

プロセスマイニングにおけるビジネスプロセスモデルの自動修正に対応したゴールモデルの修正手法

堀田 大貴[†] 平山 秀昭^{††} 早瀬 健夫^{†††} 田原 康之^{†††} 大須賀昭彦^{†††}

[†] 茨城大学 工学部 〒316-8511 茨城県日立市中成沢町 4-12-1

^{††} 目白大学 メディア学部メディア学科 〒161-8539 東京都新宿区中落合 4-31-1

^{†††} 電気通信大学 大学院情報理工学研究科 〒182-8585 東京都調布市調布ヶ丘 1-5-1

E-mail: †hiroki.horita.is@vc.ibaraki.ac.jp

あらまし ゴールモデルやビジネスプロセスモデル等の概念モデルを用いた要求分析を行う際は正確なモデルが必要とされる。これらのモデルを現状に合わせて修正する手法が存在するが、それぞれを独立して修正するために両モデルの整合性を保つのが困難である。本論文では、プロセスマイニング技術によって修正したビジネスプロセスモデルの修正内容をゴールモデルに反映する手法を提案する。ケーススタディによる評価を行い本手法の有効性を示す。
キーワード プロセスマイニング, ビジネスプロセスマネジメント, ビジネスプロセスモデル, 要求工学, ゴールモデル

Repairing Goal Models Corresponding to Business Process Model Repair Based on Process Mining Approach

Hiroki HORITA[†], Hideaki HIRAYAMA^{††}, Takeo HAYASE^{†††}, Yasuyuki TAHARA^{†††}, and Akihiko OHSUGA^{†††}

[†] Ibaraki University, College of Engineering, 4-12-1, Nakanarusawa, Hitachi, Ibaraki, 316-8511 Japan

^{††} Mejiro University, Faculty of Media Studies, Department of Media Studies, 4-31-1, Nakaochiai, Shinjuku-ku, Tokyo, 161-8539 Japan

^{†††} University of Electro-Communications, Graduate School of Information Systems, 1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585 Japan

E-mail: †hiroki.horita.is@vc.ibaraki.ac.jp

Abstract An accurate model is required when conducting a requirements analysis using a conceptual model such as a goal model and a business process model. Although there are methods for modifying these models according to the current situation, it is difficult to maintain consistency between both models in order to independently modify each model. In this paper, we propose a method to reflect the modification contents of the business process model corrected by the process mining technology in the goal model. We evaluate by casestudy and reperesent the effectiveness of our method.

Key words Process Mining, Business Process Management, Business Process Model, Requirements Engineering, Goal Model

1. ま え が き

ビジネスプロセスモデルやゴールモデル等の概念モデルは、現実の環境や目標を的確に反映してモデル化し、分析に用いられるべきである。ここで困難なことは、環境変化によって適切なモデルは変化するため、モデルを継続的に改善しなければ適

切な分析結果を得ることはできないということである。

プロセスマイニングはビジネスプロセスに関するデータを分析するための技術であり、現実に行われたビジネスプロセスの内容を捉えるために有効である [1]。モデル修正 [2] はプロセスマイニング技術の一種であり、既存のビジネスプロセスモデルを、ビジネスプロセスの実行ログに適合する形へ自動で修正

することができる。そうすることで、現実から乖離したビジネスプロセスモデルではなく、現実を捉えたビジネスプロセスモデルを得ることができる。ビジネスプロセスモデルの修正は、通常はステークホルダインタビューを行い手動で行うが、モデル修正技術を用いることで、より客観的で短時間にビジネスプロセスモデルを修正できる。しかし、ビジネスプロセスモデルは業務フローという組織の一面を記述したモデルであり、意図に関する内容は記述されていないために、この観点から as-is 分析を行うことができない。

意図を記述するためには、ゴールモデル [3] を用いることが有効である。ゴールモデルはツリー構造でシステムに対する要件を目標として記述する。抽象的なトップゴールは、それを達成するためのサブゴールへ段階的に分解されることで、ゴールがどのようにして達成されるのかが記述できる。

ゴールモデルとビジネスプロセスモデルは補完的な関係にある [4]。これらのモデルを使用することで、組織の目標とその実行フローを共に記述することができる。複数のモデルを用いる場合は両モデル間の一貫性が重要である。環境変化によってビジネスプロセスモデルにおける条件分岐の追加やイベントの削除を行う場合は、その変更がゴールモデルにおいても反映される必要がある。既存の研究では、ゴールモデルとビジネスプロセスモデルとの整合性を検証手法が提案されている [4], [5]。これらの手法を用いることでゴールモデルとビジネスプロセスモデルの整合性を検証することが可能だが、整合しない部分を検知した場合においてそれを修正する支援を行うことはできない。また、[7], [6] はゴールモデルを、[2] はビジネスプロセスモデルを修正するための手法である。これらの方法はいずれかのモデルを修正することができるが、両方のモデルの整合性を保証することはできない。以上から、両モデルの現実の状況に合わせて修正を連携して行う手法が求められている。

本稿では、プロセスマイニング技術に基づくビジネスプロセスモデル修正に対応したゴールモデル修正手法を提案した。この手法によって、ゴールモデルとビジネスプロセスモデルの整合性を保つための支援を行うことができる。本手法ではまず [2] を用いてビジネスプロセスの実行ログに合わせてビジネスプロセスモデルを修正する。次に、ゴールの分解パターンであるリファインメントパターンに基づくパターンベースのアプローチを用いてゴールモデルを修正する。ケーススタディを実施することにより、提案手法の評価を行い、有効性を確認した。

本稿は以下のように構成されている。2 節では背景について説明する。3 節で提案手法について説明する。4 節で評価結についてを説明する。最後に 5 節で本稿の結論を述べる。

2. 背景

本章ではゴールモデルとプロセスマイニングについて述べる。

2.1 ゴールモデル

ゴールモデルは、情報システムやビジネスプロセスに関するステークホルダの要求をゴールとして木構造で記述したものである。ゴール分解には 2 つのパターンがあり、AND 分解はすべての子ゴールが親ゴールを達成するためには達成されなけれ

ばならない分解方法である。OR 分解は親ゴールを達成するためにはいずれかの子ゴールが達成されなければならない分解方法である。

本研究で用いる KAOS ゴールモデル [3] では線形時相論理 (LTL) のような形式言語を用いて記述することが可能である。LTL は時間的に変化する性質を記述することができ、古典的な論理演算子 (\neg, \wedge, \vee 等) や時間演算子 (\circ (nexttime), \diamond (finally), \square (globally) 等) を用いる。LTL を用いることで、自然言語を用いるよりも厳密にゴールについて記述できるとともに形式検証を行うことが可能である。しかし、LTL を記述することは多くのビジネスアナリストにとって困難である。この問題を解決するために、ゴール分解をパターン化したりファインメントパターン [8] を用いることは有効な方法である。それぞれのパターンを用いるべき状況を把握していれば、LTL を使いこなしてゴールを記述することができなくとも、形式的なゴール記述を行うことができる。例えば、milestone-driven pattern では親ゴール ($C \Rightarrow \diamond T$) の達成を 2 つのゴール ($C \Rightarrow \diamond M$), ($M \Rightarrow \diamond T$) に分解する。その際、マイルストーン条件 M を設定することで、親ゴールを達成するための子ゴールの達成順序が定められる。

KAOS ゴールモデルからは標準的なビジネスプロセス表記法である BPMN モデルを構築することが可能であり、表 1 は KAOS ゴールモデルから BPMN モデルを導出するための規則の一部である [9]。本稿ではこれらの変換規則をゴールモデル修正のためのベースとして利用する。

表 1 リファインメントパターンに基づくゴールモデルからビジネスプロセスモデルへの変換規則

Refinement Pattern	KAOS Goal Model	BPMN Model
Milestone-driven		
Decomposition-by-cases		
Guard-introduction		

2.2 プロセスマイニング

ビジネスプロセスモデルはビジネスプロセスの実行フローを記述するモデルである。プロセスマイニングとは、情報システムを用いてビジネスプロセス実行を管理することで得られた実行履歴をイベントログとして記録し、分析を行うことで有益な情報を抽出し活用することである。例えば、プロセス発見はイベントログを入力として、それを満たすビジネスプロセスモデルを出力することである。また、適合性検査はイベントログとビジネスプロセスモデルを入力としてそれらの差分を分析する技術である。これらを用いることで、現在実行されているビジネスプロセスの実態を捉えたり、現状想定されているビジネスプロセスとイベントログとして記録されているビジネスプロセスの実態との差を把握することを支援できる。また、Fahland

らは適合性検査を用いてペトリネットによって記述されたビジネスプロセスモデルをイベントログに適合するように修正する手法を提案している [2]。これによって、ビジネスプロセスモデルがイベントログに適合するように効率的に修正できる。本稿ではこの手法を本研究の一部として用いている。

3. 提案手法

本節では提案手法について説明する。図 1 は提案手法の概要であり、ビジネスプロセスモデル修正フェーズとゴールモデル修正フェーズによって構成される。2つのフェーズを経てビジネスプロセスモデルとゴールモデル両方をイベントログに適合した形へ修正する。3.1 節、3.2 節においてそれぞれのフェーズについて説明する。

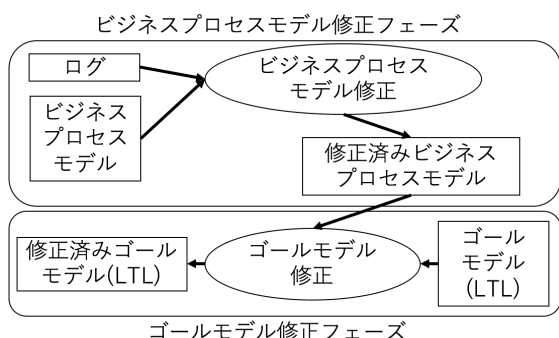


図 1 提案手法概要

3.1 ビジネスプロセスモデル修正フェーズ

ビジネスプロセスモデル修正フェーズでは、主に Fahland らの手法 [2] を用いてビジネスプロセスモデルをイベントログに適合したモデルへ修正する。この手法はプロセスマイニングにおけるイベントログの標準である XES (eXtensible Event Stream) 形式のイベントログとペトリネット形式のビジネスプロセスモデルを入力として、与えられたビジネスプロセスモデルによって許可されない振る舞いがイベントログに記録されていた時、その振る舞いを許容するようにビジネスプロセスモデルを修正するものである。

表 2 は Fahland らの手法 [2] が対応している 3 つのビジネスプロセスモデル修整パターンである。1 番上の行は initial model であり、表 3 のようなイベントログが観測された時、これらの振る舞いを許容するように下の 3 行のモデルへ修正される。例えば、Case ID 1 が実行されたときは、deleting event を行い、観測されなかったイベント B を削除し、イベントの実行順序が AC となるように修正される。Case ID 2,3 がともに観測された時は adding subprocess を行い、initial model にサブプロセスとしてイベント D が追加される。Case ID 4, 5 が観測された時は adding loop を行いイベント B の後にイベント B の手前へループするイベント D が追加される。これら 3 つの修整パターンを組み合わせることによって、イベントログに適合するためのビジネスプロセスモデルの修整が可能となる。より詳細なアルゴリズムは [2] において記述されている。

表 2 ビジネスプロセスモデル修正パターン

initial model	
deleting event	
adding subprocess	
adding loop	

表 3 イベントログ

Case ID	Event Sequence
1	AC
2	ADC
3	ABC
4	ABC
5	ABDBC

3.2 ゴールモデル修正フェーズ

本節ではゴールモデルの修正方法について説明する。このフェーズでは前フェーズにおいてイベントログに適合する形へ修正したビジネスプロセスモデルに対して、図 4 に示すパターンベースの手法によってゴールモデルが適合する形へ修正を行う。ここでは、実線・点線で囲まれたゴールとイベントが対応付けられていることを意味し、前フェーズにおいてビジネスプロセスモデルの該当箇所が修正された場合は、ゴールモデルの該当箇所を修正する。また、ここで用いられるゴールモデルは LTL によって記述されていることを前提とする。

以下では、3.2.1 節において各ビジネスプロセスモデル修整パターンに対応したゴールモデル修整について説明する。

3.2.1 ゴールモデル修正パターン

表 4 は表 2 の 3 種類のビジネスプロセスモデルの修整 (deleting event, adding subprocess, adding loop) に対応したゴールモデルの修整パターンである。最上段に記載されているのが修正前である initial model であり、これが 3 種類の修整パターンによって、下の 3 つのゴールモデルへ修正される。initial model はイベント A, B, C, D が順に実行されるビジネスプロセスモデル (A→B→C→D) に適合している。次に、各修正パターンについて述べる。

deleting event はビジネスプロセスモデルにおけるイベントの削除に対応したパターンである。ここでは、ゴール B と対応したイベント B が削除された場合を想定している。このパターンの場合にはまず、削除されたイベントに対応したゴールである B(A⇒◇B) が削除される。それとともに削除されたゴールと同一の親ゴールを持つ子ゴールの事前条件が修正される。この場合はゴール C (B⇒◇C) が C (A⇒◇C) に修正されており、「A が成り立つとき、そのうち C が成り立つ」というゴールとなっ

ている。

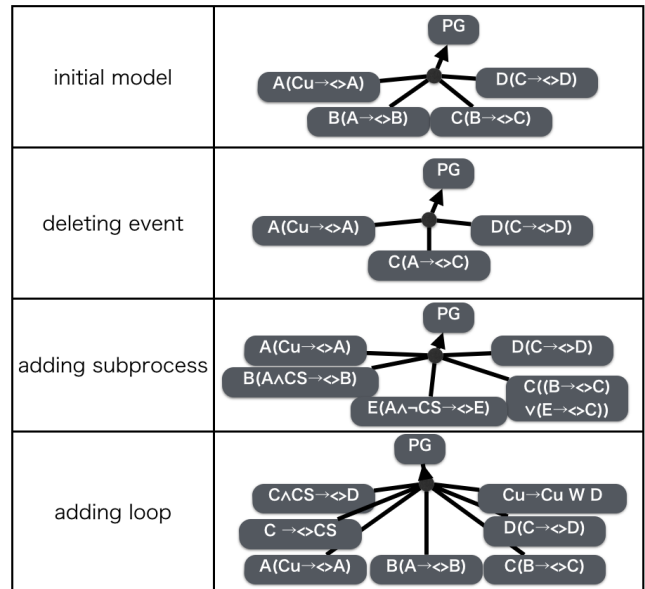
adding subprocess はビジネスプロセスモデルにおけるサブプロセスの追加に対応したパターンである。ここでは、 $(A \rightarrow B \rightarrow C \rightarrow D)$ という順番でイベントが実行されるプロセスの $A \rightarrow C$ 間に新しくイベント E というサブプロセスを追加することを想定する。ここでのゴールモデル修整にはリファインメントパターン的一种である case-decomposition pattern が用いられている。これは親ゴールをケース条件が成り立つ場合と成り立たない場合に分けて分解するためのゴール分解パターンである。[9] によってこのゴール分解パターンから XOR 分岐するビジネスプロセスモデルが導けることがわかっている。ゴールモデルの修正では新しく追加されたイベント E に関するゴール $E(A \wedge CS \Rightarrow \diamond E)$ が追加されている。このゴールにおける CS とはケース条件のことであり、ケース条件が成り立つ場合は A の次に B が実行され、ケース条件が成り立たない場合は A の次に E が実行されることを表している。またこのゴールの追加にともなって、前後のゴール B, C も修正する。B はケース条件 CS が成り立つ場合のゴールであるため、 $(A \wedge CS \Rightarrow \diamond B)$ となり、ゴール C はゴール B あるいはゴール E の後に達成される必要があるため、選言記号 \vee によって、 $(B \Rightarrow \diamond C)$ 、 $(E \Rightarrow \diamond C)$ が結ばれた $C((B \Rightarrow \diamond C) \vee (E \Rightarrow \diamond C))$ となる。

adding loop はビジネスプロセスモデルにおけるループの追加に対応したパターンである。ここでのゴールモデル修整にはリファインメントパターン的一种である guard-introduction pattern が用いられている。これはターゲット条件へ到達するために、ガード条件が必要な異なるケースへ場合分けされる場合に用いられるリファインメントパターンである。[9] によってこのゴール分解パターンからループ構造を持つビジネスプロセスモデルを導けることがわかっている。修正前のプロセス (initial model) においては start から end まで 1 つのフローしかなかったが、修正後はイベント B が実行された後にイベント C が実行されるフローだけでなく、イベント D を実行し、イベント B の前のプレースにループする (再度、イベント B を実行する) フローが追加されている。それに対応してゴールモデルでは新しく追加されたイベント D に関するゴール $(C \wedge CS \Rightarrow \diamond D)$ が追加されている。このゴールにおける CS とはガード条件の事であり、イベント C とガード条件 CS が両方成り立つときに、イベント D が実行される (ループする) ことを表している。またこの他にもループに対応する部分として、2 つのゴールが追加されている。1 つ目は $(C \rightarrow \diamond CS)$ というガード条件の成立に関するゴールであり、2 つ目は $(Cu \rightarrow Cu \wedge D)$ という D が成り立つまでの間、Cu が成り立ち続けるというゴールである。これら 3 つのゴールを追加することで、ビジネスプロセスモデルにおけるループの追加をゴールモデルへ反映することができる。

4. 評価

本節では London Ambulance Service (LAS) を題材として提案手法の評価を行う。修正前ゴールモデルと修正後ゴールモデルを対象としてゴール達成を確認する。検証には SPIN を用いてモデル検査を行う。このツールは PROMELA によって検

表 4 ゴールモデル修正パターン



査対象を記述し、LTL によって検証する性質を記述する。本研究では、ビジネスプロセスモデルを PROMELA を用いて記述し、ゴールモデル中の一部のゴールを LTL によって記述する。ビジネスプロセスモデルは Janssen の手法 [10] を使用して PROMELA の記述に変換する。

LAS は緊急通報を受けた場合に救急車を配備し、患者を病院に搬送するプロセスである。使用するゴールモデルでは、トップゴール「IncHappened \Rightarrow \diamond IncResolved」は 7 回分解されている。上から 1, 3, 4, 5 および 7 回目の分解では、milestone-driven refinement pattern を用いて分解されている。2, 6 回目の分解では、decomposition-by-cases pattern によって分解されている。評価で用いる LAS のイベントログは、CPN ツール^(注1)を用いてビジネスプロセスモデルのシミュレーションによって作成する。

4.1 ビジネスプロセスモデル修正フェーズ

まず、最初に Fahland らの手法 [2] を用いて、図 2 のビジネスプロセスモデルをイベントログに適合した形へ修正した。図 2 には、望ましくないイベント 11 が含まれているとともに、本来あるべきイベント 6 から 4 へ行くためのフローが失われている。それゆえ、このビジネスプロセスモデルは修正されるべき状況にある。図 3 は図 2 を修正した結果であり、不必要なイベントが削除されていることと、存在するべきフローが新たに追加されていることが確認できた。しかし、イベント削除は適切に行われた一方で、サブプロセスの追加は適切には行えなかった。本来はイベント 7、もしくはイベント 8, 9, 10 のどちらかのみが実行されるべきだが、図 3 では両方実行できる形になっている。そのため、この部分は手動で修正を行い図 4 を構築した。

4.2 ゴールモデル修正フェーズ

ゴールモデル修正フェーズでは、ゴールモデル修正のために

(注1) : <http://cpntools.org/>

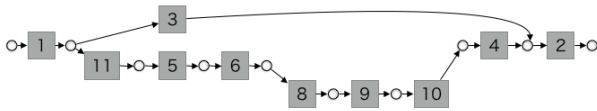


図 2 LAS のビジネスプロセスモデル (修正前)

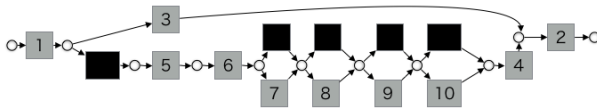


図 3 LAS のビジネスプロセスモデル (Fahland らの手法による修正後)

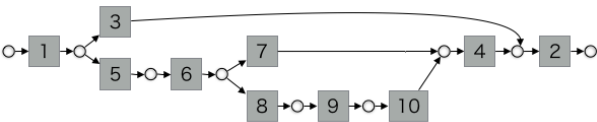


図 4 LAS のビジネスプロセスモデル (Fahland らの手法による修正後に手動修正)

図 4 のビジネスプロセスモデルを使用する。まず、ビジネスプロセスモデルの修正された部分を決定する。同様に、それに対応したゴールモデルの一部を決定する。図 6 では、両モデルの対応する部分がマークされている。これらをそれぞれ A, B と呼ぶ。マークされた部分を提案手法によって修正する。

図 7 は修正結果である。A において、イベント 11 はイベントログで実行されないため、ビジネスプロセスモデル上で削除されている。それに対応して、ゴールモデル上でも deleting event のパターンを使用してゴール 11 が削除されている。B では、イベント 7 に関するサブプロセスがビジネスプロセスモデルに追加されている。イベント 7 はイベント 6 とイベント 4 の間のサブプロセスであり、イベント 8,9,10 とどちらかのフローが実行される。したがって、ゴールモデルの対応する部分もフローの分岐に対応する必要がある。adding subprocess パターンを用いて、ケースをケース条件 (CS) を定義する 2 つのケースに分割する。ケース条件 CS が真の場合、イベント 7 が実行される。ケース条件 CS が偽の場合、イベント 8,9,10 が実行される。したがってゴールモデルでは、イベント 7 ($6 \wedge CS \rightarrow 7$) に関する新しいゴールが追加され、8 と 4 のゴールが修正される。イベント 8 については、ケース条件 CS が偽であるため、8 についてのゴールが ($6 \wedge \neg CS \rightarrow 8$) に修正される。イベント 4 については、イベント 4 を実行するための新しい条件 (イベント 7 の実行) が追加されているため、ゴール 4 の事前条件が修正され、ゴール 4 の事前条件が ($7 \rightarrow 4$) \vee ($10 \rightarrow 4$) となり、イベント 7 またはイベント 10 が実行されたら、イベント 4 が最終的に実行されなければならないことを表す。

4.3 モデル検査による検証

SPIN を使用してゴールに関する性質を検証し、修正前のゴールモデルにおけるゴールに関する性質を用いた検証と、修正後のゴールモデルに関する性質を用いた検証結果の比較を行う。修正されたビジネスプロセスモデルは正しいと仮定する。それ

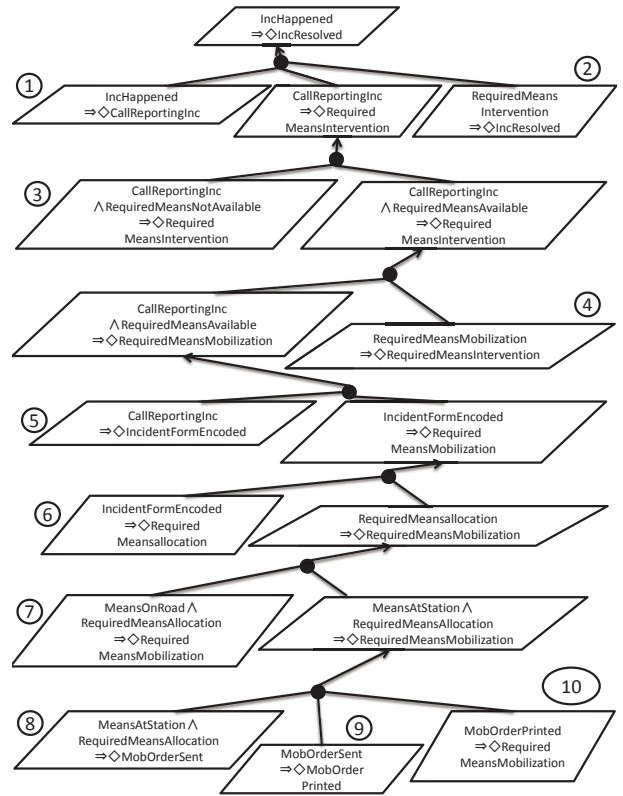


図 5 LAS のゴールモデル [9]

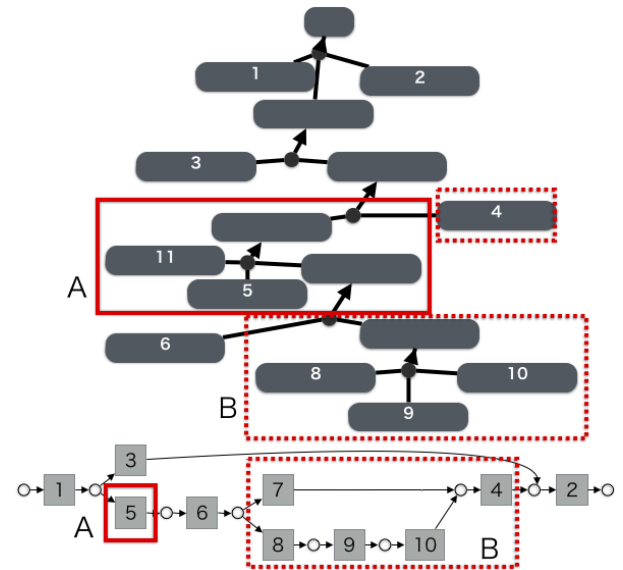


図 6 修正前ゴールモデルと修正後ビジネスプロセスモデルの対応関係

ゆえ、元のゴールモデルが正しくない場合は、モデルを修復する必要がある。ゴールモデルに関連する以下の 2 つの性質を使用した。

property 1: $\square((\text{event_1}) \rightarrow \diamond(\text{event_11}))$

property 2: $\square((\text{event_6}) \rightarrow \diamond(\text{event_8}))$

修正されたビジネスプロセスモデルにはイベント 11 が存在

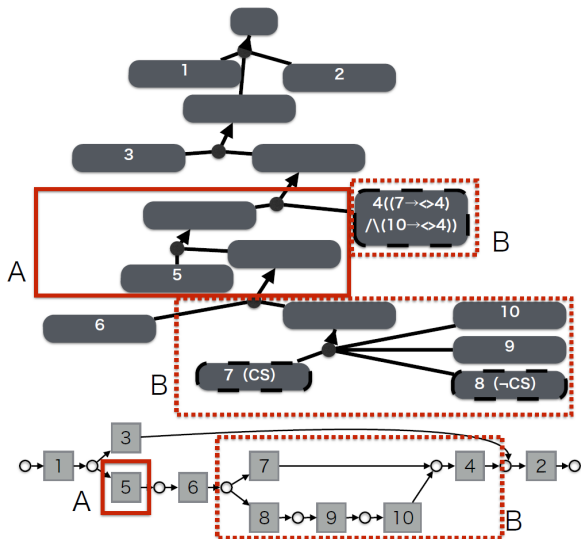


図7 修正後ゴールモデルと修正後ビジネスプロセスモデルの対応関係

しないため、性質1は修正されたビジネスプロセスモデルでは満たされない。修正されたビジネスプロセスモデルには、イベント6からイベント4までの2つのパスがあるため、性質2は修正されたビジネスプロセスモデルでも満たされない。一方、修正されたゴールモデルに関連する性質は修正されたビジネスプロセスモデルと適合している。結果として、提案手法によって適切にゴールモデルを修正できることを確認した。

5. 関連研究

本節では関連研究について記述する。

5.1 ゴールモデルとビジネスプロセスモデルの対応関係と検証

KoliadisらはBPMNとKAOSを関連付ける方法論を提案している[11]。この研究はBPMNとKAOSを関連付けることを対象としており両者のギャップ削減を図っている。grunerらはゴールモデルとプロセスモデルの整合性を一階述語論理に基づく手法で検証する手法を提案した[4]。nagelらも同様に、ゴールモデルとプロセスモデルの整合性をパターンベースの手法で検証する手法を提案している[5]。これらのように両モデルを対応付け、モデル間の整合性を検証することは重要だが、整合が取れてない場合においてはどのように両モデルを修正すればいいのか支援を行うことはできない。本稿では、両モデルを合わせて修正する手法を提案している。

5.2 ビジネスプロセスモデルからゴールモデルへの変換

Varaらはビジネスプロセスモデルをゴールモデルへ変換するガイドラインを提案している[12]。この研究は本研究とは異なりビジネスプロセスモデルの修整をゴールモデルへ反映する手法を提案しているわけではない。そのため、手法によって構築されたゴールモデルが初期モデルの形とは離れてしまう可能性がある。モデルの理解容易性を考慮すると、修正されたゴールモデルは初期モデルとできるだけ近い形になっていることが望ましい。

6. まとめ

本稿では、プロセスマイニング手法によるビジネスプロセスモデル修整に対応したパターンベースのゴールモデル修正方法を提案した。Fahlandらの手法によるイベント削除、ループ追加、サブプロセス追加にそれぞれ対応したゴールモデル修整パターンをゴール分解パターンであるリファインメントパターンを用いることで実現した。LASを題材としてケーススタディを行い、両モデルの整合性を保持する支援に有効であることを確認した。

本稿では、ビジネスプロセスモデルの修正について、コントロールフローに関する修正のみを対象としていたため、データやリソースに関する修正に対応することは今後の課題である。

文 献

- [1] van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer, 2011.
- [2] Fahland, D., van der Aalst, W.M.P.: Model repair: aligning process models to reality. Information Systems. Volume 47, pp.220-243, 2015.
- [3] van Lamsweerde, A.: Requirements Engineering: From System Goals to UML Models to Software Specifications. John Wiley Sons, 2009.
- [4] Groner, G., Asadi, M., Mohabbati, B., Gasevic, D., Boskovic, M., Parreiras, F.S.: Validation of user intentions in process orchestration and choreography. Information Systems, Volume 43, pp.83-99, 2014.
- [5] Nagel, B., Gerth, C., Engels, G., Post, J.: Ensuring consistency among business goals and business process models. Proceedings of 17th IEEE International Enterprise Distributed Object Computing Conference (EDOC2013). pp. 17-26, 2013.
- [6] Dalal Alrajeh, Axel van Lamsweerde, Jeff Kramer, Alessandra Russo and Sebastian Uchitel: Risk-driven revision of requirements models. proceedings of 38th IEEE/ACM International Conference on Software Engineering (ICSE2016),pp.855-865, 2016.
- [7] A. Cailliau and A. van Lamsweerde. Integrating exception handling in goal models. In Proceedings of the 22nd International Requirements Engineering Conference (RE2014), pp.43-52, 2014.
- [8] R. Darimont and A. van Lamsweerde: Formal Refinement Patterns for Goal-Driven Requirements Elaboration, Proceeding of Fourth ACM SIGSOFT Symposium on Foundations of Software Engineering (FSE1996), pp.179-190, 1996.
- [9] 堀田大貴, 本田耕三, 平山秀昭, 清雄一, 中川博之, 田原康之, 大須賀昭彦: リファインメントパターンを利用したKAOSゴールモデルからBPMNモデルへの変換, コンピュータソフトウェア, VOL.32, NO.4, 141-160, 2015.
- [10] Janssen, W., Mateescu, R., Mauw, S., Springintveld, J.: Verifying business processes using spin. Proceedings of the 4th International SPIN Workshop, pp. 21-36, 1998.
- [11] George Koliadis, Aditya Ghose:, Relating business process models to goal-oriented requirements models in KAOS, Proceedings of the 9th Pacific Rim Knowledge Acquisition international conference on Advances in Knowledge Acquisition and Management (PKAW2006). pp.25-39, 2006.
- [12] Jose Luis de la Vara, Juan Sanchez, and Oscar Pastor.: On the Use of Goal Models and Business Process Models for Elicitation of System Requirements. International Workshop on Business Process Modeling, Development and Support (BPMDS 2013), pp.168-181, 2013.