

ソフトウェア開発者の性別がプログラム理解速度に及ぼす影響の 予備分析

高塚 由利子^{†1} 村上優佳紗^{†1,a)} 角田雅照^{†1,b)} 中村匡秀^{†2,c)}

ソフトウェア開発において、個々のソフトウェア開発者の能力が開発プロジェクトの結果、例えば開発総工数などに与える影響は無視することができない。開発者の能力は個人により大きく異なることがいくつかの研究で指摘されている。開発者の能力差を小さくすることができれば、プロジェクトの生産性などを改善できる可能性がある。そのためには、能力差に関連する要因を明らかにし、その影響を小さくするためのソフトウェア開発支援ツールなどを用意する必要がある。本研究では、能力差に関連する要因として、開発者の性別に着目する。女性の場合、短期記憶、長期記憶とも男性よりも優れると指摘されている。そこで開発者が女性の場合、理解のために記憶力を多く必要とするプログラムでは、理解速度が速まるかを実験により確かめた。その結果、理解のために記憶力を必要とするプログラムかどうかにかかわらず、性別による読解速度の違いは小さかった。

キーワード： プログラマ、開発スキル、コードリーディング

1. はじめに

ソフトウェア開発において、個々のソフトウェア開発者の能力が開発プロジェクトの結果、例えば開発総工数などに与える影響は無視することができない。例えば、開発工数見積もりに広く用いられる COCOMO II モデル[1]では、プログラマのスキルがモデルの要因に含まれている。これは、プログラマのスキルが異なる場合、その他の条件が同じ場合でも、プロジェクトの開発工数が変化し、すなわち生産性（開発規模÷開発工数で定義される）が変化することを示している。

ソフトウェア開発者の能力は個人により大きく異なることが、いくつかの研究で指摘されている[9][10][12]。例えば、Sackman らの研究において、プログラムのデバッグ能力が、個人により大きな差があることが示されている[9]。同様に、Thelin らの研究においても、個人によりコードレビュー能力の差が大きいことが示されている[10]。もし開発者の能力差を小さくする、すなわち、能力が比較的低い開発者について、ある程度能力を高めることができれば、ソフトウェア開発プロジェクトの生産性などを改善できる可能性がある。

開発者の能力差を小さくするためには、能力差に影響する要因を明らかにし、その要因の影響を小さくするためのソフトウェア工学教育やソフトウェア開発支援ツールを用意する必要がある。例えば、個人によりコードレビュー能力に差があり、その原因が仕様と実装コードの対応をチェックしていないことが原因である[7]とする。この場合、「仕様と実装コードの対応をチェックする」ことをレビューの

チェックリストに追加することにより、レビューの能力差を小さくすることができる。ソフトウェア開発者への支援は、これまでも取り組まれてきているが、開発者ごとの特性の違い（年齢、性別など）は考慮せずに、一律に同じ方法で支援しているツールなどが多い。

本研究では、開発者ごとの特性の違いを分析し、それに基づいて開発作業を支援することを目指す。このような個別の開発者の特性に基づいて作業結果を分析した研究は非常に少ない。個人の特性に着目した研究はこれまで少なく、ここ 10 年ほどで少しずつ増えつつある[13]。また、心理学や行動経済学などの知見が、これまでのソフトウェア工学の研究では十分に活用されておらず、今後の研究分野の発展のために、それらを参照すべきであるという提言もある[6]。本研究でもそれらの知見を利用して分析する。

本研究では、開発支援のための予備的分析として、プログラム理解について、開発者個人の特性の影響を分析する。開発者の特性として性別に着目する。プログラムの読解には記憶力が必要とされる場合があるが、女性の場合、男性よりも短期記憶、長期記憶ともに優れていると指摘されており、いくつかの研究結果も存在する（[4]など）。また、他分野では男性医師と女性医師の手術後の予後を比較した研究があり[11]、ソフトウェア工学でも、GitHub などのデータを用いて性別に着目した分析が存在する[2]。ただし、性別と開発者の能力の関連について、実験により直接確かめた研究は、我々の知る限り存在しない。

分析では、性別と関連する開発者の能力は、記憶力であると仮定する。そして、女性開発者の場合、理解のために記憶力をより多く必要とするプログラムでは、そうでないプログラムと比較して理解速度が低下するかどうかを実験により確かめる。あるプログラムが、理解のためにどの程度記憶力を必要とするかを定量的に計測するために、本研究では文献[5][7]において提案されているプログラム理解容易性評価尺度を用いる。これらの尺度はプログラムの理

†1 近畿大学理工学部
Faculty of Science and Engineering, Kindai University, Japan

†2 神戸大学大学院工学研究科
Graduate School of Engineering, Kobe University, Japan

a) m.yukasa@gmail.com

b) tsunoda@info.kindai.ac.jp

c) masa-n@cs.kobe-u.ac.jp

解容易性を、プログラムの複雑度などではなく、理解に必要とする記憶力の多寡に基づいて評価している、すなわち、これらの尺度により理解が容易でないと評価されるプログラムは、理解のために記憶力がより多く必要であることを示している。

2. プログラム理解容易性評価尺度

プログラムの理解容易性を評価するために、文献[5][7]では、人間の記憶力に基づく尺度を提案している。これらの尺度では、プログラムを理解するためにメンタルシミュレーション[3]が行われることを前提としている。メンタルシミュレーションとは、プログラムの動作をコンピュータや筆記具を用いずに、思考しながら理解することであり、比較的規模の小さなコード片に対して用いられる。メンタルシミュレーションを行う場合、変数の値を記憶しておく必要があるが、多数の変数の値を同時に記憶しておくことは容易ではない。このため文献[5][7]では、多数の変数の値を記憶しておく必要のあるプログラムの場合、メンタルシミュレーションのコストが高くなり、プログラムの理解容易性が低下すると仮定している。

文献[7]では、人間の短期記憶を FIFO キューとして、メンタルシミュレーションの仮想モデル (Virtual Mental Simulation Model; VMSSM) を作成し、そのモデルに基づいて理解容易性評価尺度を定義している。基本的なアイデアは、ある変数の値を参照する際、(大きさに制限のある) キューに変数の値が記憶されている場合はコストが小さくならないとしている。逆にキューに値が記憶されていない場合、その値の変更箇所までバックトラックする必要が生じるため、コストが大きくなるとしている。

文献[7]では、VMSSM に基づき、以下の4つの評価尺度を定義している。

- ASSIGN: 変数代入に関するコスト。
- RCL: 短期記憶内の変数を思い出すコスト。
- BT_CONST: 定数をバックトラックする回数。短期記憶にない定数を得るコスト。
- BT_VAR: 変数のバックトラックの距離。短期記憶にない変数を得るコスト。

文献[5]では、上記のメトリクスが変数の更新回数に基づく再計算コストを考慮していないことと、バックトラックに関するメトリクスがプログラムの行の入れ替えに敏感すぎるのが問題点であるとし、新たな評価尺度を2つ提案している。具体的には、各変数の値の更新回数をベクトルの要素とし、ベクトルの要素の和に基づくメトリクス SUM_UPD とベクトルの要素の分散に基づくメトリクス

<pre>int i, t; t = 11; t = t - 1; i = 2; if(i < t){ i = i + 2; if(i < t){ i = i + 2; } if(i < t){ i = i + 2; if(i < t){ i = i + 2; } } if(i < t){ i = i + 2; } } System.out.println("i = " + i); (a0)</pre>	<pre>int i, t; t = 11; t = t - 1; i = 2; if(i < t){ t = t - 2; if(i < t){ i = i + 2; } if(i < t){ t = t - 2; if(i < t){ i = i + 2; } } if(i < t){ i = i + 2; } } System.out.println("i = " + i); (a1)</pre>
---	---

図 1 プログラム a0, a1[5]

<pre>int a, b, c, d, e, f, g; a = 2; b = 4; c = 3; d = 6; c = c + 4; d = d - 2; if(c < 5) e = d + 5; else e = d + 3; a = a * 2; b = b + 6; if(a > 7) f = b - 3; else f = b - 5; if(c < 5) e = d + 5; else e = d + 3; g = e + f; System.out.println("g = " + g); (b0)</pre>	<pre>int a, b, c, d, e, f, g; a = 2; b = 4; c = 3; d = 6; c = c + 4; d = d - 2; a = a * 2; b = b + 6; if(a > 7) f = b - 3; else f = b - 5; if(c < 5) e = d + 5; else e = d + 3; g = e + f; System.out.println("g = " + g); (b1)</pre>
--	--

図 2 プログラム b0, b1[5]

SUM_VAR を定義している。

3. 実験

3.1 概要

実験の目的は、女性のソフトウェア開発者の場合、理解のために記憶力をより多く必要とするプログラムを読む場合でも、そうでないプログラムと比較して理解速度が低下しにくいかどうかを確かめることである。そのために、必要とする記憶力が異なる、複数のプログラムを用意し、性別の異なる被験者がコードを理解するために掛かった時間を計測した。

表 1 各プログラムの理解容易性[5]

プログラム	ASSIGN	RCL	BT_CONST	BT_VAR	SUM_UPD	VAR_UPD
a0	12	6	0	80	7	1.25
a1	12	6	0	48	7	0.25
b0	18	8	1	30	11	0.24
b1	18	6	1	54	11	0.24

必要とする記憶量の異なるプログラムは、石黒らの研究[5]で示されているもの4つ(a0, a1, b0, b1. 図1, 図2参照)を利用した。各プログラムは20行から30行の規模である。あらかじめ指定された変数の値が、プログラム実行後にどうなるかプログラムを読んで答え、それが正しかった場合、プログラムを理解できたとした。例えばプログラムa0の場合、プログラム実行後の変数iの値を答えさせた。プログラムの理解はメンタルシミュレーションにより行うこととし、メモなどは利用させないようにした。

各プログラムで理解のために必要とする記憶力の多寡については、2章で説明した6つのメトリクス(ASSIGN, RCL, BT_CONST, BT_VAR, SUM_UPD, VAR_UPD)に基づき評価した。これらのメトリクスに基づく各プログラムの理解容易性を表1に示す(石黒らの研究[5]からの引用)。ASSIGN, BT_CONST, SUM_UPDより、プログラムb0, b1を理解するためには、比較的記憶力が必要とされることわかる。

被験者を男性グループと女性グループに分け、それぞれのグループの回答時間の平均値や中央値などを算出し比較した。被験者は、近畿大学理工学部情報学科に所属する学部生16名(男性8名, 女性8名)である。

プログラムを読む順番が実験結果に影響することを避けるために、4つのプログラムを読む順番を被験者ごとに変更した。例えばある被験者ではプログラムをa0, a1, b0, b1の順に読むとし、別の被験者ではb1, a0, b0, a1の順に読

むなどとした。

分析にあたり、以下の2つのリサーチクエスチョンを設定した。

- RQ1: 女性グループと男性グループでは、どちらがプログラムを理解する速度が速いのか?
- RQ2: 理解のために記憶力を必要とするプログラムの場合、女性グループのほうがプログラムを理解する速度が速いのか?
- RQ3: 女性グループは、記憶力を必要とするプログラムとそうでないプログラムの理解速度の差が小さいのか?

3.2 実験用ツール

実験のために、問題を出題し、回答時間や誤回答の回数を計測するためのツールを作成した。図3にツールのスクリーンショットを示す。本ツールは表計算ソフトのマクロを用いて開発した。ツールの動作を以下に示す。

1. 「回答する」をクリックすると問題とテキストボックスを表示する。
2. 問題に正解するまでテキストボックスを表示し続ける。
3. テキストボックス(問題)が表示されてから正答するまでの回答時間と誤回答の回数を記録し、被験者にも表示する。

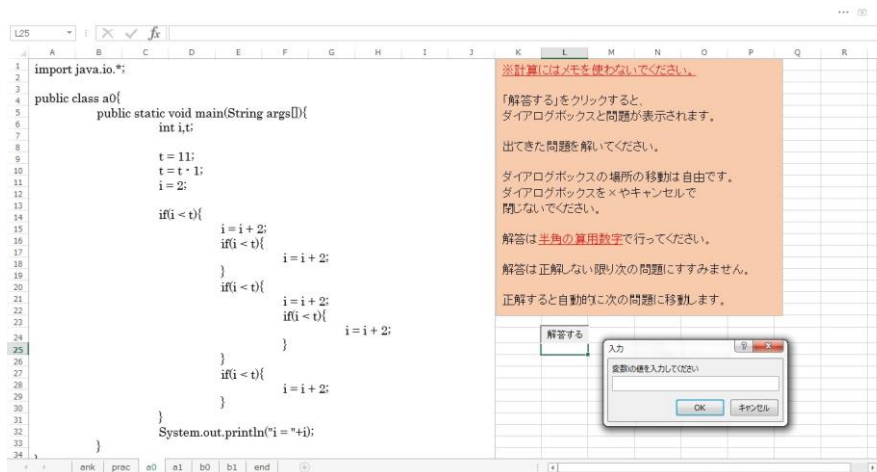


図 3 実験用ツールのスクリーンショット

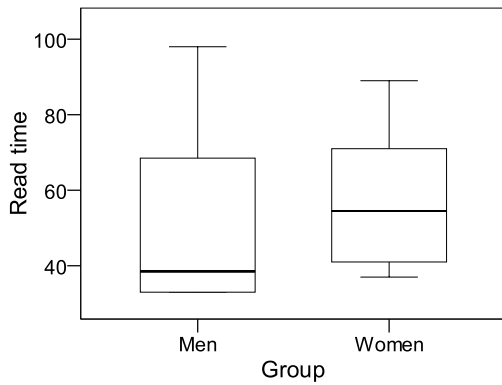


図 4 プログラム a0 の回答時間

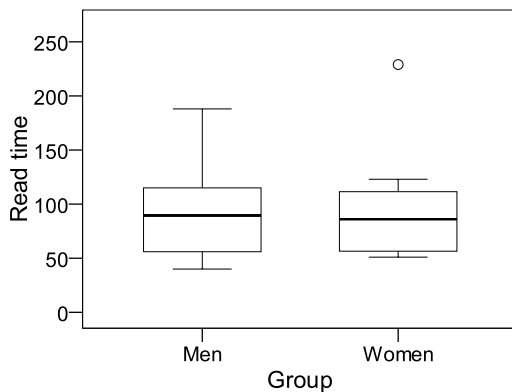


図 5 プログラム a1 の回答時間

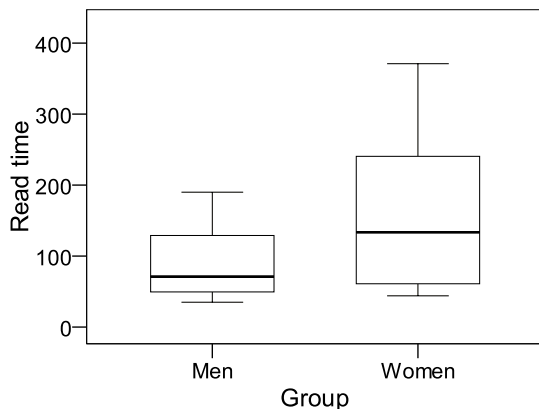


図 6 プログラム b0 の回答時間

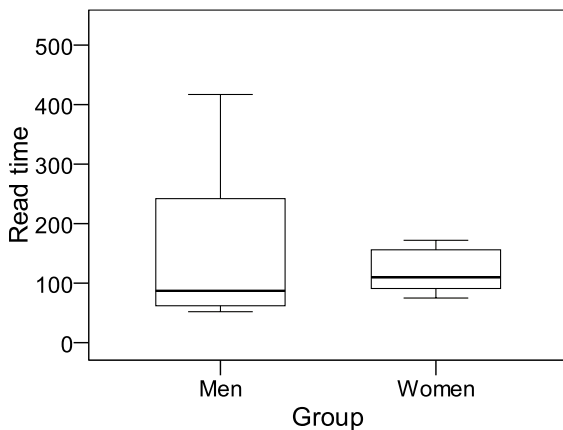


図 7 プログラム b1 の回答時間

表 2 各グループのプログラム別回答時間 (秒)

		a0	a1	b0	b1
男性	平均値	51.4	93.6	90.5	156.4
	中央値	38.5	89.5	71.0	87.0
	標準偏差	25.5	48.2	55.4	141.4
女性	平均値	57.4	98.5	160.6	120.1
	中央値	54.5	86.0	133.5	110.0
	標準偏差	18.8	58.3	114.7	37.6

表 3 各グループの回答時間の集計値 (秒)

	男性	女性	女性 / 男性
平均値	98.0	109.2	111.4%
中央値	73.0	86.0	117.8%
標準偏差	85.7	74.7	87.2%

表 4 プログラム a0 との回答時間の比

		a1 / a0	b0 / a0	b1 / a0
男性	平均値	1.83	1.68	2.61
	中央値	1.97	1.83	2.29
	標準偏差	0.45	0.31	1.03
女性	平均値	1.63	2.51	2.11
	中央値	1.52	2.47	2.10
	標準偏差	0.40	1.11	0.17

表 5 各グループの回答時間の比の集計値 (秒)

	男性	女性	女性 / 男性
平均値	2.04	2.08	102%
中央値	1.92	1.98	103%
標準偏差	0.77	0.75	98%

4. 結果

4.1 回答時間の分析

表 2 に男性、女性各グループのプログラム別の回答時間の平均値、中央値、標準偏差を示す。時間が短いグループのセルをグレーで表している。プログラム a0, b0 (b0, b1 が記憶力を必要とする) では、平均値、中央値とも男性グループのほうが小さかったが、プログラム a1 では女性グループの中央値が小さく、b1 では平均値が女性グループのほうが小さかった。すなわち、プログラムによって結果が異なり、必ずしも一方のグループの時間が短いとはいえない。

それぞれのグループの回答時間の分布を、箱ひげ図を用いて図 4 から図 7 に示す。a0 では箱の位置は両グループで差がないが、男性グループの中央値のほうが低い。a1 では箱の位置、中央値ともほとんど差がない。b0 では男性グループの箱の位置、中央値とも低かった。b1 では中央値は若干男性の方が低かったが、箱の大きさは男性の方が大き

4. 正解すると次の問題に自動的に遷移する。

かった。図からもどちらかのグループが常に時間が短いとはいえない。

全てのプログラムの回答時間を区別せずに集計した場合の基本統計量を表 3 に示す。プログラムにより結果が大きく異なるため参考にとどめるべきであるが、男性のほうが 10%ほど時間（平均値，中央値）が短く，女性の方がデータのばらつき（標準偏差）が若干小さかった（表の「女性 / 男性」の列参照）。

これらの結果より，RQ1 に対する答えは「プログラムによって傾向が異なり，必ずしも一方が速いとはいえない」，RQ2 に対する答えは「プログラムによって傾向が異なり，女性グループのほうが速いとはいえない」となる。

次に，RQ3 に答えるために，比較的記憶力を必要としないプログラム a0 と比べ，何倍の回答時間が掛かっているかを調べた。具体的には，プログラム a0 以外の回答時間÷プログラム a0 の回答時間を求めた。この値が大きいくほど，a0 と比較して回答時間が多く掛かっていることを示す。結果を表 4 に示す。プログラム a1, b1 では，女性グループのほうが比の平均値，中央値とも小さかったが，プログラム b0 では男性グループのほうが小さかった。回答時間の変化に着目した場合でも，プログラムによって結果が異なり，どちらかのグループの時間が短いとはいえなかった。

全プログラムの回答時間の比を区別せずに集計した場合の基本統計量を表 3 に示す。この場合，各グループによる回答時間の比にほとんど差がなかった。これらの結果から，RQ3 に対する答えは「プログラムによって傾向が異なり，差が小さいとはいえない」となる。

RQ に関する分析を別の観点で行うために，被験者ごとに回答時間を正規化して比較した。正規化は(回答時間 - 最小回答時間)÷(最大回答時間 - 最小回答時間)により行った。箱ひげ図を図 8, 図 9 に示す。正規化を行うと，各被験者に最も速かった場合が 0，最も遅かった場合が 1 になる。すなわち，男性グループでは，プログラム b1 が最も遅い場合が多く，女性グループでは a0 が最も速かった。女性グループは a0 を除き，分布は比較的似ており（均等に広い），被験者による傾向の違い比較的大きいといえるが，特にリサーチクエスチョンに関連して，着目すべき特徴は見られなかった。

誤回答数の基本統計量を表 6 に示す。プログラム b0 については男性グループの誤回答数が多く，b1 では女性グループの誤回答数が多かったが，これらはそれぞれのグループで回答時間が長いプログラムであり，難易度がそのまま反映されていると考えられる。女性グループでは，a0 を除き中央値が 1 を超えていた。このことから，誤回答の傾向は両グループで特に大きな差はないが，少なくとも女性グループの誤回答数が少ない傾向にあるとはいえない。

4.2 アンケートによる分析

実験後のアンケートで出題したプログラムについてア

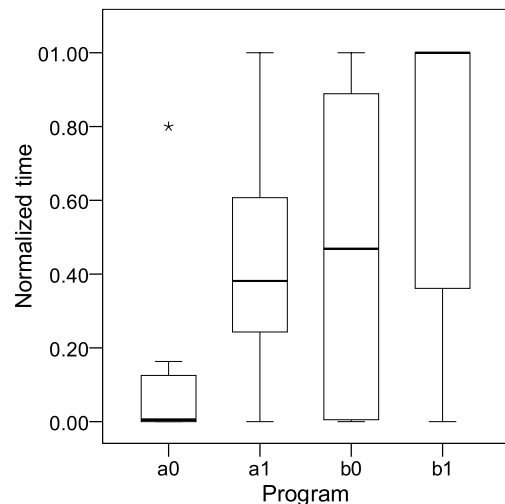


図 8 男性グループの回答時間正規化

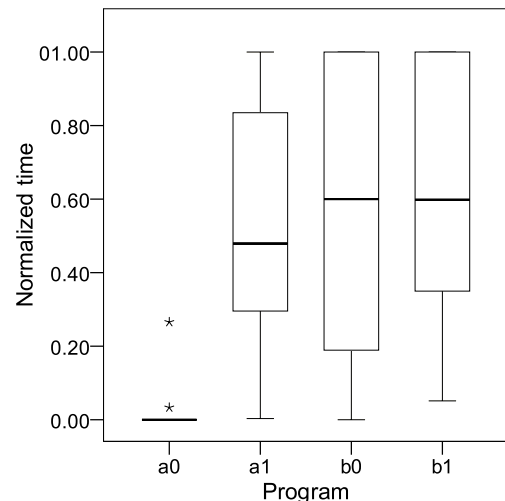


図 9 女性グループの回答時間正規化

表 6 誤回答数の基本統計量

		a0	a1	b0	b1
男性	平均値	0.38	0.50	0.38	1.88
	中央値	0.00	0.00	0.00	0.50
	標準偏差	0.74	1.07	0.74	3.00
女性	平均値	0.38	1.13	1.63	1.13
	中央値	0.00	1.00	1.00	1.00
	標準偏差	0.52	1.64	1.85	1.36

ンケートを行った。その内容は以下の 2 つである。

- I. 1 問目と 3 問目について，具体的にどの部分がどのように変更されたかについて気づいたか。
- II. 2 問目と 4 問目について，具体的にどの部分がどのように変更されたかについて気づいたか。

回答は以下の 3 つとした。

1. 気づいたうえで早く読めたと思う

表 7 アンケート結果

	問 I			問 II		
	1	2	3	1	2	3
男性グループ	50%	25%	25%	63%	25%	13%
女性グループ	38%	63%	0%	63%	25%	13%

2. 気づいたが早く読むのに役立たなかった
3. 気づかなかった

アンケート結果を表 7 に示す。問 I では、男性グループにおいて「1: 気づいたうえで早く読めたと思う」と回答した被験者が多かった。問 II では回答の比率は同一であった。この結果から、少なくとも「女性グループのほうが記憶力を有効に活用している」とはいえない。

5. おわりに

本研究では、ソフトウェア開発者のコード理解速度と関連する要因として、開発者の性別に着目した。分析では、記憶力が性別によって異なると仮定し、理解のために記憶力を必要とするプログラムとそうでないプログラムについて、性別によりプログラムの理解速度が異なるかどうかを実験により確かめた。実験の結果より、以下が明らかとなった。

- プログラムを理解する速度は、男性グループと女性グループのどちらかが速いとはいえない。
- 理解のために記憶力を必要とするプログラムに関して、男性グループと女性グループのどちらかが速いとはいえない。
- 女性グループに関して、記憶力を必要とするプログラムでは、そうでないプログラムと比較して理解速度が低下しにくいとはいえない。

今後の予定は、被験者をさらに増やして、開発者の性別とコード理解速度との関連の分析結果の信頼性を高めることである。

謝辞 本研究の一部は、日本学術振興会科学研究費補助金（基盤 C: 課題番号 16K00113, 基盤 A: 課題番号 17H00731）による助成を受けた。

参考文献

[1] Boehm, B., Clark, Horowitz, Brown, Reifer, Chulani, Madachy, R. and Steece, B.: Software Cost Estimation with Cocomo II, Prentice Hall PTR (2000).

[2] Burnett, M., Fleming, S., Iqbal, S., Venolia, G., Rajaram, V., Farooq, U., Grigoreanu, V., and Czerwinski, M.: Gender differences and programming environments: across programming populations, Proc. of International Symposium on Empirical Software Engineering and Measurement (ESEM), article 28, 10 pages (2010).

[3] Dunsmore, A., Roper, M. and Wood, M.: The role of

comprehension in software inspection, Journal of Systems and Software, vol.52, no.2-3, pp.121-129 (2000).

[4] Hill, A., Laird, A., and Robinson, J.: Gender differences in working memory networks: a BrainMap meta-analysis, Biological psychology, vol.102, pp.18-29 (2014).

[5] 石黒誉久, 井垣宏, 中村匡秀, 門田暁人, 松本健一: 変数更新の回数と分散に基づくプログラムのメンタルシミュレーションコスト評価, 電子情報通信学会技術報告, SS2004-32, pp.37-42(2004).

[6] Jørgensen, M.: What can - and should - empirical software engineering learn from empirical studies in psychology? Empirical Software Engineering and Measurement (ESEM) (2018).

[7] 栗山進, 大平雅雄, 門田暁人, 松本健一: プログラム理解度がコードレビュー達成度に及ぼす影響の分析, 電子情報通信学会技術報告, SS2004-53. pp.17-22 (2005).

[8] Nakamura, M., Monden, A., Satoh, H., Itoh, T., Matsumoto, K., and Kanzaki, Y.: Queue-based Cost Evaluation of Mental Simulation Process in Program Comprehension, Proc. of International Software Metrics Symposium, pp.351-360 (2003).

[9] Sackman, H., Erikson, W. and Grant, E.: Exploratory experimental studies comparing online and offline programming performance, Communications of the ACM, vol.11, no.1 (1968).

[10] Thelin, T., Andersson, C., Runeson, P., Dzamashvili-Fogelström, N.: A Replicated Experiment of Usage-Based and Checklist-Based Reading, Proc of International Symposium on Software Metrics, pp.246-256 (2004).

[11] Tsubawa, Y., Jena, A., Figueroa, J., Orav, E., Blumenthal, D., and Jha, A.: Comparison of Hospital Mortality and Readmission Rates for Medicare Patients Treated by Male vs Female Physicians, Journal of the American Medical Association, vol.177, no.2, pp.206-213 (2017).

[12] Uwano, H., Nakamura, M. Monden, A. and Matsumoto, K.: Exploiting Eye Movements for Evaluating Reviewer's Performance in Software Review, IEICE Transactions on Fundamentals, vol.E90-A, no.10, pp.317-328 (2007).

[13] Weber, B.: Bringing the Human into the Loop, International Conference on Product-Focused Software Process Improvement (Profes) (2017).