

[解説]

プロセスマイニング・サーベイ (第01回: 概要と基本概念)

飯島 正[†], 田端 啓一[‡], 斎藤 忍[‡]

1 はじめに

ビジネス(業務)を取り巻く環境・状況の変化が激しい昨今, ビジネスを支える情報システムも, 迅速に, その変化に追従していくことが求められている. そうした要求は, センサーデータまでも取り込んだビッグデータ活用が望まれる中で, 一層の高まりをみせており, 複合イベント処理(Complex Event Processing)とBPM(Business Process Management)の融合によるイベント駆動型(event-driven)情報システムから. さらには, データ駆動型社会システムまでもが求められている. この流れの中で, 情報システムにも現実のデータに基づいた適応(adaptation)の, 迅速化や(全面的には無理としても一部だけでも)自動化が期待されている. かといって, 情報システムを変化へ適応させることを優先することで, 新たなバグを作り込み, 安全性を損なってしまっは元も子もない. 情報システムは, 既にレガシーなサブシステムを含む強大な情報基盤として肥大化しているケースも多く, その安定性・堅牢性を維持するために取り組むべき課題も顕在化してきている. それは, 大きく分けて以下の3点から成る:

- (1) 既に, 全社的なサービス指向アーキテクチャ(SOA; Service-Oriented Architecture)へ対応によって, 頻繁で迅速なビジネスプロセスの改善(BPR; Business Process Reengineering)への取り組みはなされてきている. しかし, 複雑に有機的に相互結合し合ったサブシステムの集合体(SoS; System-of-Systems)において, その全容を理解することは決して容易ではない. 仮に, その全体像は理解

できたとしても, 全体にわたる最適化を継続的に繰り返し行っていくことは難しい.

- (2) 加えて, 個々のサブシステムだけを取り上げても, ソフトウェア技術者の世代交代が続く中, 長期に渉り使い続け, レガシーソフトウェアとなりつつあるサブシステムの内部アーキテクチャに精通した技術者を確保し続けることが困難である.
- (3) また, 情報システムを取り巻く利用環境も, 設計当時に前提としてたものから変化し続け, セキュリティなどの要求も強まり, かつ, 想定外の要素が多くを占めるようになってきている. そこで, 必ずしも設計当時のドキュメントに頼るだけではなく, 今現在, 現場で実際に稼働している情報システムの, 個々のサブシステムや, その組み合わせられた全体としての振る舞いを, あらためて理解し, その上で必要があれば, その全体を見通して振る舞いに手を加え強化(enhancement)していく手法の確立も求められている.

これらの観点から, 情報システムの理解と強化のための技術として, プロセスマイニング(**Process Mining**)技術が着目されている. この技術は, 主に西暦2000年前半から研究開発が活発化し, ようやくツールの安定化, データの標準化に向けた取り組みがなされつつあり, 実用的な普及が始まろうとしている. 本調査報告は, そうした状況を踏まえ, 全5回の連載の形で, プロセスマイニングの概念と, 手法, ならびに, ツールに関する過去から現在までの研究開発の動向, および今後の方向性に関して取り上げるものである. 今回の「概要と基本概念」編に続き, 「ツール」編, 「データ」編, 「アルゴリズム」編, 「応用」編として, 諸側面を順次取り上げていく予定である.

1.1 出発点となる情報源

プロセスマイニング技術の研究開発は, 幅広く世界中で行われており, 情報源も多岐に渡る. その中で, 取り組むにあたって, まず最初の出発点

A Survey on Business Process Mining

— 01: Overview and Basic Concepts —

Tadashi Iijima[†], Keiichi Tabata[‡], and Shinobu Saito[‡]

[†]Faculty of Science and Technology, Keio University

[‡]Nippon Telegraph and Telephone Corporation

[†]慶應義塾大学・理工学部

[‡]日本電信電話株式会社

[解説] 2016年3月24日受付.

©一般社団法人 情報システム学会

として土地勘をつけるために役立つ情報源を紹介しておく。

プロセスマイニングは、ビジネスプロセス (Business Process) のイベントログ (データ) から有益な情報を抽出し活用する技術一般を指す。その研究の流れは、元々、ペトリネットをベースとしたビジネスプロセス技術ないしワークフロー管理/自動化 (Workflow Management/Automation) 技術を先導してきた、その分野の大家であり、プロセスマイニング技術にもごく初期の段階から中心的に関わってきた、W. van der Aalst (オランダの Eindhoven University of Technology, 略称 TU/e) の著書^[1]に網羅的に詳述されている。WWW サイト「Process mining – research tools application」¹には、同グループの研究成果 (研究論文やツール、上掲書のサポートなど) を中心にした情報が掲載されており、貴重な情報源となっている。

また、プロセスマイニングの研究開発動向と今後の方向性を示すべく、IEEE のプロセスマイニング・タスクフォース²により 50 以上の組織の 75 名以上の研究者・開発者らの共著として執筆された「プロセスマイニング・マニフェスト (Process Mining Manifesto)」^[2, 3]とその日本語訳^[4]も、コンパクトながら土地勘をつけるために有益な資料である。まだ十分に中身を伴っているとは言いがたい部分も多々含まれてはいる、まさに「宣言」ないし「声明」にあたる文書だが、この分野が何を目指し/どのような問題意識を持っているのか、今後のロードマップを思い描くためにも役立つ。これに関しては後節 (第 5 節) でさらに詳しく取り上げる。

この IEEE のプロセスマイニング・タスクフォースによる 2014 年度 Best Process Mining Dissertation Award を受賞した Andrea Burattin の博士論文は、書籍として出版されており^[5]、その第一部は、この分野のコンパクトな解説といえる。

1.2 ビジネスプロセスの簡単な例

まずは、ビジネスプロセスのイメージを得るために、主にソフトウェア開発に広く活用されているモデル図的表現言語 UML (Unified Model-

ing Language)³^[6]に含まれる図面の一つであるアクティビティ図 (Activity Diagram) で記述した、ごく簡単なビジネスプロセスの例を図 1 に示す。

このプロセスは、開始イベントで開始されたプロセスが、フロー (矢印) に沿って、順に実行される。すなわち、このプロセスは、「受注」というタスク完了後に、並行して「納期回答」と「製造」というタスクに取り掛かり、その両タスクの完了後に、さらに「出荷」タスクを行い、最後に「請求」タスクを実行して、終了イベントに至るといった手順を示している。また、この手順において、各タスクを実行する主体は、スイムレーン (アクティビティ・パーティション) として分けて記載されている各ロール (役割) であり、受注、納期回答、請求といった客先と生産部門をつなぐタスクを実行する役割は正業部門が担当し、製造は生産部門、出荷は文字通り出荷部門が担当することが記述されている。

UML の体系の中で、このアクティビティ図は、ソフトウェア開発における上流工程において、業務という視点からのモデル化を意識して導入されたことは容易に想像できるが、必ずしも、ビジネスプロセス専用の表現というわけではなく、一般に「処理の手順」を表現するために、一種の拡張フローチャートとしてソフトウェア開発全体で広範に用いられている。一方、このアクティビティ図に似たビジネスプロセス専用の表現に、後節 (第 4.3 節) で詳しく述べる BPMN (Business Process Model and Notation)⁴^[7]がある。BPMN はビジネスプロセスの表記法として広く使われており、分岐や並行も表現することができる。似ているといっても、BPMN の基本構造がアクティビティ図に似ているために、アクティビティ図に馴染みがあれば、BPMN への参入障壁が低いという程度である。今や BPMN の方がずっとビジネスプロセスの記述に特化した記述要素を備えているため、基本的なプロセス定義のためには、アクティビティ図で BPMN の代用とすることができるとしても、本格的なビジネスプロセスの記述のためには BPMN を使用することが望ましい。

³本稿執筆時点の最新版は 2.5(formal/15-03-01)。 <http://www.uml.org/>

⁴<http://www.bpmn.org/>

¹ <http://www.processmining.org/>

²The IEEE CIS Task Force on Process Mining, <http://www.win.tue.nl/ieetfpm/doku.php?id=start>

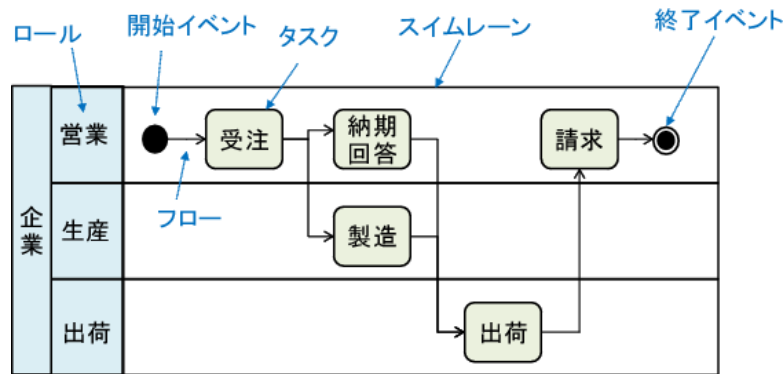


図 1: ビジネスプロセスの例

1.3 プロセスマイニング技術の三本柱

プロセスマイニング技術の中核は、大きくみればデータマイニングの一種とも言えるが、データマイニングには見られないプロセスマイニング固有の展開もある。上掲書の副題にもなっているが、プロセスマイニングは、下記の三つの活動に分類されることが多い。

- (1) プロセス発見 (process discovery)
- (2) 適合性評価 (conformance check)
- (3) 強化 (enhancement)

これらは、プロセスマイニングの三本柱ともいえるべき、重要な要素技術ともいえる(訳語は、文献^[4]に従っている)。その様子を図2に示す。

「発見」は、イベントログを入力として、それを満たすプロセスモデルを出力する技術を目指す。また、「適合性評価」はプロセスモデルとイベントログを入力として、両者の適合性を判定する技術である。プロセスモデルのいかなるトレースもイベントログと一致しないのであれば、適合性がないことになる。もう一つの「強化」は、やはりプロセスモデルとイベントログを入力として、より適合性の高いモデルを出力する技術といえる。

「発見」技術がなければ、そもそもプロセスマイニングは始まらないので、必須の中核技術と言えるが、応用面からは関心の中心は、次第に、プロセス分析やプロセス改善への直接的な応用に向けられ、「適合性評価」技術や「強化」技術へと移りつつある。本サーベイは、全5回の連載形式で、アルゴリズムの研究動向から、利用可能なツールや応用までを含む多方面から、これらの技術に対して解説を試みるものである。

2 解説の概要

本サーベイは、先に述べたように全5回で構成する予定である。今回(第一回)は、「概要/基本概念」編であり、基本概念、前提となるビジネスプロセス表現の紹介、および、プロセスマイニングマニフェストの紹介を行なう。

次回(第二回)は「ツール」編であり、プロセスマイニング関連の各種ツールについて紹介する。研究動向だけではなく、読者が参照しながら試用できるように、一部のツールに関しては、インストール方法など導入手段も含めて解説する。プロセスマイニング技術の集大成として実績のあるオープンソース・ソフトウェア ProM⁵ [8, 9] を中心に取り上げる。ProMは一種のプラットフォームに位置付けることができ、ProMが開発されて以降は、多くのアルゴリズムがProMのためのプラグインとして実装されている。その他、初期の研究過程で構築された研究用ツールである、Little Thumb⁶ [10], EMiT⁷ [11] や、その前身であるMiMo⁸ [12] と、ペトリネット合成によるモデル構築機能を有するVIP Tool⁹ [13, 14, 15, 16, 17, 18] について紹介する。また、商用ツールとして、fluxicon社のDisco¹⁰ [19] と富士通のInterstage Automated Process Discovery Service¹¹ について言

⁵<http://www.promtools.org/doku.php>, <http://www.processmining.org/prom/start>

⁶<http://www.processmining.org/discontinued/littlethumb>

⁷<http://www.processmining.org/discontinued/emit>

⁸<http://www.processmining.org/discontinued/mimo>

⁹https://www.fernuni-hagen.de/sttp/forschung/vip_tool.shtml

¹⁰<https://fluxicon.com/disco/>

¹¹<http://www.fujitsu.com/global/products/software/middleware/application-infrastructure/>

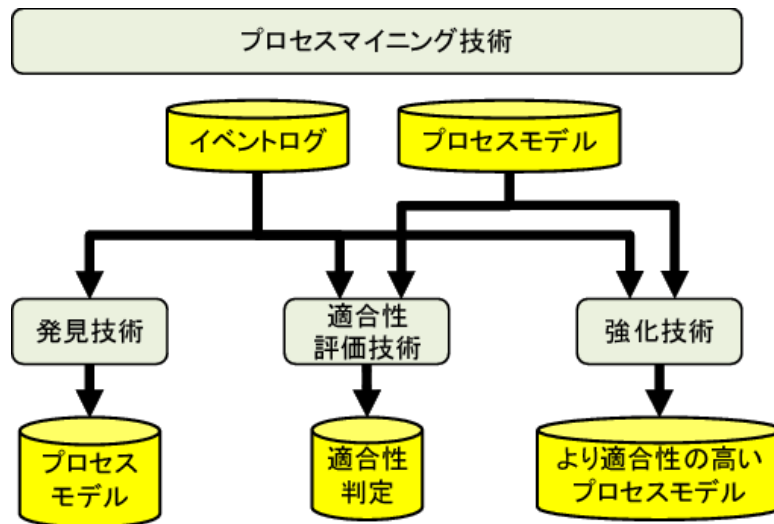


図 2: プロセスマイニング技術の三本柱

及する。

第三回は「データ」編であり、(ログ)データの側について触れる。IEEE CIS Task Force on Process Mining の XES Working Group¹²で策定されつつある、ログデータの標準規格である XES を中心に、ProM に入力できるデータ形式について紹介する。XES の参照実装として OpenXES¹³ や、イベントログの抽出ツール XESame^{14 [21]} についてもごく簡単に紹介する。

第四回は「アルゴリズム」編であり、主に「発見」技術のためのアルゴリズムについて解説する。もっとも著名な α アルゴリズムとその拡張、遺伝的アルゴリズムを適用した手法、領域 (Region) 理論に基づく手法 (状態ベースの領域、言語ベースの領域、帰納論理プログラミング)、さらに、ファジー理論に基づく手法や、それらの組合せについて解説する。

第五回は「応用」編であり、プロセスマイニングの諸技術を適用する応用先に関して、具体的な適用事例を紹介する。

interstage.com/solutions/bpmgt/bpm-services/apd/

¹²https://standards.ieee.org/develop/wg/XES_WG.html

¹³オープンエクセスと読む。 <http://www.xes-standard.org/openxes/start>

¹⁴初期には、XES Mapper の意味で XESMa と呼ばれていた [20]。 <http://www.processmining.org/xesame/start>

3 基本概念 – プロセスマイニングと周辺技術

3.1 プロセス発見におけるデータマイニングとの関係

プロセスマイニングについて話題にするとき、聴き手がデータマイニング技術を念頭におくことで、むしろ、それに引きずられすぎて、理解の妨げとなることも多い。まずは、プロセスマイニングとデータマイニングの関係を示す。この関係は、特に「プロセス発見技術」に関わってくる。

3.1.1 定義と実体 (モデルとインスタンス)/内包と外延

大量のデータから有益な情報を発見する技術であるデータマイニングという概念と、プロセスマイニングとを単純に対応付けてしまうと必ずしも適切な理解は得られない。プロセスマイニングという語のプロセスをデータと単純に置き換えて理解しようとする、「大量のプロセスを集めてきて、そこから何か有益な情報を発見する」技術というように誤解されるかもしれない。これは慎重に考えれば正しいともいえるのだが、表面的には誤解を招きがちともいえる。プロセスマイニングの三本柱のうち、「発見」技術の入力データは、一般的にいうプロセス (のモデル) というよりも、厳密に言えば、それを規定するログデータに相当する。

この点に関しては、次のように考えて欲しい。プロセスという一つの語で、その定義(モデル)と、その実体(インスタンス)とを表すことがあるため、若干の混乱が生じている。「プロセスの定義」を、これまでプロセスモデルと称してきたが、単に「プロセス」ないし「モデル」と言えばこちらを指すものとする。「スキーマ」と呼ぶこともある。一方、少々くどい言い回しだが「プロセスの実体」は、そのプロセスモデルを実行した時の一つ一つの「インスタンス」(もしくは「オカレンス」)を指し示すものとする。これは、一連の(適切な粒度と付帯情報を伴った)ログデータで表現できる。

すなわち、プロセスマイニング技術(のうちの必須な中核技術要素である「発見」技術)は、入力として「プロセスの実体」を規定するログデータの外延的な集合をとり、出力として、入力された「プロセスの実体」の集合を内包する「プロセスの定義(モデル)」を返す¹⁵。その様子を図3に示す。

ここでいうモデルは、図1で例示に使用したBPMNやUMLのアクティビティ図、ペトリネットなどの何らかのモデル記述言語で記述された「プロセスの定義(モデル)」である。多くの場合、アクティビティを実行する順序を、フローチャートのような制御フロー構造、もしくは、手続き的言語のような制御構造を持ったブロック構造、ないし、その融合的な表現で表現するものが多いが、中には宣言的な記述もありうる。

情報システムでは、その中でビジネスプロセスが「実行」される。そのビジネスプロセスに対する関与者/実行者は、人間であることも、他の情報システム、ないし、それが提供しているサービスであるかもしれない。ワークフローシステムは、その「実行」を円滑に進めるため、事前に定義したワークフローにしたがって、サービス呼び出しを自動的に行って行ったり、人間である実行者へ実行を促す(備忘録であるTodoリストに追加したりアラートを鳴らしたりする)。サービス実行の失敗に対する復旧(recovery)や補償(compensation),

¹⁵数学や論理学では、インスタンス(の外延的な集合)のことをモデルと称することがあり、どのような文脈で使われているかに関して注意が必要である。たとえば、ソフトウェア工学における検証技術の一つにモデル検査手法があるが、そこでのモデルはインスタンスに相当する。一方、同じくソフトウェア工学であっても、モデル記述言語やモデルベース開発手法などで使われるモデルは、(抽象度はまちまちであるが)分析や設計もしくは実装への変換や実装の代替として使用される定義に相当する。

実行開始や終了の遅延に対する督促、アクティビティの開始前の準備(provisioning)などを行なう。いふなれば、その手順を定義したものが、「プロセスの定義(モデル)」である。

一方、こうした「プロセスの定義」にしたがって実行される一つ一つのプロセス自体を、「プロセスの実体(インスタンス)」と呼んでいる。一つの定義に対して、実行される状況毎に、実行されるアクティビティの系列は異なる可能性がある。こうした実行系列の一つ一つが「プロセスの実体」である。

3.1.2 系列データマイニングとグラフマイニング

系列データとは、同質のデータを順序付きで並べたものであり、データマイニングの対象としての系列データには時間ないし順序と無関係な文字列やDNAも含まれるが、時間順のマーケティングデータであるバスケットデータなどもしばしば対象とされてきた。バスケットデータを対象とした相関ルールを枚挙するアルゴリズムとして、**Apriori**アルゴリズムが知られている。これは多くのデータ集合中に一定頻度以上で出現するパターン(部分系列や部分グラフ)を列挙する頻出パターンマイニングの一種である。実は、プロセスマイニング研究のごく初期(1998年)に、Aprioriアルゴリズムで有名なRakesh AgrawalらによってMining Process Models from Workflow Logsという論文^[22]が発表されており、本調査において確認できた中で、一般的なプロセスマイニングを意識した初期の研究論文といえる。

3.1.3 ソフトウェアプロセス

もっとも、Agrawalらの論文以外にも、1990年代後半より幾つかの研究発表が現れている。ソフトウェアリポジトリ操作履歴からのソフトウェア開発プロセスのモデルを発見する手法に関する論文^[23, 24, 25]がある。論文^[23, 24]では、例から有限状態マシン(FSM; Finite State Machine)を合成する手法と、確率的なマルコフ・モデルを学習する手法が使われているが、例からのFSM合成手法には、1972年に発刊された論文^[26]で提案されている方法も使われている。したがって、そうしたものを含めて使用されている手法の源流を辿ると、さらに遡ることも不可能ではない。ま

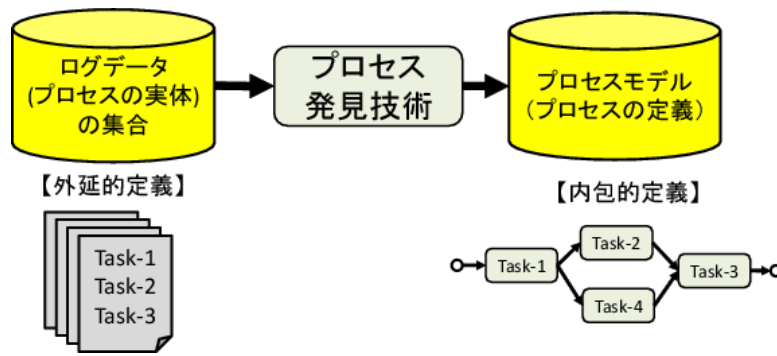


図 3: プロセス発見技術: ログデータからモデルへ (外延的定義から内包的定義へ)

た、同じく FSM の学習にニューラルネットワークも使われているが、これに関しては、次項 (第 3.1.4 項) で言及する。

3.1.4 機械学習

本稿の企画から執筆の時点 (2015 年 9 月～2016 年 3 月頃) は、第三次人工知能ブームと呼ばれている。第一次ブームは、「人工知能 (AI; Artificial Intelligence)」という分野名がダートマス会議でジョン・マッカーシーによって命名された 1956 年から 10 年ほどと言われ、第二次ブームは、「エキスパートシステム (expert system)」が火をつけ、1970 年代後半から 80 年代にかけて続いたとされている。しかし、エキスパートシステムには、専門家から知識を獲得する困難さやコストの高さ (知識獲得ボトルネック) に常に足をびっばられていた。ある意味、その知識獲得に直接的にアプローチする機械学習 (Machine Learning) 技術は、統計学などにも基礎をおく人工知能 (AI) の分野として分類されるが、特にニューラルネットワーク (Neural Network)、遺伝的アルゴリズム (Genetic Algorithm)、シミュレーテッドアニーリング (Simulated annealing) などを含むメタヒューリスティック (Metaheuristic) アプローチは計算知能 (CI; Computational Intelligence) という一分野として重要な位置を占め続けてきた。そして、1958 年の Frank Rosenblatt によるパーセプトロン (perceptron) の発表から、長らく続くニューラルネットワーク研究は、2006 年の Geoffrey Everest Hinton らの論文に始まるとされているディープラーニング (深層学習) 技術^[27]^[28]^[29]として、現在の第三次人工知能ブームを牽引している。当然のことながら、プロセスマイニングに対しても、機械学習に基礎をおく手法は、

手法の向き不向きから偏りはあるが、幅広く取り入れられている。既に名前の登場している IEEE CIS Task Force on Process Mining も、CIS すなわち Computational Intelligence Society の傘下にある。

基本的には、プロセスマイニングでの発見技術は、ログデータからモデルを「学習」しているともみなすことができる。このとき、与えられるログデータが実際に情報システムを実行して得られたものだとすると、負例は存在しないので、正例のみが与えられている教師あり学習とみなすことができる。人工的に負例を生成する手法 AGNEs (Artificially Generated Negative Events) は提案されている^[30]。

もっとも、必ずしもすべてのログデータが説明できるモデルを求めているわけではなく、ノイズを許容することが一般的である。すなわち、ログデータのうち、すべてではなくてもできるだけ多くのログデータをカバーして説明できるモデルが求められることも多くある。一方で、与えられたログデータが少ないとき、少ないログデータ (訓練データ) に過剰適応/オーバーフィッティング (overfitting) すると、一般性が乏しくなり、未知データにフィット/適合 (fit) しないという状況が頻発してしまう。この適合の度合い、すなわちフィットネス (適合度; fitness) はプロセスモデルの品質基準の一つである¹⁶。

機械学習という分野からのビジネスプロセス (ワークフロー) 獲得 (acquisition) の初期のアプローチには、Herbst らの論文^[31, 32, 33, 34, 35, 36]がある。もっとも、対象がビジネスプロセス (ワーク

¹⁶ 他にも、情報検索や推薦システムと同様に、精度 (precision) や一般性 (generalization) といった品質基準がある。また、同じログ集合を説明できるなら、より構造の単純なモデルの方が望ましい (オッカムの剃刀)。その品質基準として単純性 (Simplicity) がある。

フロー)であることから機械学習手法としては、帰納的文法学習やマルコフモデル獲得といった手法が中心であった。その後、機械学習手法の中では、遺伝的アルゴリズム [37] [38] [39] [40] [41] [42] [43] [44] や帰納論理プログラミング (ILP; Inductive Logic Programming) [45] [46] [47] [48] などが利用されている¹⁷。

前出の論文 [23, 24] では、文献 [51] の手法にしたがい FSM の学習にニューラルネットワークも使われているが、全体として分類や識別、回帰や予測に強いニューラルネットワークや SVM をこれまでプロセスマイニングへの適用した事例は少ない [52][53]。また、シミュレーテッドアニーリングの適用事例も少ない [54][55]。

本稿では、アルゴリズムの詳細に関しては、これ以上踏み込まず連載第 4 回で解説する予定である。

3.2 ソフトウェア工学におけるテスト検証技術との関係

適合性評価は、モデルをソフトウェアが満たすべき仕様とみなすと、そのソフトウェアの実行系列が仕様を満たしているかどうかを確認することに相当する (図 4)。

ソフトウェア検証 (verification) という観点からすれば、適合性診断の結果は Yes/No (適合する/しない) の二通りだけしかない。しかし、機械学習の項で説明した通り、ノイズの存在などを許し完全な適合のみを追求しないのなら、適合性は Yes/No の二通りだけではなく、適合の度合いを尺度として考慮することになる。単純には、与えられたログデータのうち、いくつのログが適合するかという適合度/フィットネス (fitness) を設定することができる (図 5)。また、一つのログデータのすべてのデータが説明できなくても、そこに含まれるデータのうち、どのくらいのデータが適合するかという許容度、および、それらの組合せも考えられる。

適合度/フィットネスは、発見されたプロセスモデルの構造の適切さの基準でもある。同じログデータに適合するモデルでも一般性が高すぎて意味をなさないと言われるプロセスモデルとしては、図 6 のように「花びら状に見えるモデル=フラ

ワーモデル」がある。このプロセスは、いろいろなログ系列にフィットするが一般性が高すぎて、ほとんど意味をなさない。これは、過剰適応/オーバーフィッティングとは逆の、アンダーフィッティング (underfitting) に相当する。

4 プロセスモデル記述言語

本節では、プロセスマイニングの前提となるプロセス定義 (プロセスモデル) を記述し (describing), 解析し (analysing), 実行する (executing, enacting) ためのプロセスモデル記述言語, プロセス定義言語, もしくはプロセスモデリング言語と呼ばれる表現 [56][57][58, 59][60][61][62][63] について簡単に解説する。

プロセスの定義 (プロセスモデル) の例として、図 1 に UML のアクティビティ図を用いて記述した例を示している。プロセスの表現としては、この例のように、複数のアクティビティ¹⁸を、制御フローチャートと同様にフロー (矢印) で結び、実行順序を規定するものが基本的である。

プロセス定義の表現としては、フロー (制御フロー) 表現 [64], ブロック (制御構造) 表現, 宣言的な表現 [57] などが考えられ、それぞれ一長一短があるが、今回は、本連載を読み進めるのに必要な最小限の前提知識として、基本的かつ直観的なフロー表現を主に取り上げるるものとする¹⁹。具体的には、WfMC (ワークフロー管理連合) における XPDL から初めて BPMN と幾つかのペトリネット (Petri Net) に基づく表現を中心に取り上げる。

4.1 WfMC と XPDL

1993 年に設立されたワークフロー管理連合 (WfMC; Workflow Management Coalition) においてワークフロー管理システムのための参照

¹⁸用語としてのアクティビティとタスクの使い分けは文脈によって異なり、単純に同義語とすることもがあるが、一般的には、アクティビティの方が粒度が大きい。そこで、アクティビティの構成要素をタスクと呼ぶことが多く、特に最小単位すなわち原子的 (atomic) なアクティビティをタスクと呼ぶこともある。たとえば、BPMN ではタスクは原子的なアクティビティであり、アクティビティはタスクかサブプロセスのいずれかとされている。また、アクティビティを事案 (case) とタスク (task) と資源 (resource) の組合せと位置付けることもある。

¹⁹但し、フロー表現とブロック表現の違いから両者のマッピングの問題についても短く言及する。また、紙数の都合上、それぞれの表記法の詳細に言及することはせず、できるだけ参考文献へのリンクを提示するものとする

¹⁷同じく ILP と略すが、整数線形計画法 (Integer Linear Programming) を用いるプロセスマイニング手法もある [49][50]

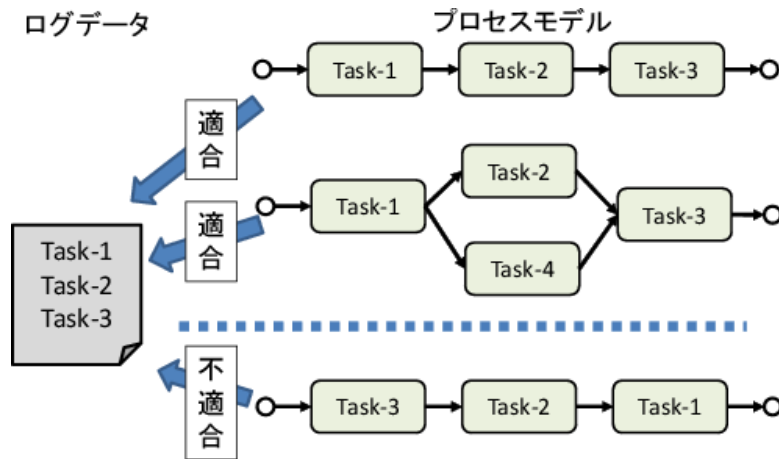


図 4: 適合性評価技術: イベントログとモデルとの適合性診断

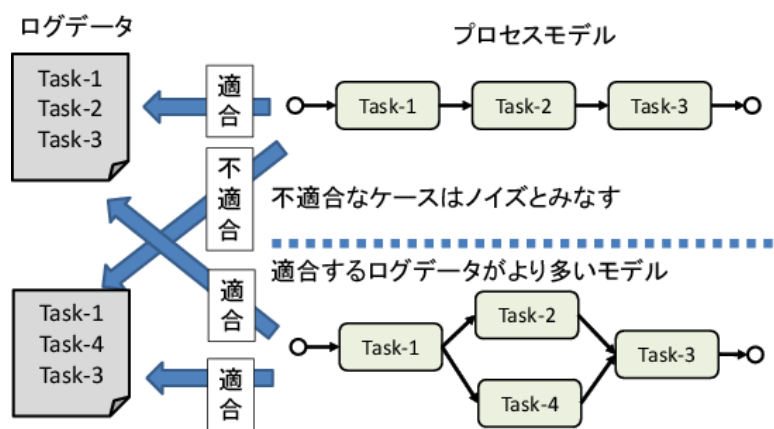


図 5: 適合性評価技術: モデルとノイズ

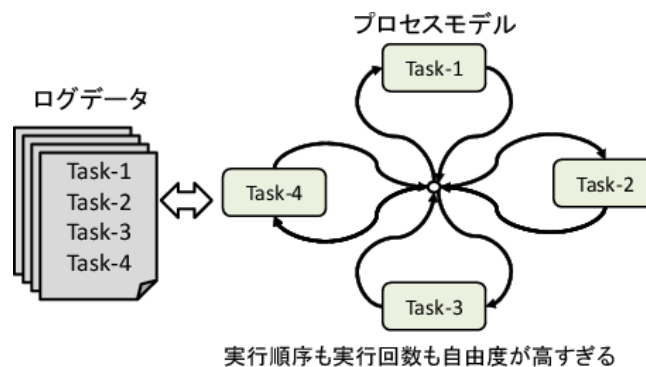


図 6: 適合性評価技術: underfitting の例 (花びら状に見えるフラワーモデル)

モデル (WfMC-TC-1003, Ver.1.1, 1995-01-19) [65] が、策定されており、ワークフローエンジンが5種類の外部のシステム/ツール/アプリケーションとの間で相互作用する API ないし情報交換フォーマットの規格も整備されつつある (文献 [65] の図 6)。そのうちの、プロセス定義ツールとのインタフェース (Interface 1; Process Definition In-

terchange) として、XML に基づくプロセス定義のための交換フォーマット **XPDL (XML Process Definition Language)** が策定されている (WfMC-TC-1025, Ver.2.00) [66]。XPDL は、XML によるテキスト表現で、後述する BPMN や YAWL のような具体的な図的表現とは独立に、プロセスを定義し交換することを可能にするを目的と

している。

本稿では、プロセス定義言語の詳細についての解説を目的としていないので、この節では、XPDLについて詳述せず基本的なプロセス定義に必要なとされる以下の、4つの経路 (routing) について、WfMCの基本資料 (WFMC-TC-1011, Workflow Management Coalition Terminology and Glossary, Issue 2.0, 1996-06^[67]; Issue 3.0, 1999-02^[68]) に基づいて紹介する。

- (a) 直列経路 (sequential routing)
- (b) 並列経路 (parallel routing)
- (c) 条件付き経路 (conditional routing)
- (d) 反復経路 (iterative routing)

プロセス定義としては、これら基本的な経路が表現できることが望まれており、したがってプロセス発見技術としても、ログデータからこれらの経路を発見することが期待されていることになる。

各経路におけるアクティビティの実行順序は下記の通りである:

- (a) 直列経路では、三つのアクティビティ task-a, task-b, task-c を順に実行する。
- (b) 並列経路では、task-a の実行後に、task-b と task-c を同時に、もしくは、任意の順序で実行した後、task-d を実行する。
- (c) 条件付き経路では、task-a の実行後に、task-b と task-c のどちらか一方を実行した後、task-d を実行する。
- (d) 反復経路では、アクティビティ task-b を1回以上任意の回数実行する。

これらは、アクティビティ間の順序を示すフローと、分岐/結合 (split/join) の対をなす制御ノードを導入することで表現できる。

- (a) 直列経路は、アクティビティを単に順序に従ってフローで連結することによって表現できる。
- (b) 並列経路は、フローが枝分かれしている部分で AND-分岐 (split), フローが合流しているところで AND-結合 (join), という、並行的な (fork/join に相当する) 制御ノードを導入しフローを明示的に制御することで表現できる。
- (c) 条件付き経路は、フローが枝分かれしている部分で OR-分岐 (split), フローが合流しているところで OR-結合 (join) という、選択的な制御ノードを導入しフローを明示的に

制御することで表現できる。OR-分岐/OR-結合のうち、特にどれか一つのフローのみを排他的 (exclusively) に遷移対象とする場合、特に XOR-分岐/XOR-結合とする。分岐には、一般に、遷移対象のフローを選択するためのゲート条件として述語 (predicate) を付与できる。²⁰ XOR-分岐/XOR-結合以外の OR-分岐/OR-結合では、複数のゲート条件が同時に成り立つことがありえる。その場合、複数のフローが同時に遷移可能となり並行実行される²¹。

- (d) 反復経路も、OR-分岐/OR-結合をつかってループバックさせることで表現できる。

こうしたフロー制御ノードを持つグラフ表現はビジネスプロセスの記述に広く使われている。XPDLでは、分岐/結合の対をなす制御ノードは、アクティビティの一種として **Route Activity** で表現でき、AND/(X)ORの区別は **gateway type** を指定することで表現できる^[66]。また、ADEPT Workflow Management System^[70, 71] や、後述するBPMNならびにYAWLも、アイコンの見た目は異なっても基本的にこのようなフロー制御ノードおよびその拡張を導入したグラフ表現を採用している。

4.2 構造化ワークフローモデル

前節 (第 4.1 節) で導入した、プロセスにおけるアクティビティの実行経路は、並列経路を除く直列経路、条件付き経路、反復経路の三つに関しては、構造化プログラミング (Structured Programming) を代表する三つの制御構造である順接 (Sequence), 条件分岐 (Selection), 反復 (Iteration) に対応する²²

そこで、構造化プログラミングと対応して、構造化ワークフローモデル (SWM; Structured Workflow Model)^[72] という概念を与えることができ、見通しの良いプロセスのためのガイドライン、ないしディシプリンとして推奨することができる。構造化ワークフローであれば、構造化

²⁰ゲート条件を導入することで決定的 (deterministic) な条件分岐を扱うことができるが、述語を与えなければ非決定的 (non-deterministic) に1つのフローが選択される

²¹ゲート条件が成立するフローのうちどれが選択されるかは非決定的である

²²余談になるが、構造化プログラミングは、goto文の排除だけを主張するものでも、順接/条件分岐/反復という制御構の利用だけを推奨したわけではない。

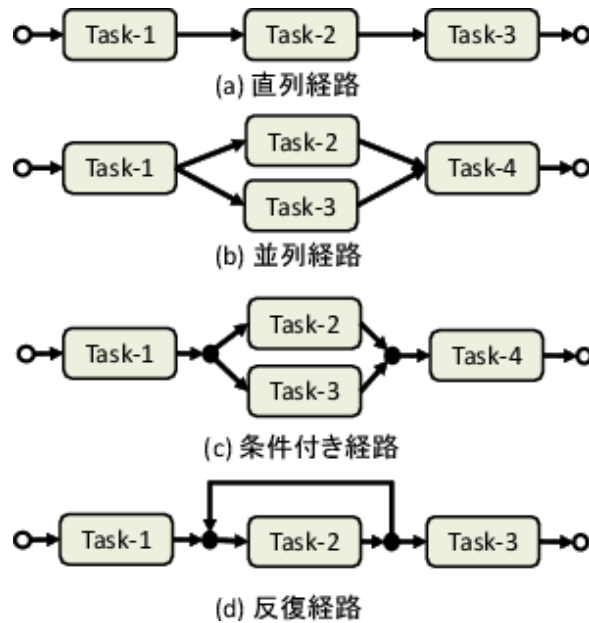


図 7: ワークフローの実行経路 (文献 [69] の図版をもとに作成)

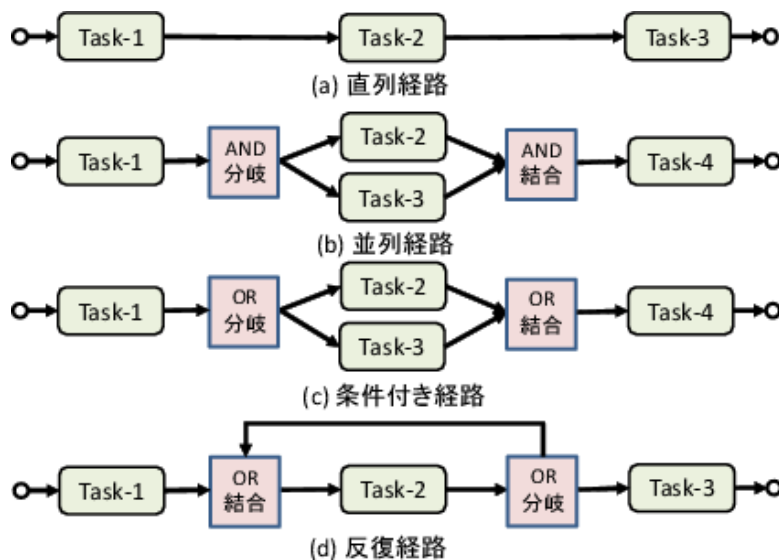


図 8: ワークフローの制御構成要素

プログラミングにおいて制御構文の導入により、入れ子になったブロック構造としてプログラムを記述できるのと同様にワークフローを理解することができる。すなわち、構造化ワークフローは、分岐/結合の制御ノードの対を、入れ子 (nesting) となるように配置することに相当し、以下のように帰納的に定義することができる [72]。

1. 単一のアクティビティのみからなるワークフローは SWM である。このアクティビティは、このワークフローの開始アクティビティでもあり、終了アクティビティでもある。

2. X と Y を SWM とする。 X の終了アクティビティから Y の開始アクティビティへのトランジションを追加して結合したワークフローも SWM である。このワークフローの開始アクティビティは X の開始アクティビティであり、終了アクティビティは Y の終了アクティビティである。
3. $X_1 \cdots X_n$ を SWM とし、 s を or-分岐とし j を or-結合とする。開始アクティビティを s と終了アクティビティを j とし、 or-分岐 s と各 X_i の開始アクティビティの間にトランジションを持ち、かつ、各 X_i の最終アク

ティビティと or-結合 j の間にトランジションを持つワークフローは、やはり SWM である。or-分岐 s の各出力トランジションにはゲート条件の述語を割り当てることができる。

4. $X_1 \cdots X_n$ を SWM とし、 s を and-分岐とし j を and-結合とする。開始アクティビティを s と終了アクティビティを j とし、and-分岐 s と各 X_i の開始アクティビティの間にトランジションを持ち、かつ、各 X_i の最終アクティビティと and-結合 j の間にトランジションを持つワークフローは、やはり SWM である。
5. X と Y を SWM とし、 s を or-分岐とし j を or-結合とする。開始アクティビティ j と終了アクティビティ s とし、or-結合 j と X の開始アクティビティの間、 X の最終アクティビティと or-分岐 s の間、 s と Y の開始アクティビティとの間、かつ、 Y の最終アクティビティと j との間にトランジションを持つワークフローは、やはり SWM である。

プロセスの定義は、通常、こうしたアクティビティの制御フローで表現されることが基本であり、そのなかで構造的なものがブロック構造(制御構造)で表現することができる。但し、実用的に使用されるプロセス定義では、各アクティビティ(やサブプロセス)に対して、使用するリソース、実行するアクター、事前条件や事後条件、起動するタイミングを示すイベントなどを記述したり、制御フロー以外にアクティビティ間でやり取りする情報(データやメッセージ、ドキュメント)のフローを追加して拡張しているものが多い。

4.3 BPMN とアクティビティ図

BPMN [73][74][75][76] は、ビジネスプロセスを定義するための標準的な図的な表現の一つであり、プロセスの直観的把握にすぐれている。元々は、複数の企業が標準案の策定を目指して参加していた BPMI(Business Process Management Initiative) から 2004 年 5 月に BPMN 1.0 として公開された標準仕様であり、Business Process Modeling Notation の略語とされていた。その後、BPMI は 2005 年 6 月 29 日に OMG(Object Management Group) との統合がアナウンスされ、BPMN の規格は、1.1, 1.2 という改版を経て、BPMN 2.0 [7]

として大きく改訂された。BPMN という名称も 2.0 より、Business Process Model and Notation の略とされている。

BPMN は図的表現を持っていることから、元々、プロセスの直観的把握性が高く、記述と分析(モデリング)にすぐれており、業務理解・要求獲得から仕様の分析・設計に重点があったと考えられる。実用的な本格使用が進むと一般的に起こりうることだが、次第に記述要素の種類も増え、曖昧性が排除されていくにつれ、後段で何らかの形で実装することを前提とした設計仕様という立場から、実行可能仕様/実行可能設計の立場へシフトしがちである。当初の BPMN が直接実行を意図していたかについては別としても、MfMC のリファレンスモデルにおいても、プロセス定義は、ワークフローエンジンで実行するための入力であるということから、それが直接実行されることは当然なことといえる。一方、BPMN で記述されたプロセス実行ということに関しては、直接実行よりも、特に BPEL(Business Process Execution Language) への変換が継続的に期待を集めてきた。このことは、OMG が標榜してきたモデル駆動アーキテクチャ(MDA; Model-driven Architecture) とも呼応する²³。

図 1 で、最初の簡単なビジネスプロセスの例を、UML²⁴ [6] のアクティビティ図で与えたが、そこで言及したように、BPMN の基本構造はアクティビティ図と似ている。とはいえ、スイムレーン(アクティビティパーティション)で区画化した制御フローという BPMN の基本構造がアクティビティ図に似ているために、アクティビティ図に馴染みがあれば、BPMN への参入障壁が低いという程度ともいえる。BPMN の方がずっとビジネスプロセスの記述に特化した記述要素を備えているため、基本的なプロセス定義のためには、アクティビティ図で BPMN の代用とすることができるとしても、本格的なビジネスプロセスの記述のためには BPMN を使用することが望ましい²⁵。BPMN のゲートウェイ、イベント、フローの表記法の一部と、アクティビティ図

²³変換のためのマッピングに関しては BPMN 2.0 の仕様書 [7] の 14 章でも言及されている

²⁴<http://www.uml.org/>

²⁵アクティビティ図は UML の一部であるためオブジェクト指向であり、BPMN はその成り立ちからプロセス指向である点が異なるという点が強調されることもあるが、そもそもアクティビティ図は UML の中でもプロセス指向的な図面であって、この点を強調することは本質的とはいえないと思われる。

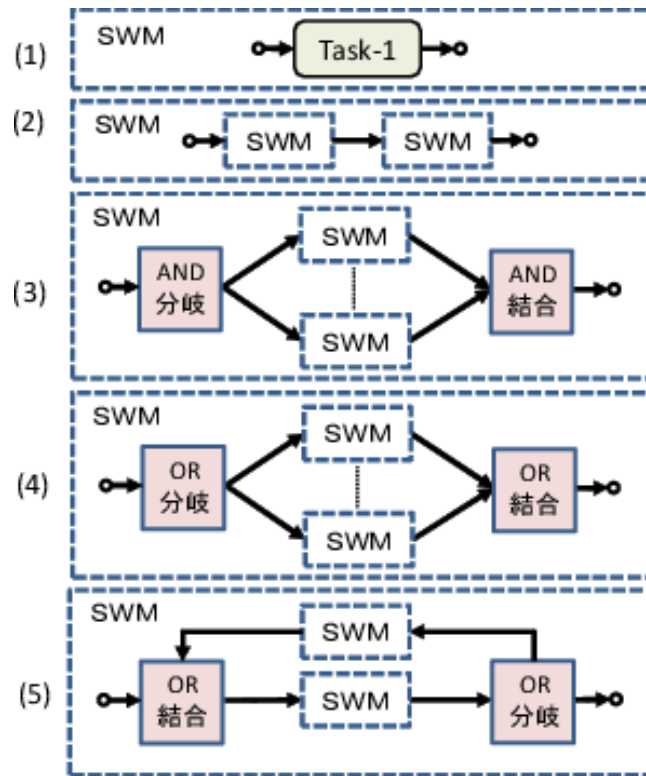


図 9: 構造化ワークフローモデル (SWM)

の条件分岐と並行分岐 (fork/join) の記法を図に示す。先に述べた通り両者の比較は文献 [77] に詳しい。BPMN は、2.0 となってさらにアクティビティ図との相違が広がり、スイムレーンの中に閉じているプロセスの定義（オーケストレーション）から、さらに主体間のコラボレーション (Collaboration) の記述力も強化され、カンバセーション図 (Conversation Diagram), コレオグラフィ (Choreography Diagram) が追加されている。これにより、最近、一層重要性が増している、組織間 (inter-organizational) コミュニケーションをプロセスとして把握し分析・設計の対象とすることが促進されると期待できる。

アクティビティ図の記述や実行 (シミュレーション) のためのツールとしては、例えば、Activiti BPM Platform²⁶, boc グループ²⁷の ADONIS²⁸, Bonitasoft 社²⁹の Bonita BPM³⁰などがある。

²⁶<http://activiti.org/>

²⁷<https://uk.boc-group.com/>

²⁸<http://en.adonis-community.com/>

²⁹<http://www.bonitasoft.com/>

³⁰<https://sourceforge.net/projects/bonita/>

4.4 ペトリネットに基づくプロセス表現

ペトリネット (Petri Net) [78][79] は、Carl Adam Petri が、1962 年に発表した離散事象システム (discrete event system) の挙動を表現できる図式表現であり、並行分散システムや通信プロトコルばかりではなく化学反応やシステム生物学に至るまで幅広く応用されている。

4.4.1 基本的なペトリネット (P/T ネット)

ここでは、トランジションにラベルを付与した、基本的なペトリネット (Place/Transition Net; P/T ネットと略す) をラベル付きの P/T ネットは、形式的には、ラベル付き有向 2 部グラフ $N = (P, T, F, W, L, m_0)$ で定義できる。

$$\left\{ \begin{array}{ll} P = \{p_1, p_2, \dots, p_{|P|}\} & \text{プレースの有限集合} \\ T = \{t_1, t_2, \dots, t_{|T|}\} & \text{トランジションの有限集合} \\ F \subseteq (P \times T) \cup (T \times P) & \text{アークの集合} \\ W : F \mapsto \{1, 2, \dots\} & \text{アークの重み} \\ L : T \mapsto \{l_1, l_2, \dots, l_{|T|}\} & \text{トランジションのラベル} \\ m_0 : P \mapsto \{0, 1, 2, \dots\} & \text{初期マーキング} \end{array} \right.$$

ここで、 $|P|$ はプレース数で、 $|T|$ はトランジション数である。

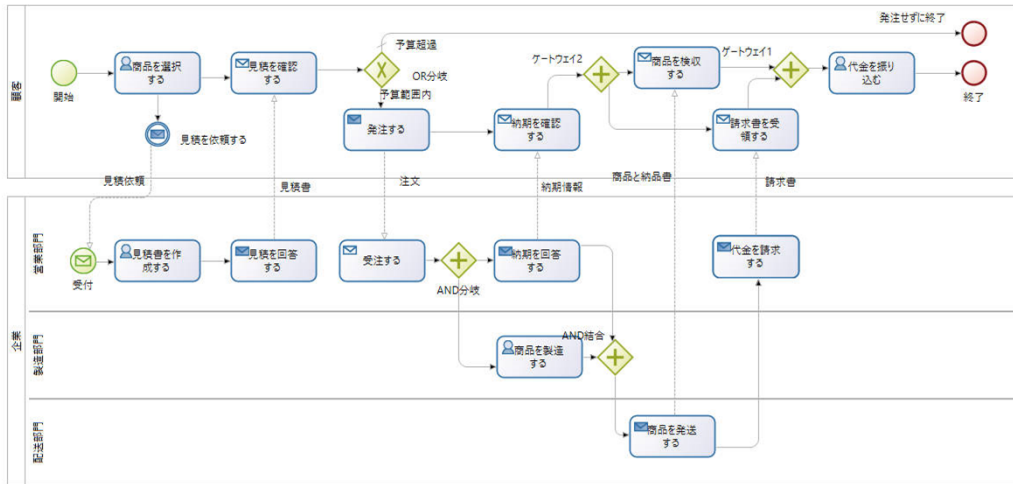


図 10: BPMN の記述例 (Bonita BPM を使用して作成した)

ゲートウェイ	AND	XOR	OR		
タスク	タスク	サービスタスク	ヒューマンタスク		
	受信タスク	送信タスク	スクリプトタスク		
イベント	開始	開始イベント	タイマー開始イベント	シグナル開始イベント	メッセージ開始イベント
	中間	スロー	キャッチ	タイマーイベント	スロー
	終了	開始イベント	停止イベント	シグナル終了イベント	メッセージ終了イベント
フロー	シーケンスフロー	メッセージフロー			

図 11: BPMN の記述要素の一部 (Bonita BPM のアイコンの一部を利用した)

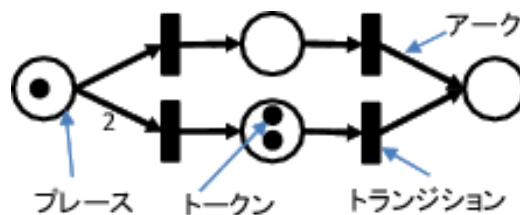


図 12: 基本的なペトリネットの例

P と T は、それぞれプレース (place) とトランジション (transition) と呼ばれるノードの有限集合であり、それぞれ白丸と四角形で図示される。 F はノード間を接続する有向アークの有限集合である。これらによりシステムの「構造」が規定される。アークには、プレースからトランジ

ションへ向かうアーク (トランジションへの入力アーク) と、トランジションからプレースへ向かうアーク (トランジションの出力アーク) の二種類がある。あるトランジション t_i を取り上げたとき、そのトランジションへの入力アークの始点となるプレースを入力プレース、出力アークの終点

となるプレースを出力プレースと呼ぶ。始点と終点を同じくするアークが複数ある場合、その本数(1以上の整数)を**重み (weight)**として、重みをラベルとする一本のアークで表現することができる(アークのラベルで重みが省略されたとき1であるとみなす)。それぞれのトランジションにはラベルを対応付けているが、これは、P/Tネットの定義としては必須ではない。ワークフローを表現する際に、アクティビティを識別するため(もしくは制御ノードの種類を識別するため)の名前として使うことを意図している。

プレース上には、小さな黒丸で図示される非負整数個のトークン(**token**)を置くことができる。 $m(p_j)$ をプレース p_j のトークン数とするとき、すべてのプレースのトークン数を列挙したベクトル $m = [m(p_1) m(p_2) \cdots m(p_{|P|})]^T$ をマーキングと呼び、その時点でのシステムの「状態」を表現する。

トークンは、後述する(トランジションの)発火則(**firing rule**)に従って、トランジション t が発火(**firing**)することにより、入力アークと出力アークを通して遷移してプレース間を移動(遷移)する。これは、入力プレース上のトークンを消費して、出力プレース上にトークンを産出する、と解釈してもよい。これによりシステムが状態の遷移を引き起こすことになる。システムの初期状態は、初期マーキング m_0 で表現される。

トランジション t_i は、 t_i のすべての入力プレースが入力アークの重み以上の個数のトークンを持つとき、マーキング m で発火可能(**fireable, enabled**)であるという。発火可能なトランジションが同時に複数存在することがありえるが、その場合、どのトランジションが発火してもよい。トランジション t_i が発火すると、そのすべての入力プレースから、対応する入力アークの重みの分だけトークンを取り除き(消費し)、すべての出力プレースに、対応する出力アークの重みの分だけトークンを追加する(産出する)。

ペトリネットの構造を、行列(**matrix**)として表現することができ(入力接続行列, 出力接続行列, および、そのから得られる接続行列), 発火前のマーキングから発火後のマーキングを計算することができる。また、ペトリネットの挙動に関して、到達可能性(**reachability**), 活性(**liveness**), 有界性(**boudedness**)といった性質を解析することが重要である。これらに関しては、本稿では紙数の都合上、説明を省略する。

ペトリネットには、望ましい性質を備えるように、構造的に制約を加えたサブクラスが複数知られている。後述するワークフローネット(**WF-net**)は、その一つである。

また、ペトリネットには時間の概念を取り入れて拡張したもの(Timed Petri Net やTime Petri Net, Timed Arc Petri Net など)や、確率の概念を取り入れたもの(Stochastic Petri Net), 離散値だけでなく連続値も扱えるもの(Hybrid Petri Net), トークンへの識別性や階層的構造化などを導入した高水準ペトリネット(High Level Petri Net)などの拡張がある。後述する色付きペトリネット(**Coloured Petri Net**)は高水準ペトリネットの一つである。

ペトリネットの記述, 解析(構造解析, 性能解析など), シミュレーション実行(トークンゲームアニメーションによる実行の可視化)のためにつかうことのできるツール類に関しては、<https://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/>に情報がある。

4.4.2 ワークフローネット (WF-net)

この節では、ペトリネットによるワークフローの表現について述べる [80][69][81][82][83][84][85][86]。

ワークフローの実行経路は、構造的には図13のような基本的なペトリネットに対応すると考えられていた [69][81]。トランジションでアクティビティを表現する。

そこで、split/joinの制御ノードをトランジションとして導入し、ペトリネットと融合した表現に、ワークフローネット(**Workflow Net**) [69][81][82]がある。ここでは、アクティビティの実行に対するトリガーとして、ユーザによるもの、メッセージの到着によるもの、タイマーによるものなど、表現力が拡張されている。このワークフローネットのための定義と解析のためのツールに、WoPeD (Workflow Petri Net Designer)³¹がある。

さらに、ワークフローとして記述できることが望ましいパターン(**Workflow Pattern**)³² [87][88]が認識され分類整理されて、基本的なペトリネットでは表現が困難であることが判明し、実用的にそれらを表現できる拡張表現とツールとして、後述するYAWL(Yet Another Workflow Language)が開発されている。

³¹<http://woped.dhbw-karlsruhe.de/woped/>

³²<http://www.workflowpatterns.com/>

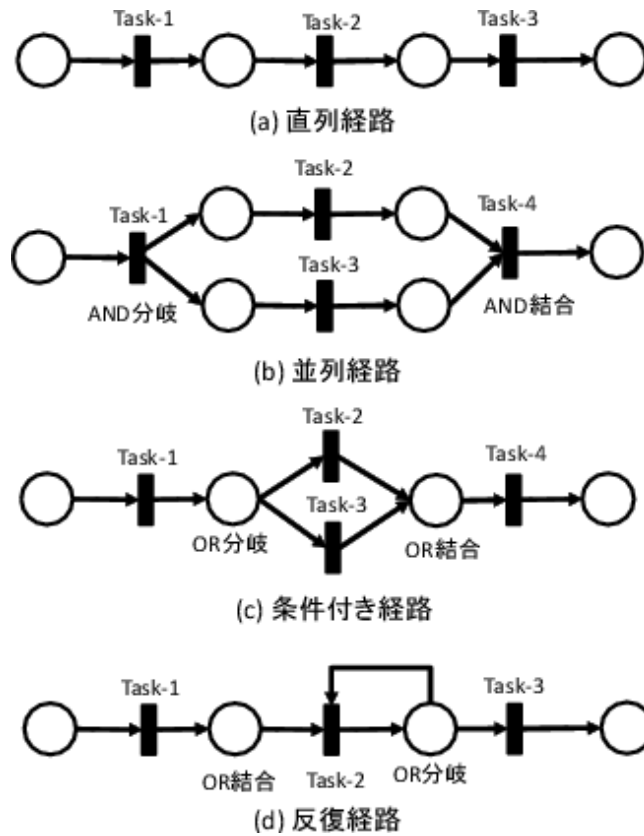


図 13: 基本的なペトリネットで表現したワークフローの実行経路 (文献 [69] の図版をもとに作成)

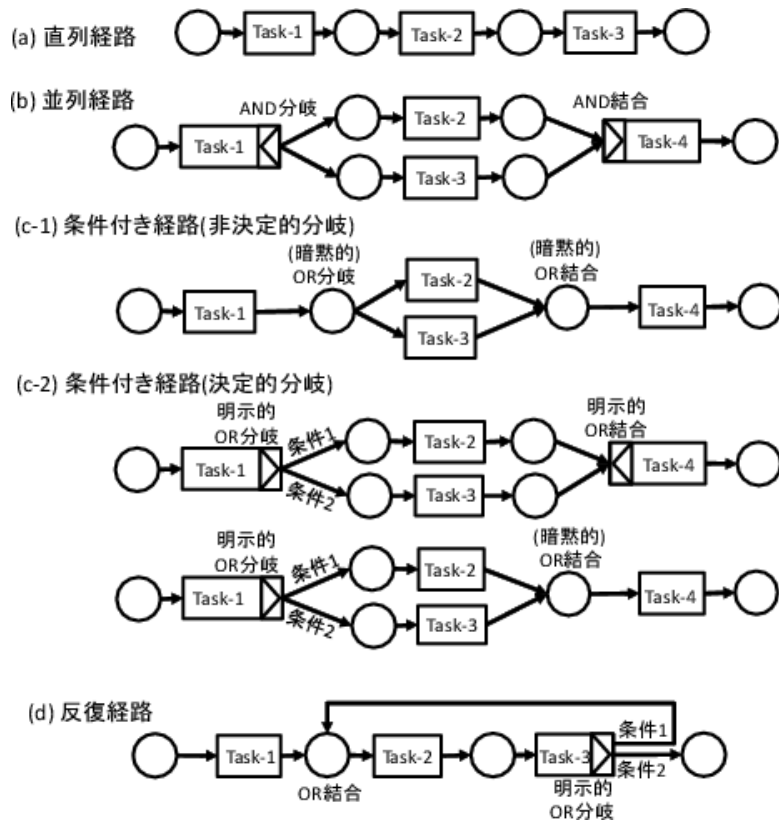


図 14: ワークフローネットで表現したワークフローの実行経路 (文献 [69] の図版をもとに作成)





(a) 自動(automatic)	
(b) 利用者(user)	
(c) メッセージ(message)	
(d) 時間(time)	

図 15: ワークフローネットにおけるトリガーの表現 (文献^[69]の図版をもとに作成)

健全性 (Soundness) 等の性質を理論的に解析するために基本的なペトリネットのサブクラスとして、ワークフローネット (WF-net) を規定する。ワークフローを表現するために望ましい性質は二つあり、これを満たすことで WF-net の定義とする。一つ目は、WF-net は入力トランジションを持たない唯一のプレース i と出力トランジションを持たない唯一のプレース o を持つことである。 i にトークンがある状態でワークフローは開始され、 o にトークンがある状態で終了する。二つ目は、使われないトランジション (アクティビティ) も、使われないプレース (条件) もないということである。したがって、どのノードも、 i から o までの経路 (path) 上にあることを意味する。これは、さらに、トランジション t を新たに追加して、プレース o からプレース i へ接続したときに、強連結となることに対応する。形式的な定義は、紙数の都合上省略するので、文献^{[69][82][89]}を参照してほしい。

ワークフローとして適切なペトリネットであるかを構造的に解析する解析/検証ツールとして、Woflan³³^{[90][91][92][93][94]}が開発されている。

4.4.3 色付きペトリネット

基本的なペトリネットの拡張である高水準ペトリネット (High Level Petri Net) の一種として、色付きペトリネット (Coloured Petri Net)^{[95][96][97][98][99]}があり、ビジネスプロセスの表現とし

て使われている。色付きペトリネットは、通常、単なる小さい黒丸で図示されるトークンに識別性を持たせたものであり、色付きペトリネットという名称は、その識別性を「トークンにつけた色」と表現したことに由来している。

ビジネスプロセスの記述、解析、シミュレーション実行のためのツールとして、CPN Tools³⁴や ExSpect³⁵がよく使われている。

4.4.4 YAWL

YAWL (Yet Another Workflow Language)³⁶^{[100][101][102]}は、Eindhoven University of Technology (TU/e) と Queensland University of Technology (QUT) を中心に開発された制御フローに基づくビジネスプロセスモデリング言語と、そのサポートシステムであり、実用的なレベルにある。先述したように、ワークフローパターン (Workflow Pattern)^{[87][88]}を記述できることを目指しペトリネットをベースに開発されたが、ペトリネットとしての意味合いは薄れている (図 17(a)は一応、ペトリネットとの互換性をとどめているが、図 17(b)の表現はむしろ、図 8の表現に近い。YAWL の記述要素の一部を図 18 に示す。

YAWL は後述する BPEL と並んで、ワークフローの実行言語としての側面も持ち、BPMN から YAWL へのマッピングも試みられている^{[103][104]}。

³⁴<http://cpntools.org/>

³⁵<http://www.exspect.com/>

³⁶<http://www.yawlfoundation.org/>

³³<http://www.win.tue.nl/woflan/doku.php>

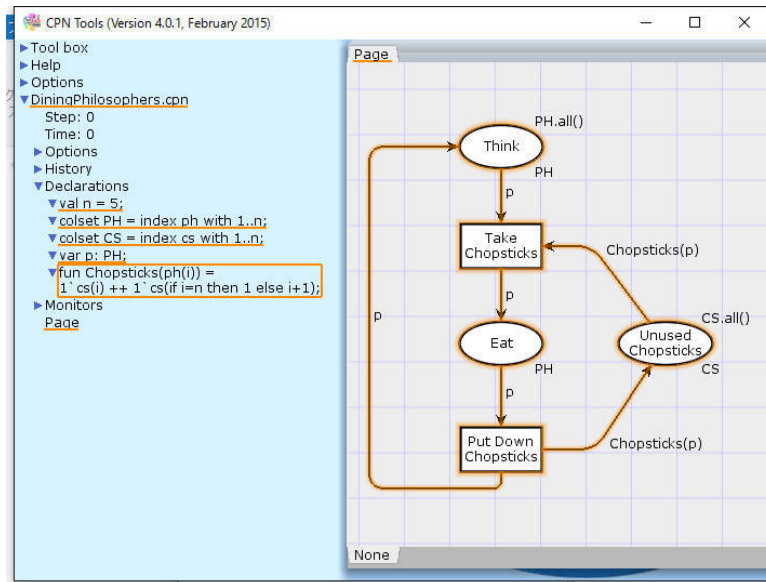
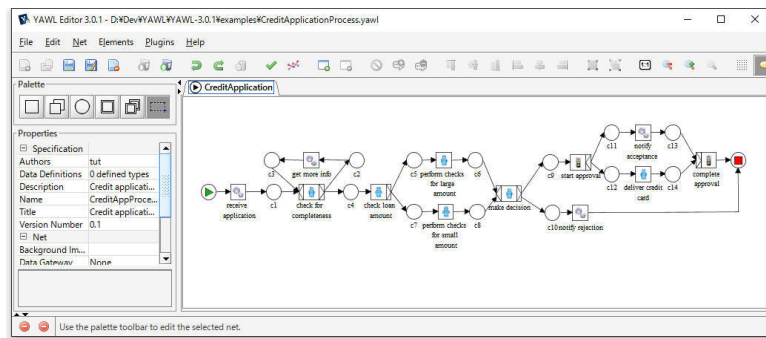
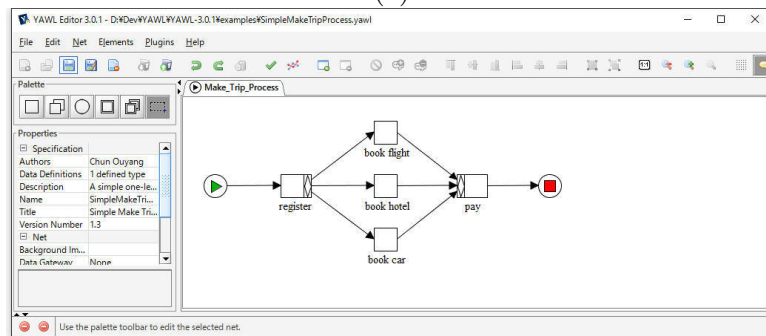


図 16: 色付きペトリネットの例 (哲学者の食事問題; CPN Tools のエディタ画面)



(a)



(b)

図 17: YAWL 記述の例 (YAWL editor の画面表示)

4.5 その他のプロセス表現

4.5.1 IDEF0 と IDEF3 およびその融合

IDEF(Integrated DEFinition) Methods family³⁷は、元々は、米空軍 (U.S. Air Force) の I-

³⁷<http://www.idef.com/>

CAM(Integrated computer-aided Manufacturing; 統合計算機支援製造) プログラムにおける定義言語として生まれ、同プログラムの終了後は、その一部が、広く一般に使われている。一応、IDEF0 から IDEF14 まで定義されてはいるが (ほとんど定義されていないものもあり)、十分に定義され、ソフトウェア工学においてよく参照されて








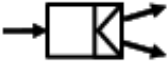


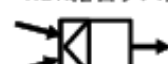


 条件(Condition)	 原子的タスク (Atomic Task)
 入力条件 (Input Condition)	 複合タスク (Composite Task)
 出力条件 (Output Condition)	 原子的タスクの複数インスタンス (Multiple instances of an Atomic Task)
	 複合タスクの複数インスタンス (Multiple instances of a Composite Task)
AND分岐タスク 	AND結合タスク 
XOR分岐タスク 	XOR結合タスク 
OR分岐タスク 	OR結合タスク 

図 18: YAWL の記述要素の一部 (文献 [101] の図版をもとに作成)

いるのは、構造化分析設計手法 (SADT; Structured Analysis and Design Technique) に基づく IDEF0(Function Modeling Method; 機能モデリング手法) ³⁸と、そのデータモデリング面を補完する IDEF1(Information Modeling Method; 情報モデリング手法) ³⁹ / IDEF1X(Data Modeling Method; データモデリング手法) ⁴⁰ である。それに準じて、IDEF3(Process Description Capture Method; プロセス定義獲得手法) ⁴¹, IDEF4(Object-Oriented Design Method; オブジェクト指向設計手法) ⁴², IDEF5(Ontology Description Capture Method; オントロジ記述獲得手法) ⁴³ が参照さ

れている。

IDEF0 ^[105] は、機能モデリング手法であるためデータフローグラフの図式表現を備えており、IDEF3 ^[106] は、プロセス定義獲得手法であるためプロセス記述言語の図式表現を備えている。IDEF3のプロセス記述言語 (IDEF3 Process Description Language) でプロセスのモデリングができることは当然といえるが、それに対して、IDEF0と IDEF3の融合手法 (Hybrid IDEF0-IDEF3) が提案されている ^{[107][108][109]}。そこでは、まず IDEF0をもとに(すなわちデータフローとして)アクティビティを定義する。必要に応じて、入力 (Input), 出力 (Output), 制御 (Control), 機構 (Mechanism) を明記する。さらにアクティビティの入出力データフローの扱いに IDEF3の制御フローのモデル化を取り入れていく。

³⁸http://www.idef.com/idefo-function_modeling_method/

³⁹http://www.idef.com/idef1-information_modeling_method/

⁴⁰<http://www.idef.com/idef1x-data-modeling-method/>

⁴¹<http://www.idef.com/idef3-process-description-capture-method/>

⁴²<http://www.idef.com/idef4-object-oriented-design-method/>

⁴³<http://www.idef.com/idef5-ontology-description-capture-method/>

4.5.2 イベント駆動プロセス連鎖 (EPC) 図

イベント駆動プロセス連鎖図 (EPC; Event-driven process chain)⁴⁴ [110][111][112] とは, August-Wilhelm Scheer (University of Saarland の Institute for Business Information Systems) らによる ARIS (Architecture of Integrated Information Systems) フレームワークのためのビジネスプロセスモデリング言語として開発され, SAP 社の ERP (Enterprise Resource Planning) パッケージ SAP R/3 に採用されている。モデリング言語のタイプとしては, やはり制御フローをベースとしている。

ノードとしては, アクティビティを中心に, そのトリガーイベントと結果イベントを記述し, それをつなげていき, コネクタで条件分岐 (Branch/Merge) や並行分岐 (Fork/Join) で制御フローが表現できる。しかし, アクティビティとして, 入力を出力に変換する機能 (Event) を記述し, 入力 (Input) や出力 (Output) となる情報やリソースを明示することで, データフローも表現できる。その他, サポートシステム (Supporting System) や, アクターとなる組織単位 (Organization Unit) もノードとして図示できる。

4.5.3 BPEL

BPEL (Business Process Execution Language) [113] [114] は, モデリング言語というよりも, その名の通り, 実行言語に相当する。そこで, 他のプロセスモデリング言語から BPEL へ変換して実行するための言語間マッピングや他の実行言語への変換 [115] が重要である。

元々は, Web サービス合成のための言語として, 2001 年に, IBM によって定義された WSDL (Web Services Flow Language) と, マイクロソフトによって定義された XLANG を, 2003 年に統合し BPEL4WS と称していたものであるが, OASIS Web Services Business Process Execution Language (WSBPEL) TC (技術委員会)⁴⁵ によって, WS-BPEL 2.0 仕様として名称が変更された。

BPEL は WSDL と XLANG の統合であるということにより, 言語マッピングの面で問題も抱えている。WSDL の構文は制御フロー指向 (グラフ指向) であり, XLANG の構文は制御ブロック指

向であり, BPEL はそのハイブリッドとなっており, どちらの構文も選ぶことができる。この経緯から BPEL のセマンティクス自体に曖昧さが残り言語マッピング上の問題となっている [116][117]。

また, 実行言語としては, 「人手で実行するタスク (manual task)」を含むワークフローの実行言語というよりも, 元々は Web サービスの自動実行 (のための一種のプログラミング言語) を目的としていたこともあり, 「人によって実行されるタスク (Human task; Manual Activity)」の面が弱かったが, その後, その方面の拡張もなされてきている [118] [119]。

4.5.4 因果関係ネット (Causal Net)

因果関係ネット (C-net; Causal Net) [120] [1] は, プロセス発見のための制御フローに基づいたプロセスモデリング表現である。タスクから流出する出力フローの根元に出力バイインディングとして, 関連しあう責務 (obligation) の生成を表示し, タスクに流入する入力フローの矢印の近くに入力バイインディングとして, 責務の解消を表示する。本連載の第 4 回で紹介する予定のプロセス発見アルゴリズムのうちのいくつか (ヒューリスティックマイニング, ファジーマイニング, ジェネティックマイニング) で使用するので, それらの解説の折にあらためて紹介する。

4.5.5 S-BPM

サブジェクト指向ビジネスプロセス管理 (S-BPM; Subject-oriented business process management) [121] [122] [123] は, サブジェクト指向プログラミング [124] の概念に基づく, ビジネスプロセスのモデリング手法である。ビジネスプロセスという文脈で言えば, サブジェクトとは, 個々の主体の側からとたえた「業務」モデルとといえるかもしれない⁴⁶。サブジェクト指向では, 「業務」を認識しモデル化するにあたって, 主体間のコミュニケーションの理解が重要となる。

前述のように, BPMN 2.0 におけるコレオグラフィ図, カンパセーション図の導入は, 主体間

⁴⁶ここでの「サブジェクト」という語は翻訳しにくい「主題」という訳語が使われることがある。もっとも S-BPM では, 主体を指すのに「サブジェクト」(主語) という語を使うこともあり, 混乱を招きやすい。特に自然言語文における主語/述語/目的語 (Subject/Predicate/Object) に言及するときには, 主体を指し示すのに, しばしば「アクター」という語の代わりに「サブジェクト」という語を用いている。

⁴⁴ <http://www.ariscommunity.com/event-driven-process-chain>

⁴⁵ <https://www.oasis-open.org/committees/wsbpel/>

のコミュニケーションによって実現されるコラボレーションを重視し、組織間にまたがる (inter-organizational) ビジネスプロセスの定義・分析に対する要求の高まりに呼応していると考えられる。主体間のコラボレーションをプロセスとして抽出したコレオグラフィもプロセスマイニングの対象として重要性を増してくることは想像に難くない。

S-BPMのモデリング言語としては、特に主体間のコミュニケーションに関して、Robin Milnerの形式手法プロセス代数 CCS (Calculus of communicating systems) [125] [126], および C.A.R.Hoareの CSP (Communicating Sequential Processes) [127] [128] に基づくPASS (Parallel Activity Specification Scheme) を用いており、モデリングツールに Metasonic 社の Metasonic Build⁴⁷がある。

4.5.6 関連する会議

毎年、活発に開催されているビジネスプロセス管理一般に関する会議 Conferences on Business Process Management⁴⁸や、Asia-Pacific Conference on Business Process Management⁴⁹などにおいても、ビジネスプロセスモデリングに関する発表が数多くなされている。また、S-BPMに関しては、2010年以降 S-BPM One⁵⁰が開催されている。

5 プロセスマイニングマニフェスト

この節では、プロセスマイニング技術の目指す方向と解決すべき課題として、どのような方向性が意識されているのかについて、土地勘を付けることを目的に一部抜粋して紹介する。

5.1 プロセスマイニングマニフェストとは

プロセスマイニングという研究分野は、「計算知能」や「データマイニング」と、「ビジネス(業務)プロセスモデリング/分析」の領域にまたがっており、両者の橋渡しを担う学際的領域である。この新しい研究分野であるプロセスマイニングを、より活発な議論を通して成熟させることを目的と

⁴⁷旧称 jPASS

⁴⁸<http://www.bpm-conference.org/BpmConference/WebHome>

⁴⁹<http://ap-bpm.org/>

⁵⁰<http://www.s-bpm-one.org/home/>

して、プロセスマイニングマニフェスト^[2, 3, 4](以降、マニフェスト)が編纂された。

マニフェストの中には、まだ若い研究分野であるプロセスマイニングが、今後どのように発展していくべきかという指針と、クリアしていく必要がある課題が示されている。

5.2 指針

Table 1 に 6 つの指針を示す。それぞれの指針を GP1 から GP6 とする。

5.2.1 GP1 イベントデータは第一級市民として扱われるべきだ

本指針は、イベントデータ(記録されたイベント)の「品質」について論じている。言うまでもないが、大半のイベントデータは、プロセスマイニングのために出力されているわけではなく、プログラムのデバッグ出力のような副産物である。そこで、マニフェストでは、イベントログ(イベントの集合)の成熟度レベルを定義している(Table 2)。マニフェストでは、レベル3以上の成熟度のイベントログはプロセスマイニングが適用可能としている。つまり、イベントログの品質をプロセスマイニングの入力たりえる水準(成熟度3以上)に引き上げていこうというのが本指針の趣旨である。

5.2.2 GP2 ログの抽出は、疑問点に応じて行うべきだ(疑問点駆動型)

イベントデータが膨大に存在するとき、その全てが宝の山というわけではない。本指針が指摘しているのは、具体的な疑問点を設定してイベントログを選択しなければ、膨大なイベントデータから有益なマイニングの成果を得ることが非常に困難であるということである。

5.2.3 GP3 並列分岐, 選択分岐, その他の基本的な制御フロー構成をサポートすべきだ

ビジネスプロセスには種々の表記法が存在している。本指針が示しているのは、プロセスマイニングでモデルを表現する上で、最低限の制御フローがサポートされている必要があるということであ

表 1: プロセスマイニングマニフェストにおける指針

番号	内容
GP1	イベントデータは第一級市民として扱われるべきだ
GP2	ログの抽出は、疑問点に応じて行うべきだ (疑問点駆動型)
GP3	並列分岐, 選択分岐, その他の基本的な制御フロー構成をサポートすべきだ
GP4	イベントはモデル要素に関連しているべきだ
GP5	現実の意図的な抽象化としてモデルを扱うべきだ
GP6	プロセスマイニングは継続的なプロセスであるべきだ

表 2: プロセスマイニングマニフェストにおけるイベントログの成熟度レベル

レベル	内容
*****	イベントログが信頼できて完全で、イベントが明確に定義されている
****	イベントが自動的に、体系的かつ信頼できる方法で記録される
***	イベントは自動的に記録されるが、体系的な取り組みになっていない
**	イベントは情報システムの副産物として自動的に記録される
*	イベントログの品質が悪い

る。最低限のフローには並分岐列, 選択分岐, その他の基本的な制御フロー構成が含まれる。

5.2.4 GP4 イベントはモデル要素に関連しているべきだ

プロセスマイニングで実施可能なことは、イベントログをマイニングしてプロセスのモデルを作成することに限らない。すでに存在するモデルや、マイニングで作成したモデルの上で、イベントをリプレイ (再生) することも可能である。これによって、イベントログとモデルの間に不一致を見付けることが可能となり、これは適合性検証と呼ばれる。本指針が示しているのは、適合性検証を可能とするために、プロセスのモデルに含まれる個々の要素と、個々のイベントとが、一対一で対応していることが望ましいということである。

5.2.5 GP5 現実の意図的な抽象化としてモデルを扱うべきだ

プロセスマイニングによって作り出されるモデルは、現実世界の一つの捉え方でしかない。言い換えれば、モデルには決まった正解がない。そこで、本指針では、モデルを見る人の利便性を考えて、モデル自体を抽象化することが望ましいと述べている。

5.2.6 GP6 プロセスマイニングは継続的なプロセスであるべきだ

本指針では、プロセスマイニングを一度きりのモデル作成に終わらせず、継続していくことを提案している。その効果の例として、Google マップをメタファーとして、プロセス内をナビゲートしたり、プロセスの「交通渋滞」情報をリアルタイムに表示したり、未来を予測したりすることが可能であるのに、現状では実施されていないと述べている。

5.3 課題

Table 3 に 11 の課題を示す。それぞれの課題を C1 から C11 とする。

5.3.1 C1 イベントデータの検索, マージ, クリーニング

本課題は、プロセスマイニングに適したイベントを抽出することの難しさについて述べている。例えば、イベントログの保存先が分散している場合に、それらをマージすることや、イベントログに異常値が含まれている場合に、その異常値を取り除くことが必要である。

表 3: プロセスマイニングマニフェストにおける課題

番号	内容
C1	イベントデータの検索, マージ, クリーニング
C2	多様な特性を有する複雑なイベントログを扱う
C3	代表的なベンチマークを作成する
C4	コンセプトドリフト (分析中にプロセスが変化している状況) の扱い
C5	プロセス発見に使用される表現バイアスを改善する
C6	フィットネス (適合度), 単純さ, 精度, 一般性など品質基準間のバランス
C7	組織横断的なマイニング
C8	運用サポートの提供
C9	他のタイプの分析と組み合わせたプロセスマイニング
C10	非専門家向けにユーザビリティを向上する
C11	非専門家向けに理解度を向上する

5.3.2 C2 多様な特性を有する複雑なイベントログを扱う

本課題は, イベントログが様々な特性を持っていて, 分析の簡単なイベントログもあれば, そうでないイベントログもあるということを描べている. 具体的には, 事例あたりの平均イベント数, 事例間の類似性, ユニークなイベント数, ユニークなパス数が挙げられている.

5.3.3 C3 代表的なベンチマークを作成する

多くの技術には, その性能を客観的かつ定量的に測定するためのベンチマークが存在する. しかしながら, プロセスマイニングはまだ新しい技術であるため, どのような指標をベンチマークに用いればよいのか, あるいは, どの程度の数値であれば優れていると言えるのかといった事実が確立されていない. 本課題は, プロセスマイニングが今後発展していく中で, 誰からも支持されるような優れたベンチマークを確立していく必要があることについて述べている.

5.3.4 C4 コンセプトドリフト (分析中にプロセスが変化している状況) の扱い

コンセプトドリフトは, プロセスマイニングの分析中に現実のプロセスが変化している状況を意味する. コンセプトドリフトが発生しているプロセスは, 時間の経過と共に変化する. プロセスマイニングの適用対象プロセスも, 定常状態のプロセスは滅多に存在しない. プロセスマイニングに

おいて, コンセプトドリフトによるプロセスの変化を取り扱う (検出・分析する) ことは非常に重要である.

5.3.5 C5 プロセス発見に使用される表現バイアスを改善する

プロセスマイニングによって発見されたプロセスは, モデルとして記述される. このとき, モデルを記述する言語によって, 表現可能な結果の範囲が異なる. このことは, プロセスマイニングの結果が, モデルの記述言語によって制約されてしまうことを意味する. 本課題は, 分析結果を表現する言語を適切に選ばなければならないことを指摘している.

5.3.6 C6 フィットネス (適合度), 単純さ, 精度, 一般性など品質基準間のバランス

本課題は, イベントログをマイニングするアルゴリズムの4つの特性について述べている. 「フィットネス (適合度)」は, イベントログの挙動をよく説明することが可能であることを意味する. 「単純さ」に関していえば, もっとも単純なモデルは最高のモデルである⁵¹が, 「精度」を始めとするその他の3つの特性と相反することが多い. また, 「一般化 (汎化)」が行われていなければ, 最初に与えられたイベントログ以外をよく説明できなくなる. 4つの特性を同時にバランスよく満たすことは難しく, そのようなアルゴリズムの開発

⁵¹このことを「オッカムの剃刀」という

が待たれる。

5.3.7 C7 組織横断的なマイニング

本課題は、現状では単一組織内で行われているプロセスマイニングを、今後、複数組織にまたがって適用していく必要があることを述べている。複数組織にまたがったプロセスマイニングには2種類ある。1つは、異なる組織のログをマージして1つのプロセスインスタンスとすることである。もう1つは、例えば地方公共団体のように複数存在する似たような組織が、同様のプロセスを実行する場合である。

5.3.8 C8 運用サポートの提供

本課題は、これまでオフラインでの履歴の分析として行われてきたプロセスマイニングを、よりリアルタイムに近い、運用サポートとして実施していく必要があると述べている。具体的には、「検出」「予測」「推奨」の3種類が運用サポート活動が挙げられている。例えば、事前に定義されたプロセスから逸脱した場合に即時にアラートを出したり、履歴データから予測モデルを構築したりといったことがある。

5.3.9 C9 他のタイプの分析と組み合わせたプロセスマイニング

本課題は、プロセスマイニング技術を、オペレーションリサーチや、視覚的な分析と組み合わせる必要があると述べている。オペレーションリサーチは、プロセスマイニングと近縁で、大規模な数学的モデルを扱う分野である。ここでの課題は、プロセスマイニングと近縁の技術をどのように連携させるかである。具体的には、履歴データに基づいてシミュレーションモデルを作成することがマニフェストでは検討されている。同様に、視覚的な分析との一体化は、プロセスマイニングをインタラクティブな技術とすることで、イベントデータからより多くの洞察を得ることを目的としている。

5.3.10 C10 非専門家向けにユーザビリティを向上する

本課題は、プロセスマイニングの結果が、専門家ではなくユーザによって日々更新されることを目指したものである。一般に、プロセスマイニングは専門家が実施するものである。しかし、すでに述べたように、プロセスは日々変化するものであるから、専門家ではなくユーザが常に実施できることが望ましい。

5.3.11 C11 非専門家向けに理解度を向上する

一般に、プロセスマイニングの結果は、専門家でないとは正しいかどうか判断できない。ユーザが出力結果を理解できない場合や、間違った結論を出してしまうこともある。そこで、本課題は、プロセスマイニングの結果を、ユーザにも理解できるようにしていくことを目指したものである。具体的には、アンダーフィッティング(一致しない傾向が高い)、オーバーフィッティング(一致しすぎていて)、データが少なすぎる、といった場合の警告を出すことが必要である。

6 おわりに

今回は、全5回連載を予定しているプロセスマイニング技術に関する解説の第一回として、連載の概要と基本概念について紹介した。今回は、プロセスマイニングの基本概念に加え、これからプロセスマイニング技術の詳細に踏み込む上での前提となる、ビジネスプロセスのモデリング言語について簡単に紹介した。さらに、プロセスマイニングマニフェストの内容の一部も合わせて紹介し、プロセスマイニング技術の研究開発の上での指針と課題を概観した。

次回は、プロセスマイニング技術の中身(データの書式やアルゴリズム)の詳細に足を踏み入れる前に、実際に小さなサンプルでツールを使用して試しながら、理解を深めることができるよう無償で使用できるツールを中心に紹介する予定である。

今回の執筆分担:

この連載解説全体と本稿(第一回)の企画は、飯島が基本となる原案を立案した上で、執筆者全員で検討/協議/調整を重ねた。連載の今回以降の回も、概ね企画原案に沿いつつも随時調整しながら

進めていく予定である。本稿の執筆分担は、第1節(図1を除く)、第2節、第3節、第4節、および第6節を飯島が担当し、第5節と第1節の図1を田端と斎藤が担当した。全体を通しての用語の統一は飯島が調整した。

文献

- [1] W. van der Aalst, Process Mining: Discovery, Conformance and Enhancement of Business Processes, Springer, 2011, <http://www.springer.com/jp/book/9783642193446>, <http://www.processmining.org/book/start>.
- [2] W. van der Aalst, A. Adriansyah, de Medeiros, et al., "Process Mining Manifesto," BPM 2011 Workshops(Part I), (eds.) F. Daniel, K. Barkaoui, and S. Dustdar, LNBIP-99, pp.169-194, Springer, 2012, The original version: <http://www.springer.com/jp/book/9783642281143>.
- [3] W. van der Aalst, A. Adriansyah, de Medeiros, et al., "Process Mining Manifesto," , 2012, The final version: http://www.win.tue.nl/ieetfpm/doku.php?id=shared:process_mining_manifesto.
- [4] W. van der Aalst, A. Adriansyah, de Medeiros, et al., "プロセスマイニングマニフェスト (最終版)," , 2012, <http://www.win.tue.nl/ieetfpm/lib/exe/fetch.php?media=shared:pmm-japanese-v1.pdf>.
- [5] A. Burattin, Process Mining Techniques in Business Environments – Theoretical Aspects, Algorithms, Techniques and Open Challenges in Process Mining, LNBIP-207 of LNBIP, Springer, 2015, <http://link.springer.com/book/10.1007/978-3-319-17482-2>.
- [6] OMG(Object Management Group), "OMG Unified Modeling Language (OMG UML), Version 2.5," , 2015-03-01 2015, <http://www.omg.org/spec/UML/2.5/PDF>.
- [7] OMG(Object Management Group), "Business Process Model And Notation (BPMN) , Version 2.0," , 2011-01-03 2011, <http://www.omg.org/spec/BPMN/2.0/PDF>.
- [8] W. van der Aalst, B. van Dongen, C. Gunther, A. Rozinat, H. Verbeek, and A. Weijters, "Prom: The process mining toolkit," Proceedings of the BPM 2009 Demonstration Track, (ed.) B.W. A.K. Alves de Medeiros, 2009, <http://ceur-ws.org/Vol-489/paper3.pdf>.
- [9] H. Verbeek, J. Buijs, B. van Dongen, and W. van der Aalst, "Prom 6: The process mining toolkit," Business Process Management Demonstration Track 2010, (ed.) M.L. Rosa, 2010, <http://ceur-ws.org/Vol-615/paper13.pdf>.
- [10] A.J.M.M. Weijters, and W.M.P. van der Aalst, "Rediscovering workflow models from event-based data using little thumb," Integrated Computer-Aided Engineering, vol.10, no.2, pp.151-162, 2003, <http://content.iospress.com/articles/integrated-computer-aided-engineering/ica00143> もしくは http://www.processmining.org/_media/publications/weijters2003.pdf.
- [11] W.M.P.v.d.A. Boudewijn F. van Dongen, "EMiT: A Process Mining Tool," Applications and Theory of Petri Nets 2004, LNCS-3099, pp.454-463, 2004, http://link.springer.com/chapter/10.1007/978-3-540-27793-4_26 もしくは http://www.processmining.org/_media/publications/dongen2004.pdf.
- [12] W. van der Aalst, T. Weijters, and L. Maruster, "Workflow Mining: Discovering process models from event logs," IEEE Transactions on Knowledge and Data Engineering, vol.16, no.9, pp.1128-1142, 2004, <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1316839>.
- [13] J. Desel, G. BJuhas, R. BLorenz, and C. BNeumair, "Modelling and validation with viptool," Business Process Management, BPM 2003, (ed.) A. ter Hofstede, LNCS-2678, pp.380-389, Springer, 2003,

- http://link.springer.com/chapter/10.1007/3-540-44895-0_26.
- [14] G. Juhas, R. BLorenz, and J. BDesel, “Can i execute my scenario in your net?,” Applications and Theory of Petri Nets 2005, (eds.) G. Ciardo, and P. Darondeau, LNCS-3536, pp.289-308, Springer, 2005, http://link.springer.com/chapter/10.1007/11494744_17.
- [15] R. Bergenthum, J. BDesel, G. BJuhas, and R. BLorenz, “Can i execute my scenario in your net? vptool tells you!,” Petri Nets and Other Models of Concurrency - ICATPN 2006, (eds.) S. Donatelli, and P.S. Thiagarajan, LNCS-4024, pp.380-389, Springer, 2006, http://link.springer.com/chapter/10.1007/11767589_21.
- [16] R. Lorenz, R. Bergenthum, J. Desel, and S. Mauser, “Synthesis of petri nets from finite partial languages,” Seventh International Conference on Application of Concurrency to System Design, 2007. ACSD 2007, pp.157-166, IEEE, 2007, <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4276275>.
- [17] R. Bergenthum, J. Desel, R. Lorenz, and S. Mauser, “Synthesis of petri nets from finite partial languages,” Applications and Theory of Petri Nets, PETRI NETS 2008, (eds.) K.M. van Hee, and R. Valk, LNCS-5062, pp.388-398, Springer, 2008, http://link.springer.com/chapter/10.1007/978-3-540-68746-7_25.
- [18] R. Lorenz, J. Desel, and G. Juhas, “Models from scenarios,” Transactions on Petri Nets and Other Models of Concurrency VII, (eds.) K. Jensen, W.M.P. van der Aalst, G. Balbo, M. Koutny, and K. Wolf, LNCS-7480, pp.314-371, Springer, 2013, http://link.springer.com/chapter/10.1007/978-3-642-38143-0_9.
- [19] C.W. Gunther, and B. Rozinat, “Disco: Discover Your Processes,” Proceedings of the Demonstration Track of the 10th International Conference on Business Process Management (BPM 2012), (eds.) N. Lohmann, and S. Moser, 2012, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.416.9550&rep=rep1&type=pdf#page=46>.
- [20] ing. J.C.A.M. Buijs, “Mapping data sources to xes in a generic way,” Master thesis, Eindhoven University of Technology, 2010, http://www.processmining.org/_media/xesame/xesma_thesis_final.pdf.
- [21] H.M.W. Verbeek, J.C.A.M. Buijs, B.F. van Dongen, and W.M.P. van der Aalst, “Xes, xesame, and prom 6,” Information Systems Evolution, (ed.) E.P. Pnina Soffer, LNBIP-72, pp.60-75, Springer, 2010, https://link.springer.com/chapter/10.1007/978-3-642-17722-4_5.
- [22] R. Agrawal, D. Gunopulos, and F. Leymann, “Mining Process Models from Workflow Logs,” Advances in Database Technology - EDBT’98, LNCS-1377, pp.467-483, Springer, 1998, <http://link.springer.com/chapter/10.1007/BFb0101003>.
- [23] J.E. Cook, and A.L. Wolf, “Automating Process Discovery through Event-Data Analysis,” ICSE ’95 Proceedings of the 17th international conference on Software engineering, pp.73-82, ACM, 1995, <http://dl.acm.org/citation.cfm?id=225021>, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.46.5709&rep=rep1&type=pdf>.
- [24] J. Cook, and A. Wolf, “Discovering Models of Software Processes from Event-Based Data,” ACM Transactions on Software Engineering and Methodology, vol.7, no.3, pp.215-249, 1998, <http://dl.acm.org/citation.cfm?id=287001>, <http://www.dtic.mil/dtic/tr/fulltext/u2/a446147.pdf>.
- [25] J. Cook, and A. Wolf, “Event-Based Detection of Concurrency,” Proceedings of the 6th International Symposium on the Foundations of Software Engineering (FSE-6), pp.35-45, 1998, <http://www.doc.ic.ac.uk/~alw/doc/>

- papers/fse98.pdf, <http://www.doc.ic.ac.uk/~alw/doc/papers/fse98.pdf>.
- [26] A. Biermann, and J. Feldman, "On the Synthesis of Finite State Machines from Samples of Their Behavior," *IEEE Transactions on Computers*, vol.21, no.6, p.592597, 1972, <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5009015>.
- [27] G.E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp.1527-1554, 2006, <http://www.mitpressjournals.org/doi/abs/10.1162/neco.2006.18.7.1527>, <http://www.cs.toronto.edu/~hinton/absps/ncfast.pdf>.
- [28] G.E. Hinton, and R.R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no.5786, pp.504-507, 2006, <http://science.sciencemag.org/content/313/5786/504>, <http://www.cs.toronto.edu/~hinton/science.pdf>.
- [29] Y. LeCun, Y. Bengio, and G.E. Hinton, "Deep learning," *Nature*, Vol. 521, pp.436-444, 2015, <http://www.nature.com/nature/journal/v521/n7553/full/nature14539.html>, <http://www.cs.toronto.edu/~hinton/absps/NatureDeepReview.pdf>.
- [30] S. Goedertier, D. Martens, J. Vanthienen, and B. Baesens, "Robust process discovery with artificial negative events," *Journal of Machine Learning Research*, 10, pp.1305-1340, 2009, <http://www.jmlr.org/papers/volume10/goedertier09a/goedertier09a.pdf>, <http://dl.acm.org/citation.cfm?id=1577113>.
- [31] J. Herbst, and D. Karagiannis, "Integrating Machine Learning and Workflow Management to Support Acquisition and Adaptation of Workflow Models," *Proceedings of the 9th International Workshop on Database and Expert Systems Applications*, p.745752, IEEE, 1998, <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=707491>.
- [32] J. Herbst, "Inducing Workflow Models from Workflow Instances," *Proceedings of the 6th European Concurrent Engineering Conference*, p.175182, Society for Computer Simulation (SCS), 1999, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.16.1945&rep=rep1&type=pdf>.
- [33] J. Herbst, and D. Karagiannis, "An Inductive Approach to the Acquisition and Adaptation of Workflow Models," *Proceedings of the IJCAI'99 Workshop on Intelligent Workflow and Process Management: The New Frontier for AI in Business*, (eds.) M. Ibrahim, and B. Drabble, pp.52-57, 1999, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1.39.866&rep=rep1&type=pdf>.
- [34] J. Herbst, "Machine learning approach to workflow management," *The 11th European Conference on Machine Learning, ECML 2000, LNCS-1810*, pp.183-194, Springer, 2000, http://link.springer.com/chapter/10.1007/3-540-45164-1_19.
- [35] J. Herbst, and D. Karagiannis, "Integrating machine learning and workflow management to support acquisition and adaptation of workflow models," *Intelligent Systems in Accounting, Finance and Management*, vol.9, no.2, p.6792, 2000, [http://onlinelibrary.wiley.com/doi/10.1002/1099-1174\(200006\)9:2<67::AID-ISAF186>3.0.CO;2-7/abstract](http://onlinelibrary.wiley.com/doi/10.1002/1099-1174(200006)9:2<67::AID-ISAF186>3.0.CO;2-7/abstract).
- [36] J. Herbst, "Dealing with concurrency in workflow induction," *European Concurrent Engineering Conference*, (eds.) U. Baake, R. Zobel, and M. AlAkaidi, SCS Europe, 2000, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.16.285&rep=rep1&type=pdf>.
- [37] A. de Medeiros, A. Weijters, and W. van der Aalst, "Using Genetic Algorithms to Mine Process Models: Representation, Operators and Results," *Technical Report WP-124*, Eindhoven University of Technology, 2004,

- <http://wwis.win.tue.nl/~wvdaalst/publications/p249.pdf>.
- [38] W. van der Aalst, A.A. de Medeiros, and A. Weijters, “Genetic Process Mining,” Applications and Theory of Petri Nets 2005, (eds.) G. Ciardo, and P. Darondeau, LNCS-3536, pp.48-69, Springer, 2005, http://link.springer.com/chapter/10.1007/11494744_5, <http://wwis.win.tue.nl/~wvdaalst/publications/p266.pdf>.
- [39] A. de Medeiros, A. Weijters, and W. van der Aalst, “Genetic Process Mining: A Basic Approach and its Challenges,” First International Workshop on Business Process Intelligence (BPI’05), (eds.) C.J. Bussler, and A. Haller, LNCS-3812, pp.203-215, Springer, 2005, http://link.springer.com/chapter/10.1007/11678564_18, <http://wwis.win.tue.nl/~wvdaalst/publications/p277.pdf>.
- [40] A.A. de Medeiros, A. Weijters, and W. van der Aalst, “Genetic Process Mining: An Experimental Evaluation,” Data Mining and Knowledge Discovery, vol.14, no.2, pp.245-304, 2007, <http://link.springer.com/article/10.1007/s10618-006-0061-7>, <http://wwis.win.tue.nl/~wvdaalst/publications/p415.pdf>.
- [41] C. Bratosin, N. Sidorova, and W. van der Aalst, “Distributed Genetic Process Mining,” IEEE World Congress on Computational Intelligence (WCCI 2010), (ed.) H. Ishibuchi, pp.1951-1958, IEEE, 2010, .
- [42] C. Bratosin, N. Sidorova, and W. van der Aalst, “Discovering Process Models with Genetic Algorithms Using Sampling,” Knowledge-Based and Intelligent Information and Engineering Systems (KES 2010), (eds.) R. Setchi, I. Jordanov, R. Howlett, and L. Jain, LNCS-6276, pp.41-50, Springer, 2010, http://link.springer.com/chapter/10.1007/978-3-642-15387-7_8.
- [43] C. Bratosin, N. Sidorova, and W. van der Aalst, “Distributed Genetic Process Mining Using Sampling,” Parallel Computing Technologies (PaCT 2011), (ed.) V. Malyszhkin, LNCS-6873, pp.224-237, Springer, 2011, http://link.springer.com/chapter/10.1007/978-3-642-23178-0_20.
- [44] J. Buijs, B. van Dongen, and W. van der Aalst, “A Genetic Algorithm for Discovering Process Trees,” IEEE Congress on Evolutionary Computation (CEC 2012), pp.1-8, IEEE, 2012, <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6256458>.
- [45] E. Lamma, P. Mello, F. Riguzzi, and S. Storari, “Applying Inductive Logic Programming to Process Mining,” Inductive Logic Programming, 17th International Conference, ILP 2007, (eds.) H. Blockeel, J. Ramon, J. Shavlik, and P. Tadepalli, LNCS-4894, pp.132-146, Springer, 2007, http://link.springer.com/chapter/10.1007/978-3-540-78469-2_16, http://research.cs.wisc.edu/machine-learning/shavlik-group/ilp07wip/ilp07_lamma.pdf.
- [46] E. Lamma, P. Mello, F. Riguzzi, and S. Storari, “Inducing declarative logic-based models from labeled traces,” Business Process Management, 5th International Conference, BPM 2007, (eds.) G. Alonso, P. Dadam, and M. Rosemann, LNCS-4714, pp.344-359, Springer, 2007, http://link.springer.com/chapter/10.1007/978-3-540-75183-0_25, <http://ds.ing.unife.it/~friguzzi/Papers/LamMelMon-BPM07.pdf>.
- [47] F. Chesani, E. Lamma, P. Mello, M. Montali, F. Riguzzi, and S. Storari, “Exploiting Inductive Logic Programming Techniques for Declarative Process Mining,” Transactions on Petri Nets and Other Models of Concurrency II, Special Issue on Concurrency in Process-Aware Information Systems, (eds.) K. Jensen, and W.M.P. van der Aalst, LNCS-5460, pp.278-295, Springer, 2009, http://link.springer.com/chapter/10.1007/978-3-642-00899-3_16, http://link.springer.com/chapter/10.1007/978-3-642-00899-3_16, http://link.springer.com/chapter/10.1007/978-3-642-00899-3_16.

- [//ds.ing.unife.it/~friguizzi/Papers/CheLamMel-TOPNOC09.pdf](http://ds.ing.unife.it/~friguizzi/Papers/CheLamMel-TOPNOC09.pdf).
- [48] F.M. Maggi, D. Corapi, A. Russo, and E.L.G. Visaggio, "Revising Process Models through Inductive Learning," Business Process Management Workshops, BPM 2010 International Workshops and Education Track, (eds.) M. zur Muehlen, and J. Su, LNBIP-66, pp.182-193, Springer, 2011, http://link.springer.com/chapter/10.1007/978-3-642-20511-8_16, http://www.processmining.org/_media/publications/bpi_maggi.pdf.
- [49] J.M.E.M. van der Werf, B.F. van Dongen, C.A.J. Hurkens, and A. Serebrenik, "Process discovery using integer linear programming," Applications and Theory of Petri Nets, 29th International Conference, PETRI NETS 2008, (eds.) K.M. van Hee, and R. Valk, LNCS-5062, pp.368-387, Springer, 2008, http://link.springer.com/chapter/10.1007/978-3-540-68746-7_24, <https://pure.tue.nl/ws/files/3092412/633716.pdf>.
- [50] J.M.E.M. van der Werf, B.F. van Dongen, C.A.J. Hurkens, and A. Serebrenik, "Process discovery using integer linear programming," Fundamenta Informaticae, vol.94, no.3-4, pp.387-412, 2009, <http://content.iospress.com/articles/fundamenta-informaticae/fi94-3-4-06>.
- [51] S. Das, and M.C. Mozer, "A Unified Gradient Descent/Clustering Architecture for Finite State Machine Induction," Advances in Neural Information Processing Systems, Proceedings of the 1993 Conference, no.6, pp.19-26, Morgan Kaufmann, 1994, <http://papers.nips.cc/paper/846-a-unified-gradient-descentclustering-architecture-for-finite-state-machine-induction.pdf>.
- [52] A. Tiwari, C. Turner, and B. Majeed, "A review of business process mining: state-of-the-art and future trends," Business Process Management Journal, vol.14, no.1, pp.5-22, 2008, <http://www.emeraldinsight.com/doi/abs/10.1108/14637150810849373>.
- [53] A.R.C. rdenas Maita, L.C. Martins, C.R.L. Paz, S.M. Peres, and M. Fantinato, "Process mining through artificial neural networks and support vector machines: A systematic literature review," Business Process Management Journal, vol.21, no.6, pp.1391-1415, 2015, <http://www.emeraldinsight.com/doi/abs/10.1108/BPMJ-02-2015-0017>.
- [54] W. Song, S. Liu, and Q. Liu, "Business process mining based on simulated annealing," The 9th International Conference for Young Computer Scientists, ICYCS 2008, pp.725-730, IEEE, 2008, <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4709063>.
- [55] D. Gao, and Q. Liu, "An improved simulated annealing algorithm for process mining," 13th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2009, pp.474-479, IEEE, 2009, <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4968104>.
- [56] H. Mili, G. Tremblay, G.B. Jaoude, E. Lefebvre, L. Elabed, and G.E. Boussaidi, "Business process modeling languages: Sorting through the alphabet soup," ACM Computing Surveys, vol.43, no.1, pp.1-56, 2010, <http://dl.acm.org/citation.cfm?id=1824799>, <http://people.ischool.berkeley.edu/~glushko/IS243Readings/BusinessProcessModelingLanguages.pdf>.
- [57] R. Lu, and S. Sadiq, "A survey of comparative business process modeling approaches," Business Information Systems, BIS 2007, (ed.) W. Abramowicz, 4439, pp.82-94, 2007, http://link.springer.com.kras1.lib.keio.ac.jp/chapter/10.1007/978-3-540-72035-5_7.
- [58] M. Havey, Essential Business Process Modeling, O'Reilly, 2005,

- <http://shop.oreilly.com/product/9780596008437.do>.
- [59] M. Havey, (監訳) 長瀬嘉秀, 永田渉, (訳) 株式会社テクノロジックアート, 詳説ビジネスプロセスモデリング – SOA ベストプラクティス, オライリージャパン, 2006, <https://www.oreilly.co.jp/books/4873112907/>.
- [60] G.M. Giaglis, “A taxonomy of business process modeling and information systems modeling techniques,” *International Journal of Flexible Manufacturing Systems*, vol.13, no.2, pp.209-228, 2001, <http://link.springer.com/article/10.1023/A:1011139719773>.
- [61] W.M.P. van der Aalst, A.H.M. ter Hofstede, and M. Weske, “Business process management: A survey,” *Business Process Management, BPM 2003*, (eds.) W.M.P. van der Aalst, and M. Weske, LNCS-2678, pp.1-12, 2003, http://link.springer.com/chapter/10.1007/3-540-44895-0_1.
- [62] R.S. Aguilar-Saven, “Business process modelling: Review and framework,” *International Journal of Production Economics*, 90, 2, p.129149, 2004, <http://www.sciencedirect.com/science/article/pii/S0925527303001026>.
- [63] B. List, and B. Korherr, “An evaluation of conceptual business process modelling languages,” *Proceedings of the 2006 ACM symposium on Applied computing(SAC)*, pp.1532-1539, ACM, 2006, <http://dl.acm.org/citation.cfm?id=1141633>, <http://www.wit.at/people/korherr/publications/acmsac2006.pdf>.
- [64] B. Kiepuszewski, A. ter Hofstede, and W. van der Aalst., “Fundamentals of control flow in workflows,” *Acta Informatica*, Vol.39, pp.143-209, 2003, <http://link.springer.com/article/10.1007/s00236-002-0105-4>, <http://wwwis.win.tue.nl/~wvdaalst/publications/p185.pdf>.
- [65] D. Hollingsworth, “The Workflow Reference Model, Issue 1.1,” Technical Report WfMC-TC-1003, WfMC(Workflow Management Coalition), 1995-01-19 1995, <http://www.wfmc.org/docs/tc003v11.pdf>.
- [66] WfMC(Workflow Management Coalition), “Process Definition Interface – XML Process Definition Language, Version 2.00,” Technical Report WfMC-TC-1025, WfMC(Workflow Management Coalition), 2005-10-03 2005, http://www.wfmc.org/docs/TC-1025_xpdl_2_2005-10-03.pdf.
- [67] WfMC(Workflow Management Coalition), “Workflow Management Coalition Terminology and Glossary, Issue 2.0,” Technical Report WfMC-TC-1011, WfMC(Workflow Management Coalition), 1996-06 1996, <http://www.aiai.ed.ac.uk/project/wfmc/ARCHIVE/DOCS/glossary/glossary.html>.
- [68] WfMC(Workflow Management Coalition), “Workflow Management Coalition Terminology and Glossary, Issue 3.0,” Technical Report WfMC-TC-1011, WfMC(Workflow Management Coalition), 1999-02 1999, http://www.wfmc.org/docs/TC-1011_term_glossary_v3.pdf.
- [69] W. van der Aalst, “The application of petri nets to workflow management,” *The Journal of Circuits, Systems and Computers*, vol.8, no.1, pp.21-66, 1998, <http://www.worldscientific.com/doi/abs/10.1142/S0218126698000043>, <http://www.workflowpatterns.com/documentation/documents/vanderaalst98application.pdf>.
- [70] M. Reichert, P. Dadam, and U. of Ulm, “ADEPTflex Supporting Dynamic Changes of Workflows Without Losing Control,” Technical Report 97-07, Ulmer Informatik-Berichte, 1997, <http://dbis.eprints.uni-ulm.de/446/1/ReDa97.pdf>.
- [71] M. Reichert, S. Rinderle, and P. Dadam, “ADEPT Workflow Management System – Flexible Support for Enterprise-Wide Business Processes Tool Presentation,” *Business Process Man-*

- agement, BPM 2003, (eds.) W.M.P. van der Aalst, and M. Weske, LNCS-2678, pp.370-379, Springer, 2003, http://link.springer.com/chapter/10.1007/3-540-44895-0_25.
- [72] B. Kiepuszewski, A.H.M. ter Hofstede, and C.J. Bussler, "On Structured Workflow Modeling," Advanced Information Systems Engineering, 12th International Conference, CAiSE 2000, (eds.) B. Wangler, and L. Bergman, LNCS-1789, pp.431-445, Springer, 2000, http://link.springer.com/chapter/10.1007%2F3-540-45140-4_29.
- [73] B. Silver, (監訳) 岩田アキラ, (訳) 山原雅人, BPMN METHOD & STYLE 第2版 BPMN 実装者向けガイド付き, 一般社団法人日本ビジネスプロセス・マネジメント協会, 2013.
- [74] M. Owen, and J. Raj, "BPMN and Business Process Management – An Introduction to the New Business Process Modeling Standard," BPTrends, 2004, <http://www.bptrends.com/publicationfiles/03-04WPBPMandBPMOwen-Raj.pdf>.
- [75] R.M. Dijkman, M. Dumas, and C. Ouyang, "Semantics and analysis of business process models in BPMN," Information and Software Technology, vol.50, no.12, p.12811294, 2008, <http://www.sciencedirect.com/science/article/pii/S0950584908000323>.
- [76] 平石邦彦, "複雑なプロセスのモデル化," 電子情報通信学会基礎・境界ソサイエティ Fundamentals Review, vol.6, no.4, pp.257-264, 2013, https://www.jstage.jst.go.jp/article/essfr/6/4/6_257/_pdf.
- [77] C.V. Geambasu, "BPMN VS. UML ACTIVITY DIAGRAM FOR BUSINESS PROCESS MODELING," Journal of Accounting and Management Information Systems, vol.11, no.4, p.637651, 2012, http://www.cig.ase.ro/articles/11_4_7.pdf.
- [78] T. Murata, "Petri nets: Properties, analysis and applications," Proceedings of the IEEE, vol.77, no.4, pp.541-580, 1989, <https://inst.eecs.berkeley.edu/~ee249/fa07/discussions/PetriNets-Murata.pdf>.
- [79] 村田忠夫, ペトリネットの解析と応用, 近代科学社, 1992.
- [80] W. van der Aalst, and K. van Hee, "Business process redesign: A petri-net-based approach," Computers in Industry, vol.29, 1-2, p.1526, 1996, <http://www.sciencedirect.com/science/article/pii/S0166361595000518>.
- [81] W.M.P. van der Aalst, "Three good reasons for using a petri-net-based workflow management system," Information and Process Integration in Enterprises, (eds.) T. Wakayama, S. Kannapan, C.M. Khoong, S. Navathe, and J. Yates, vol.428, pp.161-182, Springer, 1998, http://link.springer.com/chapter/10.1007%2F978-1-4615-5499-8_10, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.33.960&rep=rep1&type=pdf>.
- [82] W. van der Aalst, and T. Basten, "Inheritance of workflows: an approach to tackling problems related to change," Theoretical Computer Science, vol.270, no.12, p.125203, 2002, <http://www.sciencedirect.com/science/article/pii/S0304397500003212>.
- [83] W. van der Aalst, and K. van Hee, Workflow Management: Models, Methods, and Systems, The MIT Press, 2004, <http://wwwis.win.tue.nl/~wvdaalst/publications/p120.pdf>, <https://mitpress.mit.edu/books/workflow-management>.
- [84] W.M. van van der Aalst, and C. Stahl, Modeling Business Processes: A Petri Net-Oriented Approach, The MIT Press, 2011, <https://mitpress.mit.edu/books/modeling-business-processes>.
- [85] K. Salimifard, and M. Wright, "Petri net-based modelling of workflow systems: An overview," European Journal of Operational Research, vol.134, no.3, p.664676, 2001, <http://www.sciencedirect.com/science/article/pii/S016763690100003>.

- [//www.sciencedirect.com/science/article/pii/S0377221700002927](http://www.sciencedirect.com/science/article/pii/S0377221700002927).
- [86] R.D. Niels Lohmann, Eric Verbeek, “Petri net transformations for business processes a survey,” Transactions on Petri Nets and Other Models of Concurrency II, Special Issue on Concurrency in Process-Aware Information Systems, (eds.) K. Jensen, and W.M.P. van der Aalst, LNCS-5460, pp.46-63, Springer, 2009, http://link.springer.com/chapter/10.1007/978-3-642-00899-3_3.
- [87] W. van der Aalst, A. ter Hofstede, B. Kiepuszewski, and A. Barros, “Workflow Patterns,” Distributed and Parallel Databases, vol.14, no.1, pp.5-51, 2003, <http://link.springer.com/article/10.1023/A:1022883727209>.
- [88] E. Borger, “Approaches to modeling business processes: a critical analysis of BPMN, workflow patterns and YAWL,” Software & Systems Modeling, vol.11, no.3, pp.305-318, 2012, <http://link.springer.com/article/10.1007/s10270-011-0214-z>.
- [89] 山口真悟, 葛崎偉, “ネット理論の応用 – ワークフローネットとプログラムネット,” 電子情報通信学会基礎・境界ソサイエティ Fundamentals Review, vol.3, no.3, pp.52-63, 2009, https://www.jstage.jst.go.jp/article/essfr/3/3/3_3_3_52/_pdf.
- [90] D. Hauschildt, H.M.W. Verbeek, and W.M.P. van der Aalst, “WOFLAN: a Petri-net-based Workflow Analyzer,” , 1997, <http://alexandria.tue.nl/extra1/wskrap/publichtml/9715985.pdf>.
- [91] W.M.P. van der Aalst, D. Hauschildt, and H.M.W. Verbeek, “A Petri-net-based Tool to Analyze Workflows,” Proceedings of Petri Nets in System Engineering (PNSE’ 97), pp.78–90, University of Hamburg, 1997, <http://www.win.tue.nl/~hverbeek/downloads/preprints/Aalst97.pdf>.
- [92] H.M.W. Verbeek, T. Basten, and W.M.P. van der Aalst, “Diagnosing workflow processes using woflan,” The Computer Journal, vol.44, no.4, pp.246–279, 2001, <http://comjnl.oxfordjournals.org/cgi/reprint/44/4/246>, <http://www.win.tue.nl/~wvdaalst/publications/p135.pdf>.
- [93] W. van der Aalst, “Woflan: A petri-net-based workflow analyzer,” Systems Analysis Modelling Simulation, vol.35, no.3, pp.345-357, 1999, <http://www.win.tue.nl/~wvdaalst/publications/p71.pdf>.
- [94] E. Verbeek, and W.M.P. van der Aalst, “Woflan 2.0 A petri-net-based workflow diagnosis tool,” Application and Theory of Petri Nets 2000, 21st International Conference, ICATPN 2000, (ed.) D.S. Mogens Nielsen, LNCS-1825, pp.475-484, Springer, 2000, http://link.springer.com/chapter/10.1007/3-540-44988-4_28.
- [95] K. Jensen, “Coloured Petri nets,” Petri Nets: Central Models and Their Properties, Advances in Petri Nets 1986, Part I Proceedings of an Advanced Course, (eds.) W. Brauer, W. Reisig, and G. Rozenberg, LNCS-254, pp.248-299, Springer, 1987, .
- [96] K. Jensen, and L.M. Kristensen, Coloured Petri Nets: Modelling and Validation of Concurrent Systems, Springer, 2009, <http://www.springer.com/us/book/9783642002830>.
- [97] K. Jensen, Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use. Volume 1, Monographs in Theoretical Computer Science. An EATCS Series, , Springer, 1996, <http://www.springer.com/us/book/9783540609438>.
- [98] K. Jensen, Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use. Volume 2, Monographs in Theoretical Computer Science. An EATCS Series, , Springer, 1995, <http://www.springer.com/us/book/9783540582762>.
- [99] K. Jensen, Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use. Volume 3, Monographs in Theoretical

- cal Computer Science. An EATCS Series, Springer, 1997, <http://www.springer.com/us/book/9783642645563>.
- [100] A.H.M. ter Hofstede, W. van der Aalst, M. Adams, and N. Russell, (eds.), Modern Business Process Automation: YAWL and its Support Environment, Springer, 2009, <http://www.springer.com/br/book/9783642031205>.
- [101] W. van der Aalst, and A. ter Hofstede, “YAWL: yet another workflow language,” Information Systems, vol.30, no.4, p.245-275, 2005, <http://www.sciencedirect.com/science/article/pii/S0306437904000304>.
- [102] W.M.P. van der Aalst, L. Aldred, M. Dumas, and A.H.M. ter Hofstede, “Design and implementation of the YAWL system,” Advanced Information Systems Engineering, 16th International Conference, CAiSE 2004, (eds.) A. Persson, and J. Stirna, LNCS-3084, pp.142-159, Springer, 2004, http://eprints.qut.edu.au/379/1/aalst_yawls.pdf, http://link.springer.com/chapter/10.1007/978-3-540-25975-6_12.
- [103] G. Decker, R. Dijkman, M. Dumas, and L. Garca-Baños, “Transforming bpmn diagrams into yawl nets,” Business Process Management, (eds.) M. Dumas, M. Reichert, and M.C. Shan, LNCS-5240, pp.386-389, Springer, 2008, http://link.springer.com/chapter/10.1007/978-3-540-85758-7_30, <http://bpt.hpi.uni-potsdam.de/pub/Public/GeroDecker/bpmdemo2008-bpmn2yawl.pdf>.
- [104] J. Ye, S. Sun, L. Wen, and W. Song, “Transformation of BPMN to YAWL,” 2008 International Conference on Computer Science and Software Engineering (Volume:2), pp.354-359, IEEE, 2008, <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4722069>, <http://www.jsoftware.us/vol5/jsw0504-6.pdf>.
- [105] “Integration Definition for Function Modelling (IDEF0), Draft Federal Information Processing Standards Publication,” , 1993, <http://www.idef.com/wp-content/uploads/2016/02/idef0.pdf>.
- [106] R. Mayer, et al., “IDEF3 Process Description Capture Method Report,” , 1995, http://www.idef.com/wp-content/uploads/2016/02/Idef3_fn.pdf.
- [107] C. Badica, A. Badica, and V. Litoiu, “A New Formal IDEF-based Modelling of Business Processes,” Proceedings of the 1st Balkan Conference in Informatics, (eds.) Y. Manolopoulos, and P. Spirakis, p.535-549, 2003, <http://software.ucv.ro/~cbadica/cercetare/papers/bci03.pdf>.
- [108] C. Badica, and C. Fox, “Hybrid IDEF0/IDEF3 Modelling of Business Processes: Syntax, Semantics and Expressiveness,” Computer Aided Verification of Information Systems (CaVIS 2004), pp.20-22, 2004, <http://csee.essex.ac.uk/staff/foxcj/papers/C-Fox-CaVIS04-paper.pdf>, <http://chris.foxearth.org/papers/C-Fox-CaVIS04-paper.pdf>.
- [109] C. Badica, and C. Fox, “On the Application of WF-Nets for Checking Hybrid IDEF0-IDEF3 Business Process Models,” Advances in Information Systems, Third International Conference, ADVIS 2004, (ed.) T. Yakhno, LNCS-3261, pp.543-553, Springer, 2005, http://link.springer.com/chapter/10.1007/978-3-540-30198-1_55, <https://pdfs.semanticscholar.org/6106/18237af35cfe8fd573cd0c28bad13248f1ba.pdf>.
- [110] R. Davis, and E. Brabander, ARIS Design Platform – Getting Started with BPM, , chapter The Event-driven Process Chain, pp.105-125 Springer, 2007, <http://www.springer.com/us/book/9781846286124>.
- [111] W. van der Aalst, “Formalization and Verification of Event-driven Process Chains,” Information and Software Technology, vol.41, no.10, pp.639-650, 1999, <http://www.sciencedirect.com/science/>

- [article/pii/S0950584999000166,http://www.wis.win.tue.nl/~wvdaalst/publications/p74.pdf](http://www.wis.win.tue.nl/~wvdaalst/publications/p74.pdf).
- [112] J. Becker, M. Rosemann, and C. von Uthmann, “Guidelines of Business Process Modeling,” Business Process Management – Models, Techniques, and Empirical Studies, (eds.) W. van der Aalst, J. Desel, and A. Oberweis, LNCS-1806, pp.30-49, Springer, 2000, http://link.springer.com/chapter/10.1007/3-540-45594-9_3.
- [113] OASIS Standard, “Web Services Business Process Execution Language Version 2.0,” , 2007, <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf>.
- [114] OASIS Standard, “Web Services Business Process Execution Language Version 2.0 Primer,” , 2007, <http://docs.oasis-open.org/wsbpel/2.0/Primer/wsbpel-v2.0-Primer.pdf>.
- [115] A. Brogi, and R. Popescu, “From bpel processes to yawl workflows,” Web Services and Formal Methods, Third International Workshop, WS-FM 2006, (eds.) M. Bravetti, M. Nunez, and G. Zavattaro, LNCS-4184, pp.107-122, Springer, 2006, http://link.springer.com/chapter/10.1007/11841197_7.
- [116] O. Kopp, D. Martin, D. Wutke, and F. Leymann, “On the choice between graph-based and block-structured business process modeling languages,” Modellierung betrieblicher Informationssysteme (MobIS 2008), Lecture Notes in Informatics (LNI) P-141, pp.59-72, Gesellschaft für Informatik e.V. (GI), 2008, https://www.gi.de/fileadmin/redaktion/2008_LNI/lni-p-141.pdf, <http://www.iaas.uni-stuttgart.de/RUS-data/INPROC-2008-92-graph-based-vs-block-structured-modeling--mobis08.pdf>.
- [117] O. Kopp, D. Martin, D. Wutke, and F. Leymann, “The difference between graph-based and block-structured business process modelling languages,” Enterprise Modelling and Information Systems, vol.4, no.1, pp.3-13, 2009, <https://www.emisa-journal.org/emisa/article/view/54>, <https://www.emisa-journal.org/emisa/article/download/54/29>.
- [118] OASIS BPEL4People TC, “WS-BPEL Extension for People (BPEL4People) Specification Version 1.1,” , 2010, <http://docs.oasis-open.org/bpel4people/bpel4people-1.1.html>.
- [119] OASIS BPEL4People TC, “Web Services Human Task (WS-HumanTask) Specification Version 1.1,” , 2010, <http://docs.oasis-open.org/bpel4people/ws-humantask-1.1-spec-cs-01.html>.
- [120] W. van der Aalst, A. Adriansyah, and B. van Dongen, “Causal Nets: A Modeling Language Tailored Towards Process Discovery,” CONCUR 2011 Concurrency Theory, (eds.) J.P. Katoen, and B. Koenig, LNCS-6901, pp.28-42, Springer, 2011, http://link.springer.com/chapter/10.1007/978-3-642-23217-6_3.
- [121] A. Fleischmann, “What Is S-BPM?,” S-BPM ONE Setting the Stage for Subject-Oriented Business Process Management, (eds.) H. Buchwald, A. Fleischmann, D. Seese, and C. Stary, LNCIS-85, pp.85-106, Springer, 2010, http://link.springer.com/chapter/10.1007/978-3-642-15915-2_7.
- [122] E. Aitenbichler, S. Borgert, and M. Mhlhuser, “Distributed Execution of S-BPM Business Processes,” Subject-Oriented Business Process Management, Second International Conference, S-BPM ONE 2010, (eds.) A. Fleischmann, W. Schmidt, R. Singer, and D. Seese, LNCIS-138, pp.19-35, Springer, 2011, http://link.springer.com/chapter/10.1007/978-3-642-23135-3_2.
- [123] A. Fleischmann, S. Ra, and R. Singer, S-BPM Illustrated – A Storybook about Business Process Modeling and Execution, Springer, 2013, <http://www.springer.com/us/book/9783642369032>.

- [124] W. Harrison, and H. Ossher, “Subject-oriented programming(a critique of pure objects),” Proceedings of the eighth annual conference on Object-oriented programming systems, languages, and applications (OOPSLA '93), pp.411-428, ACM, 1993, <http://dl.acm.org/citation.cfm?id=165932>.
- [125] R. Milner, A Calculus of Communicating Systems, LNCS-92 of LNCS, Springer, 1980, <http://link.springer.com/book/10.1007/3-540-10235-3>.
- [126] R. Milner, Communication and Concurrency, Prentice Hall, 1989, <http://dl.acm.org/citation.cfm?id=534666>.
- [127] C.A.R. Hoare, “Communicating sequential processes,” Communications of the ACM, vol.28, no.8, p.666677, 1978, <http://dl.acm.org/citation.cfm?id=359585>.
- [128] C.A.R. Hoare, Communicating Sequential Processes, Prentice Hall, 1985, <http://dl.acm.org/citation.cfm?id=3921>, <http://www.usingcsp.com/cspbook.pdf>.

著者略歴

- [1] 飯島 正 (いいじま ただし)
慶應義塾大学 理工学部 専任講師 (管理工学科 所属). 慶應義塾大学 理工学部 計測工学科卒業 (1986 年), 同大学院 理工学研究科 修士課程 修了 (1988 年), 同博士課程 単位取得退学 (1991 年). 1990 年より (株) 東芝勤務を経て, 1992 年より 現所属 (助手を経て現職). 博士 (工学). 情報システム学会 元理事 (2007 年 大会担当理事, 2009~2013 年 理事).
- [2] 田端 啓一 (たばた けいいち)
2012 年早稲田大学 大学院 修士課程修了, 同年, 日本電信電話 (株) に入社. 以来, ソフトウェアイノベーションセンタにてソフトウェア工学の研究に従事. 専門分野: プログラム自動並列化, コンピュータアーキテクチャ, ソフトウェアテスト.
- [3] 斎藤 忍 (さいとう しのぶ)
2001 年 慶應義塾大学 大学院 修士課程修了, 同年, (株) NTT データに入社. 2015 年に日本電信電話 (株) に転籍. 現在はソフトウェアイノベーションセンタに所属. 上流プロセスに関する研究開発・技術支援に従事. 2007 年 慶應義塾大学 大学院博士課程修了. 博士 (工学). 専門分野: ソフトウェア工学, 要求工学.