修士論文

分析問合せ処理の消費エネルギーのモデル化に
関する研究

A Study on Energy Consumption Models
of Analytics Query Processing

東京大学大学院 情報理工学系研究科
電子情報学専攻

48-176436　羅 博明

指導教員　　吉永 直樹 准教授

平成 31 年 1 月 30 日 提出

本論文は東京大学大学院情報理工学系研究科に修士号授与の要件として提出した修士論文である.

内容梗概

データセンタにおける消費電力の増大は著しく，その中核であるデータベースシステムの省電力化は重要な課題である．データベースシステムの省電力については，先行研究においてプロセッサ動作モード制御が検討されている．しかし，従来の研究では特定の問合せを対象とした計測に留まっている．プロセッサ動作モードの影響を考慮した問合せ処理の電力・性能特性は，省電力化において不可欠のものであるが，未だに明らかになっていない部分が多い．本研究では，データベースの主要なワークロードの一つである分析問合せ処理を対象として，モデルベースのアプローチによりこの課題に取り組む．つまり，プロセッサ動作モードを考慮した分析問合せ処理の，問合せ処理スループットに基づいた消費エネルギーモデル構築を行う．モデル構築の前段階として，プロセッサ動作モードが分析問合せ処理に与える影響に関して議論を行い，計測実験により定性的な特性を明らかにする．そして得られた知見に基づいた消費エネルギーのモデル化手法を提案する．評価実験により 1.7 % 以下の誤差で実測値とフィットすることを示し，モデルにより分析問合せ処理の電力・性能特性を捉えることが可能になることを示す．

目次

図目次

表目次

第 1 章

序論

1.1 データベースシステムにおける省電力化

近年，様々な応用分野において大規模データの活用が注目を集めているが，同時に IT 機器による消費電力量が増え続けることが懸念されている．所謂ビッグデータブームや機械学習等に牽引される形で，日々増え続けるデータの格納・処理・分析のために IT 資源は増加する傾向にあり，それによりデータセンタにおける消費電力量も増加の一途を辿っている．米国 EPA の報告では，米国におけるデータセンタの消費電力量は，2013 年には 910 億 kWh であったが，予測では 2020 年には 1400 億 kWh にまで増加するとされている [1]．同様な傾向は欧州におけるデータセンタの消費電力量にも見られ，2007 年では 560 億 kWh だったものが，2020 年には 1040 億 kWh に達すると見込まれている [2]．こうしたデータセンタにおける消費電力量拡大を受けて，大規模データ分析基盤での省電力化の要求は高まっている．

データセンタにおける省電力化のためには，その中核を担うソフトウェアであるデータベースシステムの省電力化が重要な課題として挙げられる．データベースシステムは，大規模データの効率的かつ安全性の高い保管と処理を行うことを目的として現在も利用され続けている．そして，データベースシステム上で処理されるアプリケーションの中の一つである意思決定システムに利用されている分析問合せ処理での省電力化も重要となっている．分析問合せ処理では大量の入出力を伴う演算処理が駆動されるため，プロセッサとストレージの両側面に着目した省電力制御が必要とされている．

1.2　本研究の目的と貢献

本研究は，分析問合せ処理における省電力化に役立てるため，プロセッサ動作モードを考慮した消費エネルギーモデルの提案を行う．プロセッサ動作モードに関しては，近年のプロセッサが多くのコアを備える傾向にあり消費電力への寄与も大きくなっていることから議論と検証の余地がある．一般的に分析問合せ処理における省電力化を目的とした研究は，問合せ最適化に消費電力や消費エネルギーの指標を取り入れることが主流であるが，プロセッサ動作モードを考慮したモデル化を行っているものはなく部分的なモデル化に留まっている．また，プロセッサ動作モード制御による分析問合せ処理の省電力化の研究がいくつか存在するが，これらの研究は分析問合せ処理の特性を捉えたモデル作成には至っていない．本研究の貢献は以下が挙げられる．

- プロセッサ動作モードが分析問合せ処理に与える影響について議論し，実験を通じて省電力化の有効性と有効範囲を確認した．
- プロセッサ動作モードの影響をモデルとして表現するため，全表走査を伴う分析問合せ処理に対して資源律速に基づいた定量的なエネルギー解析モデルを作成し，実測値を用いたフィッティングによりその妥当性を示した．

1.3　本論文の構成

以降の本論文の構成は以下の通りである．

第 2 章　本研究の前提となるいくつかの基礎的な背景知識に関して説明する．

第 3 章　本研究と関連する，データベースシステムにおける省電力化に関する研究，特に分析問合せ処理における省電力化に関する研究に関して紹介する．

第 4 章　プロセッサ動作モード制御を分析問合せ処理に適用することによる効果に関して議論し，効果の検証のために行った実験とその結果を示す．

第 5 章　第 4 章の議論と結果を踏まえ，全表走査に基づく問合せを対象とした消費エネルギーモデルを提案し，モデルの妥当性を評価実験により確かめた．

第 6 章　本研究の総括と今後の課題に関して述べる.

第 2 章

分析問合せ処理とプロセッサ動作モード制御

本章では，本研究で扱う対象となる分析問合せ処理・プロセッサ動作モード制御に関してそれぞれ説明する．

2.1 分析問合せ処理

分析問合せ処理（Analytics Query Processing）とは，文字通りデータ分析を目的とした処理であり，データベース上のデータに対し関係演算や合算・平均等の操作を組み合わせることで複雑なデータ分析を実現する．分析問合せ処理ではデータベースへの書き込みは基本的には行われず，読み込み主体の処理となることが知られている．応用先として意思支援システムがあり，主にビジネスの場で現在も広く使われている．代表的な分析問合せのベンチマークとしては TPC-H [*1] が業界標準で用いられており，生成された顧客情報や商品情報，注文情報等を分析する問合せを提供する．この他にも TPC-H の代替となる TPC-DS [3] や TPC-H の派生である SSB [4] 等が存在している．

近年の主記憶容量の増大と低コスト下に伴いデータベースが主記憶装置に収まるユースケースも少なくないが，本研究では分析問合せ処理を行うシステムとして Hard Disk Drive（HDD）や Solid State Drive（SSD）のようなストレージ上にデータベースが存在するものを想定する．この理由としては，分析問合せ処理が対象とする問題の性質上，データベースサイズが数百 GB 以上程度になることが多いためである．

[*1] http://www.tpc.org/tpch/

2.2　プロセッサ動作モード制御

昨今のプロセッサはその多くが Dynamic Voltage and Frequency Scaling（DVFS）と呼ばれるプロセッサ動作モード制御機構を備えており，ある動作モードに設定するとその動作モードに対応した動作電圧と動作周波数に切り替わる．例えば，Intel 社の Xeon プロセッサ上では Intel SpeedStep Technology と呼ばれる DVFS 機能を提供しており，特定のレジスタに所望の値を書き込むことで動作モードが切り替わる [5]．プロセッサ動作モード制御は消費電力の低減に有効であることが知られている．

プロセッサ動作モード制御が消費電力に与える影響を考える上では，プロセッサの動作状態について考える必要がある．プロセッサの動作状態はアイドル状態と稼働状態に大別される．このうち，アイドル状態における消費電力は動作モードには依存せずほぼ一定であることが知られている．これは，最近のプロセッサでは C-State と呼ばれる状態に移行することで，クロックのゲーティングや供給電圧の制限を行う機能が備わっているためである．対して稼働状態における消費電力は動作周波数と動作電圧に依存するため，動作モード切り替えにより削減が可能である．プロセッサの根幹を支える技術である CMOS 回路の消費電力 P は動作周波数を f，動作電圧を V とすれば $P \propto f V^2$ の関係を持つので，稼動状態のプロセッサにおいてもこの関係が成り立つ．よって，DVFS 機能による動作モード切り替えを用いれば，動作モードを高性能なものに固定した時と比べて消費電力削減に繋がる．

しかし，動作モード切り替えは先述の通りプロセッサの処理性能を下げるものであるため，処理のスループットに対する影響が存在する．理論的にはプロセッサの処理性能は動作周波数 f に比例する．低周波数の動作モードにした時に消費電力の削減割合に比べて処理のスループットの低下割合の方が大きかった場合，消費エネルギーが増加する．そのため，プロセッサ動作モード制御がエネルギー効率向上に対して有効となるのは，プロセッサの性能低下が処理のスループットに与える影響が小さい時である．よって，マシン上で得られる情報を基に，プロセッサの性能低下が処理のスループットに与える影響が小さいと判断されるタイミングにおいて，より低周波数・低電圧となるような動作モードに切り替えることが，DVFS における基本的な方針である．

第 3 章

関連研究

3.1 データベースシステムにおける省電力化の流れ

データベースシステムは，従来スループット向上とレイテンシ低下を重点に置いた性能指向で開発されてきたが，エネルギー効率の観点を取り入れることの重要性が唱えられるようになってきた．特に，ソフトウェアレベルでの省電力化の議論は The Claremont Report において省電力性を考慮したデータベースシステムの開発の重要性が指摘されて以降本格化した [6]．ハードウェア資源の投入が一定程度進むとエネルギー効率が下がるケースや，性能向上のためのソフトウェア的工夫がエネルギー効率を下げるケースがあることを明らかにし，データベース分野における行き過ぎた性能追求の弊害を示した研究がある [7]．大きな省電力化効果が見込まれるものとして，問合せ最適化，CPU と IO のスケジューリング，データベース構築のデザインやデータ更新法が挙げられている [8]．データベースシステムの省電力化推進を目的として，エネルギーを評価指標に取り入れたベンチマークが開発されている他，従来のベンチマークの指標に消費電力・消費でネルギーを取り入れる動きがあった [9, 10]．

3.2 プロセッサ動作モード制御を利用した省電力化

プロセッサ動作モード制御は，データベース分野に限らずソフトウェアコンピューティング等において省電力化のために広く利用されている．現在普及している DVFS の制御法は，システムレベルによる制御がほとんどである．Linux が提供する Ondemand Governor は直近の CPU 利用率を基に動作モード切り替えの判断を行う [11]．CPU

利用率が低い時にはプロセッサを低速度にしても処理スループットへの影響は小さいと見込まれるため，低周波数の動作モード切り替えることで消費電力を下げる．また，CPU 利用率が高い場合でも DVFS の余地があることが示されており，中でも処理がプロセッサ律速かメモリ律速かに基づいて動作モードを切り替える手法が数多く研究されてきた．基本的には PMU（Performance Monitoring Unit）を通じて得たキャッシュやメモリに関する情報（キャッシュヒット率等）を活用するもの [12, 13] が中心だが，メモリの消費電力に基づいて判断する研究も存在する [14]．しかし，これらの研究はいずれもストレージを用いたデータベースシステムでの処理については考慮していない．

データベースシステムにおけるアプリケーションレベルでのプロセッサ動作モード制御による省電力効果は，トランザクション処理に適用されその有効性が示されている [15]．

3.3　分析問合せ処理の省電力化

分析問合せ処理の省電力化では，初期はハードウェア構成に関する議論がよく見られる [16, 17]．近年盛んに研究されているのが，省電力性を考慮した問合せ最適化である．これは，実行時間だけでなく消費電力・消費エネルギーもコストに取り入れて実行計画を選出しようというものである [18–22]．問合せ最適化に電力効率の指標を取り入れる議論は初期では 1990 年代前半に行われていたが [23]，ここ 10 年で特に多く取り組まれた．これらの研究は，問合せ最適化に必要な範囲で電力・エネルギーモデルを作成しているが，プロセッサ動作モードのような実行時システム構成まで考慮したモデル化は行うには至っていない．

そして，分析問合せ処理におけるプロセッサ動作モードに着目したものがいくつかある．プロセッサ動作モードだけでなくその他のシステム構成に関しても様々な組み合わせで問合せ処理を行い，最高周波数の動作モードに固定することがエネルギー効率の面で良い結論づけた研究が存在する [24]．しかし，それとは対照的に処理する問合せの特性によってエネルギー効率が最高となるような周波数の動作モードは変わると主張しており，問合せ毎に最適な周波数を記録することで対応することを提案した研究がある [25]．また，サーバの各コンポーネントの消費電力を詳細に取ることによる動作モードのフィードバック制御も提案されている [26]．本研究では，資源律速と

いう切り口で分析問合せ処理への影響を議論し，特性理解のための定量的なモデルを作成したという点でこれらの研究を発展させている．

第 4 章

プロセッサ動作モードが分析問合せ処理に与える影響

本章では，プロセッサ動作モード制御を分析問合せ処理に適用した時の効果に関して資源律速という観点を踏まえて議論し，実際の効果の検証のために行った実験とその結果を示す．

4.1 プロセッサ動作モードが分析問合せ処理に与える影響の議論

本節では，資源律速に着目してプロセッサ動作モード制御による影響を定性的に議論する．ここでいう資源律速とは，プロセッサの処理速度やストレージの IO 速度などの特定の資源が律速要因となることで処理全体が律速されている状態を指す．二次記憶装置を用いたシステムの場合，処理の律速要因としてはストレージの IO 性能とプロセッサの演算性能の 2 つが主になると考えられる．IO によるデータ取得速度がプロセッサによる演算処理速度よりも遅い場合には IO 律速となり，逆の場合には CPU 律速となる．

IO 律速である処理の最中は二次記憶装置によるデータ取得を待つ必要があるため，プロセッサには余裕がある状態であり CPU 利用率は一般的に低い．このような場合は，高周波数の動作モードと低周波数の動作モード間で処理スループットの差異は小さいと見込まれ，低周波数の動作モードにすることによるエネルギー効率向上が予想される．しかし，エネルギー効率向上の割合は CPU 利用率の数値に強く依存する．CPU 利用率が極端に低い処理においては，低周波数の動作モードであった場合でもほとんどの時間はアイドル状態にいるため電力削減効果は比較的小さい．この場合，実行時間はほぼ変わらないためエネルギー効率にはほとんど違いがない．したがって，効果

的な電力削減とエネルギー効率向上が見込めるのは，CPU 利用率が一定程度高い時であると予想される．

逆に CPU 律速である場合には，低周波数の動作モードでエネルギー効率は下がると予想される．これは，スループット低下の割合が動作周波数低下の割合と等しいのに対して，プロセッサの消費電力はシステム全体の消費電力の一部でしかないことから消費電力削減の割合が動作周波数低下の割合よりも低くなるためである．

CPU 利用率を u とした時に，プロセッサ動作モード制御が分析問合せ処理に与える影響としては次の三つのカテゴリのいずれかに分類することが可能である．

(i) エネルギー効率ほぼ一定: $u < u_{\text{low}}$ で，動作モードの制御によりエネルギー効率がほとんど変わらない処理

(ii) エネルギー効率向上: $u_{\text{low}} \leq u \leq u_{\text{high}}$ で，動作モードの制御によりエネルギー効率が向上する処理

(iii) エネルギー効率低下: $u_{\text{high}} < u$ で，動作モードの制御によりエネルギー効率が低下する処理

ここで，$u_{\text{low}}, u_{\text{high}}$ は動作周波数の可動域と設定粒度をはじめとしたシステム設定や，プロセッサやストレージといったコンポーネントの特性等のシステム環境に依存するパラメータである．また，問合せ処理は常にある単一の処理によって完結するわけではなく，いくつかの処理を経ることが考えられる．そのため，上の分類は問合せ中に存在する処理段階毎に適用することができる．

4.2　プロセッサ動作モードが分析問合せ処理に与える影響の測定実験

本節では，4.1 節で議論されたプロセッサ動作モード制御が分析問合せ処理に与える影響を検証するために行った，オープンソースのデータベース管理システム PostgreSQL [*2] と TPC-H ベンチマークデータセットを利用した計測実験に関して述べる．

[*2] https://www.postgresql.org/

4.2.1　測定環境

以降，測定に利用したマシンをサーバ A と呼称する．サーバ A のハードウェア構成を表 1 に示す．プロセッサは 4 コア CPU を備え，最大動作周波数は 2.8 GHz である．システム全体の消費電力の計測のために，ハードウェアの給電系統を高精度電力計 Yokogawa WT1800 の電力測定回路を経由させて接続することで，消費電力を 20 Hz で取得可能にした．

次にソフトウェア構成を述べる．OS カーネルとして Linux 3.10.0 を利用し，DVFS 機能には `cpufreq` と呼ばれるカーネルモジュールを利用した．`cpufreq` を利用することでユーザが指定した動作モードでプロセッサを動作させることが可能である．今回搭載されている Xeon E5-1603 v4 にて利用できる動作モードとしては 16 段階用意されており，それぞれが特定の動作周波数に対応している．（1.2 GHz〜1.9 GHz, 2.1 GHz〜2.8 GHz）．データベース管理システムにはオープンソースである PostgreSQL 9.6.3 を利用した．PostgreSQL のためにメモリに割り当てるキャッシュサイズ等は表 2 に示すように設定した．また，表に示していない設定値に関してはデフォルト値を利用した．分析問合せのデータセットとして dbgen 2.17.2 を利用することで，TPC-H のデータベースと問合せをスケールファクタ 100（最大のテーブルサイズが約 100 GB）で生成した．TPC-H の問合せは 22 種類あり，それぞれに固有のインデックスが振られている（Q1〜Q22）．dbgen に与える乱数を変更することで，生成される問合せのデータ選択範囲（SQL の `where` 節）が変化する．サーバ A に関するリアルタイム統計情報を

表 1　サーバ A のハードウェア構成．

プロセッサ	Intel Xeon E5-1603 v4 @ 2.8 GHz
メモリ	8GB DDR4 2133MHz × 4
OS 用ストレージ	SKhynix SC300 256GB
DB 用ストレージ	Seagate BarraCuda 2TB 7200rpm

表 2　サーバ A での測定実験における PostgreSQL のメモリ割当関連の設定．

`shared_buffers`	4096 MB
`temp_buffers`	2048 MB
`work_mem`	2048 MB

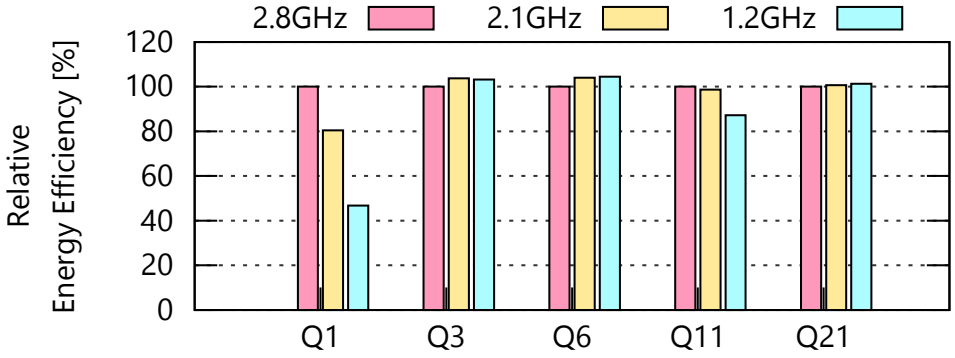
得るために，sysstat パッケージが提供する sar コマンドを通じて CPU 利用率の情報を，kernel が提供する diskstats を通じてストレージ IO の情報を，それぞれ 1 Hz で取得した．

ワークロードとしては，複数種類の分析問合せを対象に異なる動作周波数に対応する動作モードの下で処理を行った．具体的には，TPC-H に存在する 22 種類の問合せのうち 5 種類の特徴的な問合せ（Q1, Q3, Q6, Q11, Q21）を選出し，動作周波数を 2.8 GHz, 2.1 GHz, 1.2 GHz の 3 通りで処理させた際の影響を見た．また，サーバ A のプロセッサは 4 コアを備えていることから，データの選択範囲が異なる同一インデックスの問合せを多重度 4 で同時処理させる実験も行った．

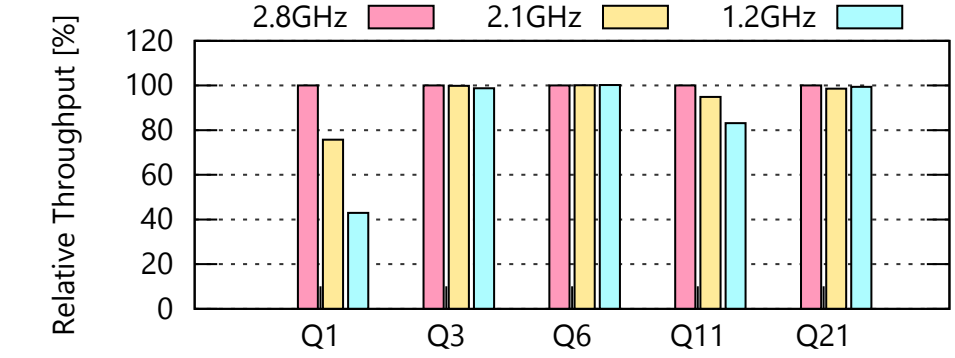
4.2.2　多重度 1 の分析問合せ処理の結果

多重度 1 で 5 種類の分析問合せ処理をさせた時のエネルギー効率・スループット・平均消費電力を図 1 に示す．高周波数の動作モードに比べ低周波数の動作モードでエネルギー効率が上がる問合せと下がる問合せが存在することが確認できる．このように問合せ毎に傾向が分かれるのは，既に述べたようにスループットの低下割合と消費電力の低下割合の大小が処理の性質毎に異なるためである．

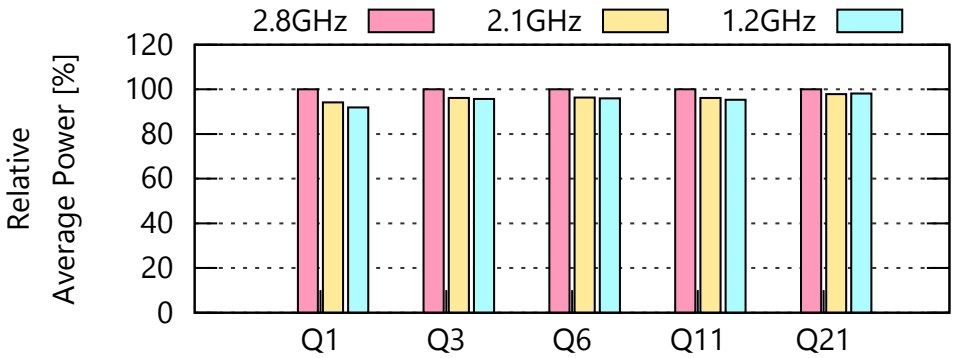
動作周波数を下げることでスループットは概ね下がる傾向にあるが，その低下割合は問合せによって大きく異なる．Q1 と Q11 は顕著なスループット低下が見られ，特に Q1 では大幅な低下が確認できる．Q1 は LINEITEM 表を全表走査して集約演算を行う問合せであり，実験環境においては CPU 律速である．つまり Q1 はカテゴリ (iii) エネルギー効率低下に属する処理のみだと言える．実際，Q1 はスループットの低下割合が動作周波数の低下割合にほぼ等しくなっている．これは，図 2 に示す Q1 の経時変化からも，どの動作モードにおいても処理全体を通して CPU 利用率は 100 % に近い値になっていることから読み取れる．Q11 の場合は，Q1 と比べるとスループットの低下割合が相対的に小さい．　図 3 に Q11 の経時変化を示すが，処理開始から 70 秒程度までは IO 律速となっており，それ以降は CPU 律速となっている．よって，Q11 はカテゴリ (iii) エネルギー効率低下とカテゴリ (ii) エネルギー効率向上に属する処理に分けられる．今回の実験条件では，カテゴリ (iii) 中の影響がカテゴリ (ii) に比べて大きかったために，スループットの低下割合が消費電力の低下割合よりも大きくなったと



(a) エネルギー効率.



(b) スループット.



(c) 平均消費電力.

図 1　サーバ A 上で多重度 1 の TPC-H 問合せ処理を行った時のエネルギー効率・スループット・平均消費電力（2.8 GHz からの相対値）.

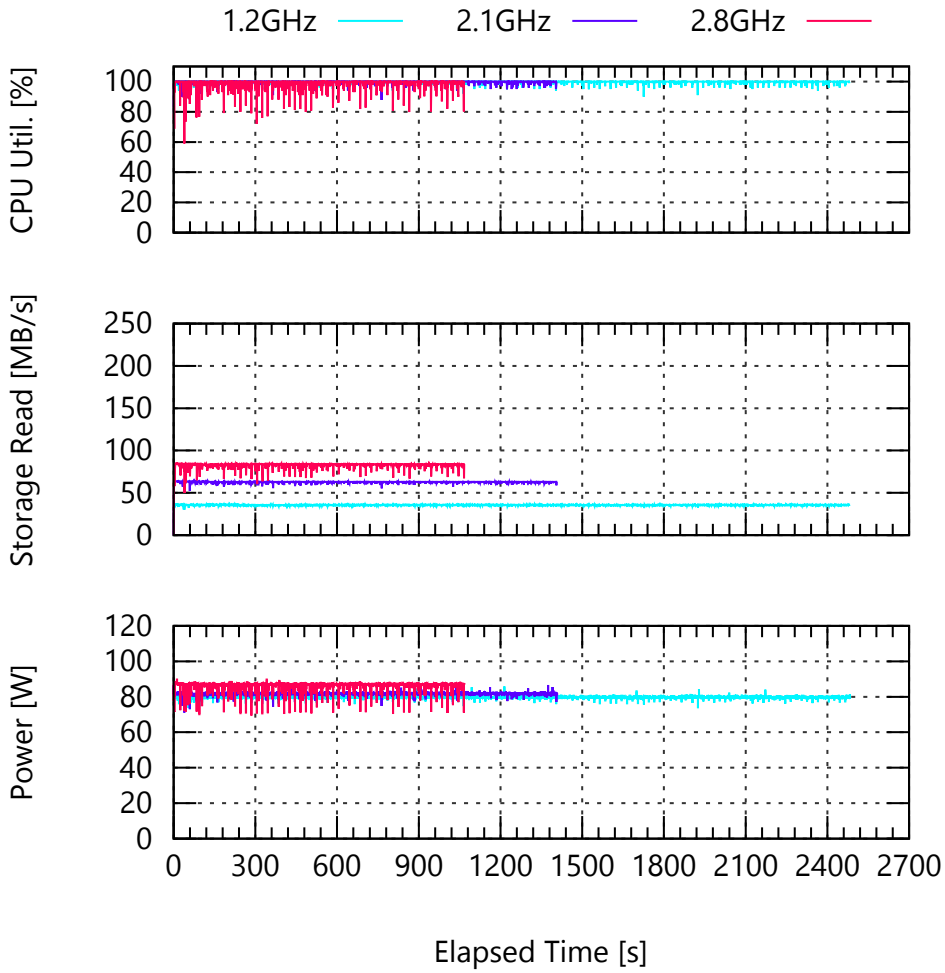


図 2　多重度 1 で Q1 を処理した時の CPU 利用率・ストレージ IO 速度・消費電力の経時変化.

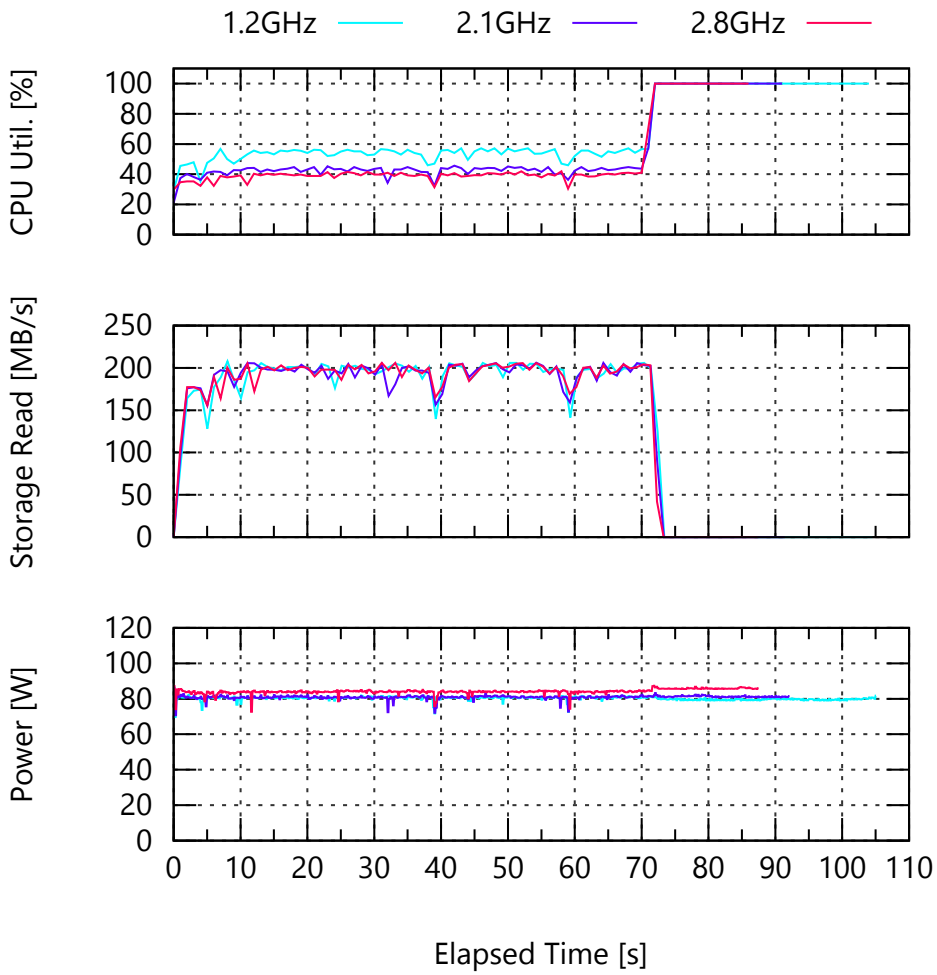


図 3　多重度 1 で Q11 を処理した時の CPU 利用率・ストレージ IO 速度・消費電力の経時変化.

解釈できる．このように動作モードの違いによりスループットが大きく影響を受けるような問合せでは，低周波数の動作モードにおけるエネルギー効率低下の傾向が見られる．2.8 GHz のモードに対する 1.2 GHz のモードのエネルギー効率の減少割合としては，Q1 が 53.2 %，Q11 が 12.8 % の減少となっている．

それに対して，スループットの低下割合が小さいのが Q3, Q6, Q21 である．それぞれの経時変化を図 4, 図 5, 図 6 に示す．　ここで，Q3 と Q6 はテーブルアクセス方法とし

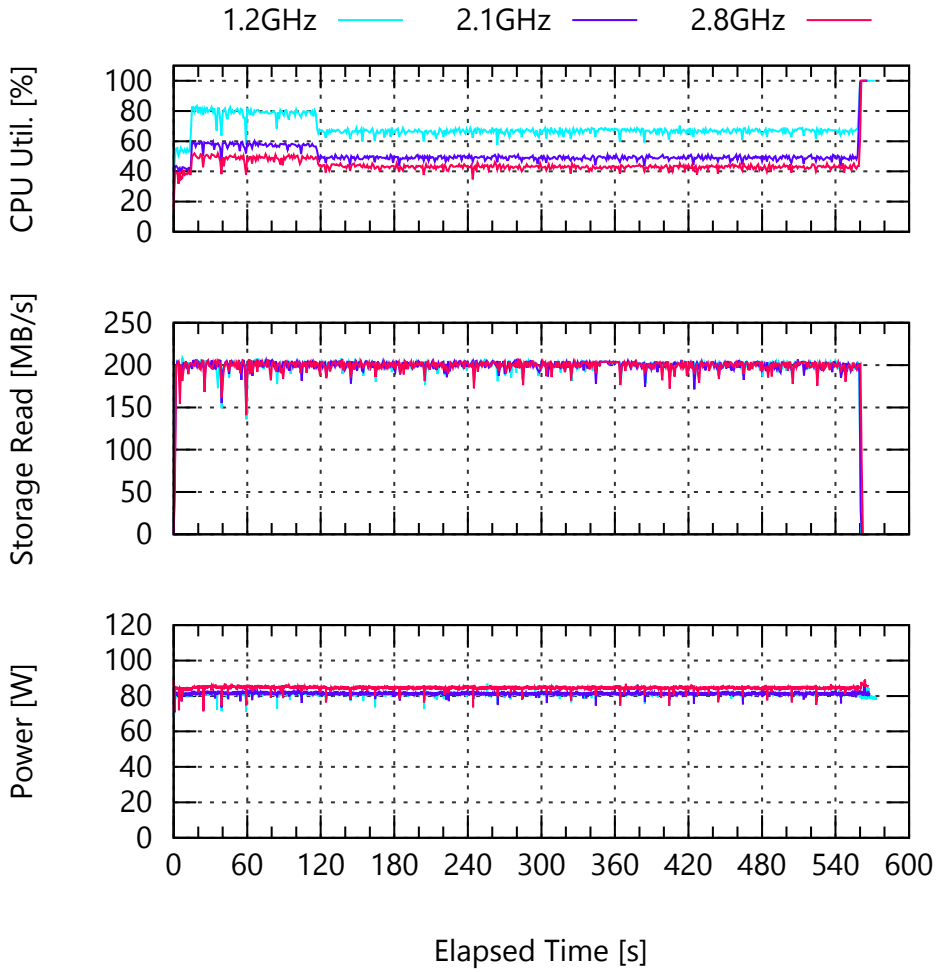


図 4　多重度 1 で Q3 を処理した時の CPU 利用率・ストレージ IO 速度・消費電力の経時変化．

てはともに全表走査のみである．それと同時に，Q1 のような複数の集約演算を行う問合せに比べると単位時間あたりのプロセッサの演算処理量が相対的に少ない．そのため全体を通じて IO 律速の処理が続くが，CPU 利用率は 40 %〜60 % 程度になっており，これらはカテゴリ (ii) エネルギー効率向上の処理のみだと言える．Q3 は最後に CPU 利用率が 100 % に達しておりカテゴリ (iii) エネルギー効率低下の処理が存在しているが，実行時間全体の 2 % 程度となっており影響が小さい（図 4）．それと同時に，低周

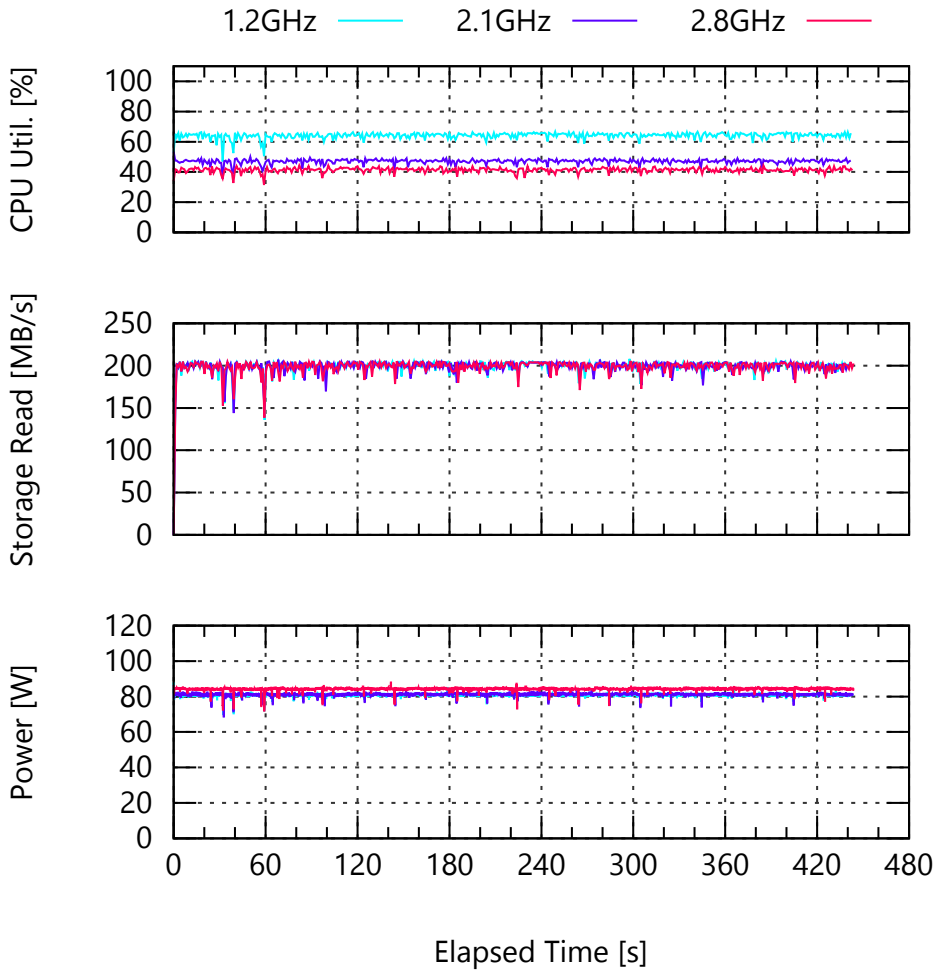


図 5　多重度 1 で Q6 を処理した時の CPU 利用率・ストレージ IO 速度・消費電力の経時変化．

波数の動作モードでもストレージ IO 速度が下がっておらず，スループットへの影響が無視できる程度に小さい．このような場合，消費電力の低下割合がスループットの低下割合を上回るので，エネルギー効率が向上する．2.8 GHz のモードに対する 1.2 GHz のモードのエネルギー効率の増加割合としては，Q3 が 3.2 %，Q6 が 4.4 % の向上となっている．対して，Q21 に関しては処理開始から 1200 秒近くまでの全表操作を伴う処理の間は CPU 利用率が 0 %〜80 % で揺れているが，その後はランダムアクセスが続いて

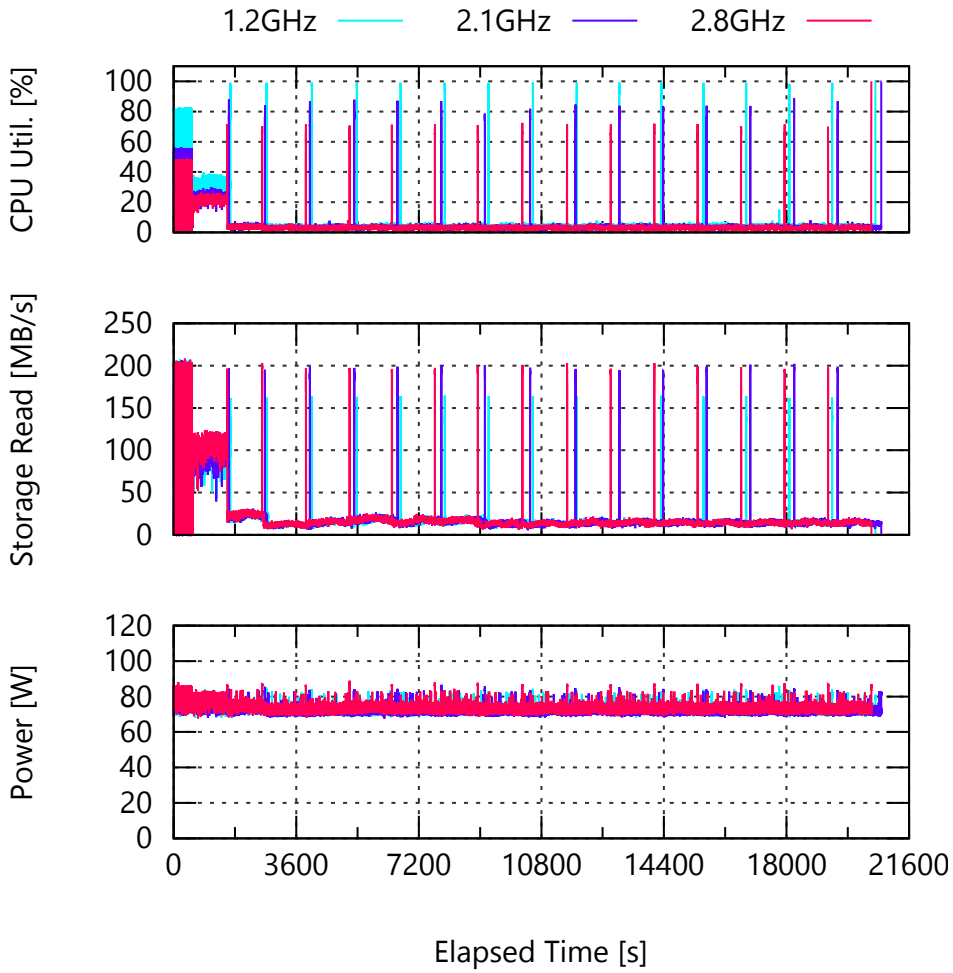


図 6　多重度 1 で Q21 を処理した時の CPU 利用率・ストレージ IO 速度・消費電力の経時変化．

おり，CPU 利用率が 10 % 以下の範囲で推移している．つまりプロセッサがアイドル状態にいる時間がこれまで見てきた問合せの中では長い．このようなカテゴリ (i) エネルギー効率不変の処理が実行時間の 90 % 以上を占めるような問合せでは，消費電力の低下割合が相対的に小さいため，2.8 GHz のモードに対する 1.2 GHz のモードのエネルギー効率の増加割合は 1.3 % となっている．

よって，エネルギー効率の変化の傾向として，4.1 節における議論と同様な区分を行うと以下の三つに大きく分けられる．

(i) エネルギー効率ほぼ一定: 動作周波数によらずエネルギー効率がほぼ一定である問合せ（Q21）

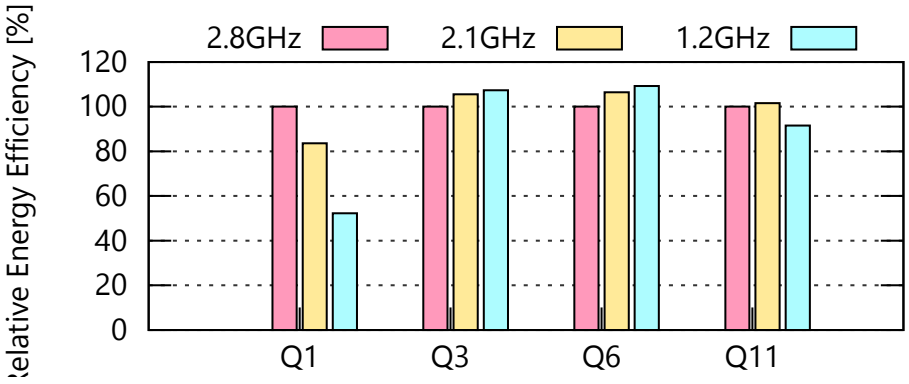
(ii) エネルギー効率向上: 動作周波数低下によりエネルギー効率が向上する問合せ（Q3, Q6）

(iii) エネルギー効率低下: 動作周波数低下によりエネルギー効率が低下する問合せ（Q1, Q11）

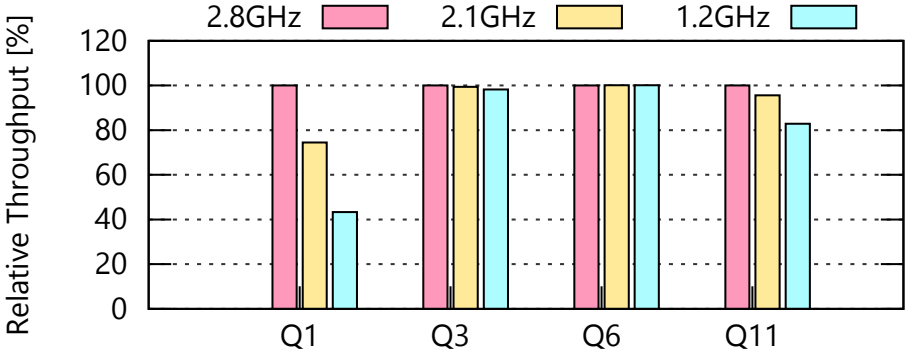
4.2.3　多重度 4 の分析問合せ処理における結果

多重度 4 で 4 種類の分析問合せ処理をさせた結果を，図 7 に示す． 計測は Q1, Q3, Q6, Q11 を対象として行った．多重度 4 の場合でも，多重度 1 の場合で確認された問合せ毎の傾向は変わらないが，エネルギー効率の変化割合が多重度 1 の場合とは異なっている．エネルギー効率が低下していた Q1 と Q11 は多重度 1 の時に比べてエネルギー効率の低下割合が小さく，2.8 GHz のモードに対する 1.2 GHz のモードのエネルギー効率の減少割合としては，Q1 が 47.7 %，Q11 が 8.5 % の減少となっている．対して，効率が向上していた Q3 と Q6 はエネルギー効率の上昇割合が多重度 1 の時に比べて大きく，2.8 GHz のモードに対する 1.2 GHz のモードのエネルギー効率の増加割合としては，Q3 が 7.4 %，Q6 が 9.3 % の増加となっている．

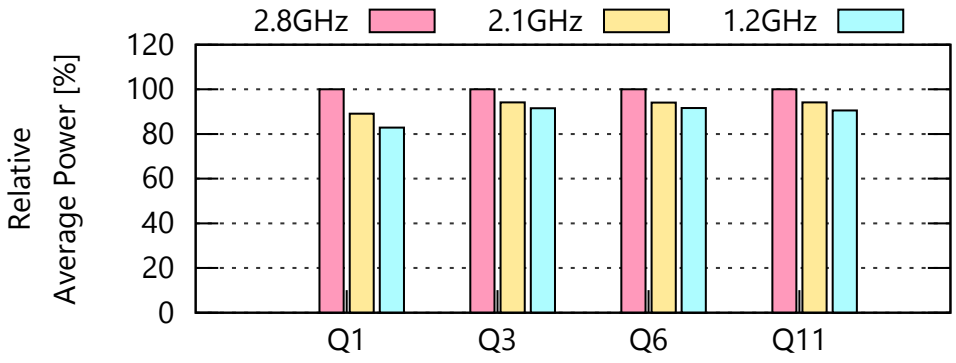
効率向上率が増大し，効率低下率が縮小した要因として，問合せが複数個同時処理されることで複数のプロセッサコアが駆動され，プロセッサの消費電力がサーバ全体の消費電力に占める割合が増加したためだと考えられる．これに対してスループットは多重度 4 にしても大きく変化していないが，その要因としては以下のことが考えら



(a) エネルギー効率.



(b) スループット.



(c) 平均消費電力.

図 7　サーバ A 上で多重度 4 の TPC-H 問合せ処理を行った時のエネルギー効率・スループット・平均消費電力（2.8 GHz からの相対値）.

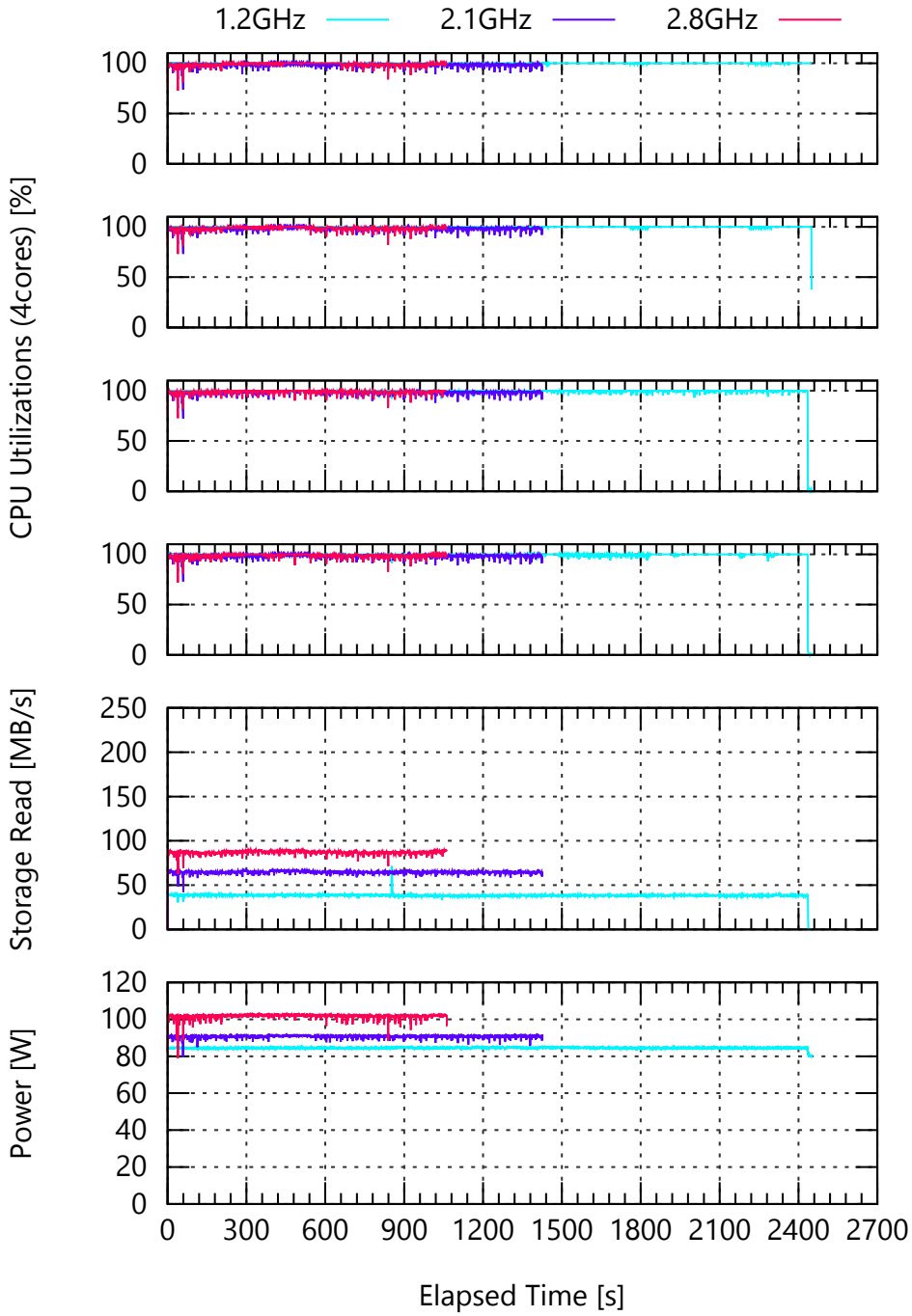


図 8　多重度 4 で Q1 を処理した時の CPU 利用率（4 コア分）・ストレージ IO 速度・消費電力の経時変化.

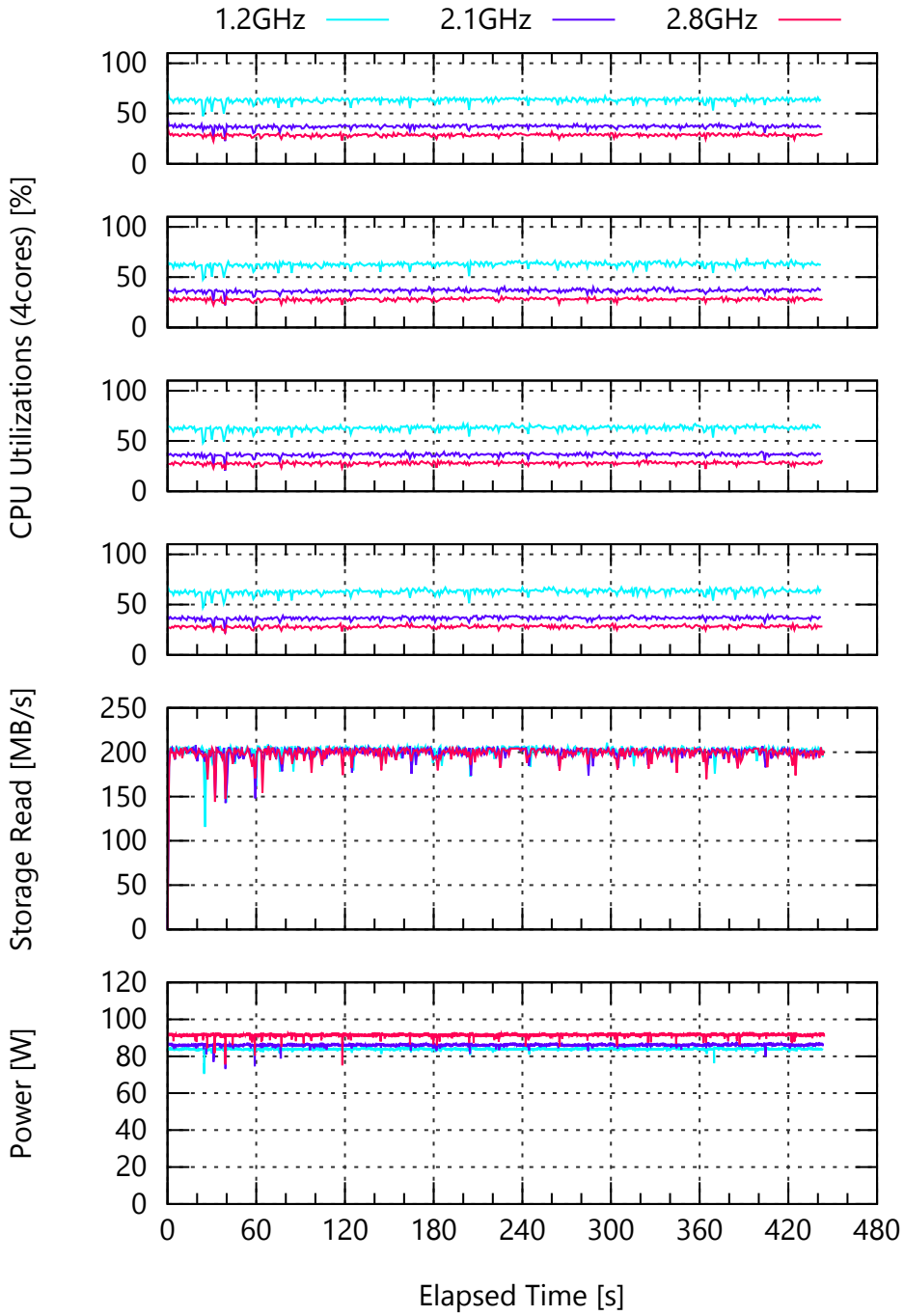


図 9　多重度 4 で Q6 を処理した時の CPU 利用率（4 コア分）・ストレージ IO 速度・消費電力の経時変化.

れる．今回は問合せの間で異なるのがデータの選択範囲のみであったが，それらの問合せは全てテーブルアクセスが全表走査のみであった．この場合，問合せ間のディスクアクセス範囲としては相違がなく，多重度 1 の時に比べてディスクアクセスを追加で行う必要はない．そのためスループットは多重度 1 の時と比べて大きく差異が出ていない． Q1 と Q6 のそれぞれの経時変化を図 8, 図 9 に示すが，いずれのコアにおいても CPU 利用率の挙動が概ね一致しており，問合せ処理の進捗がほぼ同程度の状態で進行する様子が読み取れる．また，Q11 に関しては 2.1 GHz の時のエネルギー効率は 2.8 GHz のそれよりも高いという結果になっている．以上から，マルチコア環境において問合せを同時処理した時には単独処理した時と比べてプロセッサ動作モード切り替えによる消費電力への影響が大きくなり，エネルギー効率向上が見込まれる問合せでは更なる効率の向上が可能であることが分かる．

第 5 章

プロセッサ動作モードを考慮した分析問合せ処理の消費エネルギーモデル

本章では，提案するプロセッサ動作モードを考慮した消費エネルギーモデルに関して説明する．本研究では，分析問合せの中でも基礎となる重要な問合せである全表走査に基づく問合せを対象とする．提案する消費エネルギーモデルにより，これまで計測ベースで議論されることがほとんどであったプロセッサ動作モードが分析問合せ処理に与える影響をモデルベースで検証可能になる．つまり，消費エネルギーが最小となるような動作周波数をモデルから推定可能となる．全表走査に基づく問合せを対象として作成した消費エネルギーモデルに関して説明し，モデルの有効性を確かめるために行った評価実験とその結果を示す．

5.1 全表走査に基づく分析問合せ処理におけるモデル化

データベースシステムにおいて問合せ処理の実行方式は，テーブル走査や結合等のデータベース演算子から構成される問い合わせ実行計画によって表現される．テーブル走査等の基底となるデータベース演算子は 1 回の演算実行により 1 つ以上のタプルを出力し，結合や集約演算等のデータベース演算子は入力として受け取ったタプルに基づいて演算を実行しタプルを出力する．データベース演算子は，入力に対して逐次的に演算実行が可能なパイプライン動作演算子と，全ての入力を受け取ってから演算が実行可能となるブロッキング演算子に大別される．ここで，問合せ実行計画中のデータベース演算子はブロッキング演算子を境界としてデータベース演算子をグループ化し，当該グループを実行計画基本ブロックと称することとする．実行計画基本ブロックは，

同時にパイプライン動作可能な一連のデータベース演算子から構成され，その性質上複数の実行計画基本ブロックの実行がオーバーラップすることはない．即ち，実行計画基本ブロックの単位で消費電力や消費エネルギーをモデル化することで，その組み合わせにより問合せ処理全体の消費電力や消費エネルギーを算出できる．

本研究でモデル化の対象とする全表走査に基づく問合せは，全表走査と入力タプルに対する演算処理から構成されるものを指す．当該実行計画ブロックの例としては，単純な全表走査による問合せや，ハッシュ結合のビルド処理・プローブ処理等が挙げられる．実行される問合せ処理のスループット θ（タプル毎秒）は，プロセッサの演算性能のスループット θ^{CPU} とディスクの入出力性能のスループット θ^{IO} のいずれかにより律速される．

$$\theta = \min\left(\theta^{\mathrm{CPU}}, \theta^{\mathrm{IO}}\right)$$

プロセッサの処理性能のスループット θ^{CPU} に関しては，一般的にはプロセッサの動作周波数に対して比例することが知られている．したがって，θ^{CPU} は次のように表される．

$$\theta^{\mathrm{CPU}} = af$$

a はプロセッサの性能特性やレコードあたりの演算負荷によって決まる係数である．以上から，実行計画基本ブロックで処理されるタプル数を N とすると，実行時間 T は次のように表される．

$$T = \frac{N}{\theta} = \frac{N}{\min\left(\theta^{\mathrm{CPU}}, \theta^{\mathrm{IO}}\right)} = \frac{N}{\min\left(af, \theta^{\mathrm{IO}}\right)}$$

次に，問合せ処理中のプロセッサの消費電力について考える．ここでは平均消費電力に代表させることでモデル化を行い，経時変化は考慮しないこととする．プロセッサの消費電力は，最も単純には CPU 利用率 u に対して線形で増加すると近似できるが，実際にはプロセッサの実装や負荷の種類毎に傾向は異なり，系ごとに非線形モデルを考えた方が観測データに即したモデルとなることが示されている [27]．このため本研究では，今回想定している分析問合せ処理用のデータベースシステムに対して有用性が示された文献 [27] で提案されたモデルを用いる．即ち，アイドル状態からの電力増分 ΔP^{CPU} は，$\Delta P^{\mathrm{CPU}} \propto 2u - u^r \quad (1 \leq r)$ に従うものとする．更に動作モードの違いによる影響について考えると，動作モードは動作周波数 f とそれに対応する動作電圧 V

によって規定される．動作周波数 f が高いほど，安定動作に必要な動作電圧 V は高くなることが知られており [28]，プロセッサの実装毎に f と V の関係は異なるが，一般的には V は f に対してステップ状に増加する．以上からプロセッサの消費電力 P^{CPU} は次のように表される．

$$P^{\mathrm{CPU}} = \left\{ Af\,V^2 + Bf + C(V - V_{\mathrm{idle}}) \right\} (2u - u^r) + P^{\mathrm{CPU}}_{\mathrm{idle}} \tag{1}$$

中括弧内の第 1 項は論理ゲートのスイッチングによる消費電力の項，第 2 項はショートサーキット電流による消費電力の項，第 3 項は漏れ電流による消費電力に対応する項で，A, B, C はそれぞれにかかる係数である [29]．$P^{\mathrm{CPU}}_{\mathrm{idle}}$ はアイドル状態における電力である．CPU 利用率 u に関しては，プロセッサ上で単位時間あたりに処理されるタプル数のみによって決定されると考えた場合，CPU 律速（$\theta = \theta^{\mathrm{CPU}}$）の時には CPU 利用率が 100 % となり，IO 律速（$\theta = \theta^{\mathrm{IO}}$）の時には CPU 利用率が 100 % 未満となる．最も単純なモデルとして，単位時間あたりに処理されるタプル数に比例すると考えれば，CPU 利用率 u は次のように表される．

$$u = \frac{\theta}{\theta^{\mathrm{CPU}}} = \frac{\min\left(af, \theta^{\mathrm{IO}}\right)}{af} \tag{2}$$

次にストレージによる消費電力 P^{IO} は，ストレージの稼働率が θ と θ^{IO} によって決定され，アイドル状態の消費電力からの電力変化量が稼働率に対して線形だと考えれば，次のように表される．

$$P^{\mathrm{IO}} = D\frac{\theta}{\theta^{\mathrm{IO}}} + P^{\mathrm{IO}}_{\mathrm{idle}}$$

D はストレージの稼働率が最大の時の電力増加量，$P^{\mathrm{IO}}_{\mathrm{idle}}$ はアイドル状態における電力である．

最後に他のコンポーネントによる電力消費を考える．メモリ・チップセット・電源ユニット・筐体ファンなどの電力は処理の内容に応じて多少の変動は生じるが，これらの影響はプロセッサコアやストレージによる変動と比較すると相対的に小さいため，プロセッサの動作モードには依存しない定数項 P^{others} であるものと見なす．これにより，マシンの消費電力 P は次のように書ける．

$$
\begin{aligned}
P &= P^{\mathrm{CPU}} + P^{\mathrm{IO}} + P^{\mathrm{others}} \\
&= \left\{ Af\,V^2 + Bf + C(V - V_{\mathrm{idle}}) \right\} (2u - u^r) + D\frac{\theta}{\theta^{\mathrm{IO}}} + P_{\mathrm{idle}}^{\mathrm{CPU}} + P_{\mathrm{idle}}^{\mathrm{IO}} + P^{\mathrm{others}}
\end{aligned}
$$

したがって，処理に要するエネルギー E は消費電力 P と実行時間 T の積であるから，

$$
\begin{aligned}
E &= PT \\
&= \left[\left\{ Af\,V^2 + Bf + C(V - V_{\mathrm{idle}}) \right\} (2u - u^r) + D\frac{\theta}{\theta^{\mathrm{IO}}} + P_{\mathrm{idle}}^{\mathrm{CPU}} + P_{\mathrm{idle}}^{\mathrm{IO}} + P^{\mathrm{others}} \right] \frac{N}{\theta} \\
&= N \left[\frac{Af\,V^2 + Bf + C(V - V_{\mathrm{idle}})}{\theta} (2u - u^r) + \frac{D}{\theta^{\mathrm{IO}}} + \frac{P_{\mathrm{idle}}^{\mathrm{CPU}} + P_{\mathrm{idle}}^{\mathrm{IO}} + P^{\mathrm{others}}}{\theta} \right]
\end{aligned}
$$

問合せが複数の実行計画基本ブロック i から構成される場合には，それぞれの T_i, P_i を計算することで，問合せ全体の消費エネルギー E を $E = \sum_i E_i = \sum_i P_i T_i$ から求められる．

以上の議論で最終的に得られた式に含まれるパラメータに関して，表 3 にまとめる．表中ではアイドル状態の消費電力と稼動状態の消費電力の差分をアクティブ消費電力と呼称している．

ここで，消費エネルギーを最小化するような動作周波数 f_{opt} について考える．既になされた議論を踏まえると，f_{opt} の値は資源律速に大きく依存するはずである．ここで，システム上でプロセッサ動作周波数が n 段階に調整可能（$f = f_1, \cdots, f_n$ $(f_i < f_{i+1})$）である場合に，$V(f)$ が動作周波数によって変化する（$V(f_{s_j}) > V(f_{s_j - 1})$ となる）箇所が m 箇所（$f_{s_1}, \cdots, f_{s_m}$）存在すると仮定する．IO 律速（$f > \theta/a$）の時には，消費エネルギー E は f に対して単調増加する．CPU 律速（$f < \theta/a$）の時には，$V(f)$ が変化しない限りは E は f に対して単調減少するが，もし $V(f)$ がステップ的に上昇する箇所（f_{s_j}）を跨ぐ場合には E は $f_{s_j - 1}$ において局所最小値を取る．以上から，f_{opt} は次の三つのケースに分けられる．

(a) $f_i > \theta^{\mathrm{IO}}/a$ $(i = 1, \cdots, n)$ の時，f_{opt} は最低周波数 f_1

(b) $f_b > \theta^{\mathrm{IO}}/a$ $(1 < b \leq n)$ を満たすような f_b が存在し，なおかつ $f_k < \theta^{\mathrm{IO}}/a$ $(k = 1, \cdots, b - 1)$ であるとき，f_{opt} は境界周波数 f_b または $f_{s_j - 1}$ $(s_j < b)$

表 3　プロセッサ動作モード制御の消費エネルギーモデルで用いられているパラメータ.

N	処理される総タプル数
θ^{IO}	単位時間あたりにディスクが読み込み可能なタプル数の上限 [/s]
a	単位時間あたりにプロセッサが処理可能なタプル数の上限に対応する係数 [/(s · GHz)]
r	プロセッサのアクティブ消費電力と CPU 利用率の関係の補正のための冪指数
A	回路のスイッチングによるアクティブ消費電力の係数 [W/(GHz · V^2)]
B	ショートサーキット電流によるアクティブ消費電力の係数 [W/GHz]
C	漏れ電流によるアクティブ消費電力の係数 [W/V]
D	ディスクのアクティブ消費電力の係数 [W]
V	稼動状態のプロセッサ動作電圧 [V]
V_{idle}	アイドル状態のプロセッサ供給電圧 [V]
P^{others}	プロセッサ・ディスク以外のコンポーネントの定常的な消費電力 [W]
$P^{\mathrm{CPU}}_{\mathrm{idle}}$	アイドル状態のプロセッサの消費電力 [W]
$P^{\mathrm{IO}}_{\mathrm{idle}}$	アイドル状態のディスクの消費電力 [W]

(c) $f_i < \theta^{\mathrm{IO}}/a$ $(i = 1, \cdots, n)$ の時，f_{opt} は最高周波数 f_n または f_{s_j-1}

5.2　消費エネルギーモデルの評価実験

本節では，5.1 節で記述した消費エネルギーモデルの評価実験に関して説明する．

5.2.1　測定環境

ハードウェア構成・ソフトウェア構成はサーバ A 上で 4.2 節のものと同様の構成を利用した．PostgreSQL のメモリ割当に関しては変更した（表 4）．

5.2.2　プロセッサ動作モードの基礎特性測定

測定環境における `cpufreq` モジュールを通じた動作モード制御では指定する対象は動作周波数のみであり，動作電圧に関しては直接設定あるいは計測することができないため，動作周波数（動作モード）と消費電力の関係を確かめるためのマイクロベンチマークを実施した．`stress-ng` と呼ばれるソフトウェアを用いて，16 段階の動作モードごとのアイドル状態における消費電力と，プロセッサの特定の 1 コアに CPU 利用率が 100 % となるように負荷を 20 分間与えた際の消費電力をそれぞれ計測した．仮に動作電圧が一定だった場合，式 (1) より動作周波数と消費電力は線形の関係となる．よって，動作周波数と消費電力が単純に $P^{\mathrm{CPU}} = \alpha f + \beta$ の関係を持つと考えて線形回帰モデルを適合させた．

結果を図 10 に示す． プロセッサがアイドル状態際のの消費電力値は，稼働状態と比べて 10W〜13W 程度低く，動作周波数によらずほぼ一定となっていることが分かる．稼働状態での消費電力値は，1.2 GHz から 2.7 GHz までと 2.7 GHz から 2.8 GHz にか

表 4　サーバ A でのモデル評価実験における PostgreSQL のメモリ割当関連の設定．

`shared_buffers`	8192 MB
`temp_buffers`	2048 MB
`work_mem`	8192 MB

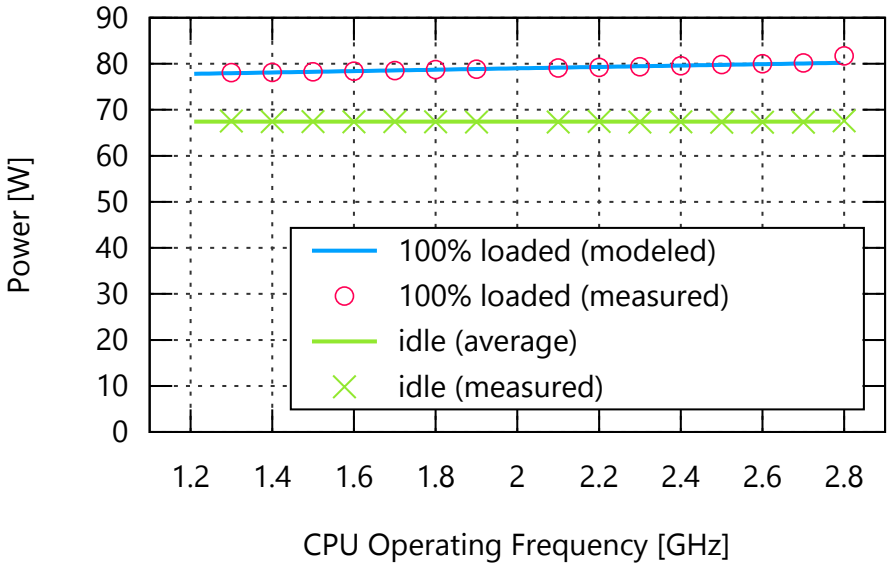


図 10　マイクロベンチマークによる動作周波数毎の平均消費電力と線形回帰モデルによるフィッティング結果.

けてでは傾向が異なることが分かる．1.2 GHz から 2.7 GHz までは線形回帰モデルとの誤差が最大で 0.15 % だが，2.8 GHz では 1.8 % で相対的に大きく外れている．これより，1.2 GHz から 2.7 GHz までは動作電圧は一定で，2.8 GHz でのみ動作電圧が増加していると推定される．よって，測定環境における動作電圧 $V(f)$ は次のように書ける．

$$V(f) = \begin{cases} V_1 & (f \leq 2.7\,\mathrm{GHz}) \\ V_2 & (f = 2.8\,\mathrm{GHz}) \end{cases} \tag{3}$$

ただし，$V_1 < V_2$ とする．

リスト 1　クエリ A の SQL.

```
1 select l_returnflag, l_linestatus, sum(l_quantity) as sum_qty, sum(
      l_extendedprice) as sum_base_price, sum(l_extendedprice * (1 -
      l_discount)) as sum_disc_price, sum(l_extendedprice * (1 - l_discount) *
       (1 + l_tax)) as sum_charge, count(*) as count_order
2 from lineitem
3 group by l_returnflag, l_linestatus;
```

リスト 2　クエリ B の SQL.

```
1 select sum(l_extendedprice * (1 - l_discount)) as revenue
2 from lineitem;
```

リスト 3　クエリ C の SQL.

```
1 select sum(l_extendedprice * (1 - l_discount)) as revenue
2 from orders, lineitem
3 where l_orderkey = o_orderkey and o_orderdate < date '1996-03-23' and
      l_shipdate > date '1994-03-23';
```

5.2.3　評価用の問合せとモデルパラメータ推定手法

TPC-H 標準の問合せを基に，全表走査に基づく問合せとして評価用クエリ A, B, C を定めた．

- クエリ **A**　5 つの集約計算と 2 つのグルーピング計算からなる `LINEITEM` の集約演算（リスト 1）
- クエリ **B**　1 つの集約計算からなる `LINEITEM` の集約演算（リスト 2）
- クエリ **C**　`ORDERS` ⋈ `LINEITEM`（リスト 3）

クエリ A とクエリ B では全表走査，クエリ C では全表走査とハッシュ結合による実行計画が用いられることを確認した．クエリ A とクエリ B は 1 つのテーブル `LINEITEM` への全表走査と集約演算を行うため，1 つの実行計画基本ブロックからなる．両者の違いは集約演算に要する計算量にあり，クエリ A は複数の集約計算とグルーピング計算を行う必要があるためクエリ B よりも必要な計算量が多い．クエリ C のハッシュ結合による結合操作を行うため 2 つの実行計画基本ブロックからなり，1 つ目の実行計画基本ブロックで `ORDERS` に対する全表走査からのハッシュ表の構築（build），2 つ目の実行計画基本ブロックで `LINEITEM` に対する全表走査からのハッシュ表を用いた結合（probe），そして集約演算が行われる．これらの問合せを 16 段階の動作モードそれぞれで処理し，実行時間，平均消費電力，消費エネルギーを計測した．そしてこれらの計測結果を基に以下のような手順でモデルパラメータを推定した．

(1) 各実行計画基本ブロックの N の値を PostgreSQL の問合せ最適化器から取得．

(2) 全ての実行計画基本ブロックの計測結果を利用して，最小二乗法でマシン固有のパラメータを推定．なお動作電圧 $V(f)$ に関しては式 (3) に従うとして推定した．

(3) 推定されたマシン固有のパラメータを当てはめ，実行計画基本ブロック依存のパラメータを最小二乗法で推定．

なお，問合せ毎の実行計画基本ブロックの数に関しては所与であるとして，実行計画基本ブロックが切り替わるタイミングを手動で指定している．具体的には，クエリ A と

クエリ B は 1 段階処理，クエリ C は 2 段階処理として扱い，各実行計画基本ブロックの実行時間・平均消費電力・消費エネルギーをモデルに与えた．

5.2.4　実測値とモデルとの比較

各評価用問合せを処理して得られた各実行計画基本ブロックの実行時間・平均消費電力・消費エネルギーの実測値と，パラメータ推定を行ったモデルが記述する関数を図 11，図 12，図 13，図 14 に示す．　どの実行計画基本ブロックにおいても，実行時間及び消費電力の傾向を捉えられていることが分かる．そして，実行時間について見ると実行計画基本ブロックによる傾向の違いが分かる．図 11 では実行時間は動作周波数の増加と共に単調減少していく．これはクエリ A ではどの動作周波数においても CPU 律速であるためである．対して，図 12 ではおおよそ 2.2 GHz, 図 14 ではおおよそ 1.7 GHz を境界として，動作周波数が低い範囲では周波数増加と共に実行時間が短くなり，動作周波数が高い範囲では実行時間がほぼ変化しない．これは，動作周波数が低い範囲では CPU 律速となっており動作モード変更による処理スループットへの影響が大きいのに対して，動作周波数が高い範囲となって IO 律速に切り替わると動作モードを高周波数にしてもスループットにほとんど影響がなくなるためである．そして図 13 に関しては，動作モードによらず IO 律速であるため実行時間はほぼ一定となっている．

これらの事項は CPU 利用率とストレージ IO 速度の経時変化のグラフからも確かめることが可能である．各問合せ処理における CPU 利用率・ストレージ IO 速度・消費電力の経時変化を図 15，図 16，図 17 に示す．　グラフは最低周波数（1.2 GHz），中程度の周波数（2.1 GHz）最高周波数（2.8 GHz）に限定して示している．図 16 では，2.1 GHz と 2.8 GHz で CPU 利用率は 100 % に達しておらず，ストレージ IO 速度の経時変化の仕方がほぼ一致している．このことから，この 2 つの動作周波数では確かに IO 律速となっており実行時間はストレージ IO スループットに依存している．一方，1.2 GHz では CPU 利用率が 100 % に達しており，ストレージ IO 速度は前述の 2 つの動作周波数の時に比べて下がっている．よって，1.2 GHz では確かに CPU 律速となっており実行時間はプロセッサの演算スループットに依存している．図 17 では，100 秒付近で異なる実行計画基本ブロックに切り替わるが，第 1 実行計画基本ブロックでは常に CPU 利用率が 100 % 未満で IO 律速なのに対し，第 2 実行計画基本ブロックでは

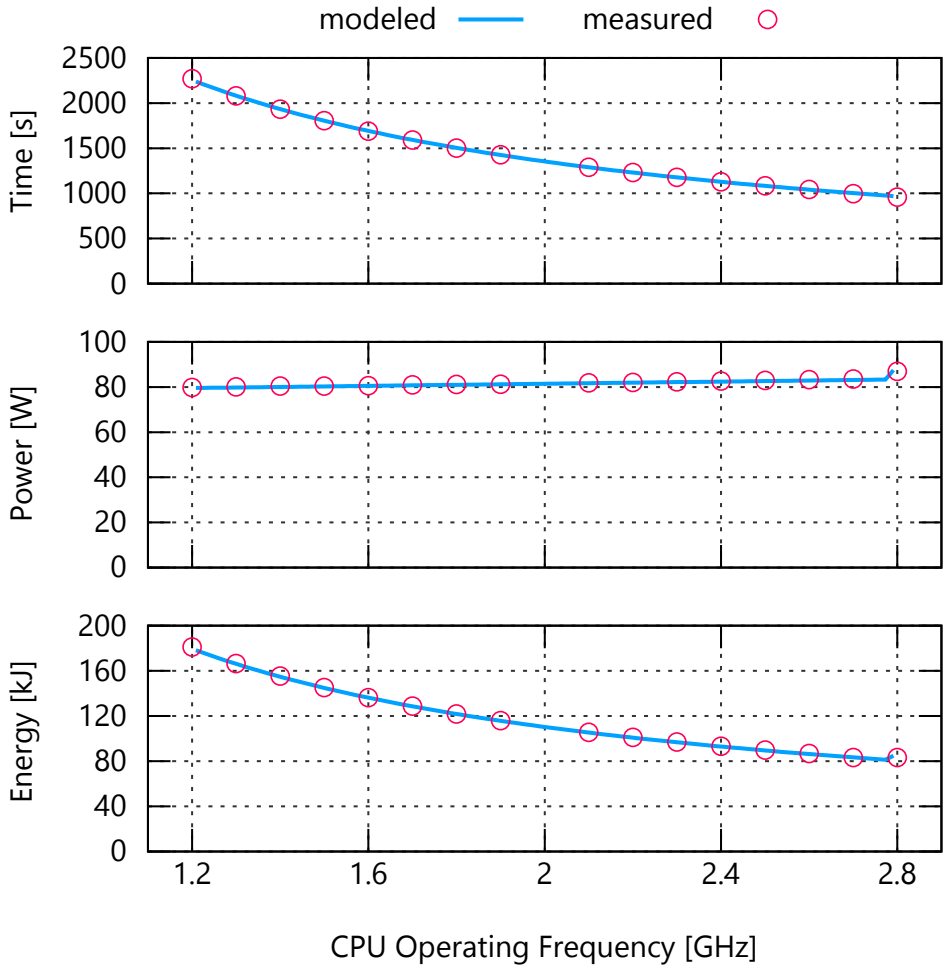


図 11　クエリ A の処理の実行時間・平均消費電力・消費エネルギーの動作モード特性の実測値とモデルが記述する曲線.

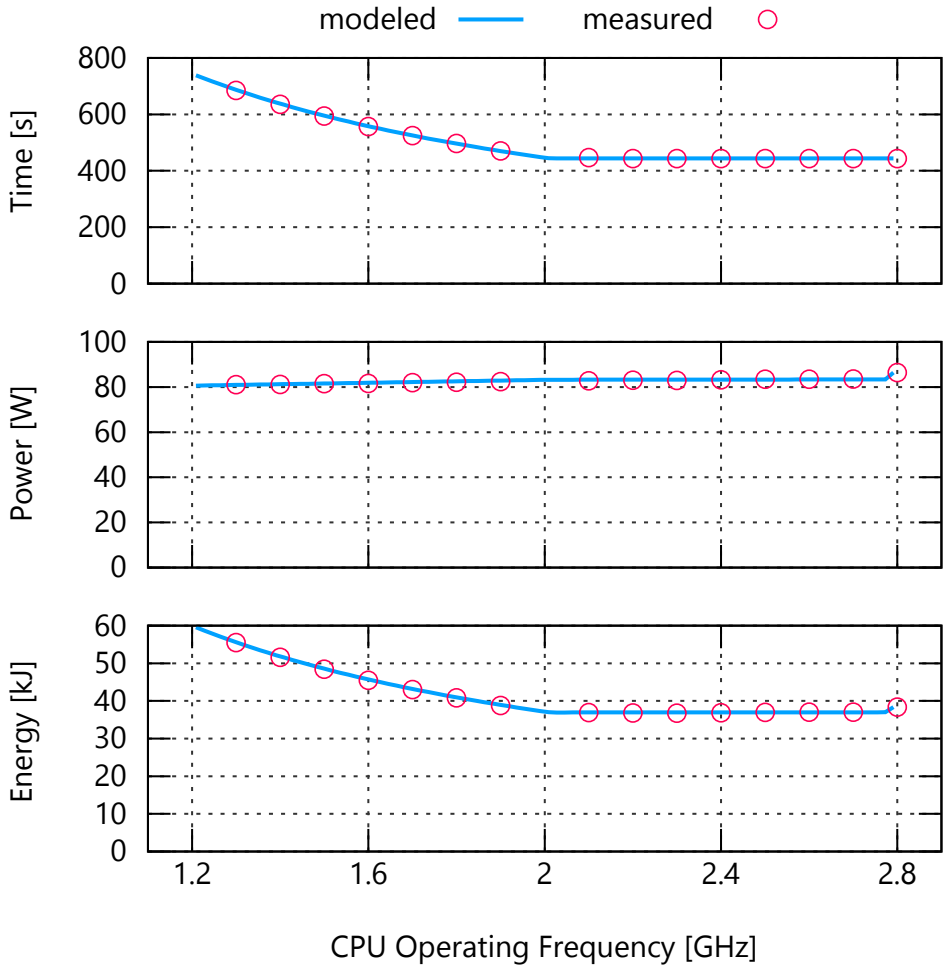


図 12　クエリ B の処理の実行時間・平均消費電力・消費エネルギーの動作モード特性の実測値とモデルが記述する曲線.

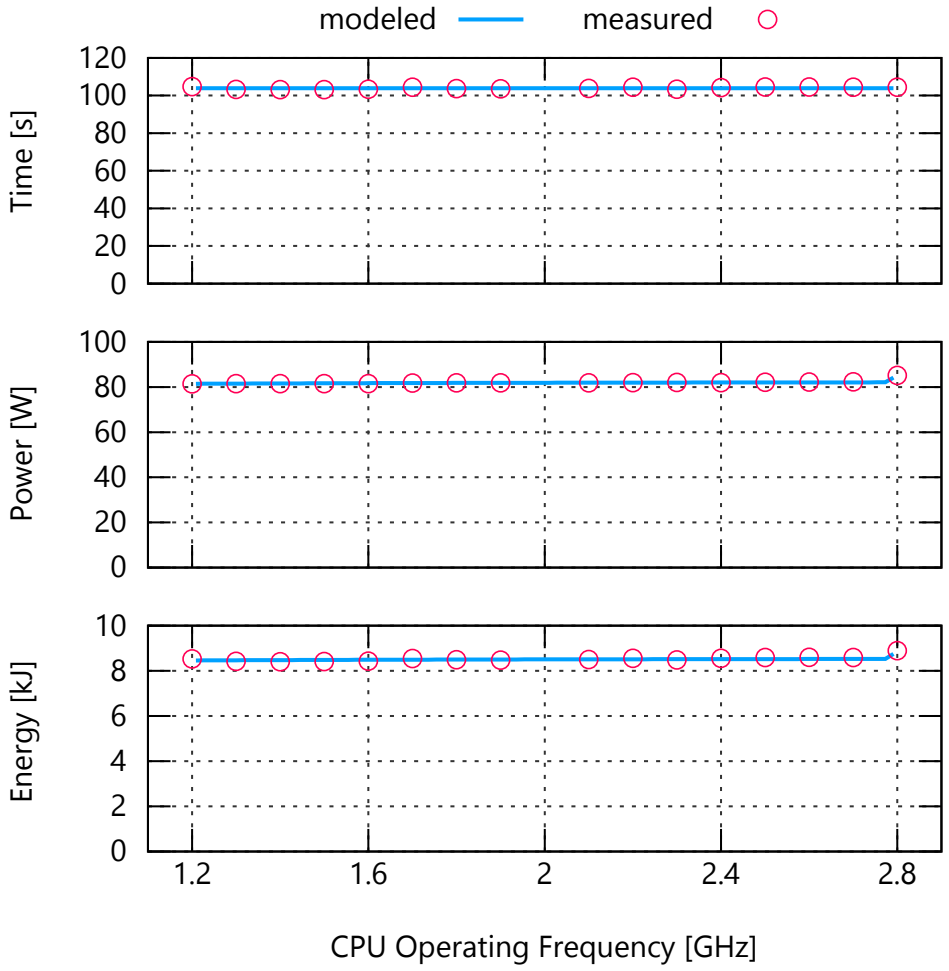


図 13　クエリ C の第 1 実行計画基本ブロックの処理の実行時間・平均消費電力・消費エネルギーの動作モード特性の実測値とモデルが記述する曲線.

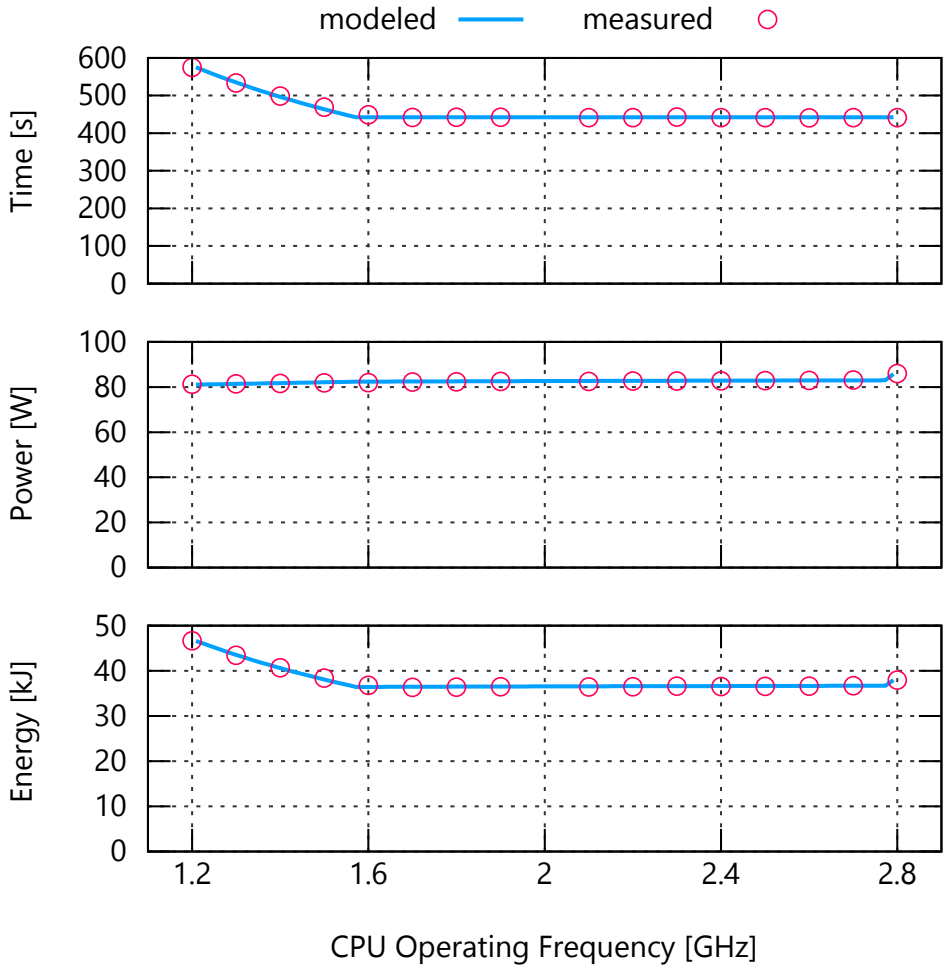


図 14　クエリ C の第 2 実行計画基本ブロックの処理の実行時間・平均消費電力・消費エネルギーの動作モード特性の実測値とモデルが記述する曲線.

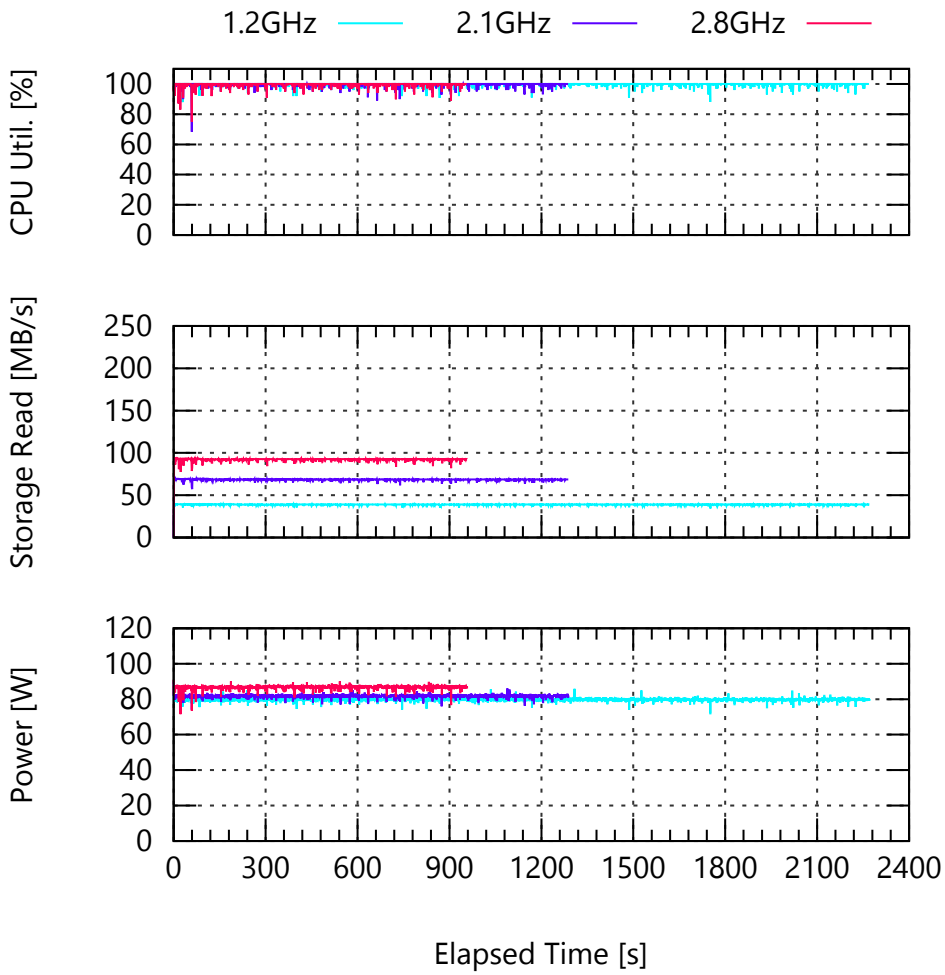


図 15　クエリ A の処理の CPU 利用率・ストレージ IO 速度・消費電力の経時変化.

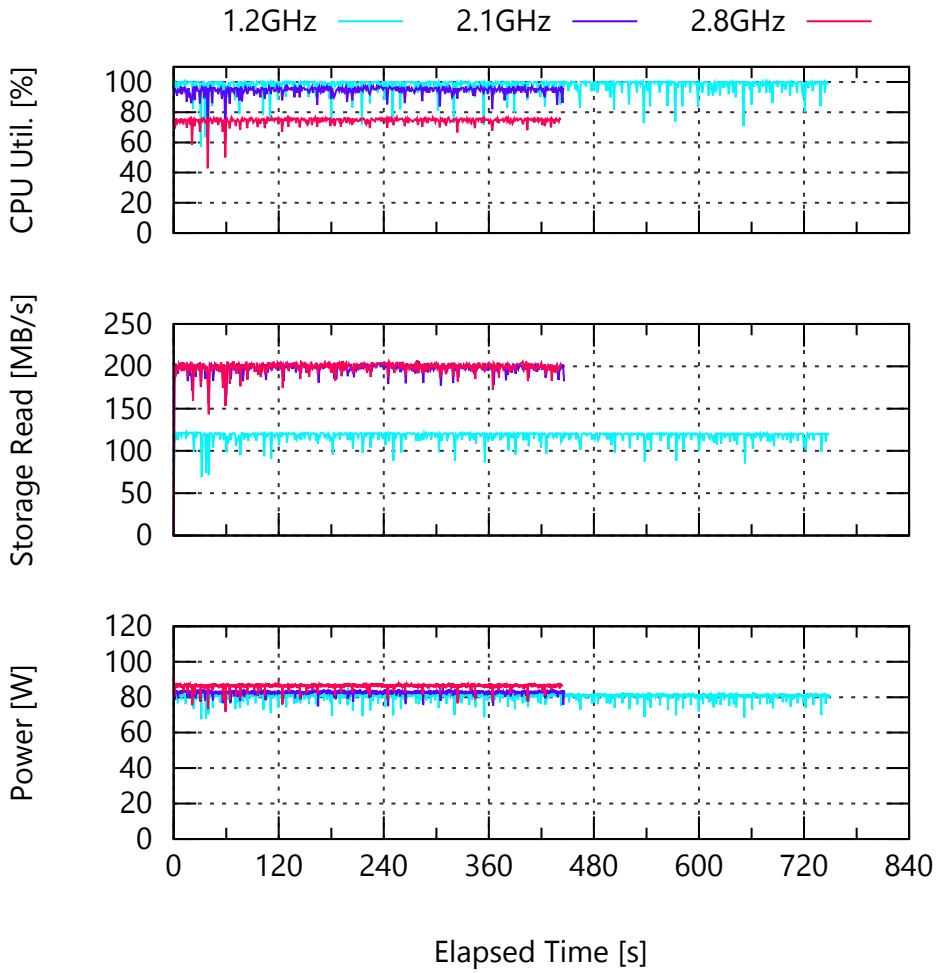


図 16　クエリ B の処理の CPU 利用率・ストレージ IO 速度・消費電力の経時変化.

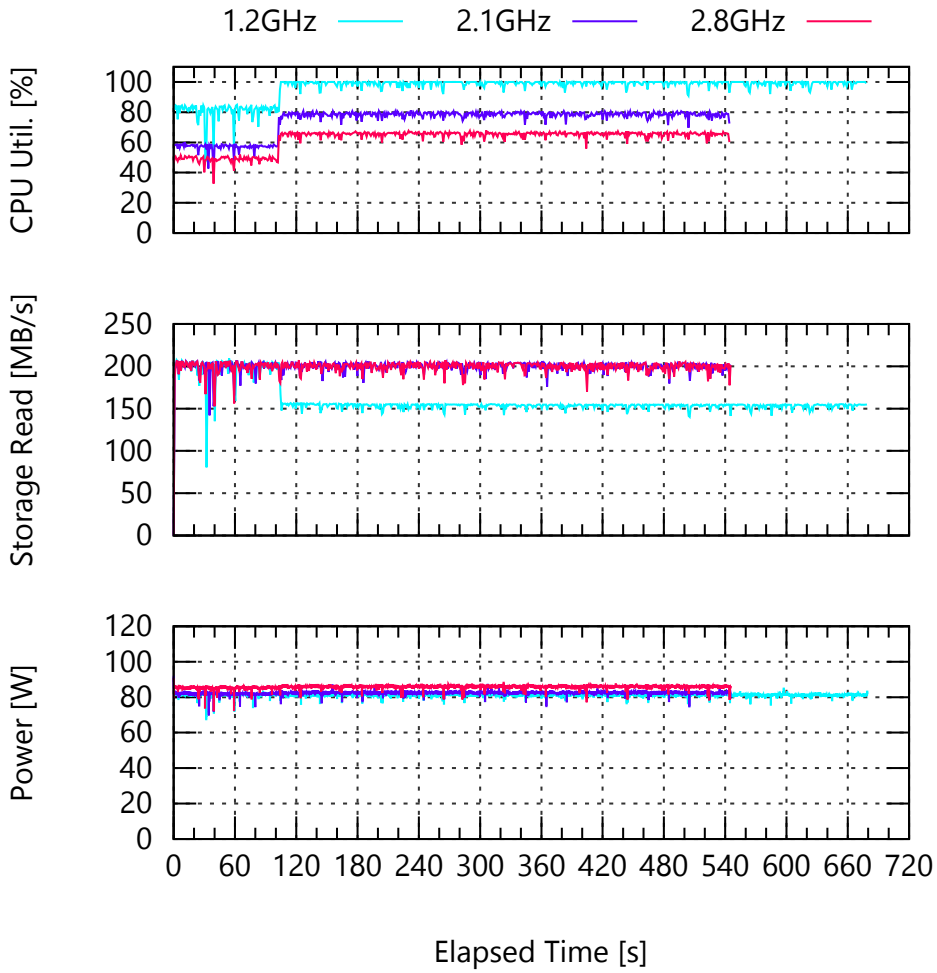


図 17　クエリ C の処理の CPU 利用率・ストレージ IO 速度・消費電力の経時変化.

図 16 と同様に 2.1 GHz と 2.8 GHz で IO 律速，1.2 GHz で CPU 律速となっている．

次に消費電力に関して見ると，動作周波数増加に従って単調増加している．しかし，増加の傾きは動作周波数の範囲によって違いがあり，CPU 律速の範囲に比べると IO 律速の範囲では電力増加の傾きが小さい．これは，式 (2) からディスク律速の場合に CPU 利用率 u は動作周波数 f に反比例するので，式 (1) における f の増加による電力増加量と打ち消し合う効果が生じていることによる．そして基礎計測実験で確かめた通り，2.8 GHz では 2.7 GHz 以下に比べて動作電圧 $V(f)$ が増加しているため，2.7 GHz から 2.8 GHz にかけては電力の傾きがそれまでと比べて急になっている．この実測結果とモデルが記述する値は概ね合致しており，実行時間と消費電力の積である消費エネルギーに関しても同様である．消費エネルギーの実測値とモデルの値の誤差は最大で 1.4 % に留まっている．

一方，CPU 律速と IO 律速の転換点の近傍においてはモデルと実測値の乖離が見られる．モデルが記述する実行時間の CPU 律速と IO 律速の転換点は微分不可能であり消費エネルギーはこの点で最小となるが，実測値では転換点の近傍において実行時間は比較的滑らかに変化しており，消費エネルギーが最小となる点がより高い動作周波数へ移る傾向がある．この挙動の原因としては，CPU 律速と IO 律速が切り替わる前後ではどちらかが完全に支配的となるのではなく，両者が混在している状態となっているからであると推察される．この挙動により，モデルを利用した消費エネルギーの極小点 f_{opt} の予測には誤差が生じる．モデルが記述するクエリ B の f_{opt} は，実際にプロセッサの動作周波数として設定可能な範囲では 2.1 GHz となる．しかし実測値としては 2.2 GHz となっている．クエリ C の第 2 実行計画基本ブロックでは，モデルの f_{opt} は律速の転換点と一致しており 1.6 Hz 前後だが，実測では 1.7 Hz に位置している．f_{opt} の予測誤差による消費エネルギーへの影響は，クエリ B においては 0.4 % の増加，クエリ C においては 1.2 % の増加となる．

総括としては，モデルが記述する値は実行時間・消費電力・消費エネルギーの実測値の傾向を捉えられていることが分かる．消費エネルギーの値の誤差は 1.65 % 以下に抑えられている他，消費エネルギーを最小化する動作周波数にはズレがあるもののその影響は 1.2 % 以下に抑えられている．これらの結果は，提案したモデルの有用性を示すものである．最後に，フィッティングされたモデルパラメータの値を表 5 に示す．

表 5　プロセッサ動作モード制御の消費エネルギーモデルでのパラメータ推定値．

(a) サーバ固有のパラメータ．

A [W/(GHz · V^2)]	1.90×10^{-1}
B [W/GHz]	1.33
C [W/V]	8.26×10^{-4}
D [W]	2.60
V_1 [V]	1.82
V_2 [V]	3.34
V_{idle} [V]	5.98×10^{-1}
$P^{\mathrm{others}} + P^{\mathrm{CPU}}_{\mathrm{idle}} + P^{\mathrm{IO}}_{\mathrm{idle}}$ [W]	7.67×10^{1}

(b) 実行計画基本ブロック依存のパラメータ．N は理論値を記している．クエリ C に関しては実行計画基本ブロックを括弧内の数字によって表している．

	クエリ A	クエリ B	クエリ C(1)	クエリ C(2)
N（理論値）	6.00×10^{8}	6.00×10^{8}	1.50×10^{8}	6.00×10^{8}
θ^{IO} [/s]	1.35×10^{6}	1.35×10^{6}	1.44×10^{6}	1.36×10^{6}
a [/(s · GHz)]	2.22×10^{5}	6.72×10^{5}	1.72×10^{6}	8.63×10^{5}
r	1.85	1.11	2.00	1.39

第 6 章

結論

本研究では，分析問合せ処理における省電力化を目的とした，プロセッサ動作モードを考慮した消費エネルギーモデルの提案を行った．消費エネルギーモデルの作成にあたり，分析問合せ処理がプロセッサ動作モードから受ける影響が資源律速によって異なることに関して定性的に議論し，実験によりこれを明らかにした．そして，この定性的な議論を議論を踏まえ，全表走査に基づく分析問合せを対象として消費エネルギーの定量的なモデルを提案し，その有効性を評価実験により確かめた．

今後の課題としては，消費エネルギーモデルを索引走査にも適用できるように拡張することが考えられる．索引走査ではストレージ IO における先読みが有効になりづらいため，問合せ実行のモデルの修正が必要となる．また，現在の消費エネルギーモデルは消費電力に関しては実行計画基本ブロックにおける平均電力のみを考慮している他，プロセッサとストレージ以外の消費電力は一定だとして扱っているため，これらが可変であるとしてモデル化を行うことも考えられる．

謝辞

本研究へ取り組むにあたり，本当に多くの方々にご指導ご支援を賜りました．

早水悠登特任助教，合田和生特任准教授には心より感謝を申し上げます．研究に行き詰まった際には，いつもお二方の指針と助言に助けられてきました．難しい問題に突き当たった際にも一緒に悩んでくださり，問題が解決する度にいつも光明が差す思いでした．お二方が居なければ，研究を形にすることさえ難しかったと思います．また，研究以外で悩んだ際にも多くの助言を頂き，励まして頂きました．お忙しい中，貴重な時間を費やして頂き本当にありがとうございました．

吉永直樹准教授には分野外ながらもいつも論文や発表に関して丁寧に目を通して頂き助かりました．また，美味しい料理や美味しいビールを数多く教わり，日々の楽しみが増えるきっかけになりました．

豊田正史教授には度々コンピュータシステムに関して相談相手になって頂き，様々な意見をぶつけ合うことができて勉強になりました．

喜連川優教授には素敵な研究室の環境を提供して頂き感謝申し上げます．また，厳しくも愛のある意見を頂くこともあり，その度に目が覚める思いでした．

秘書の皆さんには研究を円滑に運ぶために様々な助けを頂きました．また，果物やお菓子の差し入れを頂くこともありましたが，生活が荒みがちな私にとって有り難いものでした．

研究室の先輩方である，佐藤文一さん・石渡祥之佑さん・金洪善さん・佐藤翔悦さん・赤崎智さん・澤田頌子さん・陳鍵さん・大原康平さんに大変感謝しております．どんな時にも気さくに会話してくれたこと，有り難く思います．研究室生活の指針になる存在でした．皆さんの頑張る姿を追って研究ができたことを誇りに思います．

研究室の後輩方である，三條嵩明君・別所祐太朗君・福田展和君・大葉大輔君・土屋潤一郎君・蔦侑磨君・杉山普君・左天池君に感謝申し上げます．賑やかな研究室になり，楽しく毎日を過ごすことができました．研究において教わることも多々あり非常に勉強になりました．

そして，研究室同期の保田和彦君・遠田哲史君・根石将人君・佐久間仁君・清水洸希

君・張翔君に心から感謝申し上げます．分野外ながらもお互いの研究に助言しあうことで，研究者として高め合う関係を持てたことを嬉しく思います．一緒に行った日々の活動もかけがえのない思い出です．

アイドルマスターミリオンライブの野々原茜には，つらい時にも笑顔を見せ続けることの大切さを学びました．そして THE@TER CHALLENGE!!キャスティング投票企画において主人公の座を射止めたことは私の心の支えになり続けました．賞賛と共に感謝を送ります．

最後に，家族に多大なる感謝を．母は日々美味しい料理を作ってくれた他，心身共に疲弊した私のことを度々慰めてくれました．父は研究者としての心構えを説いてくれた他，人生において大切なことをたくさん教えてくれて自分の将来について思い悩む私を励ましてくれました．心が折れそうになった時に貰った二人からの励ましの言葉を胸に刻んで研究に打ち込んできました．生まれてきて良かったと心から言えます．素敵な家族に心から感謝申し上げます．

2019 年 1 月 31 日

参考文献

[1] Josh Whitney and Pierre Delforge. Data Center Efficiency Assessment. Issue paper, 2014.

[2] Paolo Bertoldi, Nicola Labanca, and Bettina Hirl. *Energy efficiency status report 2012: electricity consumption and efficiency trends in the EU-27.* Publications Office, 2012.

[3] Meikel Poess, Bryan Smith, Lubor Kollar, and Paul Larson. TPC-DS, Taking Decision Support Benchmarking to the Next Level. In *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*, pp. 582–587, 2002.

[4] Pat O'Neil, Betty O'Neil, and Xuedong Chen. Star Schema Benchmark Revise 3. June 2009.

[5] Intel. Enhanced Intel SpeedStep Technology for the Intel Pentium M Processor. White paper, March 2004.

[6] Rakesh Agrawal, Anastasia Ailamaki, Philip A. Bernstein, Eric A. Brewer, Michael J. Carey, Surajit Chaudhuri, Anhai Doan, Daniela Florescu, Michael J. Franklin, Hector Garcia-Molina, Johannes Gehrke, Le Gruenwald, Laura M. Haas, Alon Y. Halevy, Joseph M. Hellerstein, Yannis E. Ioannidis, Hank F. Korth, Donald Kossmann, Samuel Madden, Roger Magoulas, Beng Chin Ooi, Tim O'Reilly, Raghu Ramakrishnan, Sunita Sarawagi, Michael Stonebraker, Alexander S. Szalay, and Gerhard Weikum. The Claremont Report on Database Research. *Communications of the ACM*, Vol. 52, No. 6, pp. 56–65, June 2009.

[7] Stavros Harizopoulos, Mehul Shah, Justin Meza, and Parthasarathy Ranganathan. Energy Efficiency: The New Holy Grail of Data Management Systems

Research. In *4th biennial Conference on Innovative Data Systems Research (CIDR)*, September 2009.

[8] Goetz Graefe. Database Servers Tailored to Improve Energy Efficiency. In *Proceedings of the 2008 EDBT Workshop on Software Engineering for Tailor-made Data Management (SETMDM)*, pp. 24–28, March 2008.

[9] Suzanne Rivoire, Mehul A. Shah, Parthasarathy Ranganathan, and Christos Kozyrakis. JouleSort: A Balanced Energy-efficiency Benchmark. In *Proceedings of the 2007 ACM International Conference on Management of Data (SIGMOD)*, pp. 365–376, June 2007.

[10] Meikel Poess, Raghunath Othayoth Nambiar, Kushagra Vaid, John M. Stephens Jr., Karl Huppler, and Evan Haines. Energy Benchmarks: A Detailed Analysis. In *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking (e-Energy)*, pp. 131–140, April 2010.

[11] Venkatesh Pallipadi and Alexey Starikovskiy. The Ondemand Governor: Past, Present and Future. In *the Proceedings of the Ottawa Linux Symposium (OLS)*, 2006.

[12] S. Huang and W. Feng. Energy-Efficient Cluster Computing via Accurate Workload Characterization. In *Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID)*, pp. 68–75, Washington, DC, USA, May 2009. IEEE Computer Society.

[13] Robert Schöne and Daniel Hackenberg. On-line Analysis of Hardware Performance Events for Workload Characterization and Processor Frequency Scaling Decisions. In *Proceedings of the 2nd ACM/SPEC International Conference on Performance Engineering (ICPE)*, pp. 481–486, June 2011.

[14] Masahiro Miwa, Kohta Nakashima, Akira Hirai, Satoshi Kazama, Yasushi Hara, and Akira Naruse. Evaluation of Dynamic Voltage and Frequency Scaling Adap-

tation Based on Memory Power. *IPSJ Trans. Adv. Comput. Syst.*, Vol. 5, No. 5, pp. 1–9, October 2012.

[15] Yuto Hayamizu, Kazuo Goda, Miyuki Nakano, and Masaru Kitsuregawa. Application-Aware Power Saving for Online Transaction Processing Using Dynamic Voltage and Frequency Scaling in a Multicore Environment. In *Proceedings of the 24th Architecture of Computing Systems (ARCS)*, pp. 50–61, February 2011.

[16] Justin Meza, Mehul A. Shah, Parthasarathy Ranganathan, Mike Fitzner, and Judson Veazey. Tracking the power in an enterprise decision support system. In *Proceedings of the 2009 ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 261–266, 2009.

[17] Meikel Poess and Raghunath Othayoth Nambiar. Tuning servers, storage and database for energy efficient data warehouses. In *Proceedigns of the 26th IEEE International Conference on Data Engineering (ICDE)*, pp. 1006–1017, 2010.

[18] Willis Lang, Ramakrishnan Kandhan, and Jignesh M. Patel. Rethinking Query Processing for Energy Efficiency: Slowing Down to Win the Race. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering (TCDE)*, Vol. 34, No. 1, pp. 12–23, March 2011.

[19] Mayuresh Kunjir, Puneet K. Birwa, and Jayant R. Haritsa. Peak Power Plays in Database Engines. In *Proceedings of the 15th International Conference on Extending Database Technology (EDBT)*, pp. 444–455, 2012.

[20] Zichen Xu, Yi-Cheng Tu, and Xiaorui Wang. PET: Reducing Database Energy Cost via Query Optimization. *VLDB*, Vol. 5, No. 12, pp. 1954–1957, August 2012.

[21] Amine Roukh, Ladjel Bellatreche, and Carlos Ordonez. EnerQuery: Energy-Aware Query Processing. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM)*, pp. 2465–2468, 2016.

[22] 早水悠登, 合田和生, 喜連川優. ストレージ消費電力特性に基づく関係データベース演算子の省電力指向コストモデル. 第 9 回データ工学と情報マネジメントに関するフォーラム (DEIM), March 2017.

[23] Rafael Alonso and Sumit Ganguly. Energy Efficient Query Optimization. Technical report, Matsushita Info Tech Lab., 1992.

[24] Dimitris Tsirogiannis, Stavros Harizopoulos, and Mehul A. Shah. Analyzing the Energy Efficiency of a Database Server. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, pp. 231–242, June 2010.

[25] Sebastian Götz, Thomas Ilsche, Jorge Cardoso, Josef Spillner, Thomas Kissinger, Uwe Aßmann, Wolfgang Lehner, Wolfgang E. Nagel, and Alexander Schill. Energy-Efficient Databases Using Sweet Spot Frequencies. In *UCC '14*, pp. 871–876, February 2014.

[26] Ioannis Manousakis, Manolis Marazakis, and Angelos Bilas. FDIO: A Feedback Driven Controller for Minimizing Energy in I/O-Intensive Applications. *HotStorage '13*, June 2013.

[27] Xiaobo Fan, Wolf-Dietrich Weber, and Luiz Andre Barroso. Power Provisioning for a Warehouse-sized Computer. In *Proceedings of the 34th Annual International Symposium on Computer Architecture (ISCA)*, pp. 13–23, 2007.

[28] Michael J. Flynn and Patrick Hung. Microprocessor Design Issues: Thoughts on the Road Ahead. *IEEE Micro*, Vol. 25, No. 3, pp. 16–31, July 2005.

[29] Anton Beloglazov, Rajkumar Buyya, Young Choon Lee, and Albert Y. Zomaya. A taxonomy and survey of energy-efficient data centers and cloud computing systems. *Advances in Computers*, Vol. 82, , March 2011.

発表文献

査読あり国内論文誌

1. 羅博明，早水悠登，合田和生，喜連川優．プロセッサ動作モード制御による分析指向問合せ処理の省電力化効果の測定．日本データベース学会和文論文誌，Vol. 17-J，Article No. 3，2019 年 3 月．

査読あり国際会議

1. Boming Luo, Yuto Hayamizu, Kazuo Goda, Masaru Kitsuregawa. Modeling Query Energy Costs in Analytical Database Systems with Processor Speed Scaling. 29th International Conference on Database and Expert Systems Applications (DEXA 2018), Regensburg, Germany, 2018.

査読あり国内会議

1. 羅博明，早水悠登，合田和生，喜連川優．データベースシステムにおける分析指向問合せ処理のプロセッサ動作モードを考慮した消費エネルギーモデル．The 2nd. cross-disciplinary Workshop on Computing Systems, Infrastructures, and Programming (xSIG 2018)，東京，2018 年．

査読なし国内会議

1. 羅博明，早水悠登，合田和生，喜連川優．プロセッサ動作モード制御による分析指向問合せ処理の省電力化効果の測定．第 10 回データ工学と情報マネジメントに関するフォーラム（DEIM 2018），G1-2，福井，2018 年．

2. 羅博明，早水悠登，合田和生，喜連川優．分析問合せ処理の資源律速ならびに消費電力の特性に関する考察．第 11 回データ工学と情報マネジメントに関するフォーラム（DEIM 2019），長崎，2019 年（発表予定）．