

放送型データベースシステムにおけるモバイル端末の電力消費を考慮した問合せ処理に関する一考察

A Study on Query Processing Considering the Energy Consumption of Mobile Clients for Broadcast Database Systems

北島 信哉^{*} 寺田 努[†] 原 隆浩[‡]
西尾 章治郎[‡]

Shinya KITAJIMA Tsutomu TERADA
Takahiro HARA Shojiro NISHIO

近年、サーバが携帯端末や PDA などのモバイル端末にデータベースの内容を周期的に放送する放送型データベースシステムが注目されている。放送型データベースシステムにおける問合せ処理手法はオンデマンド型方式、クライアント型方式、協調型方式の 3 方式が考えられるが、これらの方式は電力消費に差がある。そこで本論文では、モバイル端末の電力消費を考慮し、3 方式の中から動的に処理方式を選択する手法を提案する。

In recent years, there has been an increasing interest in the broadcast database system where the server periodically broadcasts contents of a database to mobile clients such as portable computers and PDAs. There are three query processing methods for the broadcast database system, which consume different amounts of power resource for query processing. In this paper, we propose a new query processing method which dynamically chooses a query processing method among the three methods considering power consumption.

1. はじめに

近年、無線通信技術の発展にともない、放送型通信を用いて情報を配信する放送型情報システムが注目されている。放送型情報システムでは、サーバはクライアントへの広い帯域幅を利用して各種のデータを周期的に放送し、クライアントは必要なデータのみを選択して取得する。放送型情報システムでは、クライアント数が増加してもデータ配信のコストがほとんど変わらないため、クライアント数が多い場合に通信品質を落とさず情報配信ができ、さらに、データアクセスのスループット向上が期待できる。

^{*} 学生会員 大阪大学大学院情報科学研究科マルチメディア工学専攻 kitajima.shinya@ist.osaka-u.ac.jp

[†] 正会員 大阪大学サイバーメディアセンターサイバーコミュニティ研究部門 tsutomu@cmc.osaka-u.ac.jp

[‡] 正会員 大阪大学大学院情報科学研究科マルチメディア工学専攻 [hara,nishio@ist.osaka-u.ac.jp](mailto:{hara,nishio}@ist.osaka-u.ac.jp)

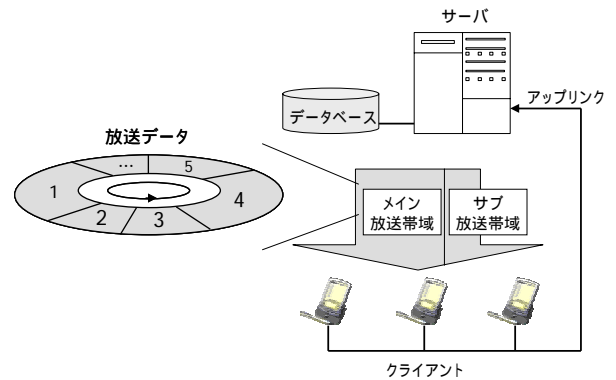


図1 放送型データベースシステム
Fig.1 Broadcast database system.

これまでに、放送型情報システムの性能向上を目的とし、多くの研究が行われている[1, 2, 3, 4]。これらの研究では、放送データを単なるデータアイテムとして扱っており、具体的な放送内容やデータ形式に基づいてシステムの効率化を行っているものは少ない。

そこで本研究では、サーバがリレーショナルデータベースの内容を繰り返し放送し、ユーザが放送されるデータベースに対して問合せを発行する環境を想定する。このようなシステムを放送型データベースシステムと呼ぶ。放送型データベースシステムにおける問合せ処理方式としては、オンデマンド型方式、クライアント型方式、協調型方式[5]の 3 方式が考えられる。これらの方式は、問合せ発生間隔や問合せ結果サイズ等の環境の変化に応じてその性能に優劣が生じるが、システム環境は常に変化し続けるため、静的に最適な方式を選択することは困難である。これまでに筆者らは、文献[6]において、クライアントからの問合せがサーバに到着した時点で、3 方式のうち応答時間が最も短い方式を選択する問合せ処理手法を提案した。しかし、この手法では問合せ処理にかかる消費電力を考慮していないため、利用できる電力に限られるモバイル環境では、電力の少ない端末の生存時間が短くなってしまおうという問題があった。

そこで本論文では、文献[6]における手法をもとに、モバイル端末の電力残量を考慮して問合せ処理方式を選択する手法を提案する。さらに、シミュレーション評価により、提案手法が従来手法と比べて、電力の少ない端末の生存時間を向上できることを示す。

以下 2 章で放送型データベースシステムについて述べる。3 章で提案手法について説明し、4 章では提案手法の性能評価を行う。最後に 5 章で本論文のまとめと今後の課題について述べる。

2. 放送型データベースシステム

本研究では、図 1 に示すように、放送型情報システムにおいてサーバがリレーショナルデータベースの内容を放送し、ユーザ(クライアント)が問合せを行う放送型データベースシステムを想定する。放送型データベースシステムは、以下に示す要素から構成される。

サーバ:サーバは、リレーショナルデータベースの内容を周期的に放送する。また、クライアントからの要求に応じて、問合せ処理を実行する。

クライアント:放送を受信するクライアントとして、記憶領域、電力資源、処理能力の乏しいモバイル端末を想定する。

ダウンリンク：サーバからクライアントへの放送帯域は、2つの帯域に分割されているものとする。サーバは、広帯域のメイン放送帯域を用いてデータベースの内容を繰り返し放送し、狭帯域のサブ放送帯域を用いてそれ以外のデータを放送する。

アップリンク：クライアントからサーバへの狭帯域の通信チャンネルが存在する。クライアントは、このアップリンクを用いて問合せをサーバに送信する。

2.1 想定環境

本研究では、街中で不特定多数のユーザに周辺情報を配信するといったアプリケーションを想定している。その一例として、ショッピングセンターにおける情報サービスが挙げられる。このサービスでは、サーバがショッピングセンター内の広告情報や店舗情報、また店舗で扱っている商品情報を含むデータベースを放送し、ユーザはモバイル端末を持ち歩きながら放送される情報を受信し利用する。

ユーザは、「商品Aの画像とその商品を扱っている店舗の地図が欲しい」といった情報検索を行いたい場合には、サーバに対して問合せを発行する。問合せの応答時間は短いほど好ましいが、ユーザは発行した問合せの応答時間を知ることができないため、ユーザが自らで問合せを処理すべきか、サーバが問合せ処理を行った結果を待つべきかの判断はサーバが行う。ユーザは、ショッピングをしながら欲しい商品を検索するため、数分程度の応答時間なら許容でき、問合せ処理にはリアルタイム性を要求しないものとする。ただし、ユーザは各問合せにデッドラインを設定でき、デッドラインの時間内にユーザが問合せの結果を得られない場合は、その問合せは失敗となる。ユーザは一定時間ショッピングセンターに滞在するが、端末の電力には限りがあり、端末の電力を使い果たしたユーザはサービスを受けられなくなる。

2.2 問合せ処理方式

放送型データベースシステムにおいて、クライアントによる問合せを処理する方式として、以下の3方式がある。

オンデマンド型方式：クライアントがアップリンクを利用して問合せをサーバに送信し、サーバが問合せ処理を行った後でサブ放送帯域を用いて問合せ結果をクライアントに配信する。オンデマンド型方式では、問合せ処理のすべてをサーバが実行し、クライアントは放送される結果を受け取るだけでよい。クライアントは問合せを処理するための記憶領域を必要とせず、問合せ結果を得るために必要な電力も小さい。また、発生する問合せ数が少ない場合、問合せ結果が放送されるまでの待ち時間が短く、クライアントはすぐに結果を取得できる。しかし、問合せが頻繁に起こる場合や問合せ結果のサイズが大きい場合にサブ放送帯域が枯渇するため、応答時間が長くなる可能性がある。

クライアント型方式：クライアントは問合せに関係するすべてのテーブルを自身の記憶領域にいったん蓄え、必要なすべてのデータが揃ってから、自ら問合せ処理を行う。クライアント型方式では、クライアント上で問合せ処理が完結するため、クライアント数が増加しても、1放送周期以内に問合せに関係する必要なすべてのデータを蓄積し、問合せ結果を得ることができる。また、アップリンクを使用しないため、アップリンクを用意できない環境でも動作する。しかし、端末のディスク容量による制約から問合せが処理できない場合がある、クライアントに大きな計算負荷がかかる、問合せ処理に大きな電力が必要になるといった問題がある。

協調型方式：クライアントは、アップリンクを利用して問

合せをサーバに送信する。問合せを受け取ったサーバは、問合せを処理し、問合せ結果に含まれるタプルに処理用の識別子を付加するとともに、クライアントがデータを処理するためのルールを作成し、サブ放送帯域を用いてクライアントに送信する。クライアントは、自分宛に送信された処理ルールをもとに、問合せ結果の作成に必要なタプルの放送開始時刻と放送終了時刻を把握し、メイン放送帯域を用いて放送されるデータベースのうち、識別子を参照して必要なデータのみを蓄積し、問合せ結果を再現する[5]。協調型方式では、クライアントは問合せ結果の作成に必要なデータのみを蓄積するため、クライアント型方式に比べてクライアントのディスク使用量を小さくでき、消費電力も小さくなる。また、処理ルールは一般に非常に小さなサイズであることから、オンデマンド型方式に比べてサブ放送帯域の占有時間を短くできる。しかし、各タプルにあらかじめ識別子領域を用意する必要があるため、放送周期が若干長くなる。また、タプルに付加された識別子は、クライアントが問合せ結果を作成し終えるまで解放されない。識別子の最大数はあらかじめ決まっているので、問合せが頻繁に起こる場合には識別子が不足し、問合せの成功率が下がってしまう。

2.3 LRT方式

2.2節で述べたオンデマンド型方式、クライアント型方式、協調型方式をそれぞれ単独で用いた場合、問合せ発生間隔や問合せ結果サイズなどのシステム環境に応じて、その性能に優劣が生じる。そこで、LRT (Least Response Time) 方式[6]では、クライアントからの問合せがサーバに到着すると、サーバはオンデマンド型方式、クライアント型方式、協調型方式の各方式を選択した場合の応答時間をそれぞれ計算し、応答時間が最も短い問合せ処理方式を選択する。

3. ELEC方式

2.3節で述べたLRT方式は問合せ処理にかかる消費電力を考慮していないため、電力の少ない端末の生存時間が短くなってしまふという問題があった。そこで本章では、端末の電力残量を考慮して問合せ処理方式を選択し、電力の少ない端末の生存時間を向上する手法を提案する。

3.1 概要

ELEC (Extended LRT considering Energy Consumption) 方式では、サーバはLRT方式に従って問合せ処理方式を選択するが、問合せを発行した端末の電力残量が閾値 TH_{on} 以下ならば、電力残量が少ないと判断し、消費電力の少ないオンデマンド型方式を優先的に選択する。一方、端末の電力残量がもう1つの閾値 TH_{cl} 以上ならば、電力残量が多いと判断し、消費電力の多いクライアント型方式を優先的に選択する。ここで、 TH_{on} 、 TH_{cl} は、 $0 < TH_{on} < TH_{cl} < 1$ なる関係を満たす範囲でのみ変化させるものとし、この関係を満たさなくなる場合には、値の増減は行わない。

x 個前にサーバに到着した問合せ Q_x を発行した端末の電力残量を P_x 、過去 q 個の問合せを発行した端末の電力残量の最大値を $P_{max}(q) (= \max(P_1, P_2, \dots, P_q))$ 、新たに問合せを発行した端末の電力残量を P_{new} とすると、問合せ処理方式選択の流れは以下のようになる。

1. LRT方式に従って問合せ処理方式を選択
2. $P_{new} < P_{max}(q) \times TH_{on}$ ならばオンデマンド型方式を選択
3. $P_{new} > P_{max}(q) \times TH_{cl}$ ならばクライアント型方式を選択

3.2 閾値の計算アルゴリズム

最適な TH_{on} 、 TH_{cl} はシステム状況によって異なるため、

状況に応じて動的に変化させる。計算の際には、問合せを発行した端末の電力残量が閾値 TH_{on} 以下の問合せのうち、実際にオンデマンド型方式を選択した問合せの割合を、問合せ数 q を用い、 $Q_i (i = 1, 2, \dots, q)$ と $Q_j^{old} (j = q+1, q+2, \dots, 2q)$ 、つまり、現在と過去の割合について比較する。これは、オンデマンド型方式を選択してもデッドラインを越えない端末の割合が、システム状況と密接に関係しているためである。

Q のうち、 $P_i < P_{max}(q) \times TH_{on}$ を満たす問合せ数を L_{max} 個、そのうちオンデマンド型方式が選択された問合せ数を L_{on} 個とする。また、 Q_j^{old} に対し、 $P_{max}^{old}(q) (= \max(P_{q+1}, P_{q+2}, \dots, P_{2q}))$ 、 TH_{on} を TH_{on}^{old} 、 TH_{cl} を TH_{cl}^{old} 、 $P_j < P_{max}^{old}(q) \times TH_{on}^{old}$ を満たす問合せ数を L_{max}^{old} 個、そのうちオンデマンド型方式が選択された問合せ数を L_{on}^{old} 個とする。

閾値の変化量は、 TH_{on} を増加させるとき、 TH_{cl} を減少させるときは W_+ 、 TH_{on} を減少させるとき、 TH_{cl} を増加させるときは W_- を用い、オンデマンド型方式を選択できる問合せには限りがあることから TH_{on} を優先的に変更する。

閾値の計算アルゴリズムを以下に示す。サーバは以下の処理の 1 ステップを、問合せが q 個到着することに実行する。初期状態は(1)とし、 TH_{on} の初期値は 0、 TH_{cl} の初期値は 1 とする。

- (1) $\frac{L_{on}}{L_{max}} \geq \frac{L_{on}^{old}}{L_{max}^{old}}$ である限りは、 $TH_{on} = TH_{on}^{old} + W_+$ とする。
条件を満たさなくなれば(2)に移る。
- (2) $\frac{L_{cl}}{L_{max}} \geq \frac{L_{cl}^{old}}{L_{max}^{old}}$ である限りは、 $TH_{cl} = TH_{cl}^{old} - W_-$ とする。
条件を満たさなくなれば(3)に移る。
- (3) $\frac{L_{on}}{L_{max}} < \frac{L_{on}^{old}}{L_{max}^{old}}$ である限りは、 $TH_{on} = TH_{on}^{old} - W_-$ とする。
条件を満たさなくなれば(4)に移る。
- (4) $\frac{L_{cl}}{L_{max}} < \frac{L_{cl}^{old}}{L_{max}^{old}}$ である限りは、 $TH_{cl} = TH_{cl}^{old} + W_+$ とする。
条件を満たさなくなれば(1)に移る。

ただし、 $L_{max} = 0$ 、または $\frac{L_{on}}{L_{max}} = 1$ ならば、 TH_{on} に W_+ を加えるとともに、 TH_{cl} から W_- を引く。

4. 評価

本章では、次に示す 3 つの評価基準を用いて、ELEC 方式の有効性をシミュレーション実験の結果から検証する。

- **問合せ成功率**：発生した全問合せのうち、クライアントが問合せ結果を受け取れたものの割合。
- **平均応答時間**：問合せが成功した場合における、クライアントが問合せを発行してから、問合せ結果を得るまでの平均時間。ただし、クライアントが問合せをサーバへ送信するのにかかる時間、サーバにおけるデータ処理にかかる時間は、十分に小さいため無視する。
- **平均生存時間**：クライアントが到着してから、予定滞在時間が経過した、もしくは端末の電力残量が 0 になったことにより退出するまでの平均時間。ただし、端末は問合せに関わる処理でのみ電力を消費するものとする。

4.1 評価モデル

本評価では、2.1 節で示したショッピングセンターにおけ

表1 問合せ成功率と平均応答時間

Table.1 Parameter configuration.

パラメータ名	値
シミュレーション時間[秒]	36000
クライアントの予定滞在時間[秒]	7200
1クライアント当たりの問合せ発生間隔[秒]	300
応答時間のデッドライン[秒]	80
ジャンル数[個]	10
ジャンル内ショップ数[個]	5
ショップ内商品数[個]	200
1タプルのサイズ[KByte]	10
1タプルに付加可能な最大識別子数[個]	200
メイン放送帯域[Mbps]	10
サブ放送帯域[Mbps]	1
処理ルールのサイズ[KByte]	1
タプル利用率の平均	0.003
タプル利用率の標準偏差	0.001

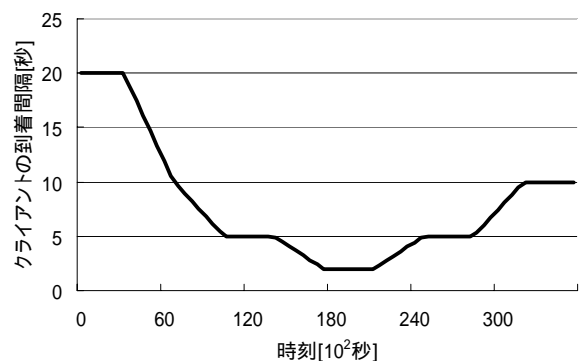


図2 クライアントの到着間隔の変化

Fig.2 Change of user arrival frequency.

る情報サービスをアプリケーション例として想定し、データベースは店舗テーブルと商品テーブルをもつものとした。表1に、評価で用いるパラメータとその値を示す。タプル利用率は、1ジャンル分のテーブルの全タプルに対する、問合せ結果に含まれるタプルの割合を表す。クライアントは、指数分布に従った間隔でショッピングセンターに到着するものとし、到着間隔の平均は図1のように変化させた。各クライアントはサービスを受けられる端末を1台ずつ所持しており、各端末の初期電力量は100単位エネルギーから1000単位エネルギーの一様分布、各端末のディスク容量は1MBから100MBの一様分布とした。

評価の際には、簡単のためすべての端末の処理速度は同じであるとし、データの書き込み、読み込み速度は、それぞれ10MB/s、15MB/sとした。また、文献[7]を参考に、問合せ処理に必要な電力は時間 t に比例するものとし、CPUにかかる電力 $E_C^h = 2t$ (高負荷時)、 $E_C^l = 0.5t$ (低負荷時)、無線LANにかかる電力 $E_W = t$ 、I/Oにかかる電力 $E_I = t$ 、 $E_O = t$ の和で表せるものとした。問合せ処理に必要な電力とは、問合せを発生した後に、必要なテーブルの受信やタプルの結合処理に必要な電力を指し、問合せの送信、問合せ結果の表示に必要な電力は、方式による差がないため考慮しない。

4.2 シミュレーション結果

図2のようにクライアントの到着間隔を変化させたときの、LRT方式とELEC方式における問合せ成功率、平均応答時間を表2に示す。また、端末の初期電力量をもとに100単位エネルギーごとに分類した平均生存時間を図3に示す。ELEC方

表2 問合せ成功率と平均応答時間
Table.2 Success rate and response time.

手法	問合せ成功率	平均応答時間
LRT方式	94.3%	44.3秒
ELEC方式	92.9%	45.8秒

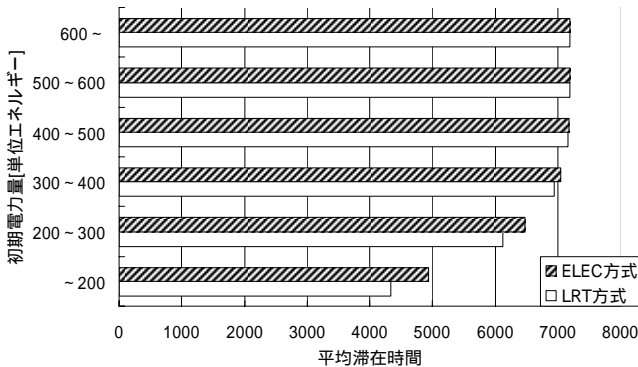


図3 平均生存時間
Fig.3 Average lifetime.

式における q は 50, W_1 は 0.002, W_2 は 0.005 とした。

図表から, ELEC方式ではLRT方式と比べ, 問合せ成功率, 平均応答時間ともわずかに性能が悪くなっているが, 初期電力量の少ない端末の平均生存時間が大きく改善していることがわかる。LRT方式では, 初期電力量が多い端末の大半は電力を使い果たすことなく予定滞在時間経過後に退出しているが, ELEC方式では, 電力残量の多い端末に優先してクライアント型方式を, 電力残量が少ない端末に優先してオンデマンド型方式を割り当てることにより, 電力残量の少ない端末の平均生存時間を向上している。

5. まとめ

本論文では, 放送型データベースシステムにおいて, モバイル端末の電力消費を考慮して, 動的に問合せ処理方式を選択する手法を提案した。提案手法では, 残存電力が多い端末には優先的に消費電力の多いクライアント型方式を割り当て, 残存電力が少ない端末には優先的に消費電力の少ないオンデマンド型方式を割り当てる。また, 提案手法の有効性を検証するために, 問合せ成功率と平均応答時間についてシミュレーション評価を行った。シミュレーション評価の結果から, 提案手法が従来手法と比べて, 電力の少ない端末の生存時間を向上できることを確認した。

今後は, 端末の処理能力を考慮した問合せ処理方式選択手法について検討する予定である。

[謝辞]

本研究の一部は, 文部科学省 21 世紀 COE プログラム「ネットワーク共生環境を築く情報技術の創出」, および文部科学省特定領域研究(16016260), 基盤研究(A)(17200006)の研究助成によるものである。ここに記して謝意を表す。

[文献]

[1] Acharya, S., Alonso, R., Franklin, M., and Zdonik, S.: "Broadcast Disks: Data Management for Asymmetric Communication Environments", Proceedings of ACM SIGMOD, pp.199-210 (1995).

[2] Acharya, S., Franklin, M., and Zdonik, S.: "Balancing Push and Pull for Data Broadcast", Proceedings of ACM SIGMOD, pp.183-194 (1997).
 [3] Aksoy, D., Franklin, M., and Zdonik, S.: "Data Staging for On-demand Broadcast", Proceedings of VLDB Conference, pp.571-580 (2001).
 [4] 箱根 聡, 田辺雅則, 石川裕治, 井上 潮: "放送型通信とオンデマンド型通信を統合した情報提供システム", 情報処理学会論文誌, Vol.40, No.10, pp.3772-3781 (1999).
 [5] 加下雅一, 寺田 努, 原 隆浩, 塚本昌彦, 西尾章治郎: "データベース放送システムのためのサーバと移動型クライアントによる協調型問合せ処理方式", 情報処理学会論文誌: データベース, Vol.44, No.SIG8 (TOD 18), pp.92-104 (2003).
 [6] 北島信哉, 寺田 努, 原 隆浩, 西尾章治郎: "放送型データベースシステムにおけるデッドラインを考慮した問合せ処理方式", 電子情報通信学会和文論文誌 D, Vol.J89-D, No.2, pp.151-162 (2006).
 [7] Mahesri, A. and Vardhan, V.: "Power Consumption Breakdown on Modern Laptop", Proceedings of International Workshop on Power-Aware Computing Systems (PACS) (2004).

北島 信哉 Shinya KITAJIMA

2005 年大阪大学工学部電子情報エネルギー工学科卒業。2006 年同大学大学院情報科学研究科博士前期課程修了。同年, 同大学大学院情報科学研究科博士後期課程に入学し, 現在に至る。放送型データベースシステムの研究に興味をもつ。日本データベース学会の学生会員。

寺田 努 Tsutomu TERADA

1999 年大阪大学大学院工学研究科博士前期課程修了。2000 年, 博士後期課程退学。同年より大阪大学サイバーメディアセンター助手。2005 年より同講師, 現在に至る。2002 年より大阪大学大学院情報科学研究科マルチメディア工学専攻助手, 2005 年より同講師を併任。工学博士。アクティブデータベース, ウェアラブルコンピューティング, ユビキタスコンピューティングの研究に従事。IEEE, 情報処理学会, 日本データベース学会の各会員。

原 隆浩 Takahiro HARA

1997 年大阪大学大学院工学研究科博士前期課程修了。同年, 博士後期課程中退学, 同大学大学院工学研究科情報システム工学専攻助手, 同大学大学院情報科学研究科マルチメディア工学専攻助手を経て, 2004 年より同大学大学院情報科学研究科マルチメディア工学専攻助教授となり, 現在に至る。工学博士。データベースシステム, モバイルコンピューティングなどの研究に従事。IEEE, ACM, 電子情報通信学会, 情報処理学会, 日本データベース学会の各会員。

西尾 章治郎 Shojiro NISHIO

1980 年京都大学大学院工学研究科博士後期課程修了。工学博士。京都大学工学部助手, 大阪大学基礎工学部および情報処理教育センター助教授, 大阪大学大学院工学研究科教授を経て, 2002 年より同大学大学院情報科学研究科教授となり, 現在に至る。2000 年より大阪大学サイバーメディアセンター長, 2003 年より大阪大学大学院情報科学研究科長を併任。データベース, マルチメディアシステムの研究に従事。現在, Data & Knowledge Engineering 等の論文誌編集委員, 本学会理事, 電子情報通信学会, 情報処理学会の各フェローを含め, ACM, IEEE など 8 学会の会員。