

## 知識工学的手法による待ち行列ネットワークの ボトルネック診断†

沢村 淳<sup>††</sup> 志田 圭介<sup>†††</sup>  
本位田 真一<sup>††††</sup> 伊藤 潔<sup>†††††</sup>

対象システムに悪影響を及ぼすものとその要因を解明し改善する作業は、一般的に手間がかかる。また、対象システムの規模が大きく複雑であれば、改善プランも数多く存在する。さらに、ある箇所を修正すると周辺への影響も大きい。このような作業は、通常、専門的な知識と経験を用いて行われている。これらの作業に対してエキスパートシステムを導入すると、評価や改善の効率を高めると期待できる。待ち行列ネットワークの形態にモデル化できるシステムを対象システムとする。また、この対象システムには過大な負荷がかかっている、すなわち、そのモデルである待ち行列ネットワークは、非定常（過負荷）状態にある、あるいはその可能性をもつシステムとする。このような待ち行列ネットワークに悪影響を及ぼすものとして、稼働率や待ち行列長が過大な窓口（ボトルネック窓口）に着目する。このような窓口の同定、その要因の解明、および改善プランの提示を行う“ボトルネック診断エキスパートシステム (Bottleneck Diagnosis Expert System: BDES)”について考察する。そこで用いられるヒューリスティックな知識は、対象システム内の窓口の接続の形状やパラメータの大小関係により詳細に分類したものである。これらは、Prolog 言語の事実や規則を用いてインプリメントされた。

### 1. はじめに

本稿では、待ち行列ネットワーク<sup>6)</sup>の形態にモデル化できるシステムを対象システムとし、待ち行列ネットワークの性能パラメータの測定平均値に対して、知識工学的手法<sup>1), 2), 7)</sup>を用いて、対象システムの性能に悪影響を及ぼすものを検出し、その要因を解明し、改善プランを提示する手法について検討する。

対象システムには過大な負荷がかかっている、すなわち、そのモデルである待ち行列ネットワークは、非定常（過負荷）状態にある、あるいはその可能性をもつシステムとする。対象システムに悪影響を及ぼすものとして、待ち行列ネットワークの中で、稼働率が過大、あるいは待ち行列長が過大な窓口（ボトルネック窓口）に着目する。このボトルネック窓口の同定、その要因の解明、および択一的に選択可能な改善プランの提示を行う“ボトルネック診断エキスパートシステム (Bottleneck Diagnosis Expert System: BDES)”

について考察する。

ボトルネックを改善すると、ある窓口の能力や到着量が増減し、その稼働率が変化する。この結果、周辺に影響が出てときには他の窓口をボトルネックとしてしまう場合もある。このように改善作業は必ずしも容易ではないため、専門的な知識と経験を組み込んだエキスパートシステムの導入が有効である。

ボトルネックの改善ではないが、システムの性能評価や改善に適用するエキスパートシステムとして、OS のモニタリングデータを診断し OS の性能改善を行う本位田らの研究<sup>4)</sup>や麻生川らの研究<sup>2)</sup>がある。また、モデル作成エキスパートシステムによって計算機システムの解析モデルを自動的に作成し、この性能評価をエキスパートシステムで行う村瀬らの研究<sup>6)</sup>がある。このように性能評価や改善に対するエキスパートシステムの導入は、評価や改善の効率を高めると期待できる。

本研究では、ボトルネック改善の作業は図1に示す流れで行う。まず、実測やシミュレーションなどの方法で性能パラメータの測定を行い、それらの平均値を求める。BDES は、この性能パラメータのうち、窓口の平均稼働率や平均待ち行列長を調べて、対象システム内に存在する1つ以上のボトルネックを検出する。ユーザが1つのボトルネックを選択すると、BDES は、そのボトルネックを改善する択一的に選択可能な1つ以上の改善プランを提示する。ユーザは、この中

† Bottleneck Diagnosis for Queuing Network Using Knowledge Engineering by JUN SAWAMURA (Faculty of Science and Technology, Sophia University), KEISUKE SHIDA (Faculty of Science and Technology, Sophia University), SHINICHI HONIDEN (Systems and Software Engineering Laboratory, Toshiba Corporation) and KIYOSHI ITOH (Faculty of Science and Technology, Sophia University).

†† 上智大学大学院理工学研究科機械工学専攻博士前期課程

††† 上智大学理工学部機械工学科

†††† (株)東芝システムソフトウェア技術研究所

††††† 上智大学理工学部一般科学研究室情報科学部門

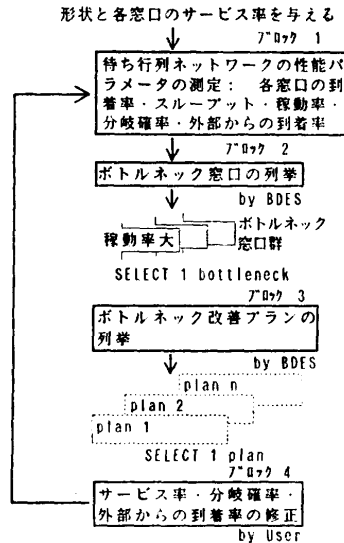


図1 BDESを用いた改善サイクル  
Fig. 1 Improvement cycle by BDES.

から任意のプランを選択し、その改善プランに示された窓口のサービス率・分岐確率および外部からの到着率の修正を検討する。他のボトルネックに対してもユーザは、同じ操作を行う。

本稿では、まず、待ち行列ネットワーク内の各窓口の稼働率の平均値と待ち行列長の平均値によってボトルネックの箇所を同定する初期診断のための知識と、その要因を解明し、改善プランを提示するための知識を整理する。次に BDES の機能について述べる。BDES には、Prolog 言語の事実や規則で記述されたボトルネック改善についての知識が組み込まれている。また、この実行例を提示する。最後に、待ち行列理論との関係を述べた後、今後の課題を述べる。

## 2. 対象システム

単一窓口の待ち行列に対して、 $\lambda$ ,  $\mu$ ,  $t$ ,  $\rho$ ,  $q$  を以下のように定義し、各々の関係を図 2, 図 3 に示す。

- $\lambda$ : 単位時間当たりに窓口に着するエンティティの平均個数 (到着率: arrival rate)
- $\mu$ : 単位時間当たりに 1 つの窓口で処理できるエンティティの平均個数 (サービス率: servicing rate)
- $t$ : 単位時間当たりに 1 つの窓口から出て来るエンティティの平均個数 (スループット: throughput)
- $\rho$ : 1 つの窓口が実際にエンティティを処理している時間の割合 (稼働率: utilization rate)

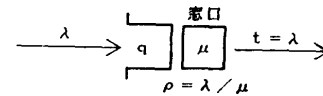


図 2  $\lambda < \mu$  の場合  
Fig. 2 Case of  $\lambda < \mu$ .

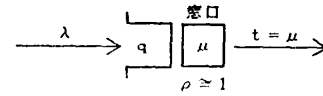


図 3  $\lambda \geq \mu$  の場合  
Fig. 3 Case of  $\lambda \geq \mu$ .

$q$ : 窓口の前で処理を待つエンティティの平均個数 (待ち行列長: queue length)

待ち行列ネットワークは複数の窓口が結合したものである。2つの窓口が直列に結合しているときには、前段の窓口からのスループット  $t$  が、後段の窓口への到着率  $\lambda$  となる。2つの窓口からのスループット  $t_1$  と  $t_2$  が合流して後段の窓口に入るときは、 $t_1$  と  $t_2$  の和が後段の窓口への到着率となる。ある窓口  $s_1$  のスループット  $t_1$  が 2 つの窓口  $s_2, s_3$  に分岐する場合、その分岐確率  $r, 1-r$  を定義する。 $s_2$  と  $s_3$  への到着率は  $rt_1, (1-r)t_1$  となる。

本稿の待ち行列ネットワークでは、その形状は固定パラメータ (診断後に変更不可能なパラメータ) とし、 $\lambda, \mu, t, \rho, q, r$  は可変パラメータ (診断後に変更可能なパラメータ) とする。

本稿の待ち行列ネットワークは、処理されるエンティティはすべて外部から到着し、それらのエンティティは処理済みになると外部へ出るオープン型の待ち行列ネットワークとする。また、エンティティは外部から待ち行列ネットワークの複数の窓口に着することが可能であるが、待ち行列ネットワーク内では、それらは区別しては扱わないため、種類に応じて途中の分岐確率が異なることがない。すなわち、待ち行列ネットワーク内部では、エンティティは 1 種類と考える。

さらに、現在の BDES で診断可能な待ち行列ネットワークの形状について主に 3 つの制約事項がある。それは、ネットワーク中の各窓口の入出力は各々 2 つ以下であること、ループ部の入口と出口は各々 1 つであること、および、各窓口の待ち行列長に容量がないことである。

本稿で対象とするシステムは、上記の性質をもった待ち行列ネットワークの形態にモデル化できるシステムであり、 $\rho$  や  $q$  が過大ないくつかの窓口をもつ、非

定常（過負荷）状態である、あるいはその可能性をもつシステムである。

### 3. ボトルネックの定義と初期診断

待ち行列ネットワークの中で稼働率が限りなく1に近づいている窓口（bottleneck）がある。そのボトルネック窓口の待ち行列長は無限に成長する恐れがある。このようなボトルネック窓口が存在する場合、待ち行列ネットワークは非定常（過負荷）状態になっている。

稼働率が過大な窓口あるいは待ち行列長が長い窓口はボトルネックとなる可能性がある。

窓口への到着率や窓口のサービス率が大きくなればつきをもつ場合、稼働率が1に近くなっても待ち行列長がときには急激に増加することが考えられる。これを考慮して、たとえば、稼働率が0.7を越える窓口はボトルネックの可能性があるとみなす。

また、稼働率が過大でなくても、待ち行列長が過大である窓口には、測定終了以降にさらにエンティティが過大に到着して、その稼働率が上昇する危険性がある。場合によっては、この過大な待ち行列長は徐々に解消されるかもしれないが、ボトルネックの発生を事前に防止するために、平均待ち行列長が1を越える窓口はボトルネックの可能性があると診断する。

以上のような基準でボトルネックの可能性があると診断された窓口が存在する場合、待ち行列ネットワークは、非定常（過負荷）状態となる危険性がある。

ボトルネック窓口が存在したり、その可能性がある場合、それらの窓口の能力（サービス率）の増加やその窓口への到着量の減少等の改善プランが必要となる。

また、待ち行列ネットワークの中で、稼働率が他の窓口と比べて非常に低い窓口は、そのネットワーク内でサービスの度合いが相対的に小さいことを示す。これは別の窓口に悪影響を与えてその別の窓口はボトルネックの可能性を与える。このような窓口も、たとえば稼働率が0.3より低いということを目安に初期診断する。

以上の目安をさらに詳しくして、ボトルネック箇所を初期診断するためのヒューリスティックな知識を以下に挙げる。

知識1：ボトルネック箇所の初期診断のための知識

- 1) 稼働率が一番高い窓口は周囲の別の窓口に対して悪影響を与える可能性がある。

- 2) 稼働率が一番低い窓口は周囲の別の窓口に対して悪影響を与える可能性がある。
- 3) 稼働率が0.9以上となる窓口はボトルネックがある。
- 4) 稼働率が0.7以上となる窓口はボトルネックの可能性がある。
- 5) 稼働率が0.7より小さいが待ち行列長が1以上の窓口はボトルネックの可能性がある。
- 6) 稼働率が0.3以下となる窓口は周囲の別の窓口に対して悪影響を与える可能性がある。

### 4. ボトルネックの要因診断のための知識

図4に示す通り、ある窓口はボトルネックの可能性があると診断された場合、 $t_1$ が過大か $\mu_2$ が過小かというバランスがその要因である。 $t_1$ を減少するために、上流からのスループットを減少する必要がある。 $\mu_2$ を増加するとスループット $t_2$ が増加し、下流に影響を与える。図5に示す通り1つ上流の窓口の $\mu_1, t_1, \rho_1, q_1$ を調査したり、下流に進んで調査する必要がある。この診断に有効な知識を見出し、ボトルネックに影響を及ぼしている様々な要因を明らかにする。

診断の知識は、 $\rho$ と $q$ の値によって、ボトルネックの可能性があると診断された場合（4.1節）と、周囲の別の窓口はボトルネックの可能性があると診断された場合（4.2節）とに分類される。その中で待ち行列の形状ごとにさらに分類される。

#### 4.1 ある窓口はボトルネックの可能性があると診断された場合

知識1により稼働率や待ち行列長の平均値からボトルネックの可能性があると診断された窓口を中心に、知識2～9は、窓口の接続の形状やパラメータの値の関係を手がかりとして、その上流や下流の窓口を調査する。これを順次、上流にさかのぼったり下流に進んで適用することにより、ボトルネックの要因を待ち行

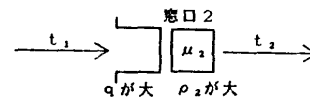


図4 ボトルネック（の可能性）がある場合  
Fig.4 Server which is (or may be) bottleneck.

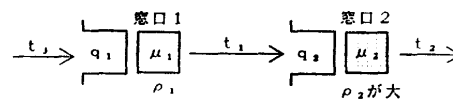


図5 直列型の場合  
Fig.5 Case where two servers are tandem.

列ネットワーク全体の中から発見する。

ある窓口でボトルネックの可能性があると診断された場合には、次の知識2でその直接的な要因と下流診断による要因を調べる。

知識2：ある窓口でボトルネックの可能性のある場合の直接的な要因と下流診断による改善

1) 直接的な要因はその到着量  $\lambda$  とサービス率  $\mu$  のバランス ( $\lambda$  が過大あるいは  $\mu$  が過小) と診断。

2)  $\mu$  の改善について (下流診断の知識) :  $\mu$  を増加してその窓口のボトルネックを改善する場合、その窓口のスループットが増加するため、下流の窓口群の稼働率、待ち行列長に影響を及ぼす。よって、 $\mu$  を改善するための知識を以下のように分類する。

2-1) その窓口が最終窓口ならば、 $\mu$  を増加。

2-2) その窓口の下流の窓口群のすべての稼働率が過大でなく、かつ、平均待ち行列長が過大でなければ、 $\mu$  を増加。

2-3) その窓口の下流の窓口群の中に稼働率が過大なものがある場合、または、平均待ち行列長が過大なものがある場合、下流にあるそのような窓口のサービス率をすべて増加することで、 $\mu$  を増加。

上の知識2の適用後、その窓口とその直上流の窓口との接続関係により次の上流診断の知識3~9を適用する。

以下の各知識において、各窓口の  $q$  が過大なものは、その  $\rho$  の値にかかわらず  $\rho$  が過大であるとみなしている。

知識3：直列型の場合

図5に示す直列型で窓口2にボトルネックの可能性があると診断され知識2が適用された後、以下に示す知識が適用される。

1)  $\rho_2$  減少のためには  $t_1$  を減少。  $t_1$  減少のためには  $t_j$  の過大さの改善が必要。

→  $t_j$  に対して知識3~9を適用。

2)  $t_j = t_{1n}$  (システムへの入力) の場合は知識14を適用。

根拠：

1-1)  $\rho_1 \cong \rho_2$  の場合、

$\mu_1$  が増加すると  $t_1$  がさらに増加。  $\mu_1$  が減少すると窓口1の待ち行列がさらに増加。

∴  $\rho_2$  がボトルネックとなる要因は、 $t_j$  が過大であると診断。

1-2)  $\rho_1 \ll \rho_2$  の場合、

$t_j = t_1$  である。  $t_j$  が増加すると  $t_1$  も増加し、窓口2

の待ち行列がさらに増加。

$t_j$  が減少すると  $t_1$  も減少し、 $\rho_2$  が減少。

∴  $\rho_2$  がボトルネックとなる要因は、 $t_j$  が過大であると診断。

知識4：合流型の場合

図6に示す合流型で窓口3にボトルネックの可能性があると診断され知識2の適用後、以下に示す知識が適用される。

1)  $t_1 \gg t_2$  の場合、 $t_1$  の方に大きな原因があると診断。

→  $t_1, \rho_3$  に対して知識3を適用。

2)  $t_1 \cong t_2$  の場合、 $t_1$  と  $t_2$  が同程度に原因となる。

→  $t_1, \rho_3$  に対して、および  $t_2, \rho_3$  に対して知識3を適用。

註：  $t_1, \rho_3$  と  $t_2, \rho_3$  のどちらに対して遡行するかはユーザが指定する。バックトラック診断機能が装備されているため、指定しなかった方に対して、後に遡行可能となる。バックトラック診断については5章で述べる。

知識5：分岐型の場合

図7に示す分岐型で、窓口2にボトルネックの可能性があると診断され知識2を適用後、以下に示す知識が適用される。

1)  $\rho_3 \cong \rho_2$  の場合

1-1) 知識2により窓口3の下流の窓口群を調査し、 $\mu_3$  が改善不可能と判断された場合には、 $t_1$  が過大

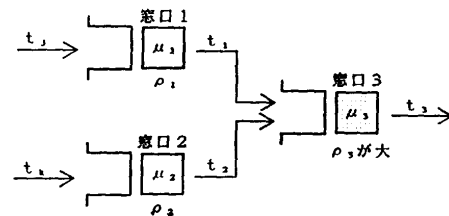


図6 合流型の場合

Fig. 6 Case where two servers join at a junction server with high utilization.

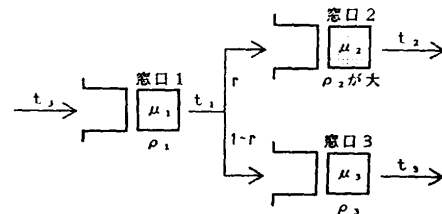


図7 分岐型の場合

Fig. 7 Case where a server branches off to two servers one of which is with high utilization.

と診断.

→ $t_1, \rho_2$  に対して知識 3 を適用.

1-2) 知識 2 により窓口 3 の下流の窓口群を調査し,  $\mu_3$  が改善可能と判断された場合には, 窓口 3 の下流のいくつかの窓口のサービス率を増加,  $\mu_3$  を増加,  $r$  を減少,  $1-r$  を増加.

2)  $\rho_3 \ll \rho_2$  の場合

知識 2 により窓口 3 の下流の窓口群を調査し,  $t_3$  が増加可能と判断された場合には, 窓口 3 の下流のいくつかの窓口のサービス率を増加,  $r$  を減少,  $1-r$  を増加.  $t_3$  が増加不可能な場合には  $t_1$  が過大と診断.

→ $t_1, \rho_2$  に対して知識 3 を適用.

知識 6 : ループ部からの出力量を改善するためにその分岐確率を変更することは実際上の意味がない.

根拠 : 図 8 に示す通り, 稼働率が過大な窓口 3 に対する窓口 2 からの到着量が確率分岐であり, 分岐の他方がループするためのものである場合, 窓口 3 への到着量を改善するために  $1-r$  を減少させる診断は, フィードバック量の変更となり, 対象システムの機能を変える.

知識 7 : 図 8 に示すループ形状で, 窓口 3 にボトルネックの可能性があると診断された場合, 知識 2, 知識 6 適用後, ループ部からの出力量  $t_{23}$  を改善するために, ループ部への到着量  $t_1$  と, 窓口 3 の稼働率  $\rho_3$  を使って遡行可能.

根拠 : 図 8 において, ループ内の窓口群の稼働率があまり過大でないときは, ループ部からの出力量はループ部への到着量に等しい. このため, ループ部からの出力量を改善する必要があるときは, その到着量を改善する.

知識 8 : 合流窓口への到着量が上流の窓口の分岐出力である場合

たとえば図 9 で窓口 3 にボトルネックの可能性があると診断されさらに知識 2 の適用後, 以下の場合に分けて知識が適用される. 知識の詳細は付録に示す.

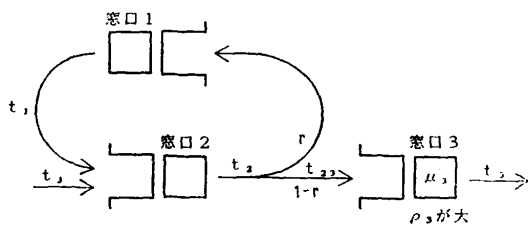


図 8 窓口でループ部分からの入力がある場合  
Fig. 8 Case where an input of a server is an output of a loop.

- 1) 到着量がループ部からの確率分岐出力である場合 (図 10 参照)
- 2) 到着量的一方が外部入力である場合 (図 11 参照)
- 3) 診断窓口がループ部に含まれる場合 (図 12 参照)
- 4) 上流の窓口の別の分岐出力が, ループ部への到着量となっている場合 (図 13 参照)

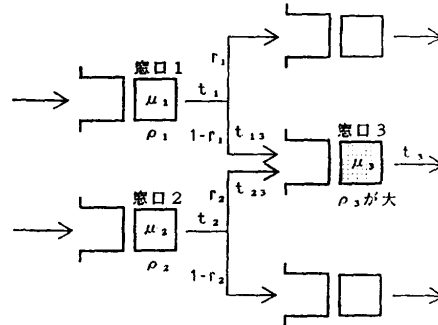


図 9 合流窓口への入力が上流の窓口の分岐出力の場合  
Fig. 9 Case where an input of a junction server is a branch of upstream server.

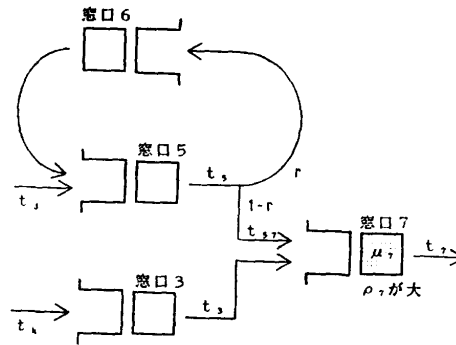


図 10 合流窓口の一方の入力がループ部分からの出力である場合  
Fig. 10 Case where one input of a junction server is the output of a loop.

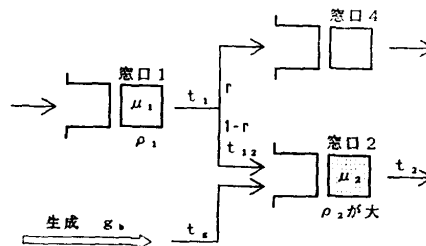


図 11 合流窓口への一方の入力が外部入力である場合  
Fig. 11 Case where one output of a junction server is an external input.

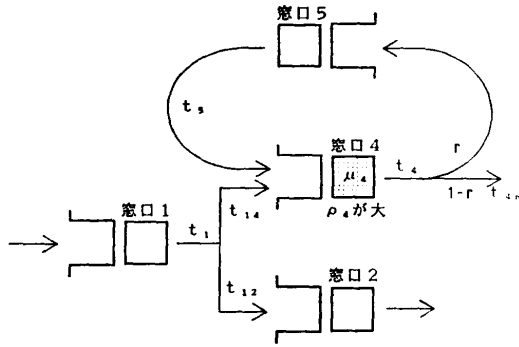


図 12 合流窓口がループ部分に含まれる場合  
Fig. 12 Case where a junction server is the part of a loop.

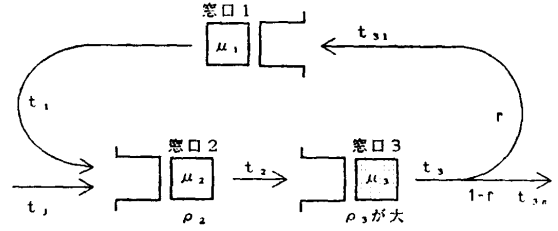


図 14 ボトルネック窓口がループ内にある場合  
Fig. 14 Case where a bottleneck server is the part of a loop.

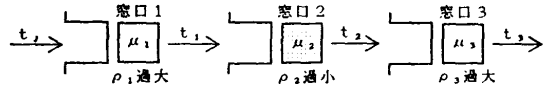


図 15 直列な窓口群の中に稼働率が過小な窓口が存在する場合

Fig. 15 Case where a server with low utilization is the part of the tandem servers.

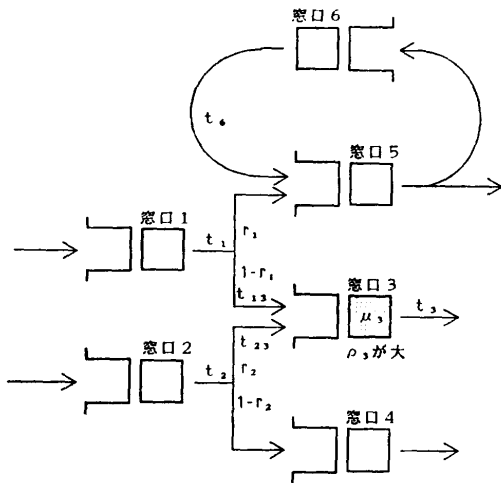


図 13 合流窓口 3 への入力に上流窓口 1 の分岐出力があり、窓口 1 の別の分岐出力がループ部の入力である場合  
Fig. 13 Case where the input to a server-3 is an output of another server-1 and another output of server-1 is an input to a loop.

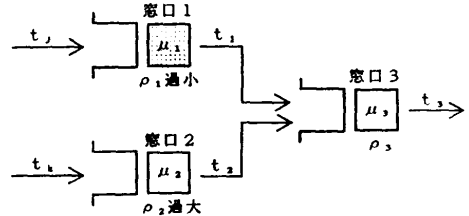


図 16 合流型の場合

Fig. 16 Case where two servers, one of which is with low utilization, join.

5) その他の場合 (図 9 参照)

知識 9 : ループ型

ボトルネックの可能性があると診断された窓口がループ内部にある場合、その窓口の稼働率とループ部への到着量を使って遡行可能 (図 14 参照)。

4.2 稼働率が過小で周囲に悪影響の可能性があると診断された場合

知識 10 : 稼働率が過小な窓口は、周囲のボトルネック窓口が悪影響を与えている可能性がある。

知識 11 : 窓口群が直列に接続している場合、稼働率が過小な窓口は前後のボトルネック窓口が悪影響を与えていない。

根拠 : 図 15 において  $\rho_2$  が過小でも  $\rho_1$  や  $\rho_3$  の過

大には何の影響も与えない。なぜなら、 $\mu_2$  を下げて  $\rho_2$  を上げたとしても、 $t_1$  と  $t_2$  はほぼ等しいままであり、 $t_2$  は減少しないため  $\rho_3$  に変化はない。 $t_1$  を上げて  $\rho_2$  を上げるためには、 $\mu_1$  を上げなければならないが、 $t_1$  が増加すると  $t_2$  がさらに増加し、 $\rho_2$  がますます過大となる。

知識 12 : 窓口群が合流型の形状で、一方の稼働率が過大で他方が過小である場合、過小な方は過大な方に悪影響を与えない。

根拠 : これは図 16 が示す通り自明である。

知識 13 : 窓口群が分岐型の形状をしていて、一方の稼働率が過大で他方が過小である場合、過小な窓口は過大な窓口が悪影響を与えている。

根拠 : 図 17 に示す通り、 $\rho_2$  の過小さが  $\rho_3$  の過大さの要因になっている。これは分岐確率  $r$  の変更を要求する。 $\rho_2$  の過小さから  $\rho_3$  の過大さを発見する。

4.3 外部入力に関する知識

知識 14 : ある窓口への到着量が過大であり、それが外部入力である場合、その外部入力が高過大であると

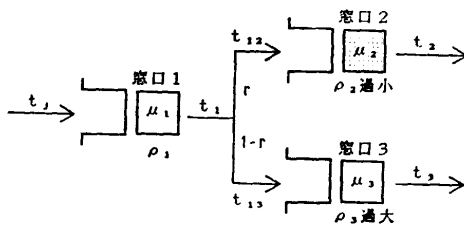


図 17 分岐型の場合

Fig. 17 Case where a server branches off to two servers one of which is with low utilization.

診断し遡行を終了して、バックトラック診断を開始する。

### 5. ボトルネック診断エキスパートシステム (BDES)

BDES は、上記の知識を Prolog 言語の事実や規則で記述し、J-3100 GT 上でインプリメントしたものである。ユーザはこれを用いて、対象システムのボトルネックの要因診断と改善プラン獲得を対話的に行う。

#### 5.1 BDES の推論制御および機能

BDES では、図 18 と図 19 に示す推論制御によって 4 章の知識を適用しボトルネック診断を行う。これらの図の中で、判断と書かれた枠と入力と書かれた枠で、ユーザは表示画面に従ってキーボード入力する。

図 18 で、まず最初に知識 1 を適用し、1 つ以上のボトルネック（の可能性がある）窓口が検出される。ユーザがその中から診断したい窓口を 1 つ選ぶと、BDES によるボトルネック診断が開始される。上流診断実行の判断でユーザが N と入力した場合、あるいは、上流への遡行を進め診断が外部入力まで至った場合（知識 14）には、制御は図 19 に示す推論制御 2 へと進む。

#### (1) バックトラック診断機能

知識 2 でボトルネック窓口のサービス率の増加をユーザが選択したとき、その窓口への到着率の減少についての知識 3-9 の適用が保留されている。また、上流に 2 つの窓口があり、それらの 2 つの窓口からのスループットが同程度であると診断された場合（知識 4）、この 2 つから 1 つの窓口をユーザが選択して上流を調査するが、未選択の窓口からの診断も保留されている。

バックトラック診断の指示を行うと、未選択の窓口や未選択のパラメータについて診断できる。これにより、あるボトルネック窓口のすべての改善プランを獲

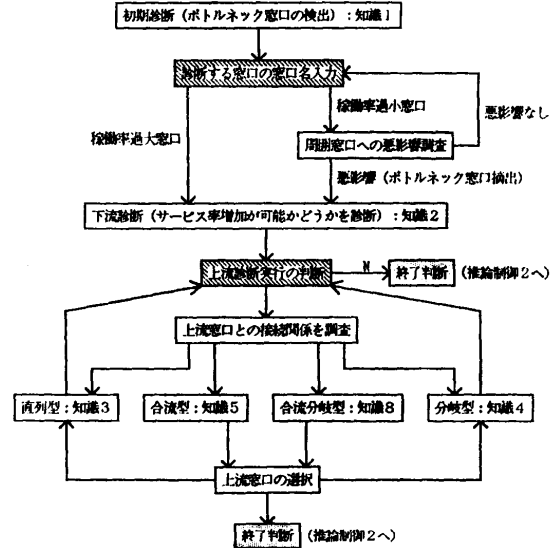


図 18 推論制御 1  
Fig. 18 Inference control 1.

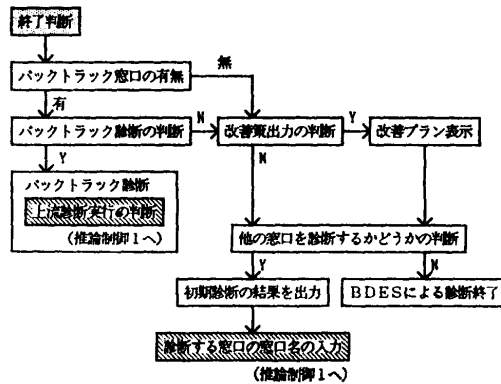


図 19 推論制御 2  
Fig. 19 Inference control 2.

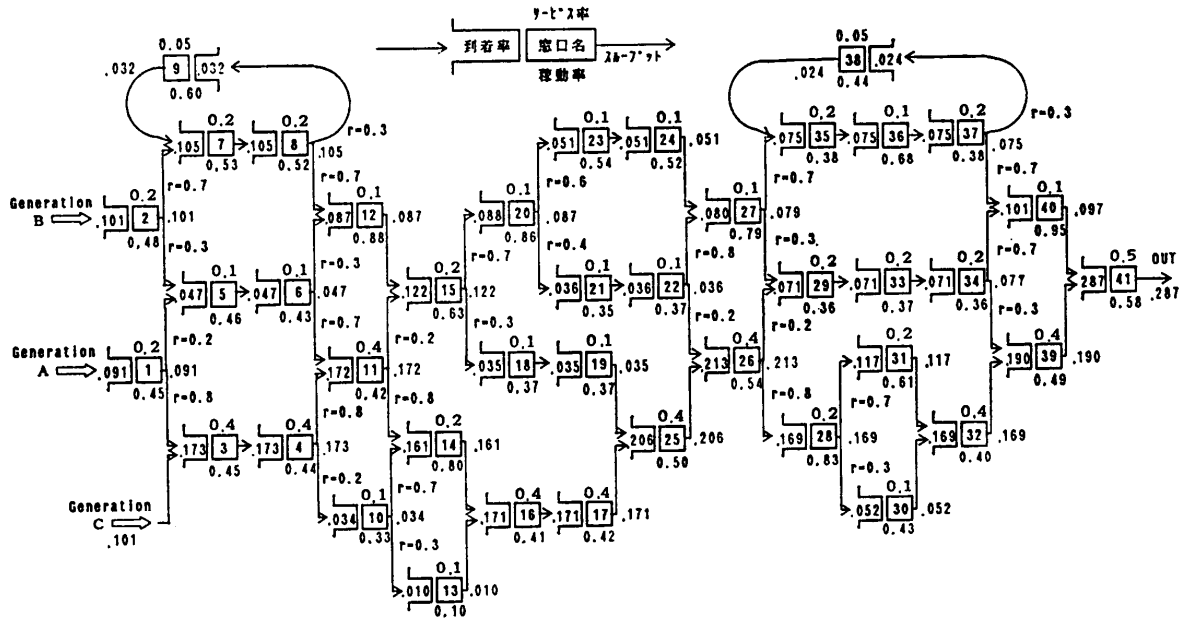
得できる。

#### (2) 改善プラン表示機能

あるボトルネック窓口の対話的な診断を終了した後、途中の診断結果を省いて、要因診断により得られた改善すべきパラメータ群をいくつかのプランに分類して出力する。ユーザは、ボトルネックを改善するために改善プランを単独で用いてもよいし、いくつかのプランを組み合わせて用いてもよい。

### 6. BDES による実行例

実行例の待ち行列ネットワークを図 20 に示す。図 20 には、既存のシミュレーション・パッケージによって得られた性能パラメータの測定平均値が書かれている。測定値の信頼性を高めるために、シミュレーショ



given: サービス率と s1, s2, s3 への到着率  
 その他はシミュレーション実行後に得られたデータ

図 20 待ち行列ネットワークのシミュレーションモデルとシミュレーション実行後の評価データ  
 Fig. 20 Queueing network simulation model and its evaluation data.

ン開始後の立ち上がり時の統計量は棄却している。  
 s40 をみるとその稼働率は極めて高く、この待ち行列ネットワーク自体は、非定常(過負荷)状態である。  
 このモデルの形状と性能パラメータを BDES に与え、対話的に診断を実行した例を図 21 に示す。  
 図 21 末尾にボトルネック窓口 s20 の改善のために 7 個のプランが出力されている。プラン 1 は、s20 の改善のために、s20 自体のサービス率のほかに、s27, s28, s36, s40 のサービス率も増加させるというプランである。BDES によるボトルネック診断はこのように局所的ではなく広域的である。他のプランも広域的であり、分岐確率の改善や外部入力 of 改善などが含まれている。

プラン 1 の診断に従って s20, s27, s28, s36, s40 の各々のサービス率を 1.3 倍に増加させてシミュレーションを再試行した結果、s20 の稼働率は図 20 にある 0.86 から 0.66 に減少した。また s41 からのスループット (s41 は待ち行列ネットワークの出口にあたるためこれはシステムのスループットになる) 0.287 から 0.291 に増加した。これは BDES の診断に従ってボトルネックが解消し、システム内の流れがよくなったことを示す。

以上の例により、ボトルネック窓口を改善するために修正すべき性能パラメータ群を、BDES により特定

できることがわかる。

### 7. 待ち行列理論との関係

本稿で対象としたオープン型の待ち行列ネットワークでしかもエンティティが 1 種類とした場合、定常状態では次のトラヒック方程式が成り立つ (たとえば文献 8))。

$$\lambda_j = \lambda_o + \sum_{i=1}^N \lambda_i P_{ij}, \quad j=1, 2, \dots, N$$

ここで  $\lambda_j$ : 窓口  $j$  への到着率

$\lambda_o$ : 窓口  $j$  への外部からの到着率

$P_{ij}$ : 窓口  $i$  から窓口  $j$  への分岐確率

この式は、定常状態であるため、窓口に到着したエンティティはよどみなくその窓口を出ることを示す。

$\lambda_o$  と  $P_{ij}$ , およびサービス率  $\mu$  がすべて既知であり、待ち行列ネットワークが定常状態であれば、上の線形式の連立で、各窓口への到着率、スループット、稼働率 ( $=\lambda/\mu$ ) が算出できる。

しかし、ボトルネックが存在するときには、待ち行列ネットワークは定常状態ではなく、この連立式は成り立たない。すなわち、この連立式はボトルネックの改善に使うことができない。

性能パラメータの中からボトルネックを検出しその原因や改善プランを捜す作業は、結局はエキスパート



待ち行列ネットワーク "netq 4" のボトルネックを診断します。

知識 1

稼働率が最大となる窓口の窓口名と稼働率と待ち行列長を出力します。

s 40      0.95      15.28

稼働率が最小となる窓口の窓口名と稼働率と待ち行列長を出力します。

s 13      0.10      0.01

稼働率が 0.9 以上の窓口の窓口名と稼働率と待ち行列長を出力します。

s 40      0.93      15.28

稼働率が 0.7 以上の窓口の窓口名と稼働率と待ち行列長を出力します。

s 12      0.88      5.78

s 14      0.80      3.11

s 20      0.86      5.78

s 27      0.79      2.6

s 28      0.83      3.74

稼働率が 0.7 以下で待ち行列長が 1 以上の窓口の窓口名と稼働率と待ち行列長を出力します。

s 9      0.6      1.51

s 15      0.63      1.02

s 36      0.68      1.52

稼働率が 0.3 以下の窓口の窓口名と稼働率と待ち行列長を出力します。

s 13      0.10      0.01

診断する窓口名を入力して下さい: s 20.

[下流との接続]

知識 2

s 20 の直下流には 2 個の窓口があり、さらに下流には 18 個の窓口があります。

[s 20 のサービス率増加について]

直下流窓口 {s 21 s 23} の稼働率が 0.7 未満なので、s 20 のサービス率を増加できる可能性があります。さらに下流の窓口の中に稼働率が 0.7 以上のものが 3 つ {s 27 s 28 s 40} あり、稼働率 0.7 未満で平均待ち行列長が 1 以上の窓口が 1 つ {s 36} あります。よって、窓口 s 20 のサービス率を上げる場合、{s 27 s 28 s 40 s 36} のサービス率も増加する必要があります。

窓口 s 20 への到着量が過大として上流を調べますか? (Y/N): y.

[接続状況]

知識 8, 知識 2

s 20 の直上流には窓口 s 15 があり、s 15 から s 20 への接続は、分岐型です。

s 15 からは s 20 以外に s 18 へ分岐出力しています。

[s 18 の状況]

s 20 に比べ、s 18 の稼働率はかなり小さいと診断できます。

s 18 の直下流には 1 個の窓口 s 19 があり、さらに下流には 12 個の窓口があります。

直下流の窓口の稼働率は 0.5 以下なので s 15 から s 18 への分岐確率を増加できる可能性があります。さらに下流の窓口の中に稼働率が 0.7 以上の窓口が 2 つ {s 28 s 40} あります。窓口 s 15 からの分岐確率を上げる場合、{s 28 s 40} のサービス率も増加する必要があります。

[分岐確率の改善について]

s 20, s 18 の関係より s 20 の稼働率を減少するために、{s 28 s 40} のサービス率を増加し、s 15 から s 20 への分岐確率を減らし、s 15 から s 18 への分岐確率を増加することができます。

分岐確率を変えられない場合は、s 15 のスループットを減少する必要があります。

s 15 を診断窓口として上流の窓口を調べますか? (Y/N): y.

..... <省略> .....

s 12 を診断窓口として上流の窓口を調べますか? (Y/N): y.

[上流との接続]

知識 8

s 12 の直上流には s 6 と s 8 の窓口があり、s 12 への接続は、各々分岐型となっています。

[到着量の減少について]

s 6 のスループットを改善した方がよいと診断できます。

<根拠>

s 8 のスループットの方が s 6 のスループットより大きいと診断できますが、s 8 の出力は、ループ窓口群からの確率分岐出力となっています。よって、この分岐確率の変更には実際上の意味はなく、s 8 のスループット改善のためにはループ窓口群への到着量を調査しなければなりません。

s 12 と s 6 の関係を調べますか? (Y/N): y.

[接続状況]

知識 5, 知識 2

s 12 の直上流には窓口 s 6 があり、2 つの窓口の接続は、分岐型です。

s 6 からは s 12 以外に s 11 へ分岐出力しています。

図 21 (その 1)

## [s 11 の状況]

s 12 に比べ、s 11 の稼働率はかなり小さいと診断できます。

s 11 の直下流には 2 個の窓口があり、さらに下流には 26 個の窓口があります。

直下流窓口の s 14 の稼働率は 0.7 以上です。また、s 15 の稼働率は 0.7 未満ですが、平均待ち行列長が 1 を越えています。さらに下流の窓口の中に稼働率が 0.7 以上のものが 4 つ {s 20 s 27 s 28 s 40} あり、稼働率 0.7 未満で平均待ち行列長が 1 以上の窓口が 1 つ {s 36} あります。また、下流の稼働率 0.7 以上の窓口で、サービス率を増加できないもの {s 20} があります。

## [分岐確率の改善について]

s 12 の稼働率を減少するために、s 6 から s 12 への分岐確率を減少し、s 6 から s 11 への分岐確率を増加することはできません。よって、s 6 のスループットが過大と診断できます。

s 6 を診断窓口として上流の窓口を調べますか？ (Y/N): y.

## [上流との接続]

知識 3

s 6 の直上流窓口は s 5 で、s 5 から他の窓口への分岐はありません。

s 5 と s 6 の接続は直列型となっています。

## [s 6 への到着量減少について]

s 6 の稼働率と s 5 の稼働率はほぼ等しいので s 6 への到着量を減少するためには s 5 への到着量の過大さを改善する必要があります。

## 〈根拠〉

s 6 への到着量は s 5 のスループットであり、s 5 から s 6 への接続が直列型なので、この s 5 のスループットを減少するためには、s 5 への到着量を減少する方法しかありません。

s 5 を診断窓口として上流の窓口を調べますか？ (Y/N): y.

## [上流との接続]

知識 8

s 5 の直上流には s 1 と s 2 の窓口があり、s 5 への接続は各々分岐型となっています。

## [到着量の減少について]

s 1 のスループットを改善した方がよいと診断できます。

## 〈根拠〉

s 1 と s 2 のスループットはほぼ等しいのですが、s 2 の分岐先がループ窓口群の入口となっているので、より局所的に到着量を減少するためには、s 1 のスループットを改善した方がよいと思われます。

s 1 と s 2 のどちらかを診断窓口として上流の窓口を調べますか？ (Y/N): y.

s 1 と s 2 のどちらの窓口を診断窓口として上流の窓口を調べますか？ : s 1.

.....〈省略〉.....

窓口 s 15 の上流窓口を s 11 とし、バックトラック診断を実行しますか？ (Y/N): y.

## [接続状況]

知識 8, 知識 2

s 15 の直上流には窓口 s 11 があり、s 11 から s 15 への接続は、分岐型です。

s 11 からは s 15 以外に s 14 へ分岐出力しています。

## [s 14 の状況]

s 14 の稼働率は、s 15 の稼働率とほぼ等しいと診断できます。

s 14 の直下流には 1 個の窓口があり、さらに下流には 13 個の窓口があります。

直下流の窓口 s 16 の稼働率は 0.5 以下なので s 11 から s 14 への分岐確率を増加できる可能性があります。さらに下流の窓口の中に稼働率が 0.7 以上の窓口が 2 つ {s 28 s 40} あります。

窓口 s 11 からの分岐確率を上げる場合、{s 28 s 40} のサービス率も増加する必要があります。

## [分岐確率の改善について]

{s 14 s 28 s 40} のサービス率を増加し、s 11 から s 15 への分岐確率を減らし、s 11 から s 14 への分岐確率を増加することで、s 20 の稼働率を減少することができます。分岐確率が変わらないなら、s 11 の上流を調べる必要があります。

s 11 を診断窓口として上流の窓口を調べますか？ (Y/N): y.

.....〈省略〉.....

s 2 を診断窓口として上流の窓口を調べますか？ (Y/N): y.

## [s 2 への到着量減少について]

知識 3

s 2 への到着量はシステムの外からのもので、上流に窓口はありません。よって、s 2 への到着量を減少するためには、この外部入力 gb の過大さを改善する必要があります。

診断窓口 s 20 のボトルネックを改善するための改善策を出力しますか？ (Y/N): y.

BDES によるボトルネック診断結果を表示します。

診断窓口: s 20 (ボトルネック窓口)

ボトルネック窓口 s 20 の改善のために変更すべきパラメータ

\*プラン 1 s 27 のサービス率増加

s 28 のサービス率増加

s 40 のサービス率増加

図 21 (その 2)

s 36 のサービス率増加  
 s 20 のサービス率増加  
 \*プラン2 s 28 のサービス率増加  
 s 40 のサービス率増加  
 s 20 への分岐確率減少  
 s 18 への分岐確率増加  
 \*プラン3 s 14 のサービス率増加  
 s 28 のサービス率増加  
 s 40 のサービス率増加  
 s 15 への分岐確率減少  
 s 14 への分岐確率増加  
 \*プラン4 s 14 のサービス率増加  
 s 28 のサービス率増加  
 s 40 のサービス率増加  
 s 11 への分岐確率減少  
 s 10 への分岐確率増加  
 \*プラン5 外部入力 gc-- 減少  
 \*プラン6 外部入力 ga-- 減少  
 \*プラン7 外部入力 gb-- 減少  
 他の窓口について調べますか? (Y/N): n.  
 BDES によるボトルネック診断を終了します。

図 21 図 20 の診断過程

Fig. 21 Diagnosis process for Fig. 20.

が試行錯誤に行わざるを得ない。エキスパートでないユーザがこれを改善するのは、簡単な作業ではない。

本稿の BDES は、このボトルネックの改善プランを発見する作業に有効である。例題で示したように 1 つのボトルネックに対して、選択できる複数の改善プランを提示して、パラメータの性質を考慮しながら、ユーザは改善プランを選択できる。サービス率、分岐確率、外部からの到着率を修正した後に、もし、定常状態になることが確認できれば、再度、性能パラメータの測定 (図 1) を行わずに、上の連立式を使ってもよい。

## 8. おわりに

本稿では、知識工学的手法を用いたオープン型の待ち行列ネットワークのボトルネック診断方法を考察し、このための知識、BDES、およびその実行例を示した。今後の課題を以下に示す。

(1) 性能パラメータの自動的な算出: BDES は、改善すべきパラメータ増減という形式で定性的な診断を行う。各パラメータの値の増減量を定量的に出力し、7章で示した式を使って、自動的にすべての性能パラメータの値を算出できれば、図 1 に示す測定を行う回数が減り、さらに改善作業の効率化が図れる。

(2) 対象システムごとの知識の導入: 本手法をより実際のエキスパートシステムとするためには、待ち行列ネットワークにモデル化できる対象システム

(たとえば、計算機ネットワーク、生産ライン) ごとのより専門的な詳細化した知識を導入する必要がある。

(3) ボトルネック基準の再検討: ボトルネック箇所の初期診断の知識として稼働率の目安を一律に 0.7 以上とした。対象システムによってこの目安が異なったり、1つのシステムの構成要素の種類によって異なる場合がある。たとえば、計算機ネットワークでは CPU は 0.7 以上とするがチャンネルは 0.2 以上を目安とする場合がある。このようにボトルネック基準をさらに検討しなければならない。

(4) 固定パラメータの導入: ユーザがあるパラメータを固定パラメータ (その対象システムでは変更が許されていないパラメータ) と指定すれば、BDES の出力する改善プランにはそのようなパラメータの増減の指示を含めないなどの検討も必要である。

(5) 改善プランの自動照合: 複数のボトルネックの改善プランの照合で、同じ分岐確率について、一方の改善プランでは増加、他方では減少という場合がある。サービス率については増加という改善プランだけであるため、照合しても矛盾しない。照合は現段階ではユーザに任されているが、この自動化を図りたい。

**謝辞** 待ち行列ネットワークの解析モデルや計算法の適用可能性について貴重な御助言を頂いた上智大学理工学部鈴木誠道教授に感謝します。

本研究は一部文部省科学研究費 (63780047) による。

## 参考文献

- 1) 新井ほか：診断型エキスパートシステム，情報処理，Vol. 28, No. 2, pp. 177-186 (1987).
- 2) 麻生川ほか：ACOS-4 性能改善エキスパートシステムの開発，情報処理学会 OS 研，No. 35 (1987).
- 3) Feigenbaum, E. A. et al. (eds.): *The Handbook of Artificial Intelligence*, William Kaufmann, Inc. (1981).
- 4) 本位田ほか：MENDEL による計算機 OS 性能診断エキスパートシステム，電気学会システム制御研，No. 86-15 (1986).
- 5) Kleinrock, L.: *Queueing Systems*, John Wiley & Sons, Inc. (1975).
- 6) 村瀬ほか：待ち行列ネットワークモデルによる計算機システムの性能評価エキスパートシステムの設計，情報処理学会 OS 研，No. 30-6 (1986).
- 7) 田中幸吉編：知識工学，朝倉書店 (1984).
- 8) Gelenbe, E. et al.: *Analysis and Synthesis of Computer Systems*, Academic Press (1989).

## 付 録

知識 8：合流窓口への到着量が上流の窓口の分岐出力である場合の詳細

- 1) 到着量がループ部からの確率分岐出力である場合

図 10 に示す通り，窓口 5 がループ部からの出力になっている場合，次の診断を行う。

- 1-1)  $t_{57} \gg t_3$  の場合， $t_{57}$  の方に原因。  
→  $t_{57}$ ,  $\rho_7$  に対して知識 7 を適用。
- 1-2)  $t_3 > t_{57}$  の場合， $t_3$  の方に原因。  
→  $t_3$ ,  $\rho_7$  に対して知識 3 を適用。
- 1-3)  $t_{57} > t_3$ ，あるいは  $t_{57} = t_3$  の場合，  
→  $t_3$ ,  $\rho_7$  に対して知識 3 を適用。または  
→  $t_{57}$ ,  $\rho_7$  に対して知識 7 を適用。

註：図 10 において，窓口 3 が窓口 7 以外の他の窓口へ分岐出力している場合，1-2)，1-3) は， $t_3$ ,  $\rho_7$  に対して知識 5 を適用して遡行可能となる。

- 2) 到着量的一方が外部入力である場合

図 11 に示す通り，窓口 2 への 1 つの到着量  $t_0$  が外部入力である場合，

→  $t_0$  と  $t_{12}$  の大小に関わらず，外部入力  $g_0$  を改善。あるいは

→  $t_0$  は不変として， $t_{12}$ ,  $\rho_2$  に対して知識 5 を適用。

根拠：図 11 において，より局所的に窓口 2 のボトルネックを改善する場合，外部入力  $g_0$  を改善した方がよい。しかし，外部入力を変更できない場合は，

$t_0 \gg t_{12}$  の場合でも， $t_{12}$  を改善。

- 3) 診断窓口がループ部に含まれる場合 (図 12 参照)

図 12 に示す通り，窓口 4 がループ内にある場合，同じループにはない上流の窓口 1 の方へ遡行。

→  $t_{14}$ ,  $\rho_4$  に対して知識 5 を適用。

- 4) 上流窓口の別の分岐出力が，ループ部への到着量となっている場合

図 13 に示す通り，窓口 1 の窓口 3 以外への分岐出力  $t_{1r1}$  が窓口 5, 6 を含むループ部の到着量となっている場合，

- 4-1)  $t_{2r2} > t_1(1-r_1)$  の場合， $t_{23}$  の方に原因。

→  $t_{23}$ ,  $\rho_3$  に対して知識 5 を適用。

- 4-2)  $t_1(1-r_1) \gg t_{2r2}$  の場合， $t_{13}$  の方に原因。

→  $t_{13}$ ,  $\rho_3$  に対して知識 5 を適用。

- 4-3)  $t_1(1-r_1) > t_{2r2}$ ,  $t_1(1-r_1) \cong t_{2r2}$  の場合，

→  $t_{23}$ ,  $\rho_3$  に対して知識 5 を適用。または

→  $t_{13}$ ,  $\rho_3$  に対して知識 5 を適用。

根拠：より局所的に  $\rho_3$  を改善するためには， $t_{23}$ ,  $\rho_3$  に対して遡行した方がよい。なぜなら， $t_{13}$ ,  $\rho_3$  に対して知識 5 を適用して遡行すると，ループ部の到着量を変更することになり，ループ部内のすべての窓口に影響を及ぼすことになる。

- 5) その他の場合 (図 9 参照)

- 5-1)  $t_{2r2} \gg t_1(1-r_1)$  の場合， $t_{2r2}$  の方に原因。

→  $t_{23}$  と  $\rho_3$  に対して知識 5 を適用。

- 5-2)  $t_{2r2} \ll t_1(1-r_1)$  の場合， $t_1(1-r_1)$  の方に原因。

→  $t_{13}$  と  $\rho_3$  に対して知識 5 を適用。

- 5-3)  $t_{2r2} \cong t_1(1-r_1)$  の場合， $t_{2r2}$  と  $t_1(1-r_1)$  が同程度に原因。

→  $t_{23}$  と  $\rho_3$ ，かつ  $t_{13}$  と  $\rho_3$  に対して知識 5 を適用。

(昭和 63 年 9 月 2 日受付)

(平成 元年 5 月 9 日採録)



沢村 淳 (学生会員)

昭和 38 年生。昭和 62 年上智大学理工学部機械工学科卒業。平成元年同大学院理工学研究科機械工学専攻博士前期課程修了。現在，(株)ソニースーパーマイクロ事業本部に勤務。在学中は，定性推論に基づいたエキスパートシステムの研究に従事。人工知能技術全般に興味を持つ。



**志田 圭介**

昭和 41 年生。平成元年上智大学理工学部機械工学科卒業。現在、同大学院理工学研究科機械工学専攻博士前期課程在学。定性推論に基づいたエキスパートシステムの研究に従事。エキスパートシステム全般に興味を持つ。



**本位田真一 (正会員)**

昭和 28 年生。昭和 51 年早稲田大学理工学部電気工学科卒業。昭和 53 年同大学院理工学研究科電気工学専攻修士課程修了。工学博士。同年(株)東芝入社。現在、同社システム・ソフトウェア技術研究所研究主務。早稲田大学非常勤講師。主として、ソフトウェア工学、人工知能の研究に従事。ソフトウェアの基礎理論に興味をもつ。昭和 61 年情報処理学会論文賞受賞。共著「ソフトウェア開発のためのプロトタイプング・ツール」。人工知能学会、日本ソフトウェア科学会、電気学会、AAAI 各会員。



**伊藤 潔 (正会員)**

昭和 26 年生。昭和 49 年京都大学工学部情報工学科卒業。昭和 54 年同大学院理工学研究科情報工学専攻博士課程修了。京都大学工学博士。昭和 54 年より上智大学理工学部勤務。昭和 60 年より助教授。現在、同理工学部一般科学研究室情報科学部門所属。主として、ソフトウェア工学、シミュレーション手法、図面理解を主とするコンピュータビジョンの研究に従事。計算機援用の様々なシステム・手法に興味をもつ。電子情報通信学会、IEEE 等会員。ISO/TC 184/SC 2 WG 3 (産業用ロボットの安全性) 委員。著書：「ソフトウェア開発のためのプロトタイプングツール」(共著)、「システムプログラム」, 訳書：「並行処理と Unix」(共訳)。