

# ソフトウェア信頼度成長モデルによる学習者特性を考慮した 原稿作成指導手法の検討

土井 崇<sup>1</sup> 奥田 隆史<sup>1,a)</sup> 井手口 哲夫<sup>1,b)</sup> 田 学軍<sup>1,c)</sup>

**概要:** 日本中の大学教員は、学生の能力向上を意識しながら期限までにある品質以上の原稿を作成するための指導をおこなっている。そのため教員は、学生の能力や意欲等の個人差に応じて、指導内容や指導方法を考慮する必要がある。例えば、学生はどの程度の期間で原稿を作成できるか、教員はどのようなタイミングで学生にアドバイスを提供する必要があるかなどを考慮することである。そこで本稿では、原稿の修正過程をソフトウェア信頼度成長モデルにより表現することで、学生の能力向上を意識した原稿作成指導について検討する。その結果より、教員に原稿作成プロセスの指標を提供することを目的とする。また、指導内容や学生の能力変化によって学生の修正数の確率的特徴が著しく変化する現象を考慮することで、より現実的なモデルとした。

## A Proposal of Teaching Methods for Academic-Document Writing Process - using Software Reliability Models with Change-Point -

TAKASHI DOI<sup>1</sup> TAKASHI OKUDA<sup>1,a)</sup> TETSUO IDEGUCHI<sup>1,b)</sup> XUEJUN TIAN<sup>1,c)</sup>

**Abstract:** In Japan, a professor teaches students at the laboratory of university in order to create a good document by the deadline. The professor is changing the document teaching methods to each student personality. Hence, we will support the professor by showing the measure of guidance. In this paper, we will consider the method of document production support, by representing document production as a software reliability growth model. Also, we will make it a realistic model in order to add the change of student skills to document writing process.

### 1. はじめに

近年、企業や若者を取り巻く経済環境の変化により、「基礎学力」や「専門知識」に加え、それらをうまく活用していくための「社会人基礎力」を大学教育において意識的に教育することが重要となっている [1]。ここで社会人基礎力とは、経済産業省が 2006 年から提唱している「職場や地域社会で多様な人々と仕事をしていくために必要な基礎的な力」のことであり 3 つの能力（「前に踏み出す力」、「考

え抜く力」、「チームで働く力」）からなる。また、社会人基礎力の教育では、「教える」視点だけでなく、「学生自身いかに経験させ、考えさせるのか」ということを重視する。このことから社会人基礎力の教育では、具体的な講義を受講するだけでなく、大学において学生に能動的な学習の機会を与えることが重要となる。

そこで本稿では、学生の社会人基礎力を教育する機会として、理工系学部の研究室でおこなわれる学会原稿作成に注目する。大多数の理工系学部の研究室では、学生・教員間での共同での原稿作成プロセスによって、ある水準以上の質を有する原稿を作成している [2]。この原稿作成プロセスは、社会人基礎力の 3 つの能力全てを活用するプロセスであるため、社会人基礎力を教育する機会として適している。以下に学会原稿作成においてどのように社会人基礎

<sup>1</sup> 愛知県立大学大学院 情報科学研究科  
Graduate School of Information Science and Technology, Aichi Prefectural University, Ibaragabasama, Nagakute-City, Aichi, 480-1198 Japan

a) okuda@ist.aichi-pu.ac.jp

b) ideguchi@ist.aichi-pu.ac.jp

c) tan@ist.aichi-pu.ac.jp

力の3つの能力を活用するかを述べる。

まず第一に「前に踏み出す力」について述べる。学会原稿作成のような限られた時間の中で目的を達成することが必要な場合、教員が一方向的に指導をおこなう方が時間が掛からないことが多い。しかし、学生に知識の伝達がおこなわれなければ、能力の向上は望めない。体験を通して発見した問題の解を求めて能動的に理論を学ぼうとすることで、実体験が確実に知識として学生の中に定着していく。そのため、本稿では学会原稿作成において学生が自ら行動し解を見つけるような指導を教員がするとする。このような指導すると仮定した場合、学生は原稿作成するために「前に踏み出す力」が必要となる。

次に「考え抜く力」について述べる。学会原稿作成では、研究結果を正確で簡潔な表現による論理的な文章で作成する能力が必要となる。ここでいう文章を作成する能力とは、事実を客観的に示し、明確に主張を述べるができること、複数の解釈を与えず冗長な表現を含まない文章を作成することのできる能力である [3]。また、文章を作成する能力は、伝えたいことを正確に表現するという意味において、伝えるべき情報がどれほど整理されているか、すなわち論理的であるかが問われる。したがって、論理的な文章力の育成と論理的な思考力の育成は表裏一体のものとして捉えられることが多い。このようなことから、学生には学会原稿作成において論理的な思考力を含む多角的な思考力が必要となり「考え抜く力」が育成される。

最後に「チームで働く力」について述べる。研究指導を考慮した学会原稿作成においては表記・文法に関する誤り指摘以上に意味的な解析による支援が必要である。しかしながら、現状の作文支援システムの多くは、表記・文法に関する誤り指摘は可能となっているが、意味的な解析が必要となる支援については、部分的に実現されるにとどまっている [4]。そのため、学会原稿に必要な記述が含まれているか、記述内容の説明が不足していないか、意味的な誤りや矛盾はないか、といった深い意味解析を必要とする支援は教員が不可欠となる。すなわち、研究指導を考慮した意味解析するために、教員は学生を指導し、学生は指導に応じて学会原稿を修正する必要がある。つまり学生は、教員とひとつのチームとして学会原稿作成に取り組むため「チームで働く力」が必要となる。

以上より、この原稿作成プロセスでは、教育の成果として単なる研究論文だけでなく、原稿を完成するために自ら行動させ、論理的に考えさせ、教員とチームで働くことによって学生の総合的な能力をも教育していると考えられる。したがって、教員等と共著の学会原稿を作成する原稿作成プロセスは、社会人基礎力が求めている3つの能力全てを活用するプロセスであるといえる。

この原稿作成プロセスを能動的な学習をおこなう機会と捉えたときの問題点は、プロセス終了時間が、当該学生の

原稿作成・修正スキルや意欲等の個人差に強く左右されることである。そこで本稿では、学生の上記スキルが完成時間に与える影響を、ソフトウェア信頼度成長モデルを応用した確率モデルを用いて検討する。また本稿では、原稿作成プロセス期間に学生の上記スキルの確率的特徴が著しく変化する現象を考慮することによって、より現実的なモデルとする。ソフトウェア信頼度成長モデルにおいてこのような現象が発生する時刻は、チェンジポイントと呼ばれる [5]。

なお本稿では、確率モデルを用いた検討結果をもとに教員に原稿作成プロセスの指標を提供することを目的とする。原稿作成プロセスにおいて教員は、明確に原稿の間違いを指摘するだけでなく、学生の能力向上を意識する必要がある。そのため、間違いを指摘するだけの教育に比べて教員に負担がかかる。そこで本稿では、教員に原稿作成プロセスの指標を提供することによって教員を補助する。具体的には、学生はどの程度の期間で論文を作成できるか、教員はどのようなタイミングで学生にアドバイスを提供する必要がありますかなどの目安を教員に提供することである。

以下、2節において、本稿に関連する研究としてソフトウェア信頼度成長モデルについて述べ、3節において、本稿で想定する原稿作成プロセスとその評価モデルを定義する。4節において、エラー数とソフトウェア信頼度成長モデルを応用した学生の上記スキルの確率モデルを提案する。次に、5節において、学生の上記スキルが原稿の作成にどのような影響を与えるかを、シミュレーションにより検証する。6節においては、シミュレーション結果より改善方法についての検討・考察をおこなう。最後に、7節でまとめと今後の課題について述べる。

## 2. 関連研究

本節では、関連研究として、2.1項にてソフトウェア信頼度成長モデルについて述べ、2.2項にてチェンジポイントを考慮したソフトウェア信頼度成長モデルについて述べる。

### 2.1 ソフトウェア信頼度成長モデル

ソフトウェア信頼度成長モデルとは、ソフトウェアテストにおいてテスト時間の経過とともにソフトウェア内に潜在するフォールト数が発見・修正・除去されて減少し、ソフトウェア故障の発生する確率が減少してソフトウェア信頼度が増加したり、ソフトウェア故障の発生時間間隔が長くなったりするソフトウェアの実行過程を記述する確率・統計論に基づく数理モデルである [6]。

ソフトウェア信頼度成長モデルは、潜在するフォールトが発見・修正・除去される過程を数理モデルによって表現するため、原稿の修正過程を数理モデルによって表現するという本稿の目的と類似している。

本稿では  $h(t)$  を強度関数、 $H(t)$  を平均値関数とする  $NHPP$ (nonhomogeneous Poisson process) モデル

$$Pr\{N(t) = n\} = \frac{\{H(t)\}^n}{n!} e^{-H(t)} \quad (n = 0, 1, 2, \dots)$$

$$H(t) = \int_0^t h(x) dx \quad (t \geq 0) \quad (1)$$

に基づくソフトウェア信頼度成長モデルを用いる [7]。ここで、 $N(t)$  は時間区間  $(0, t]$  において発生する総期待フォールト数を表す確率変数であり、 $Pr\{\cdot\}$  は確率を表す。

## 2.2 チェンジポイントを考慮したソフトウェア信頼度成長モデル

ソフトウェア開発プロセスにおけるテスト工程では、ソフトウェア故障発生時間間隔の確率的特徴が著しく変化する現象がしばしば観測される。このような現象が発生するテスト時刻はチェンジポイントと呼ばれ、これまでに提案されているソフトウェア信頼性モデルに基づいた信頼性評価精度低下の原因の1つとして考えられている [5]。

チェンジポイントを考慮したソフトウェア信頼性評価手法では、チェンジポイントの発生要因として、大別して、以下の2つの場合を想定することが多い。まず第1に、納期遅れまたは納期までに目標とする信頼度が達成不可能と予測されるとき、テスト期間後期におけるフォールト除去率の低下を抑制するとき、または、テスト計画の変更など、ソフトウェア管理者がチェンジポイントを意図的に発生させる場合である。そして第2に、フォールト発見難易度、フォールト独立性、モジュール毎のフォールト密度の違い、およびテスト技術者の学習課程により、チェンジポイントが自然に発生する場合である。

本稿では、チェンジポイント発生を考慮したソフトウェア信頼度成長モデルを応用し、原稿作成プロセスにおいてチェンジポイントが発生することを考慮したモデルを構築し、そのモデルについて検討する。

## 3. 想定環境と評価モデル

本節では、3.1 項にて想定環境について述べ、3.2 項にて想定研究室でおこなわれる原稿作成プロセスについて述べ、3.3 項にて原稿作成プロセスの評価モデルを述べる。

### 3.1 想定環境

本稿の想定する大学の理工系の研究室を図1に示す。

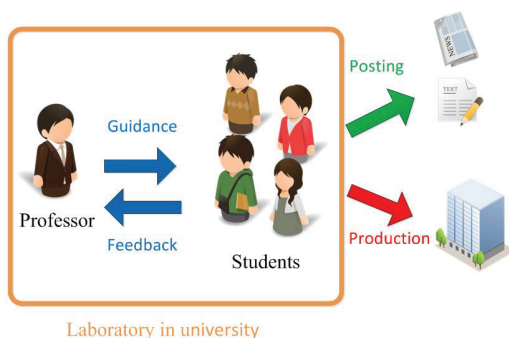


図1 大学の研究室における教育

Fig. 1 Professor and students in laboratory of university.

本稿では、教員1人に対して学生が複数人所属する大学の理工系の研究室を想定する。教員は、学生に対して指導をおこない、学生から指導内容のフィードバックを受ける。本稿では、指導内容のフィードバックとして学生作成原稿を想定している。想定研究室では、教員が学生作成原稿を完成と判断した場合、学生が論文投稿や成果発表をとする。

また本稿で想定する原稿は、他人に内容を一義に理解してもらうという明確な意図を持った文書に属するものであり、文章構成は定型かつ明確に構成され、一般には理科系文書や技術文書と呼ばれるものである [8][9]。

### 3.2 原稿作成プロセス

本稿における原稿作成プロセスの修正数の傾向とチェンジポイントの発生についての関係を図2に示す。

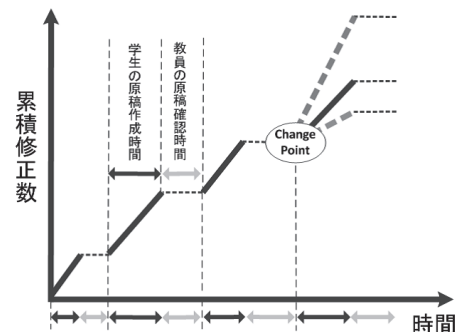


図2 原稿作成プロセスにおける修正数の推移

Fig. 2 The number of modify errors in the academic-document writing process.

本稿の原稿作成プロセスは、まず学生が原稿を作成し教員に紙や電子ファイルの形式で教員に提出し、教員が学生作成原稿の確認をおこない修正・不足箇所等の指摘をする。学生は指摘箇所はむろん指摘内容と類似した関連箇所を修正し、再度教員へ修正原稿を提出する。

この原稿作成プロセスを確率モデルすると、提出される学生作成原稿は学生の修正スキルに依存した修正数(エラー発見数)を持ち、原稿提出期限となるまでこの原稿作成プロセスを繰り返すことと表すことができる。また、教員が学生作成原稿の確認をしている際は、修正数は変化しないとする。

原稿作成プロセス期間においてチェンジポイントが発生した場合、学生の修正スキルの確率的特徴が変化する。なお、本稿の原稿作成プロセスにおけるチェンジポイント発生要因は、主に以下の2つを想定している。まず第1に、原稿提出期限に向けて原稿の修正数についての目標が達成不可能と予測されるとき、原稿作成プロセス期間後期における修正率の低下を抑制するとき、または、指導計画の変更など、指導教員がチェンジポイントを意図的に発生させる場合である。そして第2に、エラー発見難易度、および学生の学習課程により、チェンジポイントが自然に発生する場合である。

### 3.3 原稿作成プロセスの評価モデル

本稿の原稿作成プロセスモデルを図3に示す。

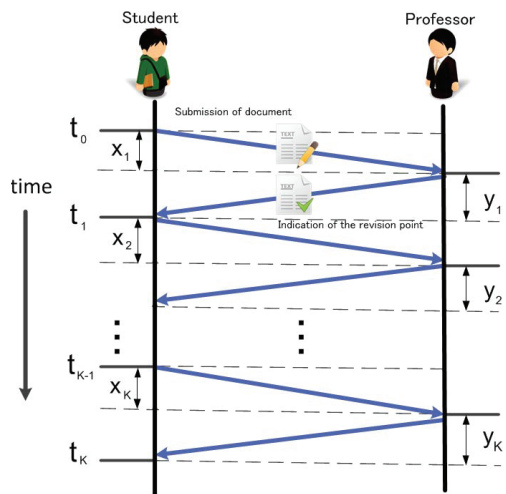


図3 原稿作成プロセスモデル

Fig. 3 Academic-document writing process.

本モデルでは学生が原稿作成を開始した時刻を  $t_0$ 、原稿提出期限となる直前の時刻を  $t_K$  とし、原稿提出期限となる直前の提出回数を  $K$  回とする。また、学生が教員に原稿を  $i$  ( $1 \leq i \leq K$ ) 回目の提出し学生に原稿が返却された時刻を  $t_i$  とする。

原稿の指導は以下の (1)~(3) に従っておこなわれる。

- (1) 学生が作成した原稿を、教員に  $i$  回目の提出をする。学生の  $i$  回目の原稿作成時間  $x_i$  は、平均作成時間  $\mu_s^{-1}$  の指数分布に従う。
- (2) 原稿は教員が確認をし、学生に対して適切な指導をおこない学生に原稿を返却する。教員の  $i$  回目原稿確認時間  $y_i$  は、平均原稿確認時間  $\mu_p^{-1}$  の指数分布に従う。
- (3) 学生は、学生の修正スキルに応じた修正を原稿におこなう。原稿提出期限となるまで、手順の (1)~(3) を繰り返す。

すなわち、 $t_i$ 、 $x_i$  と  $y_i$  は、

$$t_i = \sum_{h=1}^i (x_h + y_h) \quad (1 \leq i \leq K) \quad (2)$$

と表すことができる。

### 4. 原稿のエラー発見数と学生の修正スキル

本稿では想定原稿を、文章構成が定型かつ明確に構成されることから、自然言語の中でも形式言語に近い性質があるとし、想定原稿を複数文章（コード）から構成されるソフトウェアのプログラムコードのように捉え、学生作成原稿の修正数（エラー発見数）は、プログラムコードにおけるフォールト発見数に似た性質があるものとする。そこで、原稿作成プロセスのエラー発見事象に対して2.1項で述べたNHPPを導入する。つまり、原稿作成プロセスに

おいて時間区間  $(0, T_i]$  までの総期待エラー発見数は、ソフトウェア信頼度成長モデルのNHPPモデルにおける総期待フォールト数に相当するとする。

本稿では、簡単化のために原稿作成期間  $(0, T_K]$  を通じてチェンジポイントが1回発生すると仮定する。チェンジポイントが発生する時刻を  $\tau$  ( $0 < \tau < t_K$ ) とし、チェンジポイントを考慮したNHPPの平均値関数  $H(t_i)$  は、

$$H(t_i) = F_1(t_i)U_1(\tau - t_i) + \{F_1(\tau) + F_2(t_i - \tau)\}U_2(t_i - \tau) \quad (3)$$

に従うとする[5]。ここで、 $U_1(x)$  および  $U_2(x)$  はステップ関数を表し、それぞれ、

$$U_1(x) = \begin{cases} 0 & (x < 0) \\ 1 & (x \geq 0), \end{cases} \quad (4)$$

$$U_2(x) = \begin{cases} 0 & (x \leq 0) \\ 1 & (x > 0), \end{cases} \quad (5)$$

である。また、 $F_1(t_i)$  はチェンジポイント発生前の学生の修正スキル、 $F_2(t_i)$  はチェンジポイント発生後の学生の修正スキルとする。

本稿において学生の修正スキル  $F_n(t_i)$  ( $n = 1, 2$ ) は、時刻  $t_0$  から原稿が返却された時刻  $t_i$  までの学生の原稿作成時間  $x_i$  の総合計時間  $x(i)$  に依存する、以下の指数形信頼度成長モデルと線形信頼度成長モデルの2つを考える。

$$F_n(t_i) = a(1 - e^{-b_n x(i)}) \quad (6)$$

$$F_n(t_i) = c_n x(i) \quad (7)$$

ここで、 $a$  は原稿作成プロセス前に潜在していた原稿の総期待エラー発見数、 $b_n$  と  $c_n$  ( $n = 1, 2$ ) は学生固有のパラメータとする。学生の修正スキルは、学生の原稿に対するエラー発見数を表している。式(6)は学生のエラー発見数が残存エラー1個当りのエラー発見率に依存して変化する場合、式(7)は学生のエラー発見数が時間に比例する場合を表現している。なお、 $F_n(t_i) \geq a$  の場合、 $F_n(t_i) = a$  とする。

### 5. 数値例

本稿では、1節にあるように教員に原稿作成プロセスの指標を提供することを目的としている。そのため、学生の修正スキルであるエラー発見数をシミュレーションによって比較し評価する。また総期待エラー発見数は5回の離散シミュレーションの平均を結果とする（乱数発生は離散事象シミュレーションパッケージCsim20[10]を利用）。

なお、原稿作成プロセス前に潜在していた原稿の総期待エラー発見数は  $a = 100$ 、学生の平均原稿作成期間と教員の原稿確認期間は  $\mu_s^{-1} = 4[\text{day}]$ 、 $\mu_p^{-1} = 1[\text{day}]$  と定める。また、原稿提出期限を100日とする。本節におけるエラー発見率は、チェンジポイントが発生しない場合において、

原稿の総期待エラー発見数を 50 日程度ですべて修正する  $b = 0.08$  と  $c = 2.5$  と, 100 日程度で 8 割修正する  $b = 0.02$  と  $c = 1.0$  と定めた.

5.1 項にてチェンジポイントが発生しない場合, 5.2 項にてチェンジポイントが発生する場合についてのシミュレーション結果を示す.

### 5.1 チェンジポイントが発生しない場合

本項では, チェンジポイントが発生しない場合の学生の修正スキルを評価する. まず, 指数形修正スキルである  $b_{1,2} = 0.02$  と  $b_{1,2} = 0.08$  に対する平均エラー発見数の関係を図 4 に示す. 次に線形修正スキルである  $c_{1,2} = 1.0$  と  $c_{1,2} = 2.5$  に対する平均エラー発見数の関係を図 5 に示す.

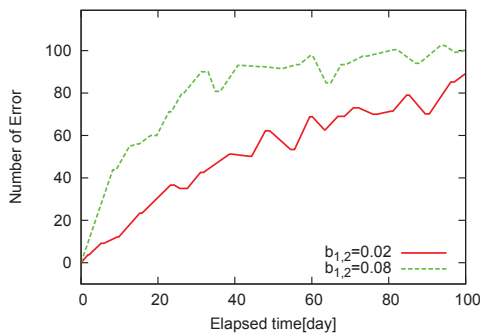


図 4 指数形修正スキル ( $b_{1,2} = 0.02, 0.08$ ).

Fig. 4 Estimated average numbers of modified errors ( $b_{1,2} = 0.02, 0.08$ ).

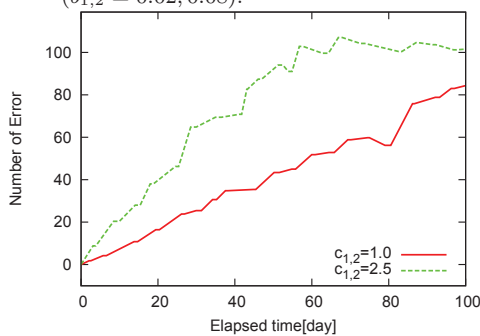


図 5 線形修正スキル ( $c_{1,2} = 1.0, 2.5$ )

Fig. 5 Estimated average numbers of modified errors ( $c_{1,2} = 1.0, 2.5$ ).

### 5.2 チェンジポイントが発生する場合

本項では, チェンジポイントが発生する場合の学生の修正スキルを評価する. チェンジポイントが発生する原稿提出の  $i$  回目を CP とする. 本項では, CP=5, 10, 15 の場合について評価する.

指数形修正スキルの場合について,  $b_1 = 0.02$  と  $b_2 = 0.08$  に対する平均エラー発見数の関係を図 6 に示す. また  $b_1 = 0.08$  と  $b_2 = 0.02$  に対する平均エラー発見数の関係を図 7 に示す.

線形修正スキルの場合について,  $c_1 = 1.0$  と  $c_2 = 2.5$  に対する平均エラー発見数の関係を図 8 に示す. また  $c_1 = 2.5$  と  $c_2 = 1.0$  に対する平均エラー発見数の関係を図 9 に示す.

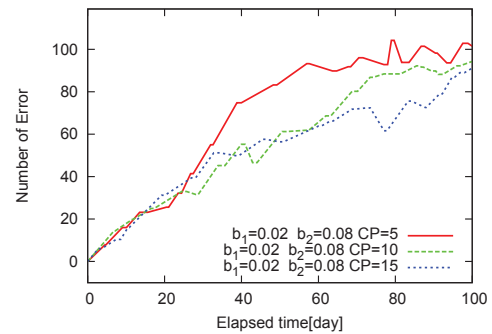


図 6 指数形修正スキルにおける CP の影響 ( $b_1 = 0.02, b_2 = 0.08$ )

Fig. 6 Estimated average numbers of modified errors ( $b_1 = 0.02, b_2 = 0.08$ ).

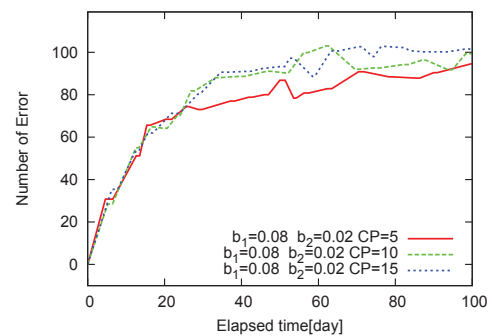


図 7 指数形修正スキルにおける CP の影響 ( $b_1 = 0.08, b_2 = 0.02$ )

Fig. 7 Estimated average numbers of modified errors ( $b_1 = 0.08, b_2 = 0.02$ ).

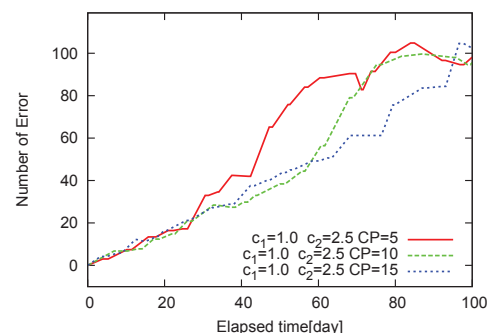


図 8 線形修正スキルにおける CP の影響 ( $c_1 = 1.0, c_2 = 2.5$ )

Fig. 8 Estimated average numbers of modified errors ( $c_1 = 1.0, c_2 = 2.5$ ).

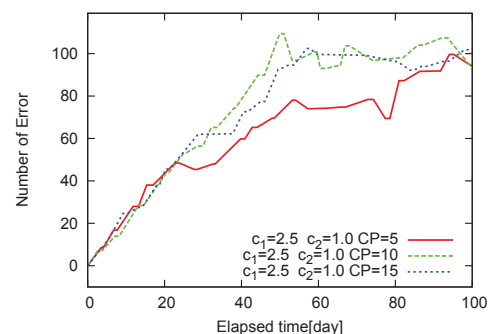


図 9 線形修正スキルにおける CP の影響 ( $c_1 = 2.5, c_2 = 1.0$ )

Fig. 9 Estimated average numbers of modified errors ( $c_1 = 2.5, c_2 = 1.0$ ).

## 6. 検討・考察：原稿作成指導への活用

本節では、6.1 項にてチェンジポイントが発生しない場合における指導例を述べ、6.2 項にてチェンジポイントが発生する場合における指導例を述べる。

### 6.1 チェンジポイントが発生しない場合

指数形修正スキルの場合には図 4 より、総エラー発見数が増加するのは速いが時間とともにエラー発見数が減少することが分かる。そのため、エラー発見数が減少してから一定量の修正を促進するような指導が必要となる。例えば、共同学習やグループ学習の機会を提供する、または研究室に属する学生同士の相互添削を義務化するなどによって一定量の修正を促進することが有効である。

線形修正スキルの場合には図 5 より、学生のエラー発見率が低いとエラー発見数が少ない期間が続くことが分かる。このエラー発見率が低い要因としていくつか考えられる。例えば、学生は自己添削を十分おこなわず指導教員に提出していることである。教員から指摘をうけても、どのような視点や根拠によって添削・修正されているか、まったくその意味合いを考えるとなく、さらには単に添削されたとおりに文章を修正するだけでは学生の能力は向上しない。そこで、技術文章チェックリストを配布するなどをおこない、文章の自己添削を徹底させることが重要となる [11]。

### 6.2 チェンジポイントが発生する場合

指数形修正スキルの場合には図 6 より、原稿提出期限直前にチェンジポイントが発生し、エラー発見率が向上したとしてもエラーがすべて発見できるとは限らないことが分かる。すなわち教員は、学生の取り組む時間とチェンジポイント後の学生のエラー発見率意識し、原稿提出期限直前に完成するようなチェンジポイントを意図的に発生することとなる。意図的にチェンジポイントを発生させる方法としては、学生が原稿を学生とともに点検・添削することがあげられる。また図 7 より、原稿作成プロセス初期にチェンジポイントが発生し、学生のエラー発見率が低下したとしても原稿提出期限まで時間がある場合は、高い確率でエラーがすべて発見できると分かる。そのため原稿提出期限まで時間がある場合、教員は具体的な指摘をせずに原稿の確認のみで学生の創造的解決や創造的活動を認めることによって学生の能力向上を促進できる可能性を示唆している。

線形修正スキルの場合には図 8・図 9 より、指数形修正スキルに比べチェンジポイントの影響が大きいと分かる。これは指数形修正スキルは初期にエラーの発見が多いのに対して、線形修正スキルは時間によらずエラーが一定に発見されるためである。そのため、指導においてもチェンジポイントの影響をより考慮する必要がある。例えば、指導においては言語情報とともに絵や図などを掲示する、または

考える時間を提供することでエラー発見数が少ないときの理解度を補助することが有効となる。また、エラー発見率が低下する要因としてエラー発見難易度の変化、学生の意欲の低下が考えられるため、現在の状態を明確に把握しつつ、過去のフィードバックを迅速に反省させるような心構えを指導することが重要である。

## 7. おわりに

本稿では、学生が原稿作成をおこなう機会を能動的な学習として捉え、学生の修正スキルによる原稿作成指導手法についてソフトウェア信頼度成長モデルを応用した NHPP モデルを構築し、原稿作成プロセスにおけるエラー発見数を表現した。また、チェンジポイントを考慮することによって、より現実的なモデルとし、シミュレーションにより原稿作成指導手法についての検討もおこなった。

今後の課題としては、原稿作成プロセスにおいて発見されたエラーは必ずしも完全にかつ確実に修正・除去されるとは限らないということを考慮する必要がある。本稿においては、提案モデルを構築する際の仮定として、検出されたエラーは完全に除去できるものとした。これはソフトウェア信頼性評価モデルにおける完全デバッグモデルと同様の仮定である。しかし、これは原稿作成プロセスにおいても非現実的であり、発見されたエラーは必ずしも完全にかつ確実に修正・除去されるとは限らない。そのため、ソフトウェア信頼性評価モデルにおける不完全デバッグ環境と同様の環境を考慮する必要がある。

## 参考文献

- [1] 「社会人基礎力」とは - 経済産業省, 入手先 ([http://www.meti.go.jp/policy/kisoryoku/kisoryoku\\_image.pdf](http://www.meti.go.jp/policy/kisoryoku/kisoryoku_image.pdf)) (2014.06.30).
- [2] 畝田 道雄, 石川 憲一, “論文作成指導の高効率化を目的とした技術文章作成能力向上のための一試み”, 工学教育 vol. 56, no.1, pp.68-71, 2008.
- [3] 稲積 宏誠, 大野 博之, “ICT を活用した日本語文章力育成への取り組み：問題点と展望”, 情報科学技術フォーラム講演論文集 vol. 10, no.4, pp.465-466, 2011.
- [4] 犬塚美輪, 大道一弘, 川島一通, “論理的コミュニケーションの評価枠組みと評価支援システムの開発”, 日本教育工学会論文誌 vol. 37, no.1, pp.67-77, 2013.
- [5] 井上真二, 山田茂, “テスト工程におけるチェンジポイントとソフトウェア信頼性評価モデルに関する一考察”, 情報処理学会研究報告. ソフトウェア工学研究会報告, vol. 158, no.8, pp.55-62, 2007.
- [6] 山田茂, 『ソフトウェア信頼性の基礎 - モデリングアプローチ』, 共立出版, 東京, 2011.
- [7] 山田茂, 『ソフトウェア信頼性モデル-基礎と応用』, 日科技連出版社, 1994.
- [8] 木下是雄, 『理科系の作文技術』, 中公新書, 1981.
- [9] 小山透, 『科学技術系のライティング技法』, 慶應義塾大学出版会株式会社, 2011.
- [10] Mesquite Software, <http://www.mesquite.com> (2013.04.09).
- [11] 畝田道雄, 石川憲一, “論文作成指導の高効率化を目的とした技術文章作成能力向上のための試みと有効性評価”, 工学教育 vol. 58, no.2, pp.58-63, 2010.