

論 文

タグ付きタスクモデルによる待ち行列ネットワークの一解法について

今井 博英[†] 角山 正博^{††} 石井 郁夫[†] 牧野 秀夫^{†††}

A Method for Analyzing Queuing Network by Tagged-Task-Model

Hiroei IMAI[†], Masahiro TSUNOYAMA^{††}, Ikuo ISHII[†], and Hideo MAKINO^{†††}

あらまし 本論文では、待ち行列ネットワークでモデル化された分散型リアルタイムシステム中に故障が発生した場合の、過渡状態における性能の変化を短時間に解析する方法を示す。本方法は、待ち行列ネットワークをタグ付きタスクモデルによってモデル化することにより過渡状態の解析を行う。タグ付きタスクモデルを用いた解析では、初期値を変化させながら極限状態確率を何度も求める必要があるが、吸収状態への到達確率である吸収確率を用いることによって短時間に状態確率を求めることができる。更に、吸収確率を求める際に、状態集合を到達可能性に基づく同値類に分割し、状態確率の小さくなった同値類を一つの代表状態に集約することによって短時間に計算する方法を示す。最後に、本方法を用いたときの計算量と実際の計算結果を示し、本方法により短時間に高精度な解析が行えることを示す。

キーワード タグ付きタスクモデル, 連続時間型マルコフ連鎖, リアルタイムシステム, 性能解析

1. ま え が き

近年、高速通信技術並びにコンピュータシステムの発達により、コンピュータネットワークを利用した様々なマルチメディア通信サービスが実現されつつある。このようなシステムは、大量の情報を効率良く処理するために、分散型のリアルタイムシステムとして実現されることが多い[1]。また、マルチメディア通信においては、タスクが制限時間以内に処理可能かどうかはサービスの品質に大きな影響を与える。このため、制限時間以内に処理を終了できずに棄却されるタスクの割合、即ちタスクの棄却率が重要なシステムの性能の指標となる。また、サービスを提供するシステムは、故障が発生した場合にも、サービスを中断することがないように、フォールトトレラント性も考慮されている。マルチメディア通信においては、リアルタイム性が必要となるため、故障が発生した場合などの過渡状態などの短期間の性能の変化も、サービスの品質に影響

する。よって、このようなシステムを設計する際には、実際にシステムを構成する前に、システムが定常状態にある場合だけでなく、故障が発生した場合などの、過渡状態における性能についても十分な解析をする必要がある。このようなシステムの、棄却率を含む性能の解析方法としては、シミュレーションによる方法の他にペトリネット [2], [3] や待ち行列 [4] を用いる方法等が提案されている。しかし、高精度な解析を行うためには、シミュレーションによる方法では非常に多くの計算を必要とするため時間的及び経済的に問題があること、また後者の方法はいずれも定常状態における解析を行うものが大部分であるため、過渡状態における解析には不向きであること等の問題点があった。この課題に対処し、過渡状態における解析を効率良く行うためのシステムのモデル化の一つの方法として、タグ付きタスクモデルに基づくシステムの解析方法が提案されている [3], [5]。これは、システム中のある一つのタスクに注目し、そのタスクの処理が終了するまでの過程をモデル化することによって、システムの性能を解析する方法である。

筆者らはこの方法を用いて、タグ付きタスクの処理過程を連続時間型マルコフ連鎖を用いてモデル化することにより、複数のプロセッサと一つの待ち行列からなるレスポンスシステムにおけるプロセッサ故障時

[†] 新潟大学大学院自然科学研究科, 新潟市

Graduate School of Science and Technology, Niigata University, 8050, Ikarashi-2, Niigata-shi, 950-2181 Japan

^{††} 新潟工科大学, 柏崎市

Niigata Institute of Technology, 1719, Kashiwazaki-shi, 945-1195 Japan

^{†††} 新潟大学工学部, 新潟市

Faculty of Engineering, Niigata University, 8050, Ikarashi-2, Niigata-shi, 950-2181 Japan

の過渡状態の性能を解析する方法を提案した[6]. 更に, タグ付きタスクモデルの性質に基づいて, 状態を集約しながら計算することにより, 短時間に吸収状態確率を求める方法を提案した[7].

本論文では, 待ち行列ネットワークで表されるような大規模なシステムを短時間に解析するために, 吸収確率行列を用いることにより故障が発生した場合の過渡状態における棄却率の変化を求める方法, 更に, 吸収確率行列を, 状態を集約しながら計算することにより, 短時間に求める方法を示す. 本論文の構成は, はじめに, 2. で解析の対象となる待ち行列ネットワークとタグ付きタスクモデルを示す. 3. では故障発生時の過渡状態における棄却率を求める方法を示し, 4. では過渡状態の解析に用いる吸収確率行列を状態の集約に基づいて短時間に求める方法を示す. 5. で計算時間について検討を行う. 更に, 6. で解析例を示し, 7. でまとめを行う.

2. 諸定義及び仮定

2.1 待ち行列ネットワーク

待ち行列ネットワーク(QN)は有向グラフを用いて表される. グラフの各々のノードは複数のプロセッサと一つの待ち行列からなるコンピュータシステムを, 有向エッジはタスクの移動を表す. QNを次のように定義する.

[定義1] (待ち行列ネットワーク)

$$QN = (V, E, \Pi) \quad (1)$$

$V = \{v_0, v_1, \dots, v_N\}$: ノードの集合

$E = \{e_{i,j}\}$: エッジの集合

$\Pi = \{\pi_{i,j}\}$: エッジのラベルの集合 \square

ここで, ノード v_0 はQNの外部を表し, エッジ $e_{i,j}$ とそのラベル $\pi_{i,j}$ はノード v_i で処理が終了したタスクが, ノード v_j へ確率 $\pi_{i,j}$ で移動することを表す.

QNのノード及びタスクについて以下のような仮定をおく.

[仮定1] (ノードとタスク)

(1) QNはフィードフォワード網であり, すべてのノードには $\pi_{i,j} > 0$ ならば, $i > j (v_i, v_j \in V; 1 \leq i \leq N; 0 \leq j \leq N)$ となるように番号を付ける.

(2) ノード v_i は K_i 個のプロセッサと長さ $L_i - K_i$ の待ち行列をもつ.

(3) ノード v_i へのタスクの到着はパラメータ $\lambda_{0,i}$ のポアソン分布に従う.

(4) ノード v_i でのタスクの処理に要する時間はパラメータ μ_i の指数分布に従う.

(5) すべてのノードにおいて, タスクは先着順優先方式で処理される.

(6) タスクの制限時間はパラメータ ν の指数分布に従う. また, 制限時間を越えたタスクは直ちに捨てられる. \square

2.2 システムモデル

QNの状態の変化を連続時間型マルコフ連鎖でモデル化したものを, システムモデルと呼ぶ.

システムモデルの状態とその状態集合を以下のように定義する.

[定義2] (システムモデルの状態)

システムモデルの状態を次式で定義する.

$$\mathbf{n} = (n_1, \dots, n_i, \dots, n_N) \quad (2)$$

ここで $n_i (0 \leq n_i \leq L_i)$ はノード v_i のタスク数を表す. また, システムモデルの状態集合を $\mathcal{S}^{(N)}$ で表す. \square

システムモデルが時刻 t で状態 \mathbf{n} にある確率を $p_{\mathbf{n}}^{(N)}(t)$ で表し, 状態確率と呼び, $\mathbf{p}^{(N)}(t) = [p_{\mathbf{n}}^{(N)}(t)]_{\mathbf{n} \in \mathcal{S}^{(N)}}$ を状態確率ベクトルと呼ぶ. また, 状態 \mathbf{n} から \mathbf{n}' への推移率を $g_{\mathbf{n}, \mathbf{n}'}^{(N)}$ で表し, 推移率行列を $\mathbf{G}^{(N)} = [g_{\mathbf{n}, \mathbf{n}'}^{(N)}]_{\mathbf{n}, \mathbf{n}' \in \mathcal{S}^{(N)}}$ で表す. 推移率は $\lambda_{0,i}, \mu_i, \nu$ を用いて表される[8].

2.3 タグ付きタスクモデル

本論文では, QNに到着した代表的なタスクの処理の過程をモデル化したタグ付きタスクモデルを用いて過渡状態の解析を行う. はじめにタグ付きタスクモデルの状態を以下のように定義する.

[定義3] (タグ付きタスクモデルの状態)

タグ付きタスクモデルは次のような3種類の状態からなる.

C : タグ付きタスクの処理が終了し, QNから出力されたことを表す状態.

F : タグ付きタスクが制限時間を満たせないか, 又は, 待ち行列に入り切らずに捨てられたことを表す状態.

$\mathbf{u} = (n_1, \dots, n_i, \dots, n_N, l, r)$: タグ付きタスクがノード $v_l (1 \leq l \leq N)$ の待ち行列の $r (0 \leq r \leq L_l - K_l)$ 番目に滞在していることを表す状態. ここで $r = 0$ はタグ付きタスクがプロセッサで処理されていることを表す. \square

タグ付きタスクモデルの状態集合を $\mathcal{S}^{(T)}$ で表す.

更に, $\mathcal{S}^{(T)}$ を吸収状態と過渡状態に分類し, それぞれ $\mathcal{S}_A^{(T)}, \mathcal{S}_T^{(T)}$ で表す. タグ付きタスクモデルの場合, 状態 C と F のみが吸収状態で, それ以外は過渡状態となる.

タグ付きタスクモデルの時刻 t における状態 $\mathbf{u} (\in \mathcal{S}^{(T)})$ の状態確率を $p_{\mathbf{u}}^{(T)}(t)$, 状態確率ベクトルを $\mathbf{p}^{(T)}(t) = [p_{\mathbf{u}}^{(T)}(t)]_{\mathbf{u} \in \mathcal{S}^{(T)}}$ と表し, 更に, 極限状態確率ベクトルを $\mathbf{p}^{(T)} = \lim_{t \rightarrow \infty} \mathbf{p}^{(T)}(t)$ で表す. また, 状態確率ベクトルと極限状態確率ベクトルを吸収状態と過渡の状態に関してブロック分けし, それぞれ, $\mathbf{p}_A^{(T)}(t), \mathbf{p}_T^{(T)}(t), \mathbf{p}_A^{(T)}, \mathbf{p}_T^{(T)}$ で表す.

状態 \mathbf{u} から \mathbf{u}' への推移率を $g_{\mathbf{u}, \mathbf{u}'}^{(T)}$ で表し, 推移率行列を $\mathbf{G}^{(T)} = [g_{\mathbf{u}, \mathbf{u}'}^{(T)}]_{\mathbf{u}, \mathbf{u}' \in \mathcal{S}^{(T)}}$ で表す. 推移率は, $\lambda_{0,i}, \mu_i, \nu$ を用いて表される [8].

2.4 システム中に発生する故障

本論文では, システム中に発生する故障に関して以下のような仮定をおく.

[仮定 2] (故障)

(1) 故障は 1 個のノード内の 1 個のプロセッサに発生する.

なお, プロセッサ故障が発生するノードを v_{iF} で表し, 故障発生時刻を 0 とする.

3. 過渡状態の解析方法

本論文では, 故障発生時刻を挟むある期間内にシステムに到着したタスクが, 制限時間を満たせないか, 若しくは待ち行列からあふれることにより捨てられる確率の変化を求める方法を示す. 時刻 t に到着したタスクが捨てられる確率を $q_F(t)$ で表し, 棄却率と呼ぶ.

3.1 吸収確率

タスク棄却率の時間的な変化を求めるためには, タグ付きタスクモデルの初期値を変更しながら極限状態確率を繰返し求める必要がある [6]. この繰返しを除き, 極限状態確率を短時間に求めるために吸収確率を用いる. 吸収確率並びに吸収確率行列を次のように定義する.

[定義 4] (吸収確率)

タグ付きタスクモデルにおいて, 初期状態が $\mathbf{u} (\in \mathcal{S}_T^{(T)})$ であったときに極限状態が $\mathbf{u}' (\in \mathcal{S}_A^{(T)})$ である確率を吸収確率と呼び, $\alpha_{\mathbf{u}\mathbf{u}'}$ で表す. 更に, 行列 $\alpha = [\alpha_{\mathbf{u}\mathbf{u}'}]_{\mathbf{u}, \mathbf{u}' \in \mathcal{S}^{(T)}}$ を吸収確率行列と呼ぶ.

□

α を過渡状態と吸収状態に関してブロック分けし, それぞれ, $\alpha_{AA}, \alpha_{AT}, \alpha_{TT}, \alpha_{TA}$ と表す. これらの

部分行列はそれぞれ次式のようになる [9].

$$\alpha_{AA} = \mathbf{I} \quad (3)$$

$$\alpha_{AT} = \alpha_{TT} = \mathbf{0} \quad (4)$$

$$\alpha_{TA} = \mathbf{H}_{TA}^{(T)} + \sum_{n=1}^{\infty} (\mathbf{H}_{TT}^{(T)})^n \mathbf{H}_{TA}^{(T)} \quad (5)$$

ここで, $\mathbf{H}^{(T)} = [h_{\mathbf{u}\mathbf{u}'}^{(T)}]_{\mathbf{u}, \mathbf{u}' \in \mathcal{S}^{(T)}}$ はタグ付きタスクモデルの隠れマルコフ連鎖の推移確率行列を表し, $\mathbf{H}_{TA}^{(T)}, \mathbf{H}_{TT}^{(T)}$ は, 吸収状態と過渡状態に関して $\mathbf{H}^{(T)}$ をブロック分けした行列を表す.

α を用いることによって, 極限状態確率は次式より求めることができる.

$$\mathbf{p}^{(T)} = \mathbf{p}^{(T)}(0)\alpha \quad (6)$$

3.2 故障前に到着したタスク

はじめに, 故障前に到着したタスクが制限時間を満たせずに捨てられる確率 $q_F(t)$ ($t < 0$) を求める手順を示す.

故障前のシステムモデルの状態集合を $\mathcal{S}^{(N,-)}$ で表し, 定常分布を $\mathbf{p}^{(N,-)}$ で表す.

故障前のタグ付きタスクモデルの状態集合を $\mathcal{S}^{(T,-)}$ で表し, 故障前のタグ付きタスクモデルの状態確率分布と推移率行列を, それぞれ $\mathbf{p}^{(T,-)}(t), \mathbf{G}^{(T,-)}$ で表す. また, 故障後のタグ付きタスクモデルの状態集合を $\mathcal{S}^{(T,+)}$ で表し, 故障後のタグ付きタスクモデルの極限状態確率分布と吸収確率行列をそれぞれ $\mathbf{p}^{(T,+)}, \alpha^{(+)}$ で表す.

更に, システムモデルの状態が $\mathbf{n} (\in \mathcal{S}^{(N,-)})$ であるときタグ付きタスクが到着した場合に, タグ付きタスクモデルの状態が $\mathbf{u} (\in \mathcal{S}^{(T,-)})$ となる確率を $\gamma_{\mathbf{n}, \mathbf{u}}^{(-)}$ で表す. また, その行列を $\mathbf{\Gamma}^{(-)} = [\gamma_{\mathbf{n}, \mathbf{u}}^{(-)}]_{\mathbf{n} \in \mathcal{S}^{(N,-)}, \mathbf{u} \in \mathcal{S}^{(T,-)}}$ と表し, システムモデルからタグ付きタスクモデルへの変換行列と呼ぶ [8]. 次に, 故障前のタグ付きタスクモデルの状態が $\mathbf{u} (\in \mathcal{S}^{(T,-)})$ であるとき故障が発生した場合に, 状態が $\mathbf{u}' (\in \mathcal{S}^{(T,+)})$ となる確率を $\phi_{\mathbf{u}, \mathbf{u}'}^{(T)}$ で表し, その行列を $\mathbf{\Phi}^{(T)} = [\phi_{\mathbf{u}, \mathbf{u}'}^{(T)}]_{\mathbf{u} \in \mathcal{S}^{(T,-)}, \mathbf{u}' \in \mathcal{S}^{(T,+)}}$ と表す [8].

なお, 刻み幅 Δ は精度を考慮してあらかじめ決めておくものとし, 計算終了時刻を t_e とする.

[手順 1] (故障前に到着したタスクの棄却率の導出)

(1) 故障前のシステムモデルの定常分布 $\mathbf{p}^{(N,-)}$ を求める.

(2) $\mathbf{p}^{(N,-)}$ から, 次式を用いて, 故障前のタグ付

きタスクモデルの初期分布 $\mathbf{p}^{(T,-)}(0)$ を求める.

$$\mathbf{p}^{(T,-)}(0) \leftarrow \mathbf{p}^{(N,-)} \mathbf{\Gamma}^{(-)} \quad (7)$$

(3) $t \leftarrow 0$

(4) 極限状態確率を次式で求める.

$$\mathbf{p}^{(T,+)} = \mathbf{p}^{(T,-)}(t) \mathbf{\Phi}^{(T)} \mathbf{\alpha}^{(+)} \quad (8)$$

棄却率 $q_F(t)$ は, 次式より得られる.

$$q_F(-t) = p_F^{(T,+)} \quad (9)$$

(5) Δ 時間後の状態確率分布を次式で求める.

$$\mathbf{p}^{(T,-)}(t + \Delta) = \mathbf{p}^{(T,-)}(t) (\mathbf{I} + \mathbf{G}^{(T,-)} \Delta) \quad (10)$$

(6) $t \leftarrow t + \Delta$ とし, $t < t_e$ ならば (4) に戻る.

3.3 故障後に到着したタスク

続いて, 故障後に到着したタスクの棄却率 $q_F(t)$ ($t > 0$) を導出する手順を示す.

故障後のシステムモデルの状態集合を $\mathcal{S}^{(N,+)}$ で表し, 故障後のシステムモデルの状態確率分布, 及び推移率行列を $\mathbf{p}^{(N,+)}(t)$, $\mathbf{G}^{(N,+)}$ で表す. また, 故障後のシステムモデルからタグ付きタスクモデルへの変換行列を $\mathbf{\Gamma}^{(+)}$ で表す. なお, 刻幅 Δ は精度を考慮してあらかじめ決めておくものとする. また, 計算終了時刻を t_e とする.

[手順 2] (故障後に到着したタスクの棄却率の導出)

(1) システムモデルの定常分布 $\mathbf{p}^{(N,-)}$ から, 次式を用いて, 故障後のシステムモデルの初期分布 $\mathbf{p}^{(N,+)}(0)$ を求める.

$$\begin{aligned} & p_{(n_1, \dots, n_{i_F}, \dots, n_N)}^{(N,+)}(0) \\ &= \begin{cases} p_{(n_1, \dots, n_{i_F}, \dots, n_N)}^{(N,-)} & (n_{i_F} < K_{i_F} - 1) \\ p_{(n_1, \dots, n_{i_F}, \dots, n_N)}^{(N,-)} + p_{(n_1, \dots, n_{i_F}+1, \dots, n_N)}^{(N,-)} & (n_{i_F} = K_{i_F} - 1) \end{cases} \end{aligned} \quad (11)$$

(2) $t \leftarrow 0$

(3) 極限状態確率を次式で求める.

$$\mathbf{p}^{(T,+)} = \mathbf{p}^{(N,+)}(t) \mathbf{\Gamma}^{(+)} \mathbf{\alpha}^{(+)} \quad (12)$$

棄却率 $q_F(t)$ は, 次式より得られる.

$$q_F(t) = p_F^{(T,+)} \quad (13)$$

(4) Δ 時間後の状態確率分布を次式で求める

$$\mathbf{p}^{(N,+)}(t + \Delta) = \mathbf{p}^{(N,+)}(t) (\mathbf{I} + \mathbf{G}^{(N,+)} \Delta) \quad (14)$$

(5) $t \leftarrow t + \Delta$ とし, $t < t_e$ ならば (3) に戻る.

4. 吸収確率の導出方法

実際のシステムを解析する場合には, タグ付きタスクモデルは大規模なマルコフ連鎖となり, 式 (5) を用いて吸収確率行列を計算するには多くの時間を要する. ここでは, タグ付きタスクモデルの状態を相互到達可能性に基づく同値類に分割し, 計算の過程で状態確率の小さくなった同値類を一つの状態に集約しながら計算することにより, 短時間に吸収確率行列を導出する方法を示す.

4.1 状態の分割

$\mathbf{u} \in \mathcal{S}^{(T)}$ から $\mathbf{u}' \in \mathcal{S}^{(T)}$; $\mathbf{u} \neq \mathbf{u}'$ へ 1 度以上の状態遷移によって到達可能な場合, $\mathbf{u} \rightarrow \mathbf{u}'$ と表す. これを用いて, 相互到達可能性を次のように定義する. [定義 5] (相互到達可能性)

状態 \mathbf{u} と \mathbf{u}' ($\mathbf{u} \neq \mathbf{u}'$) が相互到達可能であるとは, $\mathbf{u} \rightarrow \mathbf{u}'$ かつ $\mathbf{u}' \rightarrow \mathbf{u}$ の場合でありかつそのときに限る. \square

上記の関係は同値関係であることから, $\mathcal{S}_T^{(T)}$ を相互到達可能性に基づく同値類に分割することができる. このように分割した状態集合を $\mathcal{S}_1^{(T)}, \mathcal{S}_2^{(T)}, \dots, \mathcal{S}_M^{(T)}$ で表す. このとき同値類 $\mathbf{u} \in \mathcal{S}_a^{(T)}$ から $\mathbf{u}' \in \mathcal{S}_b^{(T)}$ ($\neq \mathcal{S}_a^{(T)}$) に推移可能な組が一つでも存在する場合は $a > b$ となるように番号を付ける.

4.2 同値関係を利用した吸収確率の導出

前節で求めた同値類 $\mathcal{S}_a^{(T)}$ ($a = 1, 2, \dots, M$) を集約しながら計算することによって α_{TA} を短時間で求める方法を示す.

定義により $\alpha_{u,C}$ 及び $\alpha_{u,F}$ は, 初期状態が \mathbf{u} であったときに, 極限状態が状態 C 若しくは F である確率を表す. このため, 初期分布の状態 \mathbf{u} の確率のみを 1, それ以外のすべての状態確率を 0 としてタグ付きタスクモデルの極限状態確率を計算することによって得られる状態 C と F の状態確率がそれぞれ, $\alpha_{u,C}$, $\alpha_{u,F}$ となる. 状態 \mathbf{u} が属する同値類を $\mathcal{S}_m^{(T)}$ とすると, $\mathcal{S}_a^{(T)}$ ($a = m + 1, \dots, M$) に含まれる状態には推移することはないため, 無視することができる. また, $\mathcal{S}_a^{(T)}$ ($a = 1, \dots, m$) についても, 状態確率の和が一定の値以下になった場合は, 状態確率が増加することはないため, 一つの代表状態に集約することによって計算量を減らすことができる.

続いて, α_{TA} の導出手順を示す. ここで, $\mathbf{H}_{ab}^{(T)} = [h_{\mathbf{u}\mathbf{u}'}^{(T)}]_{\mathbf{u} \in \mathcal{S}_a^{(T)}, \mathbf{u}' \in \mathcal{S}_b^{(T)}}$, $\mathbf{p}_a^{(T)}(t) = [p_{\mathbf{u}}^{(T)}(t)]_{\mathbf{u} \in \mathcal{S}_a^{(T)}}$ とする. また, 集約条件のしきい値 ε と, 計算終了条

件のしきい値 δ は、精度を考慮してあらかじめ決めておくものとする。なお、 ε は同値類を集約する場合のしきい値を表し、同値類中の状態確率の和が ε 以下になったときに、その同値類を集約する。また、 δ は、吸収確率の計算を終了するためのしきい値であり、 $1 - (p_C^{(T)}(t) + p_F^{(T)}(t))$ が δ 以下になった場合に、十分収束したとみなして、計算を終了する。

[手順 3] (α_{TA} の導出)

すべての過渡状態 $\mathbf{u} \in \mathcal{S}^{(T)}$ について以下の計算を繰り返す。ただし、 \mathbf{u} が属する同値類を $\mathcal{S}_m^{(T)}$ とする。

(1) 状態集合 \mathcal{S} と初期分布 $\mathbf{p}^{(T)}(0) = [p_{\mathbf{u}'}^{(T)}(0)]_{\mathbf{u}' \in \mathcal{S}}$, 隠れマルコフ連鎖の推移確率行列 $\mathbf{H}^{(T)}$ を次式より求める。

$$\mathcal{S} = \mathcal{S}_A^{(T)} \bigcup_{a=1}^m \mathcal{S}_a^{(T)} \quad (15)$$

$$\mathbf{H}^{(T)} = [\mathbf{H}_{ab}^{(T)}]_{a,b=1,2,\dots,m,A} \quad (16)$$

$$p_{\mathbf{u}'}^{(T)}(0) = \begin{cases} 1 & (\mathbf{u}' = \mathbf{u}) \\ 0 & (\mathbf{u}' \neq \mathbf{u}) \end{cases} \quad (17)$$

(2) $a \leftarrow m, t \leftarrow 0$

(3) $\sum_{\mathbf{u}' \in \mathcal{S}_a^{(T)}} p_{\mathbf{u}'}^{(T)}(n) < \varepsilon$ ならば、次式を用いて同値類 $\mathcal{S}_a^{(T)}$ を集約する。

$$\mathbf{H}_{ab}^{(T)} \leftarrow \frac{p_a^{(T)}(t) \mathbf{H}_{ab}^{(T)}}{\sum_{\mathbf{u}' \in \mathcal{S}_a^{(T)}} p_{\mathbf{u}'}^{(T)}(t)} \quad (b = 0, \dots, a, A) \quad (18)$$

$$\mathbf{H}_{ba}^{(T)} \leftarrow \|\mathbf{H}_{ba}^{(T)}\| \quad (b = a, \dots, m) \quad (19)$$

$$p_a^{(T)}(t) \leftarrow \sum_{\mathbf{a} \in \mathcal{S}_a^{(T)}} p_{\mathbf{a}}^{(T)}(t) \quad (20)$$

ここで、 $\|\mathbf{H}_{ab}^{(T)}\|$ は行列の要素をすべて足し合わせることを表す。更に、 $a \leftarrow a - 1$ として (3) に戻る。

(4) 1 単位時間後の状態確率分布を次式で求める。

$$\mathbf{p}^{(T)}(t+1) \leftarrow \mathbf{p}^{(T)}(t) \mathbf{H}^{(T)} \quad (21)$$

(5) $t \leftarrow t + 1$ とし、 $1 - (p_C^{(T)}(t) + p_F^{(T)}(t)) > \delta$ ならば (3) に戻る。そうでなければ、次式より $\alpha_{u,C}, \alpha_{u,F}$ を求める。

$$\alpha_{u,C} = \frac{p_C^{(T)}(t)}{p_C^{(T)}(t) + p_F^{(T)}(t)} \quad (22)$$

$$\alpha_{u,F} = \frac{p_F^{(T)}(t)}{p_C^{(T)}(t) + p_F^{(T)}(t)} \quad (23)$$

5. 計算時間

5.1 棄却率の計算時間

はじめに、吸収確率行列を用いて棄却率を導出した場合と、文献 [6] の方法を用いて棄却率を導出した場合の計算時間について検討を行う。

本論文で示した方法と文献 [6] で示した方法はともに、棄却率を求める際、タグ付きタスクモデルの初期状態確率を変化させながら極限分布を繰り返し求める必要がある。吸収確率を用いる方法では、一度吸収確率を求めた後、極限状態確率は、式 (8), (12) を用いて、ベクトルと行列の掛け算のみで求めることができる。一方、文献 [6] の方法では、

$$\mathbf{p}^{(T,+)}(t + \Delta) = \mathbf{p}^{(T,+)}(t) (\mathbf{I} + \mathbf{G}\Delta) \quad (24)$$

により $\mathbf{p}^{(T,+)}(t)$ が収束するまで計算することにより求めている。このため、棄却率 $q_F(t)$ を求める区間 $[-t_e, t_e]$ が広い程吸収確率行列を用いて計算を行ったほうが、用いない場合に比べて計算時間が短くなる。

5.2 吸収確率の導出時間

続いて、集約を行いながら α_{TA} を導出する場合と、集約を行わずに導出する場合の計算時間について検討を行う。

α_{TA} を導出する場合、集約を行わない場合に比べて、同値類が集約可能か否かを調べるための演算と同値類を集約するための演算が必要となる。ここでは、集約を行いながら α_{TA} を求めた場合と、集約を行わずに求めた場合の乗算回数を比較し、集約することにより乗算回数が減少する条件を求める。

まず、タグ付きタスクモデルの極限分布を集約を行いながら求めた場合と集約を行わずに求めた場合の乗算回数の差を求める。

ブロック $\mathbf{H}_{ab}^{(T)}$ を式 (18) により、縦方向だけ集約した行列を $\mathbf{H}_{ab}^{(T)'}$ と表し、式 (19) により、縦横両方向とも集約した行列を $\mathbf{H}_{ab}^{(T)''}$ で表す。また、集合 $\mathcal{S}^{(T)}$ の状態数を $N(\mathcal{S}^{(T)})$, 行列 $\mathbf{H}^{(T)}$ の 0 でない要素の数を $N(\mathbf{H}^{(T)})$ で表す。

集約しない場合の 1 単位時間後の確率分布の導出に必要な乗算回数と、同値類 $\mathcal{S}_{m+1}^{(T)}, \dots, \mathcal{S}_M^{(T)}$ が集約されている場合の 1 単位時間後の確率分布の導出に必要な乗算回数との差 $D_{\text{mul}}(m)$ は次式で表される。

$$D_{\text{mul}}(m)$$

$$\begin{aligned}
 &= \sum_{a=m+1}^M \left\{ N(\mathbf{H}_{aA}^{(T)}) - N(\mathbf{H}^{(T)'}_{aA}) \right\} \\
 &+ \sum_{b=1}^m (N(\mathbf{H}_{ab}^{(T)}) - N(\mathbf{H}^{(T)'}_{ab})) \\
 &+ \sum_{b=m+1}^a (N(\mathbf{H}_{ab}^{(T)}) - N(\mathbf{H}^{(T)''}_{ab})) \} \quad (25)
 \end{aligned}$$

更に、同値類 $\mathcal{S}_m^{(T)}$ が集約される時刻を t_m で表し、計算終了時刻を t_0 とすると、一つのタグ付きタスクモデルの極限分布を求める際の減少回数 $D_{mul}^{(M)}$ は次式で表される。ただし、 $t_{M+1} = 0$ とする。

$$\begin{aligned}
 D_{mul}^{(M)} &= \sum_{m=0}^M (t_m - t_{m+1}) D_{mul}(m) \\
 &= t_0 \left\{ N(\mathbf{H}) - \sum_{a=1}^M \sum_{b=1}^a N(\mathbf{H}^{(T)''}_{ab}) \right\} \\
 &+ \sum_{m=1}^M t_m \left\{ N(\mathbf{H}^{(T)'}_{mA}) - N(\mathbf{H}^{(T)}) \right. \\
 &+ N(\mathbf{H}^{(T)''}_{mm}) - N(\mathbf{H}_{mm}^{(T)}) \\
 &+ \left. \sum_{a=m+1}^M (N(\mathbf{H}^{(T)''}_{am}) - N(\mathbf{H}^{(T)'}_{am})) \right\} \quad (26)
 \end{aligned}$$

また、すべての同値類を集約するための乗算回数は次式で表される。

$$N(\mathbf{H}_{TT}^{(T)}) + N(\mathbf{H}_{TA}^{(T)}) \quad (27)$$

よって、乗算回数が減少するための条件は次式で表される。

$$\begin{aligned}
 &N(\mathbf{H}_{TT}^{(T)}) + N(\mathbf{H}_{TA}^{(T)}) \\
 &\leq t_0 \left\{ N(\mathbf{H}) - \sum_{a=1}^M \sum_{b=1}^a N(\mathbf{H}^{(T)''}_{ab}) \right\} \\
 &+ \sum_{m=1}^M t_m \left\{ N(\mathbf{H}^{(T)'}_{mA}) - N(\mathbf{H}_{mA}^{(T)}) \right. \\
 &+ N(\mathbf{H}^{(T)''}_{mm}) - N(\mathbf{H}_{mm}^{(T)}) \\
 &+ \left. \sum_{a=m+1}^M (N(\mathbf{H}^{(T)''}_{am}) - N(\mathbf{H}^{(T)'}_{am})) \right\} \quad (28)
 \end{aligned}$$

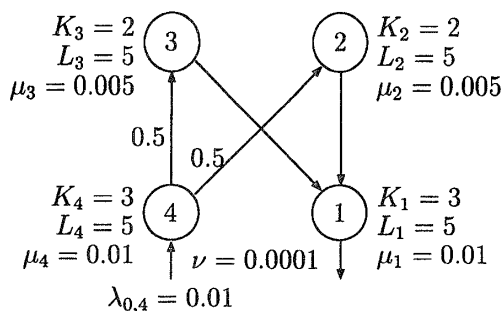
この式の右辺第2項目以降は、

$$\begin{aligned}
 N(\mathbf{H}^{(T)'}_{mA}) &\leq N(\mathbf{H}_{mA}^{(T)}) \\
 N(\mathbf{H}^{(T)''}_{mm}) &\leq N(\mathbf{H}_{mm}^{(T)}) \\
 N(\mathbf{H}^{(T)''}_{am}) &\leq N(\mathbf{H}^{(T)'}_{am}) \quad (29)
 \end{aligned}$$

であることから、0未満の値しかとらないため、乗算回数の差が減少する方向にしか作用しない。このことから計算終了時刻 t_0 が大きく、同値類 $\mathcal{S}_m^{(T)}$ が集約される時刻 $t_m (m = 1, 2, \dots, M)$ が小さいほど乗算回数が減少することがわかる。

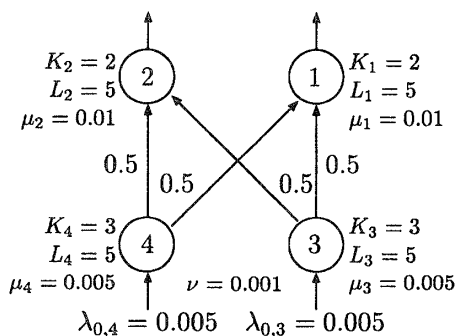
6. 解析例

図1に示す二つのQNを、本方法を用いて解析を行った結果を示す。Type1とType2を比較すると故障後のタグ付きタスクモデルの状態数は等しいが、同値類の数が大きく異なる。これは、Type1の場合、ノード1, 2, 3にタスクが到着するためには、必ずノード4を通過しなければならないという制約があるため、



Faulty node	4
Num. of states of the tagged-task model after fault	6914
Num. of cosets of the tagged-task model after fault	445

(a) Type 1



Faulty node	4
Num. of states of the tagged-task model after fault	6914
Num. of cosets of the tagged-task model after fault	17

(b) Type 2

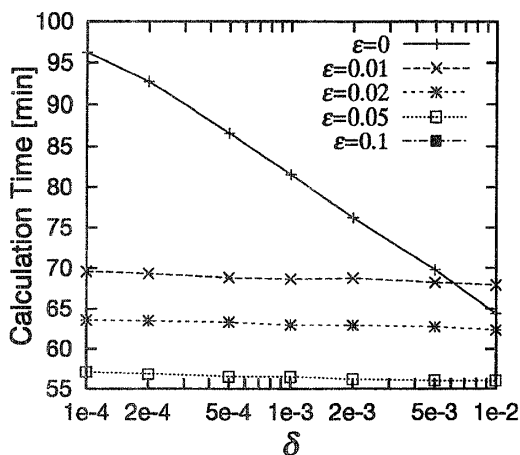
図1 待ち行列ネットワークで表されたシステム
Fig. 1 Systems represented by queuing network.

タグ付きタスクがノード 4 で処理されている場合は、タグ付きタスクの位置が変化することなしに、ノード 1, 2, 3 のタスク数が増加することはない。このため、タグ付きタスクがノード 4 で処理されていることを表す状態は同値類にまとめることができず、同値類の数が多くなる。一方、Type2 の場合はこのような制限がないため、Type1 に比べて同値類の数が少なくなる。

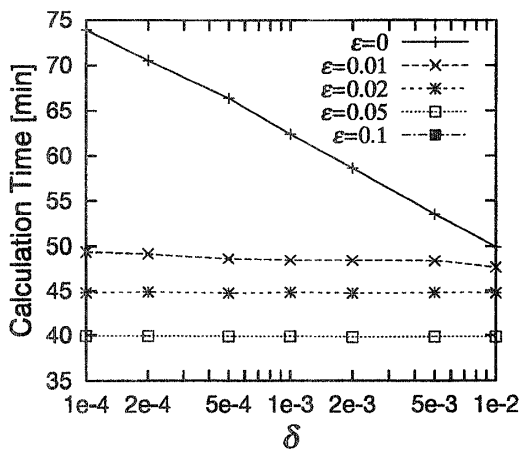
吸収確率行列 α_{TA} の計算時間の結果を図 2 に示す。集約を行いながら計算する場合は、Type1, Type2 ともに、計算時間は ε に依存しており、 δ にはほとんど影響されることがわかる。 $\delta = 0.0001$ の場合、Type1 では $\varepsilon = 0.01$ で集約を行わない場合の約 70%、 $\varepsilon = 0.1$ では約 53% の計算時間となっており、Type2 では、 $\varepsilon = 0.01$ で約 66%、 $\varepsilon = 0.1$ で約 53% の計算時間となっている。また、しきい値が同じ場合には、

同値類の数が少ない方が、集約を行いながら計算することにより計算時間が減少することがわかる。また、Type1 の場合 $\delta = 0.01$ のとき、集約を行わずに計算したほうが、 $\varepsilon = 0.01$ として集約を行いながら計算した場合よりも、計算時間が短くなっている。これは、集約条件を調べるための演算や集約を行うための演算が、状態を集約することによる演算の減少量よりも多くなるためである。

手順 1、並びに、手順 2 を用いて、棄却率 $q_F(t)$ を計算した結果を図 3 に示す。この図より、集約を行いながら計算した α_{TA} を用いた場合と、集約を行わずに計算した α_{TA} を用いた場合、すなわち、誤差を含まない場合との差は、どの場合も 0.5% 以内に収まっており、集約を行いながら計算した α_{TA} を用いて、過渡状態における棄却率を計算した場合でも、十分な精度で値が得られることがわかる。なお、本導出方法の誤差は、吸収確率の導出時に同値類を集約すること

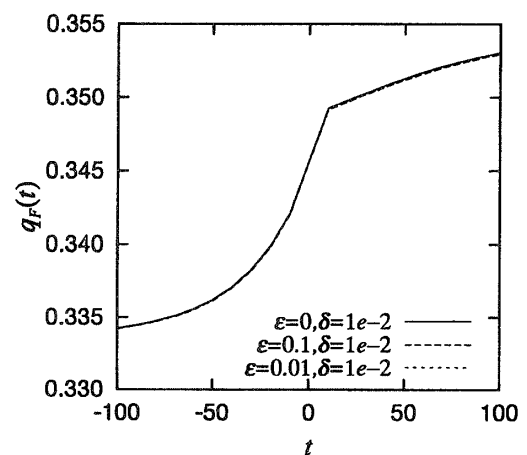


(a) Type 1

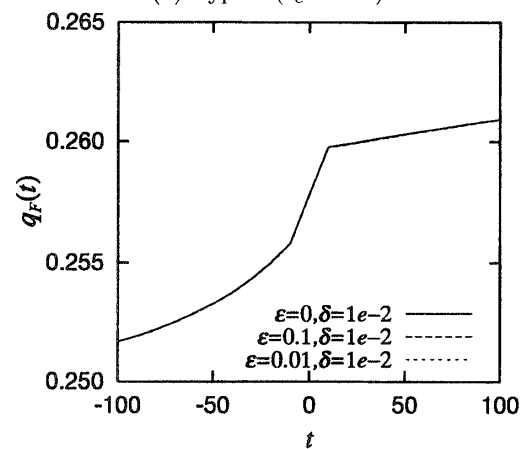


(b) Type 2

図 2 吸収確率行列の計算時間 (CPU:R10000/195 MHz)
Fig. 2 Calculation time of absorption probability matrix.

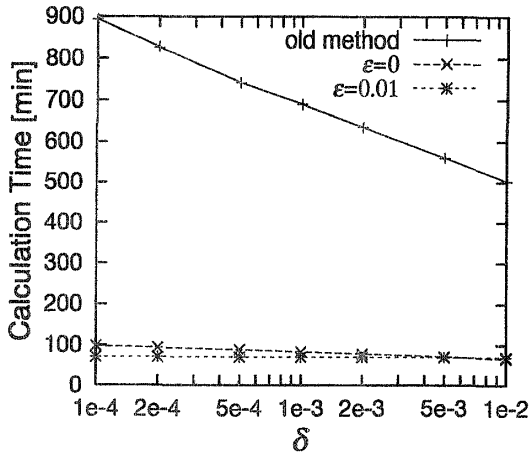


(a) Type 1 ($t_e = 100$)

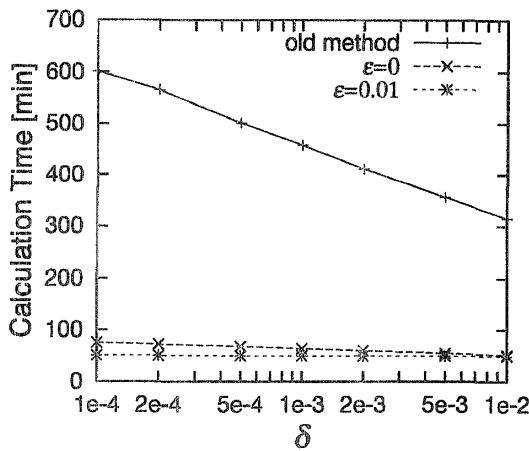


(b) Type 2 ($t_e = 100$)

図 3 棄却率
Fig. 3 Task loss ratio.



(a) Type 1 ($t_e = 100$)



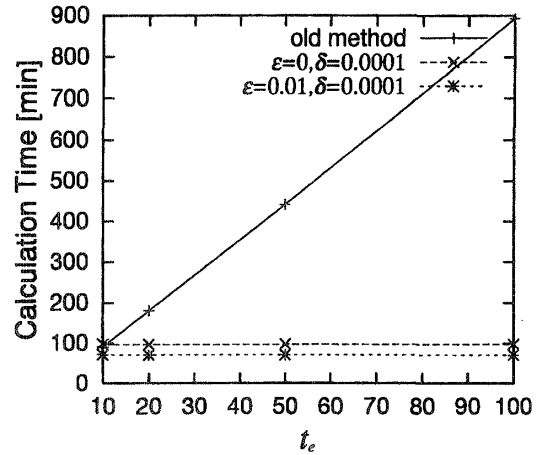
(b) Type 2 ($t_e = 100$)

図4 棄却率の計算時間 (1)

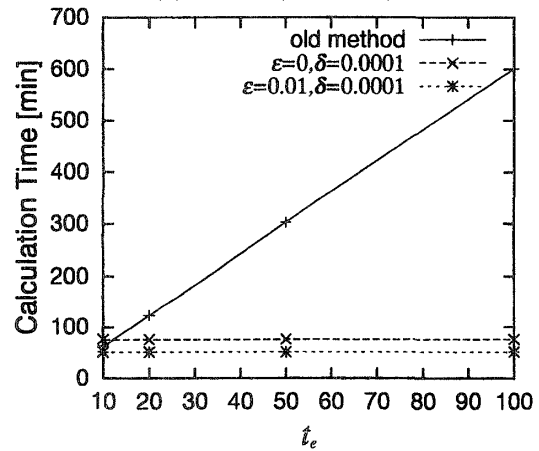
Fig. 4 Calculation time of task loss ratio (1).

によって生じ、この最大値は、 $\max\{1, \epsilon M\}$ で表せるが、集約の代表状態確率を吸収確率の計算に用いることによって相殺されるため、実際の誤差の値はこれより小さな値となっている。

文献 [6] に示した方法と、本論文で示した方法の棄却率 $q_F(t)$ の導出時間を図 4 に示す。図中の old method は文献 [6] で示した方法で求めた場合の計算時間を表す。 $\epsilon = 0, 0.01$ は吸収確率を用いて計算した場合の計算時間を示す。 $\epsilon = 0$ は、状態集約を行わずに吸収確率を求めた場合の計算時間を、更に、 $\epsilon = 0.01$ は、集約を行いながら吸収確率行列を計算した場合の計算時間を示す。文献 [6] の方法に比べ、 $\epsilon = 0$ の場合でも、Type1 の QN で $\delta = 0.0001$ の場合で約 11% となっている。また、Type2 で $\delta = 0.0001$ の場合は約 13% の計算時間で棄却率が得られており、文献 [6] の方法よりも短時間で解析が行えることがわかる。



(a) Type 1 ($\delta = 0.001$)



(b) Type 2 ($\delta = 0.001$)

図5 棄却率の計算時間 (2)

Fig. 5 Calculation time of task loss ratio (2).

最後に、棄却率の導出時間と、計算終了時刻 t_e の関係を図 5 に示す。 t_e が増加するに従って、old method では、計算時間が大幅に増加するのに比べて、本方法ではほとんど変化しておらず、求める区間 $[-t_e, t_e]$ が広いほど、本方法が有利であることがわかる。

7. むすび

本論文では、待ち行列ネットワークで表された分散型リアルタイムシステム中に故障が発生した場合の過渡状態における性能の変化を短時間に解析する方法を示した。はじめに、吸収確率行列を用いて極限状態確率を求めることにより、短時間に棄却率の過渡解を求める方法を示した。また、吸収確率行列を短時間に求めるために、タグ付きタスクモデルの状態集合を到達可能性に基づく同値類に分割し、同値類を集約しながら、タグ付きタスクモデルの吸収確率行列を計算する方法を示した。最後に、解析例を示し、短時間で十分

な精度で計算が行えることを示した。

今後はこの方法を用いて計算を行う場合の誤差の検討や、実際のシステムに本方法を適用する方法の検討などを行う予定である。

文 献

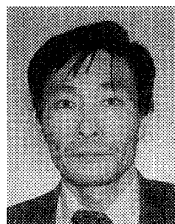
- [1] 潮田健太郎, “分散処理構成のビデオ・サーバ, 負荷分散でストリーム単価軽減,” 日経エレクトロニクス, no.637, pp.115-122, June 1995.
- [2] B. Berthomieu and M. Diaz, “Modeling and verification of time dependent systems using time Petri nets,” IEEE Trans. on Software Engineering, vol.17, no.3, pp.259-273, March 1991.
- [3] K.S. Trivedi, J.K. Muppala, and S.P. Woollet, “Real-time-systems performance in the presence of failures,” IEEE Computer, vol.24, no.5, pp.37-47, May 1991.
- [4] W. Zhao and J.A. Stankovic, “Performance analysis of FCFS and improved FCFS scheduling algorithms for dynamic real-time computer systems,” Proc. of the IEEE 10th Real-Time Systems Symp., pp.156-165, 1989.
- [5] B. Melamed and M. Yadin, “Randomization procedures in the computation of cumulative-time distributions over discrete-state Markov processes,” Operation Research, vol.32, no.4, pp.926-944, July-Aug. 1984.
- [6] 今井博英, 角山正博, 石井郁夫, 牧野秀夫, 内藤祥雄, “レスポンスシステムにおけるプロセッサ故障時の解析方法について,” 信学論 (D-I), vol.J78-D-I, no.8, pp.716-723, Aug. 1995.
- [7] 今井博英, 角山正博, 石井郁夫, 牧野秀夫, “タグ付きタスクモデルの一解法について,” 信学論 (D-I), vol.J80-D-I, no.10, pp.836-844, Oct. 1997.
- [8] 今井博英, 角山正博, 石井郁夫, 牧野秀夫, “タグ付きタスクモデルによる待ち行列ネットワークの一解法,” 信学技報, FTS97-70, Dec. 1997.
- [9] B.L. Nelson, “STOCHASTIC MODELING—Analysis and Simulation,” McGraw-Hill, Inc., 1995.

(平成 10 年 7 月 13 日受付, 11 月 2 日再受付)



今井 博英 (正員)

平 5 新潟大・工・情報卒. 平 7 同大学院修士課程了, 平 10 同大学院自然科学研究科博士後期課程了. 現在同大学院自然科学研究科助手. リアルタイムシステム, フォールトトレラントシステムの研究に従事. 平 10 工博.



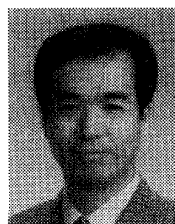
角山 正博 (正員)

昭 44 新潟大・工・電子卒. 同年東京芝浦電気(株)入社. 昭 46 横河ヒューレットパッカード(株)入社. 主に測定器の研究開発に従事. 長岡技術科学大学教務職員, 助手, 長岡工業高等専門学校教授を経て, 現在新潟工科大学教授. 論理回路の故障診断, 高信頼性並列処理システム, リアルタイムシステム等に興味をもつ. 平 2 工博. 情報処理学会, IEEE 等各会員.



石井 郁夫 (正員)

昭 38 新潟大・工・電気卒. 同大電子助手, 助教授, 同大情報教授を経て, 現在同大大学院自然科学研究科教授. この間, 画像・図形処理等の研究に従事. 現在人工現実感, ヒューマンインタフェース, 画像・図形処理装置, リアルタイムシステムなどの研究を行っている. 工博(東工大). 情報処理学会, 日本 ME 学会各会員.



牧野 秀夫 (正員)

昭 51 新潟大・工・電子卒. 昭 53 同大大学院修士課程了. 同年情報勤務. 平 2 助教授, 平 7 教授, 現在に至る. 平 1 より 1 年間カナダ・トロント大にて生体信号処理の研究に従事. 最近は, 埋込み型医用機器, 福祉機器, 画像処理, リアルタイムシステムなどの研究を行っている. 工博. IEEE, 日本 ME 学会各会員.