

## ソフトウェア開発工数予測のための フィットデータ選定方法

戸田 航史<sup>†1</sup> 門田 暁人<sup>†1</sup> 松本 健一<sup>†1</sup>

本論文では過去のソフトウェア開発プロジェクトの実績データから性能の良い開発工数予測モデル（重回帰モデル）を構築することを目的として、予測対象プロジェクトの特性に基づいて、モデル構築のためのフィットデータを自動的に選定する、もしくは、選定せずに全データを用いることを自動的に決定する方法を提案する。ISBSG データセットを用いた評価実験の結果、フィットデータの選択をいっさい行わず、すべての過去プロジェクトを用いる従来の重回帰モデルと比較すると、分母を実測値とする相対誤差 MRE の中央値では 0.452 から 0.367 へ、分母を予測値とする相対誤差 MER の中央値では 0.357 から 0.336 へと精度が向上した。本論文の結果から、予測対象の特性に合わせてフィットデータを選定する、もしくは選定しないことを決定することが重要であり、その系統的な手法として、提案方法が役立つことが示された。

### A Method for Selecting Fit Data in Software Effort Estimation

KOJI TODA,<sup>†1</sup> AKITO MONDEN<sup>†1</sup>  
and KEN-ICHI MATSUMOTO<sup>†1</sup>

To construct a better multivariate regression model for software effort estimation, this paper proposes a method to automatically select (or not to select) projects as a fit data from a given project data set based on estimation target's features. As a result of an experimental evaluation using the ISBSG data set, the proposed method showed better estimation performance than the conventional method (of constructing a regression model using all project data). The median of MRE (Magnitude of Relative Error) was improved from 0.452 to 0.367, and the median of MER (Magnitude of Error Relative) was improved from 0.357 to 0.336. This paper showed the necessity of fit data selection, and showed that the proposed method was one of the effective and systematic meant to do the selection.

### 1. はじめに

ソフトウェア開発プロジェクトの計画立案、開発要員の割当てなどにおいて、定量的工数予測手法が広く利用されている<sup>1),2)</sup>。定量的工数予測手法では、過去のプロジェクトの実績データを見積りの根拠として用いる<sup>3)</sup>。この定量的工数予測手法はモデルベース推論とメモリベース推論に大別できる。重回帰分析<sup>4)</sup>をはじめとするモデルベース推論では、プロジェクトの特性変数（開発規模、欠陥数など）を説明変数として用い、目的変数である開発工数との関係を表すモデル式を導出し、予測対象のプロジェクトに用いる。これに対し、Analogy-based Estimation<sup>1),5)</sup>をはじめとするメモリベース推論は、予測対象プロジェクトに類似した過去のプロジェクト群を選定し、それらの実績工数を用いて（加重平均などにより）予測値を得る手法である。予測モデルの性能は、モデル式の導出に用いる過去のプロジェクトデータセット、すなわちフィットデータセットの内容に大きく依存する。

本論文では、モデルベース推論の代表的な手法である重回帰モデルにおける問題を取り扱う。性能の良いモデルを得るためには、予測対象のプロジェクトと開発環境や開発プロセスの類似するプロジェクトのみを選定し、フィットデータとして用いることが重要となる。たとえば、メインフレーム系の開発プロジェクトと、組み込みソフトウェアの開発プロジェクトを混在させるべきではない。メインフレーム系の予測を行いたい場合は、フィットデータにはメインフレーム系のみを含めることが望ましい。

ところが、ソフトウェア開発は本質的に個別性が高いため、開発プロセスや業種、アーキテクチャ、開発言語などのすべてが一致するプロジェクトのみをフィットデータとして選定すると、プロジェクト件数が極端に少なくなり、モデルの予測性能がかえって低下する可能性がある。このような場合には、性質の異なるプロジェクトもフィットデータに含めざるをえない。ところが、どこまでを含めるべきであるかの一般的な基準はなく、従来ほとんど研究されていない。そのため、フィットデータの選定はモデル構築者の経験に任されてきたのが現状である。

本論文では、重回帰モデルの構築において、予測対象のプロジェクトの特性に応じてフィットデータを自動的に選定する方法を提案する。提案方法では、複数の特性を持つ予測対象プロジェクトに対し、特性が1つ以上一致する過去プロジェクト群を選び出し、それぞれを

<sup>†1</sup> 奈良先端科学技術大学院大学情報科学研究科

Graduate School of Information Science, Nara Institute of Science and Technology

フィットデータの候補とする。たとえば、予測対象のプロジェクトが業種 = 銀行、言語 = C、アーキテクチャ = 2 層クライアントサーバであるならば、業種 = 銀行の過去プロジェクトだけを集めたフィットデータセット候補を作成し、同様に、言語 = C のもの、アーキテクチャ = 2 層クライアントサーバのものだけを集めたものを作成する。さらに業種 = 銀行かつ言語 = C の過去プロジェクトだけを集めたフィットデータセット候補を作成し、同様に言語 = C かつアーキテクチャ = 2 層クライアントサーバのもの、業種 = 銀行かつアーキテクチャ = 2 層クライアントサーバのもの、そして 3 つの特性すべてが一致するものについても作成する。ただし、プロジェクト件数が非常に少なく、モデルを構築しても信頼性が低いと考えられるフィットデータ候補はこの時点で破棄する。次にこれらのフィットデータ候補群を用いて、それぞれ予測モデルを構築する。そして、予測対象のプロジェクトに対して最良の結果を与えると思われるフィットデータ候補およびモデルを、モデル構築時の適合度に基づいて決定する。ただし、予測対象プロジェクトに含まれる特性すべてが過去のプロジェクトにほとんど含まれていない場合には、プロジェクト特性を考慮しないモデル（すべての過去プロジェクトのデータを用いたモデル）の方が適合度の面で優れていることがある。その場合には、全プロジェクトをフィットデータとして採用する。

提案手法の有用性を評価するために、一般に公開されている最も大規模なプロジェクトデータセットである International Software Benchmarking Standard Group (ISBSG) が収集したプロジェクトデータ (ISBSG データ<sup>6)</sup>) を用いて実験を行った。ISBSG データは、1989 年から 2004 年の間に世界 20 カ国のソフトウェア開発企業から収集され、予測手法の有効性を評価する研究で広く用いられている<sup>7)–9)</sup>。

以降、2 章ではモデル構築におけるフィットデータ選定の問題を整理する。続いて 3 章で提案手法の詳細について説明する。4 章では提案手法の有効性を評価するための実験について説明し、5 章で実験結果とその考察を述べる。6 章でまとめと今後の課題を述べる。

## 2. 重回帰モデル構築におけるフィットデータ選定の問題

多くのソフトウェア開発企業では、多種多様なソフトウェアの開発を行っており、開発プロジェクトごとに工数の見積りを行うことが必要である。その 1 つの方法は、過去のプロジェクトで収集されたデータを用いて重回帰分析などにより工数予測モデルを構築し、予測を行うことである。

ただし、多種多様なソフトウェアの開発工数を 1 つのモデル式で表現するには限界がある。たとえば、航空電子工学関連システムと社内業務システムを比較すると、同じ 1 万 SLOC

(Source Lines Of Code) 規模のプロジェクトであっても開発のプロセスが大きく異なり、生産性に 8~18 倍の違いが生じる<sup>10)</sup>。この違いは、開発工数の差となって現れるが、この差を 1 つの重回帰モデルが吸収することは難しい。重回帰モデルでは、システム種別の違いは 0 または 1 をとる 2 値の説明変数として与えることができるが、モデル式の性質上、2 値の違いが工数に対して 8~18 倍の影響を与えるというような、乗法的な影響を表現することはできない。重回帰モデルでは、各説明変数は、目的変数に対して加法的に作用するためである<sup>4)</sup>。

また、一般に過去の開発プロジェクトデータは、工数に影響を与えるすべての変数を含んでいるとは限らず、このことも多種多様なプロジェクトを 1 つのモデル式で表現することを困難にしている。たとえば、「システムに要求される信頼性」という変数は、テスト工数に大きな影響を与えるが、変数として計測されていないこともある。一般に、システムに要求される信頼性は、システム種別と関わりが深いため、システム種別ごとに個別に工数予測モデルを構築した方が、全プロジェクトを用いたモデルよりも高い精度が期待される。

以上のことから、性能の良い重回帰モデルを構築するためには、予測対象のプロジェクトと特性が類似する、たとえばシステム種別や開発環境などが類似しているプロジェクトのみをフィットデータとして選定し、モデル構築に用いることが重要となる。しかし、ソフトウェア開発は本質的に個性性が高いため、システム種別、業種、アーキテクチャ、開発言語などのすべての特性が一致する類似プロジェクトのみをフィットデータとして選定すると、プロジェクト件数が極端に少なくなり、モデルの予測性能がかえって低下する事例が報告されている<sup>11)</sup>。このような場合には、性質の異なるプロジェクトもフィットデータに含めざるをえない。

ところが、どの特性に着目してフィットデータを選定すべきかの一般的な基準はなく、従来ほとんど研究されていない。そのため、フィットデータの選定はモデル構築者の経験に任されてきたのが現状である。たとえば本間<sup>12)</sup>は過去に行われた 1,050 件のプロジェクトのデータの分布状況を確認し、「新規開発・一括請負・Web 系開発」や「新規開発・一括請負・ビジュアル開発ツール」といったプロジェクト・タイプ約 30 種類に絞り込みを行ったうえで回帰分析を行い、数学モデルを抽出した事例を紹介している。ただし、この事例では、プロジェクト・タイプの分類を決定するにあたって、データの分布を目視してモデルの精度を高めていくと述べられており、専門家やその作業に専従する開発経験の豊富な分析者の関与が必要となる。また、本間の事例では、すべてのプロジェクトはいずれかのタイプに分類され、予測が行われることを暗に想定しているが、プロジェクトは多数の特性を持つことが

ら、一般に、プロジェクト・タイプは互いに独立しているとは限らない。たとえば、「新規開発・一括請負・Web系開発」と「新規開発・一括請負・規模＝大」は重複がある。このような分類の取扱いについては、従来決まった方法が提案されていない。

### 3. 提案するフィットデータ選定方法

本論文では、予測したいプロジェクトと一致する特性を1つ以上持ち、かつ予測精度が高くなることを期待できるフィットデータを過去のプロジェクトから自動的に選定する方法を提案する。多くの開発現場では、どのようにして最適な見積りモデルを得るかが課題となっているが、本論文は、最適な見積りモデルは最適なフィットデータから構築される、という立場から1つの解決策を与える。提案方法は、あらかじめ独立したプロジェクト・タイプ群を決めるのではなく、予測対象プロジェクトごとに、その特性に応じてフィットデータを選定する点が、本間<sup>12)</sup>の方法と異なる。また、提案方法は、つねにフィットデータを選定するのではなく、プロジェクトによっては、全データを用いることも可能としている。

図1を例に提案方法の手順を説明する。図1では、予測対象プロジェクトは言語＝COBOL、業種＝金融、開発形態＝再開発である。手順(ステップ1～4)は下記のとおりである。

提案手法では一致する特性の数を3つ以上とすることが可能であるが、説明の都合上、ここでは一致する特性の上限数が2つの場合を述べる。3つ以上の場合に適用可能な、より一般化された手順の詳細を付録Aに示す。

[ステップ1] 過去の全プロジェクトを集めたデータセット(以下、全過去データ)から、予測対象プロジェクトの特性ごとに値が一致するプロジェクト群を選定し、それぞれをフィットデータ候補とする。たとえば、開発言語という特性について、値がCOBOLであったプロジェクトを選定し、「言語＝COBOLデータ」とする。同様の作業を業種が金融(業種＝金融データ)のもの、開発形態が再開発(開発形態＝再開発データ)であるものについても行う。さらに複数の特性の組合せについてもこの作業を行う。具体的には、まず開発言語がCOBOLかつ業種が金融であるプロジェクトを選定し「言語＝COBOLかつ業種＝銀行データ」とする(図中COBOL金融データ)。同様に他の組合せについても同様の作業を行い、「業種＝銀行かつ開発形態＝再開発データ」および「開発形態＝再開発かつ言語＝COBOLデータ」を得る。ただし、選定されたデータセットに含まれるプロジェクトの件数が一定数(以降、最小フィットデータサイズと呼ぶ)に達しておらず、モデルを構築するのに十分でない、または構築しても信頼できないと考えられる場合には、そのデータセットは破棄する、図中「開発形態＝再

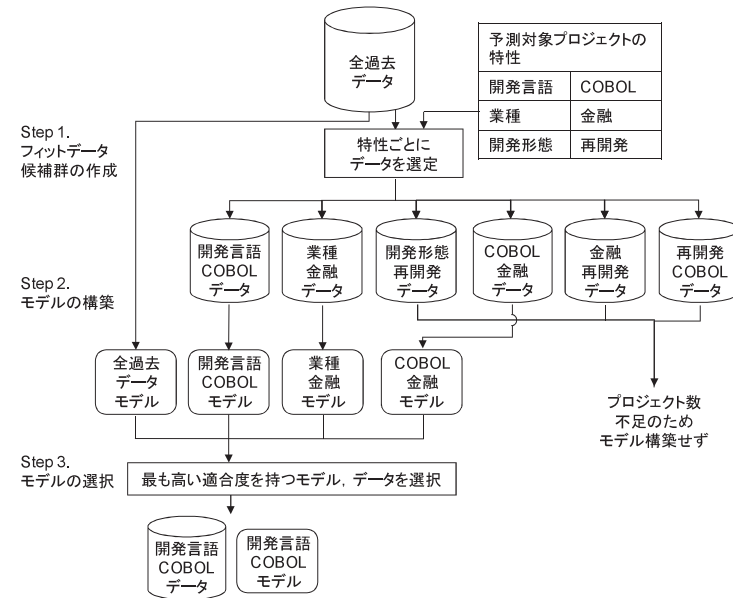


図1 フィットデータ選定方法  
Fig. 1 Procedure of fit data selection.

開発データ」およびこれを含む組合せがこれに該当する。得られたデータセット群に、全過去データを加え、フィットデータ候補群とする。全過去データを候補に加える理由は、特性ごとのモデルよりも全データを用いたモデルの方が高い精度が得られる場合があるためである。

[ステップ2] フィットデータ候補群それぞれについて重回帰モデルを構築し、各モデルのフィットデータ候補に対する適合度を表す指標(残差平方平均, 調整済み決定係数など)を記録しておく。

[ステップ3] 適合度を表す指標を基に、最も高い精度での予測が期待されるモデルおよびそのフィットデータを選定する。図1の例では、開発言語＝COBOLデータおよびモデルが選択されている。このデータおよびモデルを、最適なフィットデータおよびモデルと見なす。

[ステップ4] ステップ3で最適と見なされたモデルを予測対象プロジェクトに適用し、予

測値を得る。

以上のステップは、説明の簡単化のために予測対象プロジェクトのすべての特性が名義尺度変数として与えられる場合の手順を述べたものである。現実には、プロジェクト特性には開発期間、ファンクションポイント数 (FP 数) などの比尺度変数も含まれるため、より複雑な手順となる。具体的には、数値変数については、閾値を設けて全過去データからプロジェクトを選定する。たとえば、開発期間について 12 カ月を閾値とした場合、予測対象プロジェクトの開発期間が 10 カ月であったならば、開発期間 < 12 カ月を満たすプロジェクトを全過去データから選定し、フィットデータ候補とする。また、予測対象プロジェクトの開発期間が 15 カ月であったならば、開発期間  $\geq$  12 カ月を満たすプロジェクトを選定する。手順の詳細を付録 A に示す。本論文の評価実験 (4 章) では、比尺度変数の閾値には中央値を用いた。今回用いたデータセットの比尺度変数は分布に偏りがあるため、平均値よりも中央値がより適していると判断した。

#### 4. 評価実験

##### 4.1 概要

評価実験の目的は、現実のソフトウェア開発プロジェクトの定量的データを用いて提案手法の有効性を評価することである。実験では、ISBSG<sup>6)</sup> が収集したソフトウェア開発企業の実績データを用い、要件定義および設計が終了した時点を選定して総工数を予測した。また、実験時間の都合上、フィットデータの選定において用いる特性の組合せの数を 2 つまでに限定した。

提案手法のステップ 1 における最小フィットデータサイズを 30 に設定し、選定されたフィットデータ候補に含まれるプロジェクトが 30 件に満たない場合にはその候補を破棄するものとした。これは母集団からサンプルを抽出する場合、30 件が統計上の 1 つの目安となるためである<sup>13)</sup>。ただし、本実験の主目的は、フィットデータ選定方法がうまく働くかどうかを確認することであり、最小フィットデータサイズを 30 件とすることが最適であることを保障するものではないことに注意されたい。

ステップ 2 における各モデルのフィットデータ候補に対する適合度を表す指標としては、残差平均平方 (RMS; Residual Mean Square), 自由度調整済み決定係数 (Adj.R<sup>2</sup>; Adjusted R-square), 最小絶対偏差 (LAD; Least Absolute Deviation)<sup>14)</sup>, 赤池情報規準 (AIC; Akaike Information Criterion)<sup>15)</sup> などが知られている。しかしながら LAD, AIC については、データセットのサイズが値に影響を与えるため、フィットデータのサイズが変化する提案手法では、これらの指標を用いると、提案手法の性能を正しく評価できない可能性があ

る。このため、本実験では重回帰分析の過程で算出できる RMS, Adj.R<sup>2</sup> の 2 種類を用いた。

また、提案手法 (3 章) のステップ 2 におけるモデル構築には、従来法と同様にステップワイズ重回帰分析を用いた。

提案手法との比較対象として、フィットデータの選定を行わない、従来のステップワイズ重回帰分析、およびプロジェクト間の類似性を用いた代表的予測手法である Analogy-based Estimation (Analogy-based 法)<sup>1)</sup>, 開発種別によって適用するモデルを変更する開発種別 2 分法を用いた。Analogy-based 法における変数選択法としては、Correlation Threshold based Selection (CTS)<sup>16)</sup> を採用した。開発種別 2 分法は、フィットデータを新規開発プロジェクトと派生開発 (改造, 再利用, バージョンアップなど) プロジェクトに分割してフィットデータを作成し、新規開発プロジェクトを予測する際には新規開発プロジェクトだけのフィットデータから構築したモデルを、派生開発プロジェクトには派生開発プロジェクトだけのフィットデータから構築したモデルを適用する方法である。この方法は、プロジェクト特性に応じたモデルを人手により構築する一例として位置づけられる。

##### 4.2 ステップワイズ重回帰分析

重回帰分析では、目的変数と、それに対して影響すると考えられる複数の説明変数の関係を表す一次式を導出する。この一次式では、絶対誤差の二乗和が最小となる係数が与えられる。工数予測においては、工数が目的変数であり、それ以外の特性が説明変数の候補となる。

ステップワイズ重回帰分析は、ステップワイズ変数選択法により説明変数を決定し重回帰分析を行う手法である。ステップワイズ変数選択法は、目的変数に対して強く影響する変数を説明変数として選択する手法の 1 つである。変数選択は以下の手順で行われる。

- 1) 初期モデルを作成する。
- 2) 作成されたモデルに対して、各説明変数の係数が 0 でないかの検定を行い、指定した有意水準で棄却されない場合に変数を採択し、棄却される場合には変数を除去する。本論文では、多重共線性を回避するために、採択する変数の分散拡大要因が 10 以上の場合、またはその変数を採択することによって、他の変数の分散拡大要因が 10 以上となる場合、その変数は採択しないという操作を行う。
- 3) 2) ができなくなるまで繰り返す。

本論文では、ステップワイズ変数選択法として変数増減法を採用する。変数増減法では、変数をまったく含まないモデルを初期モデルとして変数選択を行う。これ以外にも、ステップワイズ変数選択法としては変数増加法、変数減少法、変数減増法がある。しかしながら、変数増加法は 1 度追加された変数が削除されることがないため、ある段階では有用な変数

であっても、その後、他の変数が取り込まれた段階では有用ではなくなる可能性があり、変数減少法でも同様に、いったん削除された変数が、他の変数が削除されているうちに有用になる可能性があるという問題がある。変数増減法と変数減増法のうち変数増減法を採用したのは、工数予測についての論文の多くで重回帰分析のための変数選択として用いられていること<sup>17)–19)</sup>、および、本論文で用いた、代表的な統計パッケージの1つである SPSS でも変数増減法が採用されているためである。

#### 4.3 Analogy-based Estimation

提案手法と同様、予測対象プロジェクトごとに過去データを選定し、予測に用いる方法として、Analogy-based 法<sup>1),5)</sup>がある。Analogy-based 法では、「プロジェクト特性(説明変数)が似ているプロジェクトはその工数(目的変数)も似ている」という仮定に基づき、過去プロジェクトと予測対象プロジェクトとの間の類似度を算出し、類似度の高いプロジェクトのみを用いて予測を行う。Analogy-based 法以外に、Case-based Reasoning 法<sup>20),21)</sup>、Collaborative Filtering based Estimation 法<sup>22),23)</sup>などが存在するが、類似度計算や予測値計算のアルゴリズムが異なるだけで基本的な考え方は同じである。

提案手法も Analogy-based 法も、「予測対象プロジェクトに合わせた過去データの選定を行う」という点では同種のものであると見なせることから、本実験では、提案手法と Analogy-based 法の性能比較も行う。提案手法と Analogy-based 法の大きな違いは、提案手法はモデルベース手法(予測モデル構築)のためのフィットデータ選定法であるのに対し、Analogy-based 法はメモリベース推論に分類され、モデルが構築されない点が異なる。Analogy-based 法は、近年有望視されている方法ではあるが、変数選択と変数の重み付け方法が確立されていない、類似度計算や予測値計算に多くのバリエーションがある、公開されているツールが限られているなど、容易に利用できないという課題がある。本論文では、Analogy-based 法に基づくツールとして最もよく用いられている ArchAngel<sup>24)</sup>で採用されている手法を用いた。また、変数選択法としては、CTS を用いた。ArchAngel の工数予測の手順は次のとおりである。ここでは予測値算出に用いる類似ケース数を  $k$  とおく。

- 1) 過去プロジェクトの各特性値を正規化し、特性ごとの値域の違いが予測に与える影響を除去する。
- 2) 過去プロジェクトの1つと予測対象プロジェクトの類似度を  $n$  次元空間( $n$  はプロジェクトに含まれる特性の数)におけるユークリッド距離として求める。この値が小さいほど2つのプロジェクトは類似していると見なされる。ユークリッド距離については文献 25) を参照されたい。

- 3) 2) をすべての過去プロジェクトについて行う。
- 4) ユークリッド距離が小さいものから順に  $k$  件の過去プロジェクトを選び、予測値計算アルゴリズムを用いて、予測対象プロジェクトの開発工数を算出する。本論文では予測値計算アルゴリズムとして距離の逆数を用いた加重平均を用いて計算を行う。予測値計算アルゴリズムの詳細については文献 20) を参照されたい。

#### 4.4 利用したデータ

実験には ISBSG が収集しているデータセットを用いた。工数の見積りに利用可能な実績データとしてはこれ以外にも、Kitchenham らが文献 26) で公開しているもの、Desharnais が文献 27) で公開しているものなどが存在するが、ISBSG のデータセットはこれら2つのデータセットに比べて含まれるプロジェクトの数が多く、また収集されている特性の種類も豊富であることから採用した。提案手法において最小フィットデータサイズや特性ごとにフィットデータの選択を行う都合上、提案手法の効果の有無を評価するためには十分な大きさのデータが必要であると考えられるためである。

ISBSG が収集したデータセットは、20カ国のソフトウェア開発企業の実績データによって構成されている。このデータセットには、1989年から2004年までの3,026件のプロジェクトについて、それぞれ99種類の特性値が記録されている。ただし、このデータセットには数多くの欠損値が含まれているため、全データを重回帰分析に用いることはできない。

本実験では、上記のデータセットから、文献 7), 9) にならって Data Quality Rating が C または D、すなわちデータの信頼性が低いと判定されたものを除外した。Data Quality Rating とは ISBSG に所属する専門家が、そのプロジェクトデータが信頼できるかどうかを4段階で判定したものであり、信頼できるならば A、まったく信頼できなければ D と記される。次に FP 数の測定手法が IFPUG ではないものを除外した。そのうえで欠損が多い、または致命的な欠損が存在する、たとえば計画工数などの工数に関わる特性値が欠けているようなプロジェクトや特性値を除外し、欠損を含まない232件のプロジェクト、8種類の特性を含むデータセットを作成し、実験に用いた。4.1節で述べたとおり、本実験では設計終了時を想定しているため、この8種類の特性はすべて設計終了時まで決定可能と考えられる変数のみを採用している。

これら8種類のプロジェクトの特性について、特性値名、尺度の種類、名義尺度に含まれるカテゴリ名の一部、それぞれの名義尺度に含まれる特性値の種類数を表1に示す。特性の詳細については文献 28) を参照されたい。8種類の特性変数には4種類の比尺度変数、3種類の名義尺度変数、1種類の順序尺度変数が混在している。このうち名義尺度変数およ

び順序尺度変数については、それぞれ0または1の値をとる複数の2値変数(ダミー変数)に変換したものをを用いた。この2値変数への変換手順を付録Bに示す。

実験では、データセットを無作為に半数ずつに分け、一方をフィットデータ、もう一方をテストデータとした。フィットデータを用いて、工数見積りモデルの構築および類似プロジェクトの選定を行い、テストデータを評価に用いた。この操作を独立に30回繰り返し、フィットデータとテストデータのペアを30組作成した。各データの組を用いて、目的変数である開発総工数を予測した。

#### 4.5 評価基準

実験では、予測誤差を評価する基準として、実測値を分母におく相対誤差 MRE (Magnitude of Relative Error)<sup>3)</sup>、および予測値を分母におく相対誤差 MER (Magnitude of Error Relative)<sup>29)</sup> の2つを用いる。MREが高い値ならば、予測が過大見積りであったことを示し、MERが高い値ならば、予測が過小見積りであったことを示す。それ以外の誤差指標としては絶対誤差もよく用いられているが、本論文では、MREとMERの両方を用いることで、過小見積り、過大見積りの双方を評価できることから、絶対誤差の評価は割愛する。

予測手法の評価基準として、MREとMER、それぞれの平均値(MMRE, MMER)、中央値(MdMRE, MdMER)、それらの値が0.25以下のプロジェクトが占める割合(Pred(25).MRE, Pred(25).MER)の3種6個を用いる。Pred(Percentage of prediction)とは、相対誤差が特定の値以下のプロジェクトが全体に占める割合を示すもので、Pred(25).MREは相対誤差MREが25%以下のプロジェクトが全体に占める割合を表す。したがって、Pred(25).MREが大きいほどMREでの予測精度が高いことを示す。Pred(25).MER

表1 実験用データセットに含まれるプロジェクトの特性  
Table 1 Metrics involved in experimental dataset.

特性値名	尺度	含まれる特性	特性値数
FP数	比尺度	数量データ	————
総工数	比尺度	数量データ	————
計画工数	比尺度	数量データ	————
要件定義工数	比尺度	数量データ	————
開発種別	名義尺度	新規開発, 派生開発, 再開発	3
開発言語	名義尺度	COBOL, Java, C等	24
工数計測手法	名義尺度	実作業時間集計, 報告作業時間集計等	7
工数算入業務	順序尺度	レベル1~4	4

についても同様である。

以下にMREとMERの計算式を示す。式中、実測値をEで表し、モデルから算出される値(予測値)を $\hat{E}$ で表す。

相対誤差 MRE 計算式

$$MRE = \frac{|\hat{E} - E|}{E}$$

相対誤差 MER 計算式

$$MER = \frac{|\hat{E} - E|}{\hat{E}}$$

## 5. 結果と考察

### 5.1 結果

提案手法を用いて予測した結果をMREは表2および図2に、MERは表3および図3に示す。図2、図3では、30回の試行のそれぞれで得られたMdMREおよびMdMERの分布を箱ひげ図で示している。表中および図中では、すべてのデータセットを用いてモデルを構築したものを「従来法」と表記し、提案手法において適合度を表す指標に残差平均平方を用いたものを「RMS優先」、自由度調整済み決定係数を用いたものを「Adj.R<sup>2</sup>優先」、

表2 実験結果(MRE)

Table 2 Results of experiment (MRE).

	提案手法 RMS優先	提案手法 Adj.R <sup>2</sup> 優先	従来法	Analogy-based法	開発種別2分法
MMRE	0.640	0.761	0.782	0.592	0.708
MdMRE	0.367	0.397	0.452	0.410	0.436
Pred(25).MRE	35.5	32.0	31.7	31.8	31.8

表3 実験結果(MER)

Table 3 Results of experiment (MER).

	提案手法 RMS優先	提案手法 Adj.R <sup>2</sup> 優先	従来法	Analogy-based法	開発種別2分法
MMER	0.592	0.709	0.507	0.595	0.536
MdMER	0.336	0.358	0.357	0.385	0.360
Pred(25).MER	38.7	34.8	34.7	33.2	34.0



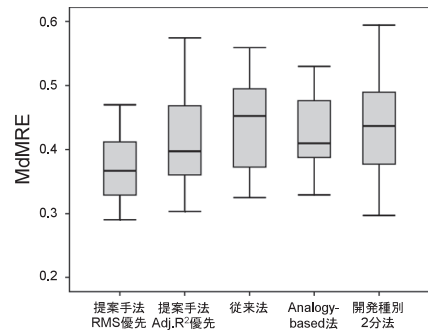


図 2 実験結果箱ひげ図 (MRE)

Fig. 2 Boxplots of experimental results (MRE).

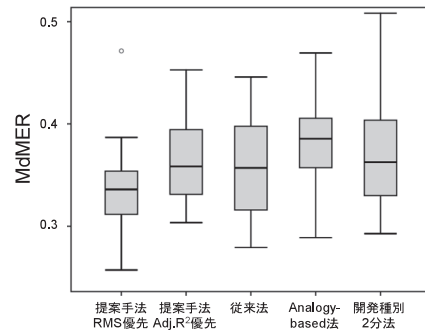


図 3 実験結果箱ひげ図 (MER)

Fig. 3 Boxplots of experimental results (MER).

Analogy-based 法を用いたものを「Analogy-based 法」、開発種別 2 分法はそのまま表記した。また表中の Pred(25) は 30 回の試行それぞれで得られた評価基準を平均したものであり、平均値、中央値および分散については 30 回の試行の結果の中央値を用いた。

まず、提案手法において RMS 優先と Adj.R<sup>2</sup> 優先を比較すると、前者の方がすべての評価基準において高い精度を示した。また、Wilcoxon 検定を行った結果、MdMRE では  $p = 0.00171$ 、MdMER では  $p = 0.000975$  と、どちらの尺度の場合でも有意水準 1% で RMS 優先の精度が高いことが確認できた。そこで、以降では、提案手法の RMS 優先と他の手法との比較を行う。

提案手法 (RMS 優先) と、従来手法を比較すると、提案手法は MRE ではすべての指標において従来手法からの精度の改善がみられた。しかしながら MER では MdMER、Pred(25).MER では精度の改善がみられたものの、MMER の精度は低下していた。Analogy-based 法との比較では、MER ではすべての評価基準において提案手法が優れていたが、MMRE では Analogy-based 法の精度が優れているという結果になった。開発種別 2 分法との比較では、従来手法との比較と同様の結果となり、提案手法がつねに優れているとはいえない。しかしながら、どの場合にも 6 種類の評価基準のうち 5 種類で提案手法の方が優れており、このような場合、過去の工数見積りに関する論文では、より多くの評価基準で勝っている方を優れているとするのが一般的であり<sup>14),25),30)</sup>、本論文でも従来手法、Analogy-based 法、開発種別 2 分法よりも提案手法の方が優れていると考えられる。Wilcoxon 検定を行った結果、提案手法と従来手法の組合せでは MdMRE は  $p = 0.000148$ 、MdMER

表 4 特性ごとのモデル選択割合 (単位: %)  
Table 4 Selection percentage of each metrics' model.

特性変数1	特性値1	特性変数2	特性値2	RMS優先	Adj.R <sup>2</sup> 優先
全データモデル	————	————	————	7.5	13.1
FP数	下位	————	————	7.8	1.8
計画工数	下位	————	————	16.1	2.0
要件定義工数	下位	————	————	11.0	0
開発種別	派生開発	————	————	5.1	22.2
開発言語	COBOL	————	————	1.9	8.3
FP数	上位	開発種別	新規開発	3.1	7.3
FP数	下位	開発種別	新規開発	2.9	6.0
要件定義工数	下位	開発種別	新規開発	7.3	1.3
要件定義工数	下位	工数算入業務	レベル1	8.0	2.5

は  $p = 0.00567$ 、提案手法と Analogy-based 法との組合せでは MdMRE は  $p = 0.00927$ 、MdMER は  $p = 0.000136$  と、有意水準 1% で提案手法の精度が高いことが示された。また提案手法と開発種別 2 分法の組合せでは MdMRE は  $p = 0.015$ 、MdMER は  $p = 0.013$  と、有意水準 5% で提案手法の精度が高いことが示された。

### 5.2 考 察

まず、提案手法に用いる適合度の指標として RMS 優先を用いた場合と Adj.R<sup>2</sup> 優先を用いた場合を比較すると、前者の方が高い精度を示した。この結果から、提案手法に用いる適合度を表す指標としては、RMS を使うことを推奨する。このような結果となった 1 つの解釈は、RMS 優先では推定値と実測値との残差が (平均的に) 小さいモデルが選択されるのに対し、Adj.R<sup>2</sup> 優先では推定値と実測値との相関係数が大きいモデルが選択されるが、相関係数は外れ値に影響を受けやすいため、Adj.R<sup>2</sup> 優先の精度が低下した可能性がある。

以降では、RMS 優先と Adj.R<sup>2</sup> 優先の違いの詳細を明らかにするために、各プロジェクト特性について、フィットデータ選定基準として採用された回数を調査した結果を表 4 に示す。表 4 中の値は、それぞれのフィットデータが最終的に選択された回数が全体に占める割合を示した。ただし、RMS 優先および Adj.R<sup>2</sup> 優先において選択された割合がどちらも 5% に満たないものについては省略している。

表 4 から、RMS 優先では、開発規模、システム化計画工数、要件定義工数などの比尺度の特性、特にそれが下位 (中央値よりも小さい) のプロジェクトがフィットデータとして

選定される傾向にあり、特に、計画工数が下位のプロジェクトのみを選定してフィットデータとするケースが多かった。一方、Adj.R<sup>2</sup> 優先では、開発種別や開発言語などの名義尺度の特性、そしてすべての過去プロジェクトをフィットデータとするケースが多くみられた。実験結果において RMS 優先が Adj.R<sup>2</sup> 優先よりも高い精度で予測ができたことを考えると、規模が小さいプロジェクトの予測では、開発種別などの名義尺度の変数よりも規模や工数などの比尺度の変数が共通するプロジェクトを選定し、フィットデータとした方が、高い精度で予測が可能だと考えられる。

RMS 優先, Adj.R<sup>2</sup> 優先に共通する特徴としては、開発種別が派生開発の場合にはその特性を含むプロジェクトがフィットデータとして選択されているのに対して、新規開発の場合には、開発種別が新規開発かつ、それ以外の特性を持つという 2 つの条件を満たすプロジェクトがフィットデータとして選択されている。このことは、新規開発プロジェクトの総工数を予測する場合には、過去プロジェクトから新規開発プロジェクトを抽出し、さらに FP 数や計画工数、要件定義工数などの総工数と密接に関わりのある特性によって選定を行うことが精度を高める可能性があることを示唆している。

また特筆すべきこととして、RMS 優先, Adj.R<sup>2</sup> 優先の双方においてすべての過去データをフィットデータとする場合が少なからず存在している。このことは、フィットデータを特性によって選択することが、つねに精度を高めるわけではないことを示している。一般に、特性に基づくフィットデータの選定は予測精度を高めると考えられがちであるが、この結果は、予測対象のプロジェクトによってはすべての過去のプロジェクトをフィットデータとして扱った方が精度が高くなる可能性を示唆しており、興味深い。予測精度を高めるためには、フィットデータの選定において、すべての過去のプロジェクトをフィットデータ候補の 1 つとして含めることが重要であると考えられる。

次に、提案手法 (RMS 優先) と従来手法を比較した場合に、MER の平均値である MMER において提案手法が低い精度を示した原因について考察を行う。考えられる原因として、提案手法では、予測結果に外れ値が存在したことがあげられる。平均値は中央値よりも外れ値に影響を受けやすいことから、MdMER では高い精度を示しているも、MMER では低い精度となったと考えられる。実績データセットには多くの場合、外れ値となるようなプロジェクトが含まれており、提案手法では、そのようなプロジェクトをうまく予測できていない可能性がある。しかし、MdMER では高い精度を示していることから、外れ値とならない多くのプロジェクトをより高精度で予測できており、提案手法の有用性は高いと考える。

## 6. おわりに

本論文では、過去のソフトウェア開発プロジェクトの実績データを用いて重回帰分析により開発工数予測モデルを構築するという状況において、予測対象プロジェクトの特性に基づいて、モデル構築のためのフィットデータに適したプロジェクト群を選定する方法を提案した。フィットデータの選択を行わない (全過去プロジェクトを用いる) 従来の重回帰分析と比較すると、相対誤差 MRE の中央値では 0.452 から 0.367 へ、相対誤差 MER の中央値では 0.357 から 0.336 へと精度が向上した。また Analogy-based 法と比較した場合にも、多くの評価指標で精度が高いことが示された。

本実験の結果から、予測対象の特性に合わせてフィットデータを選定することが重要であり、それを系統的に行う一手法として、提案手法が役立つことが示された。同時に、フィットデータの選択結果から、選定時の候補の中にすべての過去データを用いるケースを含めることもまた、精度を高めるうえでは重要であることが示された。

実際のソフトウェア開発における提案手法の適用可能性としては、提案手法は 1 つ以上のプロジェクト特性が記録されていれば利用できるもので、原理上はソフトウェア開発のどの時点においても適用可能である。特に高い精度での見積りが可能となる業務要件設計終了以降<sup>31)</sup>であれば、開発プロセスの進行にともない、利用可能な特性の種類は増加する。開発プロセスの節目節目において提案手法を適用することでプロジェクトの特性により即したフィットデータとモデルが選択可能となり、精度の向上が期待できる。

ソフトウェア開発企業内で提案手法を適用する場合に精度をさらに向上させる方法としては、評価実験では 30 件としたモデル構築のための最小フィットデータ数を企業で蓄積されているデータに合わせて適切に変更することが考えられる。それに加えて、フィットデータに含まれる特性を結合する、たとえば開発言語の C と C++ を 1 つのものとして扱う、といった作業を適切に行うことでも、精度の向上が可能であると考えられる。

提案手法の適用結果のソフトウェア開発へのフィードバック方法としては、まず、開発初期の時点で予測を行い、予測結果を開発現場に示し、納期と予測工数から調達すべき人員の数を決定することが考えられる。そして、開発が進むとともに繰り返し予測を行うことで、予測工数と実際の工数の乖離から、進行中のプロジェクトに問題があるかどうかを判定することで、問題の発生の把握に役立つ。また、人手による工数見積りが併用されている場合には、モデルによる予測結果と比較することで、人手による見積りの確からしさを確認したり、見積りの再検討をするのに役立つ。さらに、予測工数とともに、選定されたフィット



データ集合および、対象プロジェクトと一致した特性値を開発現場に提供することで、予測の根拠の確認が可能となり、予測結果が開発現場に受け入れられやすくなると期待される。

なお、現代のソフトウェア開発は、派生開発が多く、そのような開発プロジェクトでは、提案手法による工数予測に先立って、母体システムの調査を行い、母体の変更箇所や規模を明確にしておくことが望ましい。開発の契約の観点からは、母体の調査を含む上流工程を準委任契約とし、開発規模の見積りを行い、改めて請負契約を結ぶことが奨励される<sup>32)</sup>。

提案手法は、モデルベースの予測手法であれば、ニューラルネットなどにも適用可能である。また、2クラス判別問題のための線形判別モデルやロジスティック回帰モデルなどにも応用可能である。今後、重回帰分析以外の予測手法に対する提案方法の有効性評価が課題となる。

謝辞 本研究の一部は、文部科学省「e-Society 基盤ソフトウェアの総合開発」および「次世代 IT 基盤構築のための研究開発」の委託に基づいて行われた。

### 参 考 文 献

- 1) Boehm, B.W.: Software Engineering Economics, *IEEE Trans. Softw. Eng.*, Vol.10, No.1, pp.4–21 (1984).
- 2) Briand, L., Langley, T. and Wiczorek, I.: A replicated Assessment and Comparison of Common Software Cost Modeling Techniques, *Proc. 22nd International Conference on Software Engineering (ICSE'00)*, pp.377–386 (2000).
- 3) Conte, S.D., Dunsmore, H.E. and Shen, V.Y.: *Software Engineering Metrics and Models*, The Benjamin-Cummings Pub. Co. (1986).
- 4) 奥野忠一, 久米 均, 芳賀敏郎, 吉澤 正: 多変量解析法, 日科技連 (1971).
- 5) Walkerden, I. and Jeffery, R.: An Empirical Study of Analogy-based Software Effort Estimation, *Empirical Software Engineering*, Vol.4, No.2, pp.135–158 (1999).
- 6) International Software Benchmarking Standards Group: ISBSG Estimating Benchmarking and Research Suite Release 9, ISBSG (2004).
- 7) Jeffery, R., Ruhe, M. and Wiczorek, I.: Using Public Domain Metrics to Estimate Software Development Effort, *Proc. 7th International Software Metrics Symposium (METRICS'01)*, pp.16–27 (2001).
- 8) Sentas, P., Angelis, L., Stamelos, I. and Bleris, G.: Software productivity and effort prediction with ordinal regression, *Information and Software Technology*, Vol.47, No.1, pp.17–29 (2005).
- 9) Mendes, E., Lokan, C., Harrison, R. and Triggs, C.: A Replicated Comparison of Cross-company and Within-company Effort Estimation Models using the ISBSG Database, *Proc. 11th IEEE International Software Metrics Symposium (METRICS'05)*, p.36 (2005).
- 10) Putnam, L. and Myers, W.: *Measures for Excellence*, PRENTICE HALL (1992).
- 11) 柿元 健, 角田雅照, 大杉直樹, 門田暁人, 松本健一: 協調フィルタリングによる工数見積もり手法におけるデータ数と見積もり精度の関係の分析, ソフトウェア工学の基礎 XII, 日本ソフトウェア科学会 (FOSE2005), pp.77–86 (2005).
- 12) 本間周二: CSK に見る FP 法の実践 プロジェクト定量化で品質と見積り精度向上, 日経 IT プロフェッショナル, Vol.7 (2004).
- 13) Spiegel, M.R. (著), 氏家勝巳, 土井 誠 (訳): マグロウヒル大学演習統計, p.73, オーム社 (1995).
- 14) Lokan, C.: What Should You Optimize When building an Estimation Model? *Proc. 11th IEEE International Software Metrics Symposium (METRICS'05)*, p.34 (2005).
- 15) Akaike, H.: A New Look at The Statistical Model Identification, *IEEE Trans. Automatic Control*, Vol.19, No.6, pp.716–723 (1974).
- 16) 瀧 進也, 戸田航史, 門田暁人, 柿元 健, 角田雅照, 大杉直樹, 松本健一: プロジェクト類似性に基づく工数見積りに適した変数選択法, 情報処理学会論文誌, Vol.49, No.7, pp.2338–2348 (2008).
- 17) Mendes, E. and Kitchenham, B.: Further Comparison of Cross-company and Within-company Effort Estimation Models for Web Applications, *Proc. 11th IEEE International Software Metrics Symposium (METRICS'05)*, pp.348–357 (2004).
- 18) Shepperd, M. and Schofield, C.: Estimating Software Project Effort Using Analogies, *IEEE Trans. Softw. Eng.*, Vol.23, No.12, pp.736–743 (1997).
- 19) Briand, L. and Jürgen, W.: Modeling Development Effort in Object-Oriented Systems Using Design Properties, *IEEE Trans. Softw. Eng.*, Vol.27, No.11, pp.963–986 (2001).
- 20) Kadoda, G., Cartwright, M., Chen, L. and Shepperd, M.: Experience Using Case-Based Reasoning to predict Software Project Effort, *Proc. 4th International Conference on Empirical Assessment & Evaluation in Software Engineering*, pp.1–34 (2000).
- 21) Angelis, L. and Stamelos, I.: A Simulation Tool for Efficient Analogy Based Cost Estimation, *Empirical Software Engineering*, Vol.5, No.1, pp.35–68 (2000).
- 22) Ohsugi, N., Tsunoda, M., Monden, A. and Matsumoto, K.: Effort Estimation Based on Collaborative Filtering, *Proc. 5th International Conference on Product Focused Software Process Improvement (Profes2004)*, Bomarius, F. and Iida, H. (Eds.), Vol.3009, pp.274–286 (2004).
- 23) 角田雅照, 大杉直樹, 門田暁人, 松本健一, 佐藤慎一: 協調フィルタリングを用いたソフトウェア開発工数予測方法, 情報処理学会論文誌, Vol.46, No.5, pp.1155–1164 (2005).

- 24) Angel Project: Shepperd, M., Bournemouth University (online). available from <http://dec.bournemouth.ac.uk/ESERG/ANGEL/> (accessed 2008-07-22)
- 25) Mendes, E., Watson, I., Triggs, C., Mosley, N. and Counsell, S.: A Comparative Study of Cost Estimation Models for Web Hypermedia Applications, *Empirical Software Engineering*, Vol.8, No.2, pp.163-196 (2003).
- 26) Kitchenham, B., Pfleeger, S.L., McColl, B. and Eagan, S.: An Empirical Study of Maintenance and Development Estimation Accuracy, *Journal of Systems and Software*, Vol.64, Issue 1, pp.57-77 (2002).
- 27) Desharnais, J.M.: Analyse statistique de la productivité des projets informatiques partie de la technique des point des fonction, Unpublished Masters Thesis (1989).
- 28) International Software Benchmarking Standards Group: The Benchmark Release 6, International Software Benchmarking Standards Group (2000).
- 29) Kitchenham, B., MacDonell, G., Pickard, M. and Shepperd, M.: What Accuracy Statistics Really Measure, *IEEE Proc. Software*, Vol.148, No.3, pp.81-85 (2001).
- 30) Jørgensen, M.: Experience with the Accuracy of Software Maintenance Task Effort Prediction Models, *IEEE Trans. Softw. Eng.*, Vol.21, No.8, pp.674-681 (1995).
- 31) 原田 晃, 幕田行雄, 石川貞裕, 大野 治, 楠本真二, 井上克郎: ファンクションポイント法を応用した早期見積技法の提案とそのシステム化, 電子情報通信学会論文誌, Vol.J89-D, No.4, pp.755-766 (2006).
- 32) 経済産業省商務情報政策局情報処理振興課:「情報システムの信頼性向上のための取引慣行・契約に関する研究会」報告書 モデル取引・契約書 (2007).

## 付 録

### A フィットデータ選定手順の詳細

; Step 1 ... フィットデータセット候補群の構築

過去プロジェクトの集合を  $P(p_1, \dots, p_{p_{\max}})$  とし, 予測対象プロジェクトを  $p_i \notin P$  とする. プロジェクト特性の集合を  $V$  とし,  $V$  のべき集合の部分集合  $S = \{X \mid X \in \wp(V) \wedge X \neq \emptyset \wedge |X| \leq m_{\max}\}$  とする. ここで,  $m_{\max}$  は, 予測対象プロジェクトと一致する特性の最大値である.

for 全ての  $s_i \in S$  について {

フィットデータ候補となるプロジェクト集合  $F_i = \{p \mid (v \in s_i) R(p, v) = R(p_i, v)\}$  を作成する.

ここで,  $R(p, v)$  はプロジェクト  $p$  の特性  $v$  の値を表す.

}

; Step 2 ... 予測モデル群の構築

予測モデルを構築するうえで必要な最小プロジェクト数を  $min\_projects$  とする

for 全ての  $F_i$  について {

if  $|F_i| \geq min\_projects$  ならば, 予測モデル  $m_i$  をプロジェクト集合  $F_i$  を用いて構築する.

}

予測モデル  $m_{all}$  を全プロジェクト集合  $P$  を用いて構築する.

Step 2 で構築されたすべての予測モデルの集合を  $M$  とする.

; Step 3 ... 最良のフィットデータおよびモデルの選定

モデル  $m$  の適合度を  $GoF(m)$  とする.

すべての  $m \in M$  について  $GoF(m)$  を算出し, 最良の  $GoF(m)$  を持つモデル  $m_s$  およびそのフィットデータ  $F_s$  を選定する.

; Step 4 ... モデルの適用

Step 3 で選定された最良のモデル  $m_s$  を予測対象プロジェクト  $p_t$  に適用し, 予測結果を得る.

### B 2 値変数変換手順

名義尺度, 順序尺度の 2 値変数への変換の説明を, 開発言語を例にとって行う.

開発言語の中に「COBOL」「C」の 2 つが含まれている場合, 説明変数  $x$  は言語が COBOL のときには 1, 言語が C のときには 0 をとる. 同様に言語が「COBOL」「C」「JAVA」のときにはその自由度  $3 - 1 = 2$  に対応して 2 つの定義変数  $x_1, x_2$  を導入し, その値は以下の表のようになる.

開発言語	$x_1$	$x_2$
COBOL	1	0
C	0	1
JAVA	0	0

これが名義尺度, 順序尺度を 2 値変数に変換する手続きである.

(平成 20 年 7 月 23 日受付)

(平成 21 年 9 月 11 日採録)



戸田 航史

平成 16 年大阪大学基礎工学部卒業。平成 18 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了。現在、同大学博士後期課程在籍。ソフトウェアメトリクスの研究に従事。IEEE 会員。



門田 暁人 (正会員)

平成 6 年名古屋大学工学部電気学科卒業。平成 10 年奈良先端科学技術大学院大学情報科学研究科博士後期課程修了。同年同大学助手。平成 16 年同大学助教授。平成 19 年同大学准教授。平成 15~16 年 Auckland 大学客員研究員。博士 (工学)。ソフトウェアメトリクス, ソフトウェアプロテクション, ヒューマンファクタの研究に従事。電子情報通信学会, 日本ソフトウェア科学会, IEEE, ACM 各会員。



松本 健一 (正会員)

昭和 60 年大阪大学基礎工学部情報工学科卒業。平成元年同大学大学院博士課程中退。同年同大学基礎工学部情報工学科助手。平成 5 年奈良先端科学技術大学院大学助教授。平成 13 年同大学教授。工学博士。エンピリカルソフトウェア工学, 特に, プロジェクトデータ収集/利用支援の研究に従事。電子情報通信学会, 日本ソフトウェア科学会, ACM 各会員,

IEEE Senior Member .