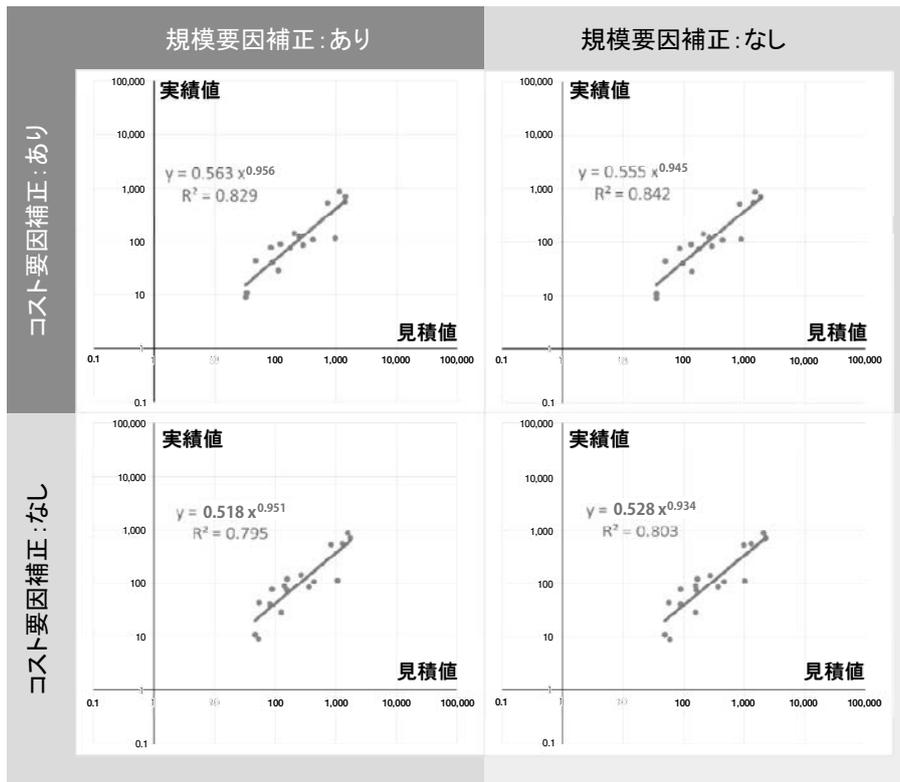


図8 モデル適用結果(ソフトウェア規模単位:FP、主開発言語COBOL)

開発言語(開発における利用率が、機能量でみて50%以上となっているプログラミング言語)で対象プロジェクトを層別し、モデルを適用してみた。ここでは、対象プロジェクトの多くで利用されていたJava、VB.NET、および、COBOLによる層別の結果を図-6、7、8にそれぞれ示す。なお、図-4と図-5の結果を受け、ソフトウェア規模単位はいずれもFPとした。図-6は、Javaを主開発言語とするプロジェクト78件への適用結果である。図-5で示した場合と比べ、決定係数の値は更に大きくなり、回帰式の係数 $b$ の値も、ほぼ1となっている。Javaを主開発言語とするプロジェクトのみを対象とすることで、プロジェクトの規模に影響されずに見積り可能であることになる、ただし、回帰式の係数 $a$ の値が、図-5で示した場合よりも更に小さくなっていることから、常に工数をより大きく(4倍程度)見積もってしまう傾向にあることも分かる

図-7は、VB.NETを主開発言語とするプロジェクト23件への適用結果である。決定係数は、ソフトウェア規模をKLOCとした場合と同程度まで小さくなっている。図-8は、COBOLを主開発言語とするプロジェ

クト17件への適用結果である。対象プロジェクトが少ないが、決定係数の値はいずれも0.8弱かそれ以上であり、これまで示した適用結果の中では最も大きい。

## 5 考察

4章で示したとおり、モデル式による工数見積値を説明変数( $x$ )、開発プロジェクトにおける工数実績値を被説明変数( $y$ )とした場合、決定係数 $R^2$ の値が0.7前後となり、両者の間には、ある程度の関係性が認められた。また、ソフトウェア規模の単位をFPとした場合には、回帰式 $y=ax^b$ において $b \approx 1$ となり、プロジェクトの規模や工数の大小に見積り結果が影響されるにくいことも分かった。

こうした適用結果は予想を上回るものであり、少なくとも現時点では、COCOMO IIそのものやその利活用を否定することはできないと思われる。ただし、「関係性がある」ということと、「実用上十分な見積り精度がある」ということは別である。モデルの利活用に向けた次の段階では、見積り精度とその向上の議論が

必要となってくる。参考までに、現状での見積り精度を確認してみた。対象となるプロジェクト数が多く、決定係数の値が比較的高かった「ソフトウェア規模単位：FP、主開発言語：Java」という条件の場合(図-6参照)、対象プロジェクト78件における工数の見積り精度(見積値/実績値)の平均値は5.21、標準偏差は4.08である。つまり、(式2)による見積りでは、工数を平均で5.21倍も大きく見積もってしまうことになる。この点が将来解消されると仮定し、各プロジェクトの見積り精度をその平均値で割ったとしても、標準偏差は0.78である。度数分布を図-9に示すが、見積り値の範囲は、実績値の1/3～3倍程度であることがわかる。

なお、COCOMO構築時にも参考にされたデータでは、ソフトウェア規模やその開発工数の見積り精度は、ソフトウェア開発プロジェクトのごく初期においては0.25～4倍、基本設計開始時点でも0.67～1.5倍程度であるとされている[Boehm]。また、モデルによる見積りだけに頼ることに限界があるとして、「専門家の経験や合議に基づく推定」、「類似プロジェクトの実績データに基づく類推」、「WBS(Work Breakdown Structure)等に基づく積算」などを併用するアプローチも報告されている[IPA2]。

今後の議論においては、規模要因とコスト要因による補正が、見積り結果にほとんど影響を与えなかったという結果にも注目すべきである。もちろん、(式2)で用いた規模要因やコスト要因は、COCOMO IIが定義する計22個のうち、およそ1/3にあたる8個のみである。見積りに用いた8個の要因についても、プロジェクト特性との対応付けに見直すべき点があるかもしれない。

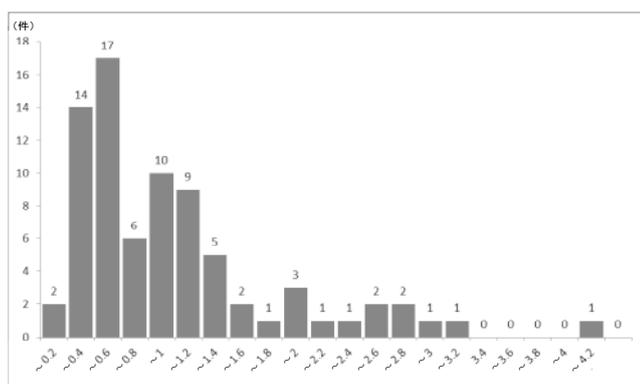


図9 見積り精度の度数分布(ソフトウェア規模単位:FP、主開発言語Java)

い。また、規模要因やコスト要因以外でも、ソースコードの自動生成・変換や再利用・流用に関するパラメータは、必要なデータが得られなかったため(式2)には含まれていない。COCOMO IIに本来備わっている補正機能が十分に発揮された結果とすることはできない。

COCOMO IIを検証する、という立場からすれば、まずは、モデルパラメータ算出に必要なデータを揃えることに注力すべきなのかもしれない。その一方で、モデル化の基本的な考え方は踏襲するとしても、ソフトウェア開発の現状やデータ収集コストなどを考慮して、規模要因やコスト要因を大幅に見直すというアプローチもある。更には、規模要因とコスト要因による補正が、見積り結果にほとんど影響を与えなかったという今回の結果を重視し、それら要因による見積り補正に固執するよりも、対象プロジェクトの層別など、モデル適用方法に工夫を加えるべきという考え方もある。例えば、今回の適用においても、主開発言語で見積り対象プロジェクトを層別した3つの場合(図-6、7、8)では、いずれにおいても、規模要因とコスト要因による補正により、決定係数の値がわずかではあるが高くなっている。この結果をどのように解釈すべきかの議論が、見積り値の補正や対象プロジェクトの層別に関する議論の端緒となるかもしれない。

## 6 まとめ

本稿では、提案されて10年余りが経過している米国生まれの工数見積りモデルCOCOMO IIが、今日の日本のソフトウェア開発プロジェクトにも適用可能かどうかを、実際のソフトウェア開発プロジェクトのデータを用いて検証した。経済調査会リポジトリが提供するソフトウェア開発プロジェクト388件のデータにCOCOMO IIを適用した結果、見積り工数と実績工数の間には関係性があり、ソフトウェア規模の単位としてKLOCではなくFPを用いることで、より強い関係性が見られることなどが分かった。今後、見積り精度やモデル適用方法の改善に向けた更なる議論や適用実験の実施が待たれるが、そうした取り組みに一定の意義のあることが今回確認された。

【参考文献】

[Boehm] B.W. Boehm: Software Engineering Economics, Prentice-Hall, 1981.

[IPA] 情報処理推進機構：“2012年度「ソフトウェア産業の実態把握に関する調査」調査報告書”，2013年.

[IPA2] 情報処理推進機構 ソフトウェア・エンジニアリング・センター編：“ソフトウェア開発見積りガイドブック：ITユーザとベンダにおける定量的見積りの実現”，オーム社，2006年.

[ERR] 経済調査会，ソフトウェア開発データリポジトリの分析，2010年。  
 ※経済調査会が，毎年実施している「ソフトウェア開発に関する調査」で収集したプロジェクトデータのうち，平成13年度～18年度までの6年分について分析結果をとりまとめたもの。PDF版は経済調査会のWebサイト <http://www.zaikeicho.or.jp/>の「研究成果」で確認できる。

[Kikuchi] 菊地 奈穂美，飯泉 純子，亀田 康雄，細川 宣啓，渡辺 千恵子，大槻 繁，“見積り法COCOMO II概説”，SEC journal, No.12, pp.34-43, 2008年。  
<http://www.ipa.go.jp/files/000024479.pdf>

[Monden] 門田 暁人，松本 健一，大岩 佐和子，押野 智樹：  
 “生産性に基づくソフトウェア開発工数予測モデル”，経済調査研究レビュー，Vol.11, pp.32-37, 2012年9月.

[Park] R, E, Park: “Software Size Measurement: A Framework for Counting Source Statements,” CMU/SEI-92-TR-020, 1992.  
<http://www.sei.cmu.edu/reports/92tr020.pdf>

[QSM] Quantitative Software Management: “Function Point Languages Table.”  
<http://www.qsm.com/resources/function-point-languages-table>

[SPR] Software Productivity Research: “SPR Programming Languages Table,” Ver.PLT2007c, Dec. 2007.

[CII] University of Southern California, COCOMO II Model Definition Manual, Ver.2.1, 2000.  
[http://csse.usc.edu/csse/research/COCOMOII/cocomo2000.0/CII\\_modelman2000.0.pdf](http://csse.usc.edu/csse/research/COCOMOII/cocomo2000.0/CII_modelman2000.0.pdf)

[Walston] C. Walston and C. Felix: “A method of programming measurement and estimation,” IBM Journal, 16, 1, pp.54-73, 1977.

付録COCOMO II 規模要因・コスト要因一覧

|                    |                             | 非常に低い<br>VL               | 低い<br>L | 中位<br>N | 高い<br>H | 非常に高い<br>VH | 極めて高い<br>XH |      |
|--------------------|-----------------------------|---------------------------|---------|---------|---------|-------------|-------------|------|
| 規模要因<br>(スケールファクタ) | プロダクトの先例性 (PREC)            | 6.20                      | 4.96    | 3.72    | 2.48    | 1.24        | 0.00        |      |
|                    | 開発の柔軟性 (FLEX)               | 5.07                      | 4.05    | 3.04    | 2.03    | 1.01        | 0.00        |      |
|                    | アーキテクチャ/リスクの早期解決の必要性 (RESL) | 7.07                      | 5.65    | 4.24    | 2.83    | 1.41        | 0.00        |      |
|                    | チーム凝集度 (TEAM)               | 5.48                      | 4.38    | 3.29    | 2.19    | 1.10        | 0.00        |      |
|                    | プロセス成熟度 (PMAT)              | 7.80                      | 6.24    | 4.68    | 3.12    | 1.56        | 0.00        |      |
| コスト要因<br>(コストドライバ) | プロダクト要因                     | 信頼性の要求 (RELY)             | 0.82    | 0.92    | 1.00    | 1.10        | 1.26        |      |
|                    |                             | データベースの規模 (DATA)          |         | 0.90    | 1.00    | 1.14        | 1.28        |      |
|                    |                             | プロダクトの複雑性 (CPLX)          | 0.73    | 0.87    | 1.00    | 1.17        | 1.34        | 1.74 |
|                    |                             | 再利用性の要求 (RUSE)            |         | 0.95    | 1.00    | 1.07        | 1.15        | 1.24 |
|                    |                             | 文書化の要求 (DOCU)             | 0.81    | 0.91    | 1.00    | 1.11        | 1.23        |      |
|                    | プラットフォーム要因                  | 実行時間の制約 (TIME)            |         |         | 1.00    | 1.11        | 1.29        | 1.63 |
|                    |                             | 主記憶容量の制約 (STOR)           |         |         | 1.00    | 1.05        | 1.17        | 1.46 |
|                    |                             | プラットフォームのバージョン変更頻度 (PVOL) |         | 0.87    | 1.00    | 1.15        | 1.30        |      |
|                    | 要員の要因                       | 分析者の能力 (ACAP)             | 1.42    | 1.19    | 1.00    | 0.85        | 0.71        |      |
|                    |                             | プログラマの能力 (PCAP)           | 1.34    | 1.15    | 1.00    | 0.88        | 0.76        |      |
|                    |                             | 要員の継続性 (PCON)             | 1.29    | 1.12    | 1.00    | 0.90        | 0.81        |      |
|                    |                             | アプリケーションの経験 (APEX)        | 1.22    | 1.10    | 1.00    | 0.88        | 0.81        |      |
|                    |                             | プラットフォームの経験 (PLEX)        | 1.19    | 1.09    | 1.00    | 0.91        | 0.85        |      |
|                    |                             | 言語およびツールの経験 (LTEX)        | 1.20    | 1.09    | 1.00    | 0.91        | 0.84        |      |
|                    | プロジェクト要因                    | ソフトウェアツールの使用 (TOOL)       | 1.17    | 1.09    | 1.00    | 0.90        | 0.78        |      |
| 複数拠点開発 (SITE)      |                             | 1.22                      | 1.09    | 1.00    | 0.93    | 0.86        | 0.80        |      |
| 開発期間の要求 (SCED)     |                             | 1.43                      | 1.14    | 1.00    | 1.00    | 1.00        |             |      |