

## 因果推論に基づく開発プロセス評価に関する考察

夏目 珠規子<sup>†</sup> 野々村 琢人<sup>†</sup> 艸薙 匠<sup>†</sup> 小島 昌一<sup>\*\*</sup> 藤田 和也<sup>\*\*\*</sup>

株式会社東芝<sup>†</sup> 東芝ソフトウェア・コンサルティング株式会社<sup>\*\*</sup> 東芝トレーディング株式会社<sup>\*\*\*</sup>

### 1. はじめに

プロジェクトを成功に導くためには、「技術・人・プロセス」の3要素が大きな影響を及ぼす。そのうちの一つである「プロセス」の評価を行うためには、プロセス診断モデルであるCMMI<sup>®</sup>が一つの指標として用いられる。CMMI<sup>®</sup>とは組織の開発プロセスの成熟度(ML: Maturity Level)を5段階で表現したモデルである。従来のプロセス評価では、CMMI<sup>®</sup>等のモデルを用い、専門家がプロセスの強み弱みを検出する方法がとられてきた。しかし評価は時間・コスト制約から調査対象の開発例を多く網羅することは難しい。また、評価は定性的な評価であり、開発プロセスの中身(質)を示すデータを用いて定量的な評価が行われているとは限らない。そこで、開発中に収集されたデータから定量的に組織の開発プロセス成熟度を評価する方法を提案し、事例を元に評価方法の検証を行った結果を述べる。

### 2. 因果推論とは

データを活用して物事の間接関係を把握するためには、一般的にデータの相関を確認する。しかし、相関があるからと言って、原因と結果の間接関係を把握できているわけではない。因果関係とその効果をデータ間の関係から抽出するための理論としては因果推論がある。ソフトウェア開発プロセスにおいては、プロセス間の原因と結果が複雑に絡み合っており、相関関係から因果関係を読み解くには、組織の背景事情やプロセス管理の専門知識を必要とする。そこで、本論文ではプロセス間の因果関係を把握するために、因果推論の理論を用いる。因果推論によって構築された因果モデルは、プロセスの関連を表現しており、「成熟した組織であるほど理想の因果関係に近づくはずである」という仮説を立てている。

### 3. 因果推論に基づく開発プロセスの評価

2章の仮説の下、定量的にプロセスを評価する

Study on the evaluation of the organization maturity level based on causal inference

<sup>†</sup>Mikiko Natsume, Takuto Nonomura, Takumi Kusanagi, Corporate Software Engineering Center Toshiba Corporation.

<sup>\*\*</sup>Shoichi Kojima, Toshiba Software Consulting Corporation.

<sup>\*\*\*</sup>Kazuya Fujita, Toshiba Trading Inc.

方法を提案する。次に評価ステップを示す。

(1) ソフトウェア開発における理想的な因果モデルを定義する。

開発中に収集できる各工程での基本的なメトリクスをもとに、理想的な因果モデルを定義する。メトリクス間の因果は、作業の時間的な順序関係とメトリクス間の関連性から決定する。

(2) 開発データをもとに因果モデルを構築する。

ベイジアンネットワークの構造学習アルゴリズムを活用し、評価対象であるプロセスの因果モデルを構築する。

(3) (1)で定義した理想的なモデルと(2)で構築されたモデルを比較し、プロセスを評価する。

2つのモデルを比較することで、メトリクス間の因果の破綻を捉え、より理想的な因果モデルに近いかによってプロセスを評価する。因果の破綻のタイプは、因果関係が矛盾する「逆転」、2つの間に対応がなくなる「消失」、時間的/意味的な矛盾はないが新しく関係が発生する「順方向の発生」、矛盾ありの「逆方向の発生」の4つに分類できる。

### 4. 検証事例

本論文で提案する方法は、因果関係の破綻が起きている場合はプロセスが適切でないという仮定のもとに成り立つ。この仮説が正しいかどうかを検証するために、CMMI<sup>®</sup>のML2とML3相当の両組織のデータをもとに因果モデルを構築し、検証を行った。検証の際には、開発ステップ数、単体・結合検査数/不具合数、総合検査数/不具合数、遅れ日数の6つのメトリクスを定義し、理想的な因果モデルを図1のように定義した。

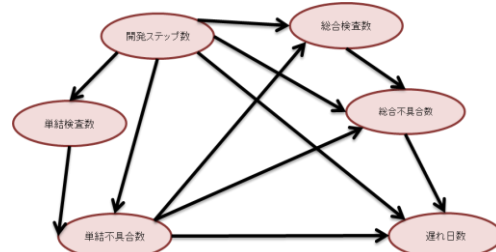


図1 ソフトウェア開発における理想的な因果モデル  
続いて、データをもとに因果モデルを構築する。モデルの構築には統計ソフト R のベイジアンネットワーク deal パッケージを用いた。ML2 と

ML3 相当の組織の因果モデルを図 2, 図 3 に示す。

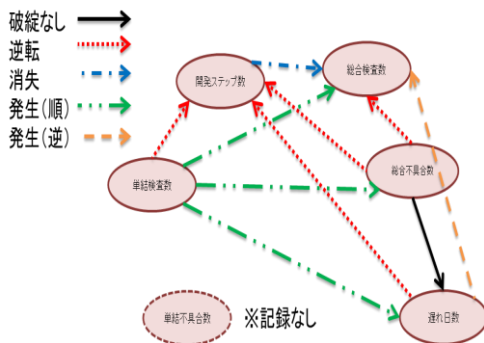


図 2 ML2 相当の組織の因果モデル

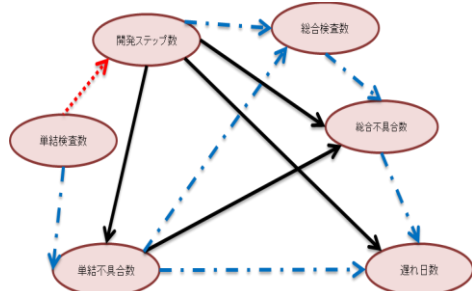


図 3 ML3 相当の組織の因果モデル

図 2 を理想的な因果モデル (図 1) と比較すると、正しい因果が把握できるのは以下の 1 件だけである。

- ・ [総合不具合数] が [遅れ日数] の原因
- 一方、図 3 を理想的なモデル (図 1) と比較すると、以下の 4 件の正しい因果が把握できた。
- ・ [開発ステップ数] が [単結不具合数] の原因
  - ・ [開発ステップ数] が [遅れ日数] の原因
  - ・ [開発ステップ数] が [総合不具合数] の原因
  - ・ [単結不具合数] が [総合不具合数] の原因

上記の通り、ML2 と ML3 の組織の結果を比較すると、成熟度レベルが高い組織の方が理想的な因果モデルに近いことが分かる。また、因果の破綻のタイプの重大性に応じて重みを設定することで、理想モデルとの近さを定量化できる。例えば、表 1 のように重み付けをすると以下の通り定量化できる。

表 1 因果破綻のタイプと辺の数

タイプ	重み	辺の数	
		ML2 相当	ML3 相当
逆転	10	4	1
消失	5	1	6
発生 (順)	3	3	0
発生 (逆)	7	1	0

100 点を基準とした時のプロセススコア：

$$100 - \Sigma (\text{重み} \times \text{辺の数})$$

ML2 の組織のプロセススコア：39 点

ML3 の組織のプロセススコア：60 点

上記の結果は、本論文で扱う「成熟した組織であるほど理想の因果関係に近づくはずであ

る」という仮説をサポートしている。また、この検証結果を組織の状況と比較してみると以下の通り考察できる。

＜ML2 相当の組織の特徴＞

- ・ 上流のプロセスが弱く、出荷間際の仕様変更や不具合修正による後戻りが頻発。

＜ML3 相当の組織の特徴＞

- ・ 上流での品質確保に力を入れた改善活動が行われているため、下流での後戻りが少ない。

ML2 相当の組織では後戻りの発生が因果モデルで表現されており、理想とは逆の因果関係が検出されているのに対し、ML3 相当の組織は上流で対策が行われていることで理想的な因果モデルに近い結果となった。

## 5. まとめ

本論文では原因と結果が複雑に絡み合うソフトウェア開発プロセスにおいて、プロセスを特徴づけるメトリクスを定義することでメトリクス間の因果関係を明らかにし、定量的にプロセスを評価するためのアイデアを紹介した。また、データを用いて本アイデアの仮説を検証した結果、組織のプロセス成熟度レベルが高いほど、理想的な因果モデルに近づくことが確認された。本論文で提案する評価方法を用いることで、組織間、改善の前後など複数のプロセスの能力を相対的に評価することが可能になり、データによってプロセスの中身 (質) を客観的に評価することができる。さらに、因果の破綻が起きているメトリクス間を特定することで、組織の弱みの候補が抽出でき、改善箇所の選定に役立てることもできる。今後は、データでの検証を進めるとともに、点数付けの方法についても検討を進めていきたい。また、因果関係を明らかにすることで、改善の結果と施策の関係も明確になり、改善の効果の把握がしやすくなるため、プロセス改善の効果測定の仕組みについても検討していきたい。

## 参考文献

- [1] 宮川雅巳：統計因果推論-回帰分析の新しい枠組み、朝倉書店、2004。
- [2] 古山恒夫：“ソフトウェアプロジェクトにおけるリスクの因果関係の分析”，情報処理学会研究報告，p. 47-54，2001。
- [3] 板橋吉徳，落水浩一郎：“ソフトウェア開発組織のプロセス特性とソフトウェア品質との因果関係の定量的な推定法”，電子情報通信学会技術報告，p. 1-4，2010。
- [4] M. Chrissis, M. Konrad, S. Shrum: “開発のための CMMI®”，2010 カーネギーメロン大学 CMU/SEI-2010-TR-033, 2010, CMMI, CMM Integration and Capability Maturity Model are registered in the U.S. Patent and Trademark Office.