情報処理学会論文誌　数理モデル化と応用　Vol.9　No.2　20–32 (Aug. 2016)

AND合流ゲートウェイと連結するアクティビティの平均潜在待ち時間とサービス時間の推定法とその検証

野ヶ山 尊秀[1,2,a)]　高橋 治久[2,b)]

受付日 2016年2月9日，再受付日 2016年3月31日，
採録日 2016年5月4日

概要：開始時刻または完了時刻のみが記録された単一時刻イベントログからは，アクティビティの待ち時間とサービス時間は計算できない．このようなログから統計的に平均潜在待ち時間とサービス時間を推定する方法が知られているが，AND 合流ゲートウェイが含まれる場合には仮定にない並列実行の同期待ち時間が含まれるため適用できない．本論文では，指数分布を仮定して AND 合流ゲートウェイの確率モデルを構築し，平均待ち時間とサービス時間を EM アルゴリズムを用いて統計的に推定する方法を提案する．人工的に生成させたログを用いた数値実験により，提案手法が正しく推定できることを示した．

キーワード：プロセスマイニング，性能分析，待ち時間，サービス時間，EM アルゴリズム，指数分布，フォーク・ジョイン，単一時刻イベントログ

Verification of Estimation of Average Latent Waiting and Service Times of Activities Connected with AND-join Gateway

Takahide Nogayama[1,2,a)]　Haruhisa Takahashi[2,b)]

Received: February 9, 2016, Revised: March 31, 2016,
Accepted: May 4, 2016

Abstract: Waiting and service times of activities are not readily available in most event logs that only record either the start or the completion events in activities. Authors had proposed a method of estimating the average waiting and service times from such event logs. However, the probabilistic model of the method does not match a process that includes And-join gateway. This paper proposes a probabilistic model of AND-join gateway with exponential distribution and a novel method that estimates the average waiting and service times with expectation and maximization (EM) algorithm. Our experimental results using artificially generated logs indicated that estimated results of the average latent waiting and service times enable practical applications with sufficient accuracy.

Keywords: process mining, performance analysis, waiting time, service time, EM algorithm, exponential distribution, fork-join, single timestamp event log

1. はじめに

ビジネスプロセスのモデル化と応用の研究は，欧米において活発に研究されており[*1]，プロセスマイニングと呼ばれている．ビジネスプロセスは，作業の基本単位であるアクティビティと条件分岐や並列実行を矢印でつないだフローチャートとしてモデル化される．ビジネスプロセスを実行するシステム，ワークフローマネジメントシステム（WFMS）やビジネスプロセスマネジメントシステム（BPMS）などは，プロセス指向型情報システム（PAIS）[5]と呼ばれており，プロセスの実行の履歴をイベントログとして詳細に記録する．プロセスマイニングは，主にこのイ

[1] 日本アイ・ビー・エム株式会社東京基礎研究所
IBM Research – Tokyo, IBM Japan Ltd., Chuo, Tokyo 103–8510, Japan
[2] 電気通信大学
The University of Electro-Communications, Chofu, Tokyo 182–8585, Japan
[a)] nogayama@jp.ibm.com
[b)] takaharuroka@uec.ac.jp

[*1] 主な国際会議として Business Process Management（BPM），Conference on Advanced Information Systems Engineering（CAiSE）などがある．

© 2016 Information Processing Society of Japan

ベントログから知識を獲得するための技術である [9], [22].

イベントログからは，アクティビティの待ち時間やサービス時間といった性能指標を得ることができる．こうした指標を用いた性能分析は人件費や設備費に直接関係するため，ビジネスプロセス改善のうち特に重要な工程である．

しかし，多くのプロセスシステムは不完全なイベントログを生成するため，こうした性能指標は信頼できるものにはならないか，計算不可能であることが多い．性能指標に関わるイベントログの不完全さには主に以下の 2 つが考えられる：

(1) アクティビティに対して 1 つの時刻しか記録されずアクティビティ所要時間が計算できない場合．多くのプロセスシステムでは，ログの記録の目的が監査や問題判別であるため，開始または終了時刻のみが記録される．その 1 つの時刻が開始時刻なのか終了時刻なのか分からない場合もある．たとえば，Kuo ら [10] は開始時刻のみが記録される医療システムに直面した．他にも，プロセスマイニングの普及を意図したマニフェスト [9], [22] でも同様のイベントログを典型的な例としてあげている．

(2) 制御がアクティビティに移った正確な時刻や，制御を他のアクティビティに移した正確な時刻が記録されない場合．たとえば，処理依頼を受け取ったにもかかわらずすぐに受領処理を行わなかった場合や，アクティビティが終了したのちにすぐ次のアクティビティに処理依頼を出さずに溜めてしまった場合は，ログのうえでは遷移時間が長くなる．

BPMS ベンダにとってこのような不完全イベントログの分析は，他社との差別化を図るため必須である．たとえば，ある顧客が古いプロセスシステムを更改するとき，同時に現行システムの問題を改善した新システムの設計を望むことが多い．現行システムの不完全イベントログを分析し，移行案に加えて改善案を提案できれば，BPMS ベンダはシステム移行プロジェクトを獲得できる見込みが高まる．

我々は，このような不完全なイベントログであっても統計的に分析することでサービス時間と待ち時間の平均値を推定する方法を提案した [14], [23]．この方法の本質は，遷移時間が遷移元アクティビティの潜在サービス時間と遷移先アクティビティの潜在待ち時間によって構成されると仮定することと，分岐または合流ゲートウェイを通る遷移時間は同じ確率分布に従う潜在待ち時間と潜在サービス時間を共有するという特徴を利用することにある．この仮定は BPMN で定義されているほとんどの要素間の遷移で成り立つが，AND 合流ゲートウェイ[*2]を経由するアクティビティ遷移では成り立たない．なぜならば，先に到着したすべてのサブプロセスが最後に到着したサブプロセスを待つためにそれより前に到着したサブプロセスには同期待ち時間が加わるため，確率モデルに不整合がでてくるためである．

本研究では，こうした現実的な問題を解決するため文献 [23] を拡張し指数分布を採用した，AND 合流ゲートウェイを含むプロセスの不完全イベントログから，EM アルゴリズムにより平均潜在待ち時間とサービス時間を算出する方法を提案し，計算機実験によって効果を検証した．

2. 関連研究

イベントログから性能を要約する研究は多く行われている．たとえば，Ferreira は医療プロセス内の各アクティビティにおける所要時間の最大値，最小値，平均値を算出してプロセスモデル上に可視化した [6]．所要時間に加え，イベントログから抽出できる多くの性能指標について，Lanz ら [11] がよく整理し分類している．また，多くのビジネスプロセスの性能指標やその上での分析法について，Hornix [8] がよくまとめている．

性能の要約データは，プロセスの再設計時のほかにも実行中のプロセスの予測や制御にも用いられることがある．van der Aalst ら [19], [20] は，要約データをもとにして，実行中のプロセスインスタンス全体の所要時間の予測方法を提案している．また，Rogge-solti ら [16] はアクティビティの平均サービス時間を用いて実行中のプロセスの異常を予測する方法を提案している．予測だけでなく，動的にプロセスに介入することでより良いプロセスに変えることもできる．たとえば，Sindhgatta ら [18] はアクティビティの担当者ごとの平均時間を用いて，実行中のアクティビティの適切な担当者を割り当てる手法を提案している．

待ち行列理論によるビジネスプロセスの性能分析も研究されている．トランザクションタイプが記録されたイベントログの上では，アクティビティの処理依頼の到着や処理の開始や完了の時刻が観測されるため，待ち時間やサービス時間を算出できる．これらの情報を用いて平均行列長などをシミュレーションによって算出できる [17], [21]．シミュレーションでは到着過程，サービス（担当者）の数，アクティビティ内の行列の構造などはあらかじめ仮定する必要がある．しかし，ビジネスプロセスでは担当者のスキルレベルや数が時間帯や日によって変わるため，単純な待ち行列理論で扱うことは難しい．また，これらの仮定を補強する情報は通常イベントログに記録されないため，それ以外の情報源，たとえば詳細を知る者へのインタビューや設計文書が必要となる．

ビジネスプロセスの待ち時間とサービス時間を指数分布と仮定すれば，プロセス所要時間を連続マルコフ連鎖を用いた相型分布 [24] によってモデル化できる．標本から相型分布のパラメータを EM アルゴリズムを用いて推定する方

[*2] OR 合流ゲートウェイは AND 合流と XOR 合流のいずれかとして動作するため，AND 合流として動作したときの OR 合流ゲートウェイも同様であるが，本論文ではそれも含めて AND 合流ゲートウェイと呼ぶことにする．

© 2016 Information Processing Society of Japan

法 [1] が知られており，プロセス所要時間の標本から内部の指数分布のパラメータを推定できる．この方法を適用する場合，隠れ確率変数である推移パスと滞在時間が観測されないという仮定の下で推定を行う．

ビジネスプロセスでは単一時間イベントログが記録されることが多く，その場合正確なプロセス全体の開始時刻と終了時刻が観測されない．すなわち相型分布を推定するためのプロセス所要時間が分からないため，上記アルゴリズムを直接適用することができない．一方，ビジスプロセスのイベントログでは，アクティビティの推移パスと部分的な滞在時間は観測される．滞在時間を確率変数と見なせば，この観測された複数の確率変数はパラメータを共有しており，互いに独立ではない．このため観測された滞在時間に対してそれぞれ独立にパラメータを推定（たとえば Asmussen らの方法 [1] を用いて）しても，同じパラメータに対して異なる結果が得られてしまう．本論文では，得られた標本を直接表現するような確率モデルを構築することで，プロセス所要時間が分からなくても，観測される互いにパラメータを共有する確率変数から隠れた確率変数を推定できる方法を提案している．

相型分布は非常に高い自由度を持ち広範な非負の確率分布を近似できる．しかしその自由度の高さとは対照的に近似対象は 1 つの確率変数であるため，ある確率分布を同程度に近似可能な，内部構造やパラメータが異なる相型分布が複数存在可能になる．本論文の方法を相型分布のパラメータ推定だととらえれば，いくつかの隠れ確率変数を観測することで拘束条件が強まり，より正確な相型分布のパラメータ推定を行う方法であると解釈することができる．ビジネスプロセス改善では実際に改善できる対象は個々のアクティビティであるため，プロセス所要時間の確率分布の近似よりも個々のパラメータの推定精度や安定性が重要となる．このため，提案手法はビジネスプロセス改善に向くといえる．

提案手法では M ステップでコスト関数を最大化するパラメータを求めるため，指数関数と多項式関数を両方含んだ非線形方程式 (8) の解が必要になる．この方程式は解析的に解けないため，数値計算によって近似解を求めるか，勾配方向に移動する一般化 EM アルゴリズムを用いる必要がある．提案手法では一般化 EM アルゴリズムを採用することで数値計算を回避した．一方，一般の相型分布のパラメータ推定である Asmussen らの方法 [1] でも M ステップにおいて連立微分方程式の解が必要で，解析的に解けないため数値計算により近似解を求め，尤度関数の近似関数である $\mathcal{Q}$ の極大点にパラメータを更新する．どちらのアプローチも本質的な違いはないが，前者は方程式のソルバを必要としないため実装が簡便になるという利点があり，後者は性質の良いパラメータ上を移動するため更新ステップ数が少なくて済むという利点がある．なお，指数分布と XOR 分岐と合流をモデル化した文献 [23] では M ステップでの更新パラメータは解析的に求まる．ガンマ分布と XOR 分岐と合流をモデル化した文献 [14] では M ステップでの更新パラメータは解析的には求まらないため，数値計算により解くことで EM アルゴリズムを構成している．本論文では指数分布と XOR と AND の分岐と合流をモデル化した結果，M ステップでの更新パラメータは解析的に求められなかった．これらの比較から，指数分布と XOR 分岐と合流で構成される相型分布までが M ステップが解析的に求まる問題のクラスで，要素となる確率分布か構造のどちらかに複雑さが加わると M ステップが解析的に求まらないクラスになると考えられる．

3. ビジネスプロセス

組織が何らかの目的のために行う一連の工程を**ビジネスプロセス**と呼ぶ．たとえば，工場での製品の製造工程や，病院での診断や治療の工程，保険会社での保険金の受け取り申請，などがビジネスプロセスである．

何度も繰り返し行われる同様のビジネスプロセスは文書化することで多くの恩恵（たとえば，属人性や誤りの減少，効率化，例外の検出，品質の向上，プロセスシステムの自動生成など）が得られる．ビジネスプロセスを何らかの記法で記述したものを**プロセスモデル**と呼ぶ．これまでの多くの研究や標準化の試みの結果（詳しくは文献 [4]）として，プロセスモデルの表記法は Business Process Modeling Notation（BPMN）[15] が主流となっている．

図 1 に，典型的に頻繁に用いられる BPMN の基本要素を示した．BPMN ではアクティビティは角丸長方形で表現し，ゲートウェイはひし形で表現する．ゲートウェイは，分岐と合流がつねに対になって使用される．XOR 分岐ゲートウェイはフローでつながっているいずれかの 1 つの要素へ制御を移し，XOR 合流ゲートウェイはいずれかのフローからきた制御を次の要素に渡す．一方，AND 分岐ゲートウェイはフローでつながっているすべての要素に

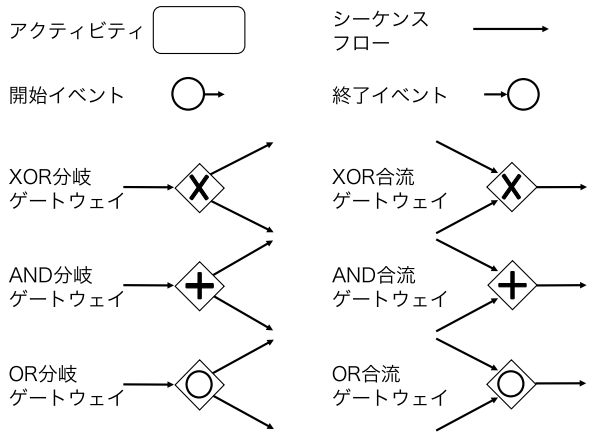

図 1　BPMN の基本要素
Fig. 1　Basic elements of BPMN.

© 2016 Information Processing Society of Japan

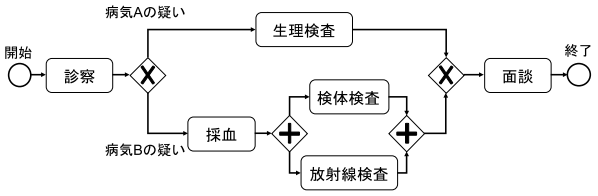

図 2　病院での診断プロセスの例
Fig. 2　Example medical diagnosis process on hospital.

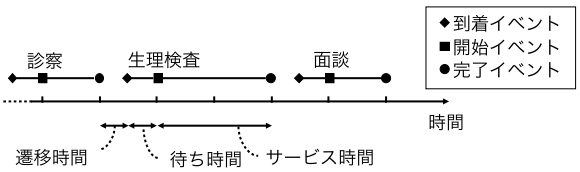

図 3　病気 A の疑いがある場合の診断プロセスの時系列イベント
Fig. 3　Time sequence events of the diagnosis process when a patient is suspected case of disease A.

制御を移し，並列に実行させる．AND 合流ゲートウェイはすべてのフローから制御が戻るまで待ってから次の要素に制御を渡す．OR 分岐と合流ゲートウェイは，XOR と AND のどちらかとして振る舞う．どちらとして振る舞うかは分岐の時点で決定され，合流時は決定された振舞いに従う．

たとえば，**図 2** に BPMN で示すような診断プロセスがあったとする．まず患者が病院に到着してから医師が診察を行い，病気 A か病気 B のどちらの疑いがあるかを判定する．病気 A の疑いがある場合は血液の検査と放射線写真の撮影が必要で，独立に実行可能であるためそれらは採血の後に並列実行される．病気 B の疑いがある場合は心電図や超音波による検査が行われる．最後にどちらの場合でも検査結果を受け取った医師と面談を行う．患者が病気 A と病気 B のどちらの疑いになるか排他的であるため，XOR 分岐と合流ゲートウェイで表現される．並列実行する検体検査と放射線検査は AND 分岐と合流ゲートウェイで表現される．

具体的な 1 つのビジネスプロセスの実行は**プロセスインスタンス**と呼ばれる．プロセスインスタンスで発生したイベントを記録したログを**イベントログ**と呼ぶ．ログ内では異なるプロセスインスタンスを区別するために**ケース ID** という固有の識別子を用いる．病院の例では，1 人の患者に対して行った一連の医療行為がプロセスインスタンスである．病気 A の疑いがある場合を**図 3** に，病気 B の疑いがある場合を**図 4** に可視化した．

3.1　性能指標

こうしたイベントからは様々な性能指標を算出することができ，性能改善の際に非常に重要な役割を果たす．プロセスインスタンスの開始から完了までを**プロセス所要時間**，アクティビティの最初のイベントから最後のイベントまでを**アクティビティ所要時間**，アクティビティ内の最初のイベントからサービスの開始までを**待ち時間**，サービスの開始からアクティビティ内の最後のイベントまでを**サービス時間**と呼ぶ．図 3 の例では，患者が窓口に到着してから診察が開始されるまでの時間が待ち時間，診察の開始から完了までの時間がサービス時間である．

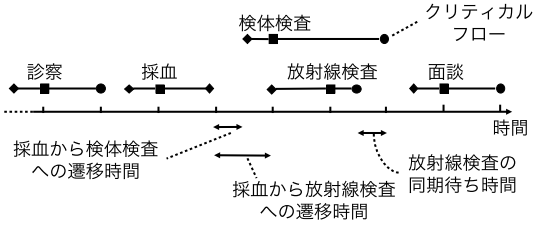

図 4　病気 B の疑いがある場合の診断プロセスの時系列イベント
Fig. 4　Time sequence events of the diagnosis process when a patient is suspected case of disease B.

遷移時間はアクティビティの外で要した時間の総称で，遷移元アクティビティの最後のイベントから遷移先アクティビティの最初のイベントまでの時間間隔である．図 3 の例では，患者が診察を終え生理検査の窓口に移動するまでが遷移時間である．AND 分岐ゲートウェイがある図 4 では，採血から放射線検査への遷移と採血から検体検査への遷移の 2 通りの遷移時間を算出できる．同様に AND 合流ゲートウェイでは放射線検査から面談への遷移と検体検査から面談への 2 通りの遷移時間を算出できる．この例では放射線検査が先に終わり，検体検査の結果を待って次の面談が開始する．

AND 合流ゲートウェイの遷移元のシーケンスフローのうち，最後に完了したフローを AND 合流ゲートウェイの**クリティカルフロー**と呼ぶ．残りすべてのフローはクリティカルフローのアクティビティの完了まで待たなければならない．この待ち時間を**同期待ち時間**と呼ぶ．この例では，放射線検査の完了から検体検査の完了までの時間が同期待ち時間である．

導出した性能指標を用いて，シミュレーションなどを用いてさらに別の性能指標（たとえば行列の長さ）を求める場合があるが，注意が必要である．待ち行列理論ではこうした加工はむしろ積極的に行われるが，それは分析対象の挙動がかなり分かっていて，かつ詳細なログが取得できるからである．たとえばコンピュータ内部の処理を待ち行列ネットワークでモデル化しシミュレーションなどでボトルネックなどを発見する研究が行われてきている．こうした研究では，CPU 数があらかじめ分かっていて変化しない，というようないくつかの仮定を一般的に用いる．それに比べて，ビジネスプロセスの対象は人間の手作業によるものが多く含まれる傾向にある．複雑な意思決定プロセスをブラックボックス化して 1 つのアクティビティと見なしたり，スキルレベルが著しく異なる複数の担当者がサービ

© 2016 Information Processing Society of Japan

スを提供するアクティビティや，利用可能な担当者の数が時間帯によって大幅に変わるアクティビティなどがごく当たり前にビジネスプロセスに用いられている．加えて，到着したプロセスを窓口に割り当てる方法に仮定を置けないことも多い．先入れ先出しの行列が使われていることもあれば，担当者がたまっているプロセスの中から好きなものを選んで開始するといった場合もある．このような状況から，ビジネスプロセスの性能分析に待ち行列理論を適用できる範囲は限られてくる．その典型的な例が後述するログの不完全性で，シミュレーションのもととなる平均サービス時間や平均待ち時間などの性能指標そのものが取得できないことが多い．こうした事情により，ビジネスプロセスの性能分析はデータの代表値（たとえば平均値や最大値や最小値）の算出にとどまらざるをえないことが多い．

3.2　単一時刻イベントログ

性能指標の算出には，詳細なイベントが記録されている必要があるが，実際には一部のイベントしか観測されないことが多く，算出できる性能指標も限られる．そうしたイベントログは**不完全イベントログ**と呼ばれる．典型的に多く見られるのは，1 つのアクティビティに対して 1 つの状態遷移と時刻のみが記録されたイベントログである．このような不完全イベントログを**単一時刻イベントログ**と呼ぶ．たとえば，稟議プロセスのような承認したことだけが重要な意味を持つビジネスプロセスの場合，完了時刻のみが記録されることがほとんどである．一方，作業がどのような結果をもたらすか予測しにくいビジネスプロセスの場合，作業の開始時刻のみを記録することが多い．たとえば，医療プロセスにおいて投薬の開始後に患者の状態が変化したことにより医師の診察にプロセスを移行した場合，投薬の完了は記録されないか，記録されたとしても役に立たず，むしろ投薬の開始時刻に価値がある．実際に，Kuo ら [10] は開始時刻のみが記録される医療システムに直面したと報告している．

例として，開始イベントのみが記録された単一時刻イベントログを**表 1** に示し，**図 5** と**図 6** に可視化した．こうした単一時刻イベントログからはアクティビティの遷移時間しか算出できないことが分かる．また，完了イベントがないため，AND 合流におけるクリティカルフローが分からなくなり，加えて同期待ち時間も算出できない．

AND 合流ゲートウェイは主にプロセスの並列実行による高速化のために用いられ，依存関係のないアクティビティの直列実行が性能上ボトルネックになった場合などに採用される．最も簡便に適用可能で効果も大きい性能改善であるため，ビジネスプロセス改善の現場では頻繁に利用される．一方，並列実行に設計されたビジネスプロセスの性能改善はさらに難しくなる．待ち時間，サービス時間に加えて，同期待ち時間やクリティカルフローによる影響を考慮

表 1　診断プロセスで観測される単一時刻イベントログの例
Table 1　Example single timestamp event log observed in the hospital process.

ケース ID	時刻	アクティビティ
1	2015-08-02 10:10:00	診察
	2015-08-02 10:30:00	生体検査
	2015-08-02 10:55:00	面談
2	2015-08-01 08:50:00	診察
	2015-08-01 09:10:00	採血
	2015-08-01 09:25:00	検体検査
	2015-08-01 09:30:00	放射線検査
	2015-08-01 09:55:00	面談
⋮	⋮	⋮

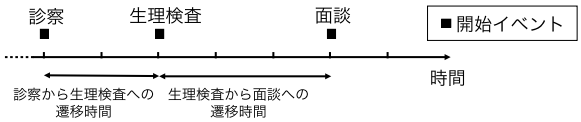

図 5　病気 A の疑いがある場合の診断プロセスの時系列イベント（開始イベントのみが記録された場合）
Fig. 5　Time sequence events of the diagnosis process when a patient is suspected case of disease A (Start events only).

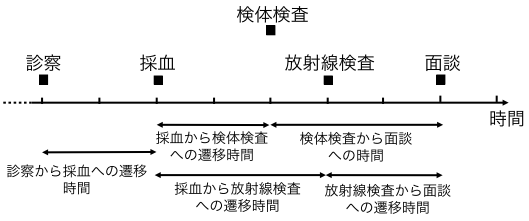

図 6　病気 B の疑いがある場合の診断プロセスの時系列イベント（開始イベントのみが記録された場合）
Fig. 6　Time sequence events of the diagnosis process when a patient is suspected case of disease B (Start events only).

しなければならないからである．それにもかかわらず不完全イベントログの影響でこれらの性能指標が利用できないため，並列実行の性能改善はきわめて困難な課題となり，解決が望まれている．Leemans ら [12] も，アクティビティの状態遷移を利用したプロセスディスカバリと性能分析の研究において，単一時刻イベントログが多いことに触れ，状態遷移を推定する方法も提案している．彼らはまた，そうしたイベントログの上で AND 合流ゲートウェイが存在したときの同期待ち時間の観測の難しさについても触れている．

4.　潜在待ち時間とサービス時間

本論文は，単一時刻イベントログのようなデータにおいても平均待ち時間と平均サービス時間を算出するため，潜在待ち時間とサービス時間という性能指標を導入する．ある遷移時間が観測されたとき，その遷移時間の前半が遷移

© 2016 Information Processing Society of Japan

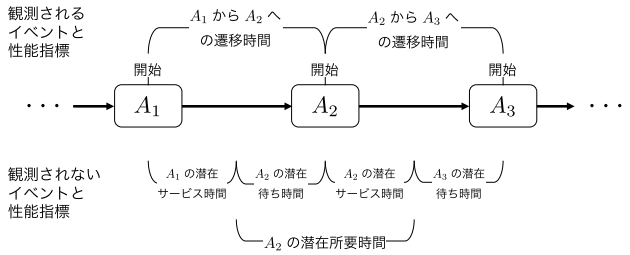

図 7　単一時刻イベントログにおける潜在待ち時間と潜在サービス時間

Fig. 7　Latent waiting and service times on single timestamp event log.

元アクティビティによって消費された時間であると仮定し，潜在サービス時間と呼ぶ．その遷移時間の後半が遷移先アクティビティによって消費された時間であると仮定し，潜在待ち時間と呼ぶ．いい換えれば，観測された遷移時間が遷移元の潜在サービス時間と遷移先の潜在サービス時間の和によって構成されていると仮定する．**図 7** に，簡単な直列フローを成すプロセスで単一時刻イベントログが観測された場合の，潜在時間を示した．

我々は，遷移時間を統計的に分析することで潜在サービス時間と潜在待ち時間の平均値を推定する方法を提案した [14], [23]．しかし，これらの手法は AND 合流ゲートウェイを通る遷移時間については扱えない．なぜならば，同期待ち時間が含まれるためである．

5. 確率モデル

本章では，アクティビティの遷移時間を潜在待ち時間と潜在サービス時間に分解し，AND 合流ゲートウェイの定式化を行う．

潜在待ち時間と潜在サービス時間の確率密度関数として，処理時間の分布として最もシンプルで多く用いられている指数分布を用いる．パラメータ $\lambda > 0$ の指数分布に従う確率変数 $X > 0$ の確率密度関数は $p(X;\lambda) = \frac{1}{\lambda}e^{-\frac{X}{\lambda}}$ である．これを $X \sim \mathrm{Exp}(\lambda)$ と書く．X の期待値 $E[X]$ は λ と等しい．またある時点よりも前に処理が完了した確率は分布関数 $p(x \leq X;\lambda) = 1 - e^{-\frac{X}{\lambda}}$ により得られる．

いま，ビジネスプロセスに N 個のアクティビティ $A_1, \ldots, A_N$ があり，各アクティビティの潜在待ち時間 W_i は $\mathrm{Exp}(\beta_i)$ に，潜在サービス時間 S_i は $\mathrm{Exp}(\alpha_i)$ に従うとする．

AND 合流ゲートウェイを通らない A_i から A_j への遷移は，文献 [23] と同様に $T_{ij} = S_i + W_j$ と仮定すれば同時確率密度関数は

$$p(T_{ij}, W_j) = p(S_i = T_{ij} - W_j)p(W_j) \tag{1}$$

となる．S_i と W_j は観測されないため，W_j に関して周辺化を行い

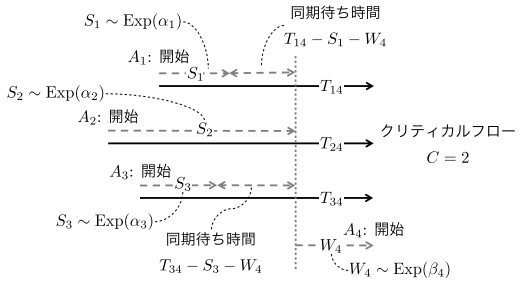

図 8　クリティカルフローが $C = 2$ と仮定した場合の AND 合流ゲートウェイの確率モデル

Fig. 8　The probabilistic model on AND-join gateway with an assumption critical flow $C = 2$.

$$p(T_{ij}) = \int_0^{T_{ij}} p(T_{ij}, W_j)dW_j \tag{2}$$

を得る．

AND 合流ゲートウェイを通る場合は複数の遷移時間を 1 つの事象として扱う．他のゲートウェイと組み合わせたとき AND 合流ゲートウェイの遷移元アクティビティは実際のプロセスに依存して決まり，事前には決まらない．そのため同じ AND 合流ゲートウェイでも，観測された遷移元アクティビティ集合と遷移先アクティビティの組合せごとに異なる確率モデルを構成する．

あるとき観測された遷移元アクティビティの添え字集合を I，遷移先アクティビティを A_j，そのときクリティカルフローとなった遷移元アクティビティの添え字を $C \in I$，観測された遷移時間を集合 $\boldsymbol{T}_{Ij} = \{T_{ij} | i \in I\}$ とする．たとえば，**図 8** に図示したようなイベントログでは，遷移元添え字集合は $I = \{1, 2, 3\}$，遷移先は A_4，遷移時間集合は $\boldsymbol{T}_{Ij} = \{T_{14}, T_{24}, T_{34}\}$ である．

A_c の遷移時間 T_{cj} と潜在時間 S_c，W_j の関係は $T_{cj} = S_c + W_j$ であるため同時確率密度関数は式 (1) と同じである．クリティカルフローとならない $i \in I$ の遷移時間 T_{ij} と潜在時間 S_i，W_j の関係は $S_i \leq T_{ij} - W_j$ である．これは A_i の潜在サービスが $T_{ij} - W_j$ よりも前に終了したため，クリティカルフローにならなかったことを意味し，同時確率密度関数は $p(T_{ij}, W_j) = p(S_i \leq T_{ij} - W_j)p(W_j)$ となる．図 8 の例では，クリティカルフローを $C = 2$ と仮定した場合の関係を図示している．T_{24} のみが，$S_2 + W_4$ の関係にあり，それ以外の遷移時間は $T_{i4} \geq S_i + W_4$, $i \in \{2, 4\}$ の関係にある．

関係するすべての事象 $\boldsymbol{T}_{Ij}$，S_i，W_j，C の同時確率密度関数は，$p(W_j)$ とすべての $P(S_i)$（$i \in I$）の積

$$p(\boldsymbol{T}_{Ij}, C, W_j) = p(W_j) \prod_{i \in I} \begin{cases} p(S_i = T_{ij} - W_j) & i = C \\ p(S_i \leq T_{ij} - W_j) & i \neq C \end{cases}$$

となる．クリティカルフロー C と潜在時間 S_i，W_j は観測されないため，周辺化することで

© 2016 Information Processing Society of Japan

$$p(\boldsymbol{T}_{Ij}) = \sum_{C \in I} \int_0^{\min(\boldsymbol{T}_{Ij})} p(\boldsymbol{T}_{Ij}, C, W_j) dW_j \qquad (3)$$

を得る．ここで，W_j のとりうる値は 0 から遷移時間 $\boldsymbol{T}_{Ij}$ の最小値 $\min(\boldsymbol{T}_{Ij})$ である．なぜならばそれよりも大きい場合，最後に開始した遷移元アクティビティの開始時刻よりも前に遷移先アクティビティが開始することになってしまうからである．

6. パラメータ推定

本論文では，最尤原理に基づき観測された遷移時間を最も尤もらしく説明するパラメータ $\widehat{\alpha}_i$，$\widehat{\beta}_i$（$i = 1, \ldots, N$）を求め，そのパラメータを用いてアクティビティ A_i の平均潜在待ち時間 $E[W_i] = \widehat{\alpha}_i$ とサービス時間 $E[S_i] = \widehat{\beta}_i$ を推定する．単一時刻イベントログでは，アクティビティ A_i の平均潜在所要時間を $E[W_i] + E[S_i]$ として推定する．

6.1 最尤推定

AND 合流ゲートウェイを通らない A_i から A_j への遷移を (i, j) を用いて表し，あるイベントログ内で観測されたすべての AND 合流ゲートウェイを通らない遷移の集合を $\mathcal{T}_1$ とする．そのイベントログにおいて，遷移 $(i, j) \in \mathcal{T}_1$ の遷移時間 $t_{ij}^{(1)}, \ldots, t_{ij}^{(n_{ij})}$ が互いに独立に n_{ij} 個観測されたとする．これらの遷移時間の尤度を L_1 とする．

AND 合流を通る複数のアクティビティ A_i（$i \in I$）から A_j への遷移を (I, j) を用いて表し，あるイベントログ内で観測されたすべての AND 合流ゲートウェイを通る遷移の集合を $\mathcal{T}_2$ とする．そのイベントログにおいて，遷移 $(I, j) \in \mathcal{T}_2$ の遷移時間集合 $\boldsymbol{t}_{Ij}^{(1)}, \ldots, \boldsymbol{t}_{Ij}^{(n_{Ij})}$ が互いに独立に n_{Ij} 個観測されたとする．これらの遷移時間の尤度を L_2 とする．

観測された遷移時間の尤もらしさを計る対数尤度 $\log L = \log L_1 + \log L_2$ は，前章の式 (2) と式 (3) を用いて

$$\log L_1 = \sum_{(i,j) \in \mathcal{T}_1} \sum_{k=1}^{n_{ij}} \log p(t_{ij}^{(k)}; \boldsymbol{\theta}) \qquad (4)$$

$$\log L_2 = \sum_{(I,j) \in \mathcal{T}_2} \sum_{k=1}^{n_{Ij}} \log p(\boldsymbol{t}_{Ij}^{(k)}; \boldsymbol{\theta}) \qquad (5)$$

となる．ここで，$\boldsymbol{\theta} = \{\alpha_1, \ldots, \alpha_N, \beta_1, \ldots, \beta_N\}$ とした．この尤度関数を最大にする $\widehat{\boldsymbol{\theta}}$ が求めたいパラメータである．しかし，この最大化問題は解析的に解くことができない．そのため，本研究では expectation maximization（EM）アルゴリズムを用いてこの問題を解く．

6.2 EM アルゴリズム

EM アルゴリズム [3] は観測されない確率変数が含まれる最尤推定問題を逐次的に解く手法である．適当な初期値からスタートし，E ステップと M ステップを繰り返し適用してパラメータを更新することで，より尤度の高いパラメータに更新できることが証明されている [2], [7], [13]．この節では，観測される確率変数を T_{ij}，観測されない確率変数を S_i，W_j，C として，EM アルゴリズムにあてはめる．アルゴリズムは，まずパラメータ $\boldsymbol{\theta}$ に適当な値を初期値として与え，M ステップで新たな $\boldsymbol{\theta}'$ に逐次更新し，尤度関数の極大点に到達したら停止する．

E ステップでは，現在のパラメータ $\boldsymbol{\theta}$ をもとにして次のパラメータ $\boldsymbol{\theta}'$ の完全対数尤度関数の期待値 $\mathcal{Q}$ を導出し，不完全対数尤度関数 (4), (5) を近似する．すなわち，式 (4) の期待値 $\mathcal{Q}_1(\boldsymbol{\theta}, \boldsymbol{\theta}')$ は

$$\sum_{(i,j) \in \mathcal{T}_1} \sum_{k=1}^{n_{ij}} \int_0^{t_{ij}^{(k)}} p(W_j | t_{ij}^{(k)}; \boldsymbol{\theta}) \log p(t_{ij}^{(k)}, W_j; \boldsymbol{\theta}') dW_j,$$

式 (5) の期待値 $\mathcal{Q}_2(\boldsymbol{\theta}, \boldsymbol{\theta}')$ は

$$\sum_{(I,j) \in \mathcal{T}_2} \sum_{k=1}^{n_{Ij}} \sum_{c \in I} \int_0^{\min(\boldsymbol{t}_{Ij}^{(k)})} p(C, W_j | \boldsymbol{t}_{Ij}^{(k)}; \boldsymbol{\theta}) \log p(\boldsymbol{t}_{Ij}^{(k)}, C, W_j; \boldsymbol{\theta}') dW_j$$

となり，あわせて $\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}') = \mathcal{Q}_1(\boldsymbol{\theta}, \boldsymbol{\theta}') + \mathcal{Q}_2(\boldsymbol{\theta}, \boldsymbol{\theta}')$ となる．ここで $p(W_j | \boldsymbol{t}_{Ij}) = p(\boldsymbol{t}_{Ij}, W_j) / p(\boldsymbol{t}_{Ij})$ である．

M ステップでは，期待値 $\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}')$ を最大化する $\boldsymbol{\theta}'$ を求める．$\mathcal{Q}$ は上に凸な関数であるため，極大値条件 $\frac{\partial \mathcal{Q}}{\partial \boldsymbol{\theta}'} = \mathbf{0}$ を満たすパラメータを求めればよい．$\mathcal{Q}_1(\boldsymbol{\theta}, \boldsymbol{\theta}')$ の偏微分を式 (6)，式 (7) に，$\mathcal{Q}_2(\boldsymbol{\theta}, \boldsymbol{\theta}')$ の偏微分を式 (8)，式 (9) に示した．

この条件式は解析的に解くことはできない．そこで一般化 EM アルゴリズムと呼ばれる緩和を採用する．すなわち M ステップでの極大化をあきらめ，$\mathcal{Q}$ が増加する $\boldsymbol{\theta}'$ に更新すればよいことにする．勾配方向を用いて極大点を探索すればよいため，たとえば勾配法や準ニュートン法を用いればよい．

6.3 経験則による初期値の設定

勾配法や EM アルゴリズムのような最適化アルゴリズムは局所的最適解を求めることができるが，それが大域的最適解であるとは限らない．局所的最適解が複数存在する場合，初期値に依存して求まる解が異なるためである．本研究では，標本から小さなコストで計算できる解を求めパラメータ推定アルゴリズムの初期値として用いる．

AND 合流ゲートウェイを通過しない i から j への遷移時間 t_{ij} が観測された場合，仮定より遷移元の潜在待ち時間 s_i と遷移先の潜在サービス時間 w_j の和で構成されている．特に前提知識がない場合 t_{ij} に対する s_i と w_j の構成比率は分からない．このような場合，構成比率が等しいと仮定すれば，真の構成比率との誤差の期待値が最小になる．すなわち 2 つの潜在時間は等しく $s_i = w_j = t_{ij}/2$ であったと考える．

AND 合流ゲートウェイを通過する遷移元アクティビティ集合 I からアクティビティ j への遷移時間 t_{Ij} が観測され

© 2016 Information Processing Society of Japan

$$\frac{\partial \mathcal{Q}_1}{\partial \alpha'_h} = \sum_{\substack{(i,j)\in\mathcal{T}_1 \\ i=h}} \sum_{k=1}^{n_{ij}} \left(-\frac{1}{\alpha'_h} + \frac{1}{\alpha'^2_h} \left(t_{hj}^{(k)} - \frac{1}{p(t_{hj}^{(k)};\boldsymbol{\theta})} \int_0^{t_{hj}^{(k)}} p(t_{hj}^{(k)}, W_j;\boldsymbol{\theta}) W_j dW_j \right) \right) \tag{6}$$

$$\frac{\partial \mathcal{Q}_1}{\partial \beta'_h} = \sum_{\substack{(i,j)\in\mathcal{T}_1 \\ j=h}} \sum_{k=1}^{n_{ij}} \left(-\frac{1}{\beta'_h} + \frac{1}{\beta'^2_h p(t_{ih}^{(k)};\boldsymbol{\theta})} \int_0^{t_{ih}^{(k)}} p(t_{ih}^{(k)}, W_h;\boldsymbol{\theta}) W_h dW_h \right) \tag{7}$$

$$\frac{\partial \mathcal{Q}_2}{\partial \alpha'_h} = \sum_{\substack{(I,j)\in\mathcal{T}_2 \\ I\ni h}} \sum_{k=1}^{n_{Ij}} \left(\frac{1}{p(\boldsymbol{t}_{Ij}^{(k)};\boldsymbol{\theta})} \sum_{C\in I} \int_0^{\min(\boldsymbol{t}_{Ij}^{(k)})} p(\boldsymbol{t}_{Ij}, C, W_j;\boldsymbol{\theta}) \left\{ \begin{array}{ll} \left(-\frac{1}{\alpha'_h} + \frac{t_{hj}^{(k)} - W_j}{\alpha'^2_h} \right) & C = h \\ \left(-\frac{t_{hj}^{(k)} - W_j}{\alpha'^2_h (e^{(t_{hj}^{(k)} - W_j)/\alpha'_h} - 1)} \right) & C \neq h \end{array} \right\} dW_j \right) \tag{8}$$

$$\frac{\partial \mathcal{Q}_2}{\partial \beta'_h} = \sum_{\substack{(I,j)\in\mathcal{T}_2 \\ j=h}} \sum_{k=1}^{n_{Ij}} \left(-\frac{1}{\beta'_h} + \frac{1}{\beta'^2_h p(\boldsymbol{t}_{Ih}^{(k)};\boldsymbol{\theta})} \sum_{C\in I} \int_0^{\min(\boldsymbol{t}_{Ih}^{(k)})} p(\boldsymbol{t}_{Ih}, C, W_h;\boldsymbol{\theta}) W_h dW_h \right) \tag{9}$$

た場合，クリティカルフローとなった場合は前述の 2 つの時間の和で構成され，ならなかった場合は同期待ち時間を加えた 3 つの時間で構成される．どちらの場合かは本論文で前提としている不完全イベントログからは分からないことと，遷移元アクティビティが複数あった場合 1 つを除いてすべてがクリティカルフローにならないことから，すべての遷移時間が 3 つの時間の和によって構成されそれらが等しいと考え，観測された t_{ij}（$i \in I$）に対して $s_i = w_j = t_{ij}/3$ をそれぞれの実現値として考える．

このようにすべての観測された遷移時間に対して経験則によって算出した実現値の平均値をパラメータの初期値とする．

7. 実験

この章では，真の値が分かっているデータを AND 合流ゲートウェイで生成し，提案手法が潜在平均待ち時間とサービス時間を推定できることを実験により示す．提案手法は GNU Octave を用いて実装した．また，一般化 EM アルゴリズムの更新には Octave に標準添付される準ニュートン法を関数 sqp を通じて利用した．

7.1 標本数と推定精度の関係

AND ゲートウェイを含む単純なビジネスプロセスを用いて，標本数に対する提案手法の推定精度について実験により評価する．**図 9** に示す AND 合流ゲートウェイを構成し，A_1 の潜在サービス時間を $S_1 \sim \mathrm{Exp}(1)$，潜在待ち時間を $W_1 \sim \mathrm{Exp}(1)$，A_2 の潜在サービス時間を $S_2 \sim \mathrm{Exp}(2)$，潜在待ち時間を $W_2 \sim \mathrm{Exp}(2)$，A_3 の潜在待ち時間を $W_3 \sim \mathrm{Exp}(3)$ とする．観測した時間をもとに推定すべき真の値は，A_1 の平均潜在サービス時間は $E[S_1] = \alpha_1 = 1$，A_2 の平均潜在サービス時間は $E[S_2] = \alpha_2 = 2$，A_3 の平均潜在待ち時間は $E[W_3] = \beta_3 = 3$ である．AND 合流ゲートウェイを通過する遷移時間のセットを乱数を用いて生成し，そのデータをもとに推定を行った．

図 10 に，生成したプロセスインスタンス数 25，50，100，200，400，800，1,600，3,200 における推定値の 10 回の実験の平均値を棒グラフとして示した．誤差バーは，10 回の実験の標準偏差を示す．

この図より，標本数が増えると推定値が真の値 $(\alpha_1, \alpha_2, \beta_3) = (1, 2, 3)$ に近づき，推定値の分散も小さくなることが分かる．

図 11 に，この実験で使用したイベントの平均的な時間軸での工程を示した．プロセスが開始ノードから開始し，AND 分岐ゲートウェイによって 2 つの並列プロセスに分割される．A_1 の潜在待ち時間に平均 1 秒，潜在サービス時間に平均 1 秒消費した後，多くの場合もう 1 つのプロセスを待つ．A_2 の潜在待ち時間に平均 2 秒，潜在サービス時間に平均 2 秒消費した後，AND 合流ゲートウェイで 2 つのプロセスを合流させた後に A_3 の潜在待ち時間に平均

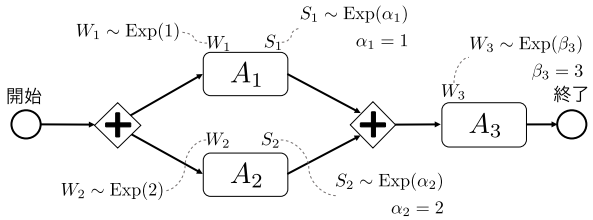

図 9　実験に使用した AND 合流ゲートウェイと確率分布
Fig. 9　An AND-join gateway and probabilistic distributions used in the experiment.

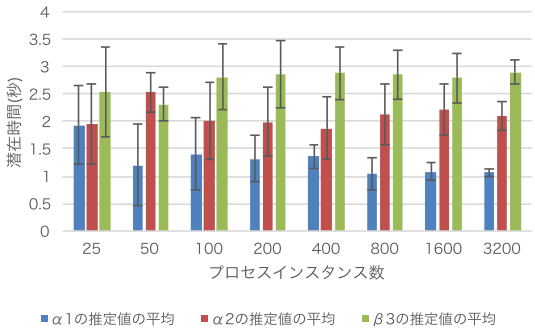

図 10　標本数と推定値の関係
Fig. 10　Estimators v.s. sample size.

© 2016 Information Processing Society of Japan

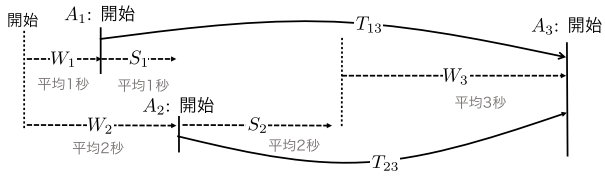

図 11　AND 合流ゲートウェイ（図 9）の平均的なイベントの関係
Fig. 11　An AND-join gateway and probabilistic distributions used in experiment.

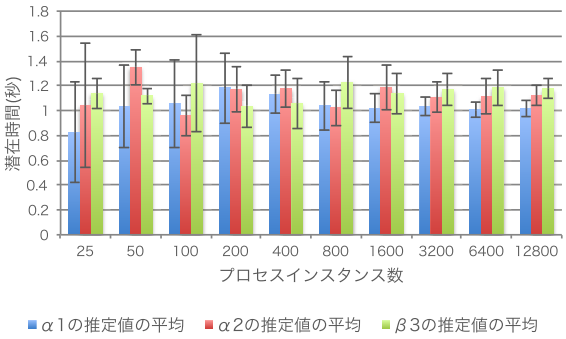

図 12　標本数と推定値の関係（差異の小さなパラメータの場合）
Fig. 12　Estimators v.s. sample size with similar parameters.

3 秒消費する．この場合観測可能な性能指標は，A_1 の開始時刻から A_3 への開始時刻の時間間隔 T_{13} と，A_2 の開始時刻から A_3 への開始時刻の時間間隔 T_{23} のみである．

このビジネスプロセスでは，平均的に T_{13} は T_{23} よりも大きくなる．このため，初期値を 6.3 節で示した方法で設定した場合，A_1 の潜在サービス時間 W_1 が A_2 の潜在サービス時間 W_2 よりも大きくなる．真の解に近づくためには，逐次更新の途中でこの大小関係を逆転させなければならない．標本数が少ないとき，この大小関係を崩すことができずに逐次更新が収束する場合があったため，この問題には局所解が存在していることが実験により分かった．一方，標本数を多いときはこの大小関係を逆転させ真の解付近までパラメータが移動する．

7.2　標本数と推定精度の関係（差異の小さいパラメータ）

前節では，ある程度大きさの異なるパラメータのもとで実験を行ったが，この実験では前節に比べ差異の小さいパラメータの場合について，標本数と推定誤差の関係を評価する．前節と同様に，図 9 に示す AND 合流ゲートウェイを構成し，一部のパラメータを $(\alpha_1, \alpha_2, \beta_3) = (1, 1.1, 1.2)$ に変更した．その後乱数を用いて遷移時間を生成し，そのデータをもとに推定を行った．**図 12** に，生成したプロセスインスタンス数 25，50，100，200，400，800，1,600，3,200，6,400，12,800 における推定値の 10 回の実験の平均値を棒グラフとして示した．誤差バーは，10 回の実験の標準偏差を示す．この図より，標本数が増えると推定値が真の値 $(\alpha_1, \alpha_2, \beta_3) = (1, 1.1, 1.2)$ に近づき，推定値の分散も小さくなることが分かる．しかし前節と比べ，パラメータどうしが異なる値を持つことを知るには多くの標本が必要になることが分かる．

一般に，統計的推定ではパラメータどうしの小さな差分を高い確信度で推定するには，多くの標本が必要になることが知られている．通常の最尤法による推定を行う場合は，小さな標本数での予備実験の結果と求める確信度から，そのために必要な最小限の標本数を類推することができる．製薬のように動物実験を行う場合に，必要以上の動物を薬効確認の実験に晒すことを避けるためにこのような方法がとられる．しかし本論文では EM アルゴリズムによる逐次更新によりパラメータを推定するため，必要な標本数を類推することが困難である．そのため，前節や本節のように標本数を変え何度か実験することにより必要な標本数を類推する必要がある．

7.3　様々な大小関係のパラメータに対する推定精度の変化

本節では，図 9 の α_1，α_2，β_3 に大きさが異なる真の値 5 と 1 を設定し，どのような大小関係のときに推定が正しく動作するかを検証する．5 と 1 を代入した異なる真の値 α_1，α_2，β_3 の 7 通りの組合せについてそれぞれプロセスインスタンスを 800 生成し，推定値の 10 回の実験の平均値と標準偏差と真の値との誤差の平均値と汎化誤差を**表 2** に示した．ここで汎化誤差の平均値とは，推定値と真の値との差の絶対値の 10 回の実験の平均値である．

設定 1，2，3，4 のとき，小さな誤差で推定できたが，設定 5，6，7 のときは，大小関係が真の値と異なるパラメータが推定された．設定 1，2，3，4 に共通する点は，遷移先の平均潜在待ち時間が，遷移元の平均潜在サービス時間と同じかそれよりも大きいことである．一方，設定 5，6，7 に共通する点は遷移先の平均潜在待ち時間が，遷移元の平均潜在サービス時間よりも小さいことである．このことから，遷移元の平均潜在サービス時間が遷移先の平均潜在待ち時間に比べて大きいような場合に初期値による問題がおきてくると考えられる．

7.4　初期値と推定精度の関係

本節では，前節で推定誤差が大きかった設定 5，6，7 について，真の解の近くから推定を始めた場合に真の解に収束するのかどうかを検証する．前節と同様にプロセスインスタンス数 800 で標本を生成し，真の解付近から推定を開始させた 10 回の実験の平均値と標準偏差と汎化誤差を**表 3** に示した．

この実験では 3 つの設定すべてにおいて，真の値に近い値を推定できた．このことから，前節での推定誤差が大きかった解が局所解であり，性質の良い初期値から開始すれば推定誤差が小さい推定値に収束することが分かった．図 11 で示したように，真の大小関係とは異なる大小関係の遷移時間が観測されることがある．たとえば，この図で

© 2016 Information Processing Society of Japan

表 2　異なるパラメータの下での推定値と誤差（プロセスインスタンス数 800，経験則による初期値）

Table 2　Estimators and errors over various true parameters (800 process instances, heuristic initialization).

設定	パラメータ	真の値	初期値	推定値の平均値	汎化誤差の平均値	対数尤度の平均値
1	α_1	1	1.29	1.01±0.20	0.13	−0.817893
	α_2	1	1.01	0.92±0.10	0.12	
	β_3	1	1.33	1.07±0.15	0.14	
2	α_1	1	2.77	0.99±0.28	0.22	−1.382378
	α_2	1	2.63	1.02±0.13	0.10	
	β_3	5	3.09	5.06±0.19	0.17	
3	α_1	1	4.05	1.57±0.25	0.57	−1.570569
	α_2	5	4.05	4.31±0.43	0.69	
	β_3	5	4.61	5.42±0.61	0.57	
4	α_1	5	3.87	4.68±0.91	0.74	−1.556994
	α_2	1	3.87	1.22±0.27	0.29	
	β_3	5	4.40	5.16±0.95	0.74	
5	α_1	1	2.58	0.49±0.17	0.51	−1.334855
	α_2	5	2.43	1.20±0.14	3.80	
	β_3	1	2.86	4.59±0.23	3.59	
6	α_1	5	2.44	1.71±1.00	3.29	−1.305542
	α_2	1	2.28	0.40±0.13	0.60	
	β_3	1	2.69	4.08±0.96	3.08	
7	α_1	5	3.38	2.64±0.75	2.36	−1.46871
	α_2	5	3.30	2.00±0.76	3.00	
	β_3	1	3.81	4.87±0.30	3.87	

表 3　異なるパラメータの下での推定値と誤差（プロセスインスタンス数 800，真の値に近い初期値）

Table 3　Estimators and errors over various true parameters (800 process instances, initialized by near true parameters).

設定	パラメータ	真の値	初期値	推定値の平均値	汎化誤差の平均値	対数尤度の平均値
5	α_1	1	2.00	0.95±0.22	0.18	−1.356998
	α_2	5	4.00	5.08±0.22	0.20	
	β_3	1	2.00	1.06±0.21	0.19	
6	α_1	5	4.00	4.91±0.24	0.18	−1.289309
	α_2	1	2.00	1.03±0.30	0.23	
	β_3	1	2.00	1.03±0.31	0.25	
7	α_1	5	4.00	4.55±1.02	0.97	−1.464521
	α_2	5	4.00	5.04±0.67	0.54	
	β_3	1	2.00	1.16±0.33	0.28	

は一見して A_1 の潜在サービス時間のほうが A_2 よりも大きいと推定することが自然である．すなわち，AND 合流の問題は遷移元のアクイティビティの潜在サービス時間について対称性があり，パラメータの大小関係が入れ替わる境界線上に尤度関数の極大点を分けるような谷が存在し，尤度関数が双峰になっていると考えられる．

設定 6，7 では真の値に近い推定値がより大きな対数尤度を持った．しかし設定 5 では誤差が大きい局所解による推定値のほうが大きな対数尤度を持った．このことから，対数尤度の大小で局所解どうしの良し悪しを決めることができないことが分かった．

7.5　遷移元と遷移先の潜在時間の大小関係と精度との関係

7.3 節において，推定誤差が大きかった設定 5，6，7 では，遷移先の潜在待ち時間のパラメータ $\beta_3 = 1$ が遷移元のパラメータに対して小さく，推定誤差が小さかった設定 1，2，3，4 では $\beta_3 = 5$ と遷移元のパラメータに対して同じ大きさであるという違いがあることが分かる．このことから，遷移元の潜在サービス時間に対して，遷移先の潜在待ち時間が小さいときに誤差が大きくなり，大きいときに誤差が小さくなるという仮説を立てることができる．

本節では，この仮説を確認するため，遷移先の潜在待ち時間を変化させ，推定精度との関係を評価する．遷移元の潜

© 2016 Information Processing Society of Japan

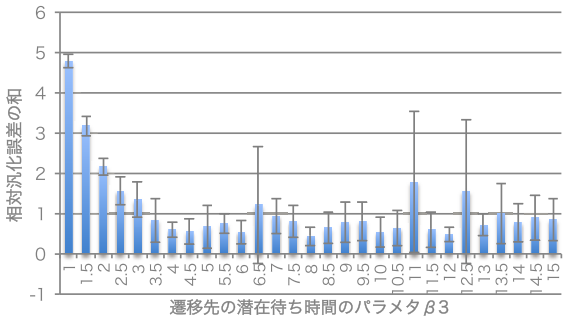

図 13　遷移先の潜在待ち時間の大きさと推定誤差との関係
Fig. 13　Estimation errors v.s. latent waiting time.

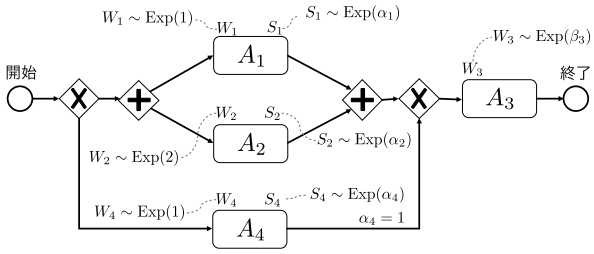

図 14　実験に使用した AND 合流ゲートウェイと確率分布
Fig. 14　An AND-join gateway and probabilistic distributions used in experiment.

在サービス時間のパラメータは 7.3 節の設定 5 の $\alpha_1 = 1$, $\alpha_2 = 5$ を用い，遷移先の潜在待ち時間のパラメータ β_3 を 1 から 15 まで 0.5 ずつ変化させた．それぞれのパラメータのもとで，プロセスインスタンスを 800 生成して相対汎化誤差の和を計測し，10 回の実験の平均値と標準偏差を**図 13** に示した．この実験ではパラメータの値が大きく異なるため，比較のため相対汎化誤差の和 $\widehat{\alpha_1}/\alpha_1 + \widehat{\alpha_2}/\alpha_2 + \widehat{\beta_3}/\beta_3$ を計測した．

この実験では，遷移先の潜在待ち時間が小さいとき推定誤差が大きく，大きいとき推定誤差が小さくなった．これは，共有している確率変数の割合が大きくなるため，β の推定が簡単になったからだと考えられる．たとえば，仮に遷移先の潜在待ち時間が遷移元の潜在待ち時間の影響がゼロといえるほど大きかった場合，1 変数の推定問題となり簡単な問題になる．一方，仮に遷移先の潜在待ち時間がゼロに近づけば，オーバラップがほとんどなくなり，互いに独立な複数の変数の推定問題となり，難しい問題となる．これが，7.3 節で推定誤差に違いが起きた理由であると考えられる．

7.6　プロセスモデルの構造と推定精度の関係

本節では，図 9 に新たにアクティビティ A_4 を加えた**図 14** のプロセスモデルを用いて，AND 合流と XOR 合流の両方の遷移時間が観測された場合の推定精度について検証する．7.3 節では，AND 合流ゲートウェイに起因する不完全さにより，初期値による問題が起きていたが，別のパスからのフローが観測され，β_3 を共有するような標本が得られた場合，β_3 の推定精度が向上することが期待される．このプロセスモデルから観測された標本の尤度関数は，XOR 合流ゲートウェイの尤度 $\log L_1$（式 (4)）と AND 合流ゲートウェイの尤度 $\log L_2$（式 (5)）の和で構成される．前節までの実験では尤度は $\log L_2$ のみで構成されていたため，この実験では 2 つの尤度関数の和を目的関数としたときの挙動を検証することができる．

A_4 を通るプロセスインスタンスを 400 生成し，A_1，A_2 を通るプロセスインスタンスを 400 生成し，合計 800 個のプロセスインスタンスを標本として生成した．真の解付近から推定を開始させた 10 回の実験の平均値と標準偏差と汎化誤差を**表 4** に示した．7.3 節と同様に経験則による初期値を用いたにもかかわらず，すべての設定で良い推定値に収束した．このことから，AND 合流ゲートウェイ単体が観測された場合よりも，部分的にオーバラップするように別のフローが観測された場合，初期値の問題が緩和され，推定精度が向上することが分かった．7.3 節の実験では設定 5，6，7 において，真の値が 1 の β_3 が 2.5 以上の大きな値に推定されてしまったために他のパラメータの推定にも影響を与え，すべてのパラメータの推定値の誤差が大きくなった．しかし，本節では A_4 から A_3 へのフローが観測され，設定 5，6，7 ではこの遷移時間の期待値は $E[S_4 + W_3] = \alpha_4 + \beta_3 = 2$ となり，当然観測された遷移時間も平均が 2 になるような標本が得られる．したがってこのフローの尤度関数は β_3 について，0 から 2 の間のどこかに極大点を持つような関数になる．このような尤度関数が加わることで，β_3 に関する極大点の位置が 2.5 以上の位置からより小さな値に移動したと考えられる．

8.　おわりに

本研究では，AND 合流ゲートウェイを含むビジネスプロセスからの不完全なイベントログであっても，平均潜在待ち時間とサービス時間を推定する方法を提案した．これまで現実問題として典型的に観測されるイベントログが不完全であるためにできなかった性能分析が，提案手法を利用することで可能になった．このような性能指標はビジネスプロセスの性能改善だけでなく，性能に関する分析の特徴量としても用いることができるため，多くの応用が期待できる．

また，人工的に生成させたログを用いた数値実験により，提案手法が正しく推定できることを示した．この研究で取り組んだ課題は，待ち時間，サービス時間，クリティカルパス，同期待ち時間が観測されないという厳しい条件であるにもかかわらず，正しい推定値を得ることができることが分かった．また実験の結果，対象となるプロセスモデルに依存して初期値による問題がおきることが分かった．この問題は，同じパラメータを共有するような異なるパスも追加で観測されれば緩和されることが分かった．現実問題

© 2016 Information Processing Society of Japan

表 4　AND 合流と XOR 合流の両方を含むプロセスモデルにおいて，異なるパラメータの下での推定値と誤差（プロセスインスタンス数 800，経験則による初期値）

Table 4　Estimators and errors over various true parameters on the process model which has AND-join and XOR-join (800 process instances, heuristic initialization).

設定	パラメータ	真の値	初期値	推定値の平均値 ± 標準偏差	汎化誤差の平均値	対数尤度の平均値
1	α_1	1	1.26	0.98±0.17	0.15	−1.067418
	α_2	1	1.03	1.14±0.21	0.17	
	β_3	1	1.31	0.92±0.21	0.16	
	α_4	1	1.33	1.05±0.26	0.21	
2	α_1	1	2.72	1.23±0.58	0.44	−1.816994
	α_2	1	2.60	0.95±0.26	0.22	
	β_3	5	3.29	4.84±0.22	0.20	
	α_4	1	3.79	0.99±0.17	0.14	
3	α_1	1	4.04	1.49±1.44	0.76	−1.963659
	α_2	5	4.00	4.64±1.23	0.66	
	β_3	5	4.38	4.99±0.31	0.23	
	α_4	1	4.14	0.92±0.19	0.17	
4	α_1	5	3.88	5.18±0.42	0.34	−1.948688
	α_2	1	3.85	0.94±0.21	0.17	
	β_3	5	4.25	4.90±0.24	0.20	
	α_4	1	4.16	1.05±0.11	0.08	
5	α_1	1	2.47	1.61±1.35	0.80	−1.416955
	α_2	5	2.26	4.74±0.84	0.40	
	β_3	1	2.35	0.89±0.21	0.19	
	α_4	1	1.39	1.09±0.22	0.18	
6	α_1	5	2.34	5.07±0.29	0.25	−1.380908
	α_2	1	2.07	1.16±0.38	0.31	
	β_3	1	2.21	0.87±0.29	0.26	
	α_4	1	1.37	1.09±0.27	0.24	
7	α_1	5	3.14	4.67±0.75	0.52	−1.492248
	α_2	5	2.92	5.01±0.61	0.47	
	β_3	1	2.92	1.11±0.19	0.18	
	α_4	1	1.38	0.87±0.23	0.22	

として，AND 分岐と合流ゲートウェイだけが単体で用いられるようなビジネスプロセスは稀であり，いくつかのアクティビティやゲートウェイ要素が使われることが多いため，同じパラメータを共有する複数のパスが観測されると思われる．

本論文では説明のため単一時刻イベントログに焦点を当てて論じたが，提案手法は遷移時間を分解する手法であるため，完全なイベントログにおいても適用可能である．

本論文では時間間隔の確率分布として指数分布を仮定した．この仮定は多くの場合で現実的だが，より複雑な場合は整合性を保てなくなる．たとえば複数の小さなタスクがアクティビティ内に内包されているが外からそれを観測できない場合，処理時間は指数分布の和になる．こうした一般の処理時間の分布としてはガンマ分布が知られており，提案手法をガンマ分布をもとにした手法に拡張すればより多くの場面で整合性を失うことなく推定が可能になるため，今後の研究課題である．

参考文献

[1] Asmussen, S., Nerman, O. and Olsson, M.: Fitting phase-type distributions via the EM algorithm, *Scandinavian Journal of Statistics*, Vol.23, No.4, pp.419–441 (1996).

[2] Csiszàr, I. and Tusnady, G.: Information geometry and alternating minimization procedures, *Statistics and Decisions*, No.1, pp.205–237 (1984).

[3] Dempster, A.P., Laird, N.M. and Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society, Series B*, Vol.39, No.1, pp.1–38 (1977).

[4] Dumas, M.: From models to data and back: The journey of the BPM discipline and the tangled road to BPM 2020, *BPM 2015*, *LNCS*, Vol.9253, Springer, Heidelberg (2015).

[5] Dumas, M., van der Aalst, W.M.P. and ter Hofstede, A.H.M.: *Process-Aware Information Systems: Bridging People and Software through Process Technology*, Wiley (2005).

[6] Ferreira, D.R.: Performance analysis of healthcare processes through process mining, *ERCIM News 89*, pp.18–19 (2012).

© 2016 Information Processing Society of Japan

[7] Hathaway, R.J.: Another interpretation of the EM algorithm for mixture distributions, *Statistics & Probability Letters*, Vol.4, No.2, pp.53–56 (1986).
[8] Hornix, P.T.: Performance Analysis of Business Processes through Process Mining, Master's Thesis, Eindhoven University of Technology (2007).
[9] IEEE Task Force on Process Mining: Process mining manifesto, *BPM 2011 Workshops*, *LNBIP*, Daniel, F., Barkaoui, K. and Dustdar, S. (Eds.), Vol.99, pp.169–194, Springer, Heidelberg (2012).
[10] Kuo, Y.-H., Leung, J.M.Y. and Graham, C.A.: Simulation with data scarcity: Developing a simulation model of a hospital emergency department, *Proc. 2012 Winter Simulation Conference*, pp.1–12, IEEE (2012).
[11] Lanz, A., Weber, B. and Reichert, M.: Time patterns for process-aware information systems, *Requirements Engineering*, Vol.19, No.2, pp.113–141 (2014).
[12] Leemans, S.J., Fahland, D. and van der Aalst, W.M.: Using Life Cycle Information in Process Discovery, *BPM 2015 Workshops*, *LNBIP*, Springer, Heidelberg (2015).
[13] Neal, R.M. and Hinton, G.E.: A new view of the EM algorithm that justifies incremental and other variants, *Learning in Graphical Models*, pp.355–368, Kluwer Academic Publishers (1993).
[14] Nogayama, T. and Takahashi, H.: Estimation of average latent waiting and service times of activities from event logs, *BPM 2015*, *LNCS*, Vol.9253, pp.172–179, Springer, Heidelberg (2015).
[15] OMG: Business Process Model and Notation (BPMN) (2010).
[16] Rogge-solti, A. and Kasneci, G.: Temporal anomaly detection in business processes, *BPM2014*, *LNCS*, Sadiq, S., Soffer, P. and Hagen, V. (Eds.), Vol.8659, pp.234–249, Springer, Heidelberg (2014).
[17] Senderovich, A., Weidlich, M., Gal, A. and Mandelbaum, A.: Queue mining — Predicting delays in service processes, *CAiSE 2014*, *LNCS*, Jarke, M., Mylopoulos, J., Quix, C., Rolland, C., Manolopoulos, Y., Mouratidis, H. and Horkoff, J. (Eds.), Vol.8484, pp.42–57, Springer, Heidelberg (2014).
[18] Sindhgatta, R., Dasgupta, G.B. and Ghose, A.: Analysis of operational data for expertise aware staffing, *BPM 2014*, *LNCS*, Sadiq, S., Soffer, P. and Hagen, V. (Eds.), Vol.8659, pp.317–332, Springer, Heidelberg (2014).
[19] van der Aalst, W.M.P., Pesic, M. and Song, M.: Beyond process mining: From the past to present and future, *CAiSE 2010*, *LNCS*, Pernici, B. (Ed.), Vol.6051, pp.38–52, Springer, Heidelberg (2010).
[20] van der Aalst, W.M.P., Schonenberg, M.H. and Song, M.: Time prediction based on process mining, *Information Systems*, Vol.36, No.2, pp.450–475 (2011).
[21] Zerguini, L.: On the estimation of the response time of the business process, *17th UK Performance Engineering Workshop* (2001).
[22] 加藤光幾（訳）：IEEE Task Force on Process Mining：プロセスマイニングマニフェスト，入手先 ⟨http://www.win.tue.nl/ieeetfpm/lib/exe/fetch.php?media=shared:pmm-japanse-v1.pdf⟩.
[23] 野ヶ山尊秀：アクセスログの滞在時間を処理時間と次の遷移ページの前処理時間とに分解する方法，電子情報通信学会技術研究報告，LOIS，ライフインテリジェンスとオフィス情報システム，Vol.114, No.32, pp.39–43 (2014).
[24] 牧本直樹：待ち行列アルゴリズム—行列解析アプローチ，朝倉書店 (2001).

野ヶ山 尊秀　（正会員）

昭和 54 年生．平成 14 年電気通信大学電子情報通信学科卒業．平成 16 年同大学大学院博士前期課程修了．同年日本アイ・ビー・エム株式会社入社，東京基礎研究所に所属．機械学習による故障検知，プログラムの性能分析と高速化，ビジネスプロセス分析，セキュリティデータの異常検知の研究に従事．IEEE Task Force on Process Mining，電子情報通信学会各会員．

高橋 治久　（正会員）

昭和 27 年生．昭和 50 年電気通信大学電気通信学部通信工学科卒業．昭和 52 年同大学大学院修士課程修了．昭和 55 年大阪大学大学院工学研究科博士後期課程修了．博士（工学）．同年豊橋技術科学大学助手．昭和 61 年電気通信大学講師を経て，現在，同教授．現在形式ニューラルネットワーク，学習等の研究に従事．昭和 59 年度電子情報通信学会学術奨励賞受賞，電子情報通信学会，国際ニューラルネットワーク学会各会員．

© 2016 Information Processing Society of Japan