

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4460319号
(P4460319)

(45) 発行日 平成22年5月12日(2010.5.12)

(24) 登録日 平成22年2月19日(2010.2.19)

(51) Int.Cl.

F I

G 0 6 F 11/34 (2006.01)

G 0 6 F 11/34

S

請求項の数 16 (全 17 頁)

(21) 出願番号 特願2004-30038 (P2004-30038)
 (22) 出願日 平成16年2月6日(2004.2.6)
 (65) 公開番号 特開2005-222343 (P2005-222343A)
 (43) 公開日 平成17年8月18日(2005.8.18)
 審査請求日 平成18年10月10日(2006.10.10)

(73) 特許権者 000005108
 株式会社日立製作所
 東京都千代田区丸の内一丁目6番6号
 (74) 代理人 110000350
 ポレール特許業務法人
 (74) 代理人 100068504
 弁理士 小川 勝男
 (74) 代理人 100086656
 弁理士 田中 恭助
 (74) 代理人 100094352
 弁理士 佐々木 孝
 (72) 発明者 ▲高▼野 英樹
 神奈川県川崎市麻生区王禅寺1099番地
 株式会社日立製作所 システム開発
 研究所内

最終頁に続く

(54) 【発明の名称】 チューニング制御方法及びシステム

(57) 【特許請求の範囲】

【請求項1】

階層構成をもつ複数のプログラムから構成される多階層システムの性能のチューニング制御システムであって、前記チューニング制御システムは、計算機によって実行されるプログラムの構成要素として、

前記多階層システムを構成する各プログラムの動作のパラメータを設定するパラメータ設定部と、前記各プログラムを起動、停止するプログラム起動・停止部と、前記多階層システムの性能情報を計測する性能情報計測部と、前記多階層システムを構成するプログラム実行中に発生する不定期の処理が実行される時間帯を監視する不定期処理監視部と、前記パラメータ設定部、前記プログラム起動・停止部、前記性能情報計測部及び前記不定期処理監視部を起動し、前記多階層システムを稼働させることによって実験する実験制御部と、前記性能情報の計測結果に対して次の実験に使用する前記パラメータの組合せである次パラメータセットを決定する次パラメータセット決定部と、所定の終了条件を満たすまで前記次パラメータセットを決定し前記実験を繰り返すように前記実験制御部および前記次パラメータセット決定部を制御するチューニング制御部と
 を有し、

前記実験制御部は、前記不定期処理監視部が報告する前記時間帯にかかる前記性能情報を性能計測の対象から除外し、

前記次パラメータセット決定部は、前記時間帯以外の前記性能情報が最適化されるように前記パラメータセットを決定することを特徴とするチューニング制御システム。

10

20

【請求項 2】

前記多階層システムは、Webサーバ、アプリケーションサーバ及びデータベースサーバの各プログラムを有するWebシステムであることを特徴とする請求項 1 記載のチューニング制御システム。

【請求項 3】

前記チューニング制御システムは、前記性能情報計測部を含み前記多階層システムに対して同時に複数のリクエストを発行することによって負荷をかけるプログラム構成要素として負荷かけツールを有することを特徴とする請求項 1 記載のチューニング制御システム。

【請求項 4】

前記チューニング制御部がパラメータ数 + 1 個の初期パラメータセットを用いて初期実験を行うよう制御して各初期パラメータセットに対応する性能情報を取得した後に、前記次パラメータセット決定部は、取得した前記性能情報に滑降シンプレックス法を適用して前記性能情報を最適化するようなパラメータ空間上の 1 点に向かってパラメータセットを収束させることを特徴とする請求項 1 記載のチューニング制御システム。

【請求項 5】

所定回数の前記実験によって最良のパラメータセットが変更されないとき、前記チューニング制御部は、前記最良のパラメータセットについてもう一度実験を行い、計測された性能情報と以前に計測された前記最良のパラメータセットに対応する性能情報との平均値を、前記最良のパラメータセットの性能情報として前記次パラメータセット決定部に制御を渡すことを特徴とする請求項 1 記載のチューニング制御システム。

【請求項 6】

前記チューニング制御部は、前記パラメータ数 + 1 個より多い個数の前記初期パラメータセットを用いて前記初期実験を行うよう制御した後に、前記性能情報の良い順に前記パラメータ数 + 1 個の初期パラメータセットを選択して前記次パラメータセット決定部に制御を渡すことを特徴とする請求項 4 記載のチューニング制御システム。

【請求項 7】

前記次パラメータセット決定部は、算出された前記次パラメータセットの各パラメータの値を所定の整数倍することによって前記次パラメータセットを補正することを特徴とする請求項 4 記載のチューニング制御システム。

【請求項 8】

前記次パラメータセット決定部は、同一の最良点のパラメータセットが連続したとき、当該パラメータセットの各パラメータをパラメータ値に比べて小量だけランダムに増減するような補正をすることを特徴とする請求項 4 記載のチューニング制御システム。

【請求項 9】

階層構成をもつ複数のプログラムから構成される多階層システムの性能のチューニング制御方法であって、前記チューニング制御方法は、計算機によるプログラム実行によって実現され、

前記多階層システムを構成する各プログラムの動作のパラメータを設定するステップと、

前記各プログラムを起動するステップと、

前記多階層システムの性能情報を計測するステップと、

前記多階層システムを構成するプログラム実行中に発生する不定期の処理が実行される時間帯を監視するステップと、

前記性能情報を計測するステップで計測された性能情報から、前記監視するステップで報告された前記時間帯にかかる前記性能情報を性能計測の対象から除外するステップと、

前記各プログラムを停止するステップと、

前記性能情報の計測結果に対して次の前記多階層システムの稼動に用いる前記パラメータの組合せである次パラメータセットを決定するステップとを有し、前記次パラメータセットを決定するステップは、前記時間帯以外の前記性能情報が最適化されるように前記パ

10

20

30

40

50

ラメータセットを決定するものとし、

所定の終了条件を満たすまで前記ステップを順に繰り返すことを特徴とするチューニング制御方法。

【請求項 10】

前記多階層システムは、Webサーバ、アプリケーションサーバ及びデータベースサーバの各プログラムを有するWebシステムであることを特徴とする請求項9記載のチューニング制御方法。

【請求項 11】

前記多階層システムを構成する各プログラムが起動された後に、前記多階層システムに対して同時に複数のリクエストを発行することによって負荷をかけることを特徴とする請求項9記載のチューニング制御方法。

10

【請求項 12】

前記パラメータを設定するステップは、最初にはパラメータ数 + 1 個の初期パラメータセットの各々を設定し、前記初期パラメータセットについて前記性能情報を計測し、前記初期パラメータセットについての前記性能情報を取得した後に、前記次パラメータセットを決定するステップは、取得した前記性能情報に滑降シンプレックス法を適用して前記性能情報を最適化するようなパラメータ空間上の1点に向かってパラメータセットを収束させるように前記次パラメータセットを決定することを特徴とする請求項9記載のチューニング制御方法。

【請求項 13】

20

前記ステップを順に繰り返し実行している間に所定回数の前記多階層システムの稼動によって最良のパラメータセットが変更されないことを検出したとき、前記最良のパラメータセットについても一度前記多階層システムを稼動させ、計測された性能情報と以前に計測された前記最良のパラメータセットに対応する性能情報との平均値を、前記最良のパラメータセットの性能情報として前記次パラメータセットを決定するステップを実行することを特徴とする請求項9記載のチューニング制御方法。

【請求項 14】

前記次パラメータセットを決定するステップは、最初にはパラメータ数 + 1 個より多い個数の前記初期パラメータセットの各々を設定し、前記初期パラメータセットについて前記性能情報を計測し、前記初期パラメータセットについての前記性能情報を取得した後に、前記性能情報の良い順に前記パラメータ数 + 1 個の初期パラメータセットを選択して前記次パラメータセットを決定するステップを実行することを特徴とする請求項12記載のチューニング制御方法。

30

【請求項 15】

前記次パラメータセットを決定するステップは、算出された前記次パラメータセットの各パラメータの値を所定の整数倍することによって前記次パラメータセットを補正することを特徴とする請求項12記載のチューニング制御方法。

【請求項 16】

前記次パラメータセットを決定するステップは、同一の最良点のパラメータセットが連続したとき、当該パラメータセットの各パラメータをパラメータ値に比べて小量だけランダムに増減するように補正をすることを特徴とする請求項12記載のチューニング制御方法。

40

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、階層構成をもつ複数のプログラムから構成される多階層システムの動作パラメータを最適化するチューニング方式に関するものである。

【背景技術】

【0002】

企業の統廃合や様々な事業展開の積み重ねにより、それを支える企業情報システムは複

50

雑になってきている。このような環境にあって、少ないコストで高いパフォーマンスを実現するためには、連携して動作する複数のプログラムがバランス良く動作するようにパフォーマンスチューニングを実施することが重要である。

【 0 0 0 3 】

プログラムの動作やハードウェアの動作を詳細に熟知したエキスパートであれば、システムを稼働させたときに計測可能な情報と、その計測情報にある特徴があったときに設定すべきパラメータのルールを予めファジィ推論ベースとして構築することでチューニングを実施することは可能である。例えば特許文献 1 は、このようなパフォーマンスチューニング方式を開示する。

【 0 0 0 4 】

しかし複雑化したシステム全体を熟知するようなエキスパートはなかなか存在しないし、存在したとしても、チューニングに必要な全てのルールを定義することは非常に困難である。

【 0 0 0 5 】

【特許文献 1】特開平 1 1 - 2 4 9 9 3 4 号公報

【発明の開示】

【発明が解決しようとする課題】

【 0 0 0 6 】

複数のプログラムから構成される多階層システムに対して、各プログラムの最適な動作パラメータを見つけることは困難である。

【課題を解決するための手段】

【 0 0 0 7 】

本発明は、階層構成をもつ複数のプログラムから構成される多階層システムの性能のチューニング制御技術であって、この技術は、計算機によるプログラム実行によって実現され、多階層システムを構成する各プログラムの動作のパラメータを設定し、各プログラムを起動し、多階層システムの性能情報を計測し、多階層システムを構成するプログラム実行中に発生する不定期の処理が実行される時間帯を監視し、計測された性能情報から、不定期の処理が実行される時間帯にかかる性能情報を除外し、各プログラムを停止した後に、性能情報の計測結果に対して次の多階層システムの稼働に用いるパラメータの組合せである次パラメータセットを決定し、この次パラメータセットを決定するステップは、前記時間帯以外の性能情報が最適化されるようにパラメータセットを決定するものとし、所定の終了条件を満たすまで上記のステップを順に繰り返すチューニング制御技術を特徴とする。

【発明の効果】

【 0 0 0 8 】

本発明によれば、最適な動作パラメータを求めるのが困難な複数のプログラムから構成される多階層システムであっても、誰でも容易に最適なパラメータセットを求めることができる。

【発明を実施するための最良の形態】

【 0 0 0 9 】

図 1 は、本実施形態のチューニングシステムの構成図である。多階層システム 1 1 0 は、チューニングの対象とするシステムであり、そのシステムの構成要素であるプログラム 1 1 2 - 1 , 1 1 2 - 2 , ... 1 1 2 - n が稼働する。各プログラム 1 1 2 - 1 ~ 1 1 2 - n は、そのプログラムの動作を設定するための動作パラメータであるパラメータ 1 1 1 - 1 ~ 1 1 1 - n を有する。

【 0 0 1 0 】

これらパラメータをチューニングするためのチューニング制御システムは、チューニング制御部 1 0 1、実験制御部 1 0 2、次パラメータセット決定部 1 0 3、パラメータ設定部 1 0 4、プログラム起動・停止部 1 0 5 および性能情報計測部 1 0 6 の各プログラムを含む。パラメータ設定部 1 0 4 は、各パラメータ 1 1 1 - 1 ~ 1 1 1 - n に値を設定する

10

20

30

40

50

。プログラム起動・停止部 105 は、多階層システム 110 のプログラム 112 - 1 ~ 112 - n を起動、停止するプログラムである。性能情報計測部 106 は、多階層システム 110 に対してリクエストを送出するなどして稼働させ、その性能情報を計測する。実験制御部 102 は、パラメータ設定部 104、プログラム起動・停止部 105 および性能情報計測部 106 を動作させて、実験を実施するプログラムである。次パラメータセット決定部 103 は、最適化アルゴリズムを利用し、計測した性能情報に基づいて次に実験すべきパラメータを求めるプログラムである。チューニング制御部 101 は、実験制御部 102 及び次パラメータセット決定部 103 を制御してチューニングのための実験を遂行するプログラムである。以下プログラム 112 - 1 ~ 112 - n のいずれかを指すときには、プログラム 112 のように表記することがある。他の構成要素についても同様である。

10

【0011】

プログラム 112 - 1 ~ 112 - n 及びチューニング制御システム側の 101 から 106 までの構成要素は、計算機で稼働可能なプログラムであり、1つ又は複数の計算機に配置することができる。図 2 は、使用される計算機 200 のハードウェア構成を示す図である。ハードディスク 203 は、各プログラム又はパラメータなどの各種データを保存する。中央処理装置 202 は、これらハードディスク 203 に蓄えた各種データを必要に応じてメモリ 201 に読み込む。プログラムを実行する場合も、プログラムをまずメモリ 201 に読み込み、中央処理装置 202 により実行する。ビデオメモリ 204 は、表示に用いられるデータを格納するメモリであり、ビデオメモリ 204 上のデータの変更は、表示装置 205 に反映される。

20

【0012】

図 3 にチューニングシステムのシーケンス図を示す。チューニング制御部 101 は、実験制御部 102 に実験要求を行う。実験制御部 102 は、パラメータ設定部 104 にパラメータ設定要求を行う。パラメータ設定部 104 は、各構成要素のプログラム 112 - 1 ~ 112 - n が管理する設定ファイルなどにパラメータを設定する。次に実験制御部 102 は、プログラム起動・停止部 105 に対してプログラム起動要求を行い、構成要素のプログラム 112 - 1 ~ 112 - n を起動する。次に実験制御部 102 は、性能情報計測部 106 に性能情報計測要求を発行する。性能情報計測部 106 は、プログラム 112 - 1 ~ 112 - n を稼働させることによって多階層システム 110 の性能情報を計測する。ここで多階層システム 110 は、プログラム 112 - 1 ~ 112 - n が実行される 1 台またはそれ以上の計算機 200 を稼働させ実際の処理を行う。実験制御部 102 は、計測した性能情報を取得する。次に実験制御部 102 は、プログラム起動・停止部 105 に対してプログラム停止要求を行い、プログラム起動・停止部 105 は、構成要素のプログラムを停止する。チューニング制御部 101 は、次パラメータ決定部 103 に対して次パラメータを要求する。次パラメータ決定部 103 は、計測した性能情報を用いて次に実験すべきパラメータセットを求めてチューニング制御部 101 に返す。チューニング制御部 101 は、このパラメータを用いて再度実験要求からの処理を繰り返す。チューニング制御部 101 は、終了条件を満たすまで実験を繰り返すことによって性能計測値が最も良くなるパラメータセットを求める。

30

【実施例 1】

40

【0013】

図 4 は、Web システムに対して性能のチューニングを実施するシステムの構成図である。システムは、Web システム 440、チューニング制御プログラム 402 を稼働させる計算機 401 および負荷かけ計算機群 420 から構成される。Web システム 440 は、計算機 450、計算機 460 および計算機 470 から構成され、これらの計算機はそれぞれネットワークで接続される。計算機 450 上では Web サーバ 451 が稼働し、計算機 460 上では AP (アプリケーション) サーバ 461 が稼働し、計算機 470 上では DB (データベース) サーバ 471 が稼働する。この例では、Web システム 440 が 3 階層の多階層システム 110 に相当し、Web サーバ 451 がプログラム 112 - 1 に、AP サーバ 461 がプログラム 112 - 2 に、DB サーバ 471 がプログラム 112 - 3 に

50

各々相当する。これらのプログラムが連携して動作するWebシステム440がチューニング対象である。負荷かけ計算機群420は、計算機421-1～421-3から構成される。計算機421-1～421-3は、各々負荷かけツール431-1～431-3の各プログラムを稼働させる。負荷かけツール431-1～431-3の各々は、Webシステム440のWebサーバ451へ同時に多数のリクエストを発行してWebシステム440の各プログラムに負荷をかけ、レスポンスを受け取るまでのレスポンスの状況から性能情報を計測する。すなわち負荷かけツール431-1～431-3の各々は、性能情報計測部106を含む。チューニング制御プログラム402は、負荷かけ計算機群420およびWebシステム440を制御する。チューニング制御プログラム402は、チューニング制御部101、実験制御部102、次パラメータセット決定部103、パラメータ設定部104およびプログラム起動・停止部105を含む。

10

【0014】

実施例1では、チューニングの対象とするパラメータとして、Webサーバ451の最大同時処理数、APサーバ461の最大同時処理数、およびAPサーバ461がDBサーバ471とのコネクションを保持する最大プール数の3つとする。最大同時処理数は、リクエスト発行に対応してそのプログラムで同時に実行されるプロセス、スレッドまたはタスクの最大数である。チューニング制御プログラム402は、Webシステム440の性能を最適にするようなパラメータの組合せ(パラメータセット)を求める。本実施例は、最適化アルゴリズムとして、滑降シンプレックス法を用いる。

【0015】

20

図5は、チューニング制御プログラム402の動作を設定する設定ファイルのデータ例を示す図である。最大実験回数(max_experiment_times)は、実験を打切るときの最大実施回数である。パラメータセットは、本実施例では3次元のパラメータ空間上の点によって表現できる。距離(distance)は、最良点のパラメータセットと他の3つのパラメータセットの各々との間の最適化されたときの距離を設定する。この3つの距離がすべてdistanceの設定値以内になったとき、性能情報が最適値に収束したと判断される。初期パラメータセット(init_param.1～4)は、実験開始時に与えられるパラメータであり、ここではパラメータ数+1だけ用意される。パラメータセットの値は、上記の3つのパラメータを順に設定する。設定ファイルは、計算機401の記憶装置に格納される。

【0016】

30

図6は、チューニング制御プログラム402の処理手順を示すPAD(problem analysis diagram

)である。チューニング制御プログラム402のチューニング制御部101は、まず設定ファイルから初期パラメータセットを読み込む(ステップ601)。次にチューニング制御部101は、ステップ603の処理をすべての初期パラメータセットについて繰り返し、性能情報を取得する(ステップ602)。ここでは4つの初期パラメータセットがあるからステップ603を4回繰り返す。性能情報は、スループット又は平均レスポンスタイムである。スループットは、単位時間に行うことができるリクエストの数である。平均レスポンスタイムは、負荷かけツール431がリクエストを発行してからレスポンスを受け取るまでの所要時間の平均値である。ステップ603では、チューニング制御部101は、実験制御部102を呼び出す。実験制御部102は、パラメータ設定部104を起動し、Webシステム440に初期パラメータセットを設定し、Webシステム440を稼働して性能情報を取得する実験を行う。

40

【0017】

次にチューニング制御部101は、終了条件に達するまでステップ605とステップ606の処理を繰り返し実行するよう制御する(ステップ604)。ここで終了条件とは、上記の距離の条件を満足するか、または実験回数が最大実験回数に達することである。ステップ605では、チューニング制御部101は、次パラメータセット決定部103を起動する。次パラメータセット決定部103は、先の実験結果から次の実験のためのパラメータセットを決定する。パラメータセットの決定には滑降シンプレックス法が適用される

50

。ステップ606では、ステップ603と同様に、チューニング制御部101は、実験制御部102を呼び出す。実験制御部102は、パラメータ設定部104を起動し、決定された次パラメータセットをWebシステム440に設定し、Webシステム440を稼働して性能情報を取得する実験を行う。終了条件に達したとき、チューニング制御部101は、これまで計測した中で性能情報が最良であったパラメータセットを最適なパラメータセットとして表示装置205に出力する(ステップ607)。

【0018】

図7は、ステップ603およびステップ606の処理のシーケンスを示す図である。実験制御部102は、パラメータ設定部104を介してWebサーバ451、APサーバ461およびDBサーバ471に各々パラメータを設定する。次に実験制御部102は、プログラム起動・停止部105を介してDBサーバ471、APサーバ461およびWebサーバ451の順で各プログラムを起動する。次に実験制御部102は、負荷かけツール431-1~431-3に対して負荷かけを要求する。負荷かけツール431-1~431-3は、Webシステム440に繰り返しリクエストを発行する。各負荷かけツール431は、複数スレッドを用いてほぼ同時に複数のリクエストを発行する。性能情報計測部106は、返ってきたレスポンスの時刻から性能情報を算出し、算出された性能情報を実験制御部102に報告する。次に実験制御部102は、プログラム起動・停止部105を介してWebサーバ451、APサーバ461およびDBサーバ471の各プログラムを停止する。後述するようにステップ605で複数の次パラメータセットが決定された場合には、次パラメータセットの数だけ実験を繰り返す。

【0019】

図8は、ステップ605の次パラメータセットを決定する処理の手順を示すPADである。次パラメータセット決定部103は、実験に用いたパラメータセット及び計測した性能情報を取得する(ステップ801)。以下に述べるように収縮の場合には、複数のパラメータセットと各々に対応する性能情報を取得する。次に次パラメータセット決定部103は、パラメータセット群の各パラメータセットを性能情報が良い順に配列に格納する(ステップ802)。従って配列の先頭には最良点のパラメータセットが配置される。性能情報がより良いとは、スループットの場合にはスループットがより大きいことである。レスポンスタイムの場合にはレスポンスタイムがより小さいことである。ここでパラメータセット群とは、滑降シンプレックス法で用いるパラメータ数+1個のパラメータセットのことである。今の例ではパラメータセット群は4個のパラメータセットから成る。すでに配列にパラメータセット群が格納されている場合には、この例では既存のパラメータセット群とステップ801で取得したパラメータセットのうち性能情報が良いもの4個を選択してこの配列に再格納する。次に次パラメータセットを保持するバッファを作成する(ステップ803)。次に次パラメータセット決定部103は、実験の種類を取得し、そのバッファに格納する(ステップ804)。初期状態における実験の種類は、「null」が格納される。

【0020】

実験の種類が「null」であった場合(ステップ810)、次パラメータセット決定部103は、実験の種類を「反射」に変更し(ステップ811)、後述の手順によってパラメータ空間上の今対象としている点の2倍反射の点を求める(ステップ812)。このときの引数は-1,0である。次に次パラメータセットのバッファに反射の位置を格納する(ステップ813)。次に次パラメータセット決定部103は、チューニング制御部101に次パラメータセットを返す(ステップ860)。

【0021】

実験の種類が「反射」であった場合(ステップ820)、反射の位置の性能情報が最悪点よりも良い場合(ステップ821YES)、次パラメータセット決定部103は、最悪点を反射の位置の点に置き換える(ステップ823)。反射の位置の性能情報が最良点よりも良い場合(ステップ823YES)、次パラメータセット決定部103は、実験の種類を「2倍反射」に変更する(ステップ824)。次に後述の手順によってパラメータ空

間上の4点のうちの最悪点の2倍反射の位置を求める(ステップ812)。このときの引数は-2.0である。次に次パラメータセット決定部103は、次パラメータセットのバッファに2倍反射の位置を格納する(ステップ825)。

【0022】

反射の位置の性能情報が最良点よりも悪い場合(ステップ823NO)で、反射の位置の性能情報が2番目に悪い点より悪い場合(ステップ826YES)、次パラメータセット決定部103は、実験の種類を「内側反射」に変更する(ステップ827)。次に後述の手順によってパラメータ空間上の4点のうちの最悪点の内側反射の位置を求める(ステップ812)。このときの引数は0.5である。次に次パラメータセット決定部103は、次パラメータセットのバッファに内側反射の位置を格納する(ステップ828)。反射の位置の性能情報が2番目に悪い点と同じかより良い場合(ステップ826NO)、実験の種類を「null」に変更し、もう一度、次パラメータセットの決定(ステップ605)を実行する。

【0023】

実験の種類が「2倍反射」であった場合(ステップ830)、2倍反射の位置の性能情報が最悪点よりも良いならば(ステップ831)、次パラメータセット決定部103は、最悪点を2倍反射の位置の点に置き換える(ステップ833)。次に次パラメータセット決定部103は、実験の種類を「null」に変更し、もう一度、次パラメータセットの決定(ステップ605)を実行する。

【0024】

実験の種類が「内側反射」であった場合(ステップ840)、内側反射の位置の性能情報が最悪点より良いならば(ステップ841YES)、次パラメータセット決定部103は、最悪点を内部反射の位置の点に置き換える(ステップ842)。次に次パラメータセット決定部103は、実験の種類を「null」に変更し(ステップ843)、もう一度、次パラメータセットの決定(ステップ605)を実行する。内側反射の位置の性能情報が最悪点よりも悪ければ(ステップ841NO)、次パラメータセット決定部103は、実験の種類を「収縮」に変更する(ステップ844)。次に後述の手順によってパラメータ空間上の4点から収縮の位置を求める(ステップ845)。このときの引数は0.5である。次に次パラメータセット決定部103は、次パラメータセットのバッファに収縮の点の位置(複数)を格納する(ステップ846)。

【0025】

実験の種類が「収縮」であった場合(ステップ850)、次パラメータセット決定部103は、最良点以外の点を収縮の位置の点に置き換える(ステップ851)。次に次パラメータセット決定部103は、実験の種類を「null」に変更し(ステップ852)、もう一度、次パラメータセットの決定(ステップ605)を実行する。

【0026】

各処理ステップから最後のステップ860に達したとき、次パラメータセット決定部103は、チューニング制御部101に次パラメータセットを返す。

【0027】

図9は、ステップ812の反射の位置を求める処理の手順を示すPADである。ここでは、簡単のためにパラメータセット群が(a1,b1,c1) (a2,b2,c2) (a3,b3,c3) (a4,b4,c4)で、この中の最悪点が(a2,b2,c2)であった場合について説明する。引数としてfacを取る。これは反射の距離を表す引数であり、「反射」の位置の場合は-1.0、「2倍反射」の場合は-2.0、「内側反射」の場合は0.5を取る。以下の手順で計算する。

【0028】

a=a1+a2+a3+a4	(数1)
b=b1+b2+b3+b4	(数2)
c=c1+c2+c3+c4	(数3)
fac1=(1.0-fac)/3	(3はパラメータ数) (数4)
fac2=fac1-fac	(数5)

10

20

30

40

50

```

ar=a×ac1-a2×fac2          ( 数 6 )
br=b×fac1-b2×fac2          ( 数 7 )
cr=c×fac1-c2×fac2          ( 数 8 )

```

(ar,br,cr)が反射の位置の点になる。

【 0 0 2 9 】

図 1 0 は、ステップ 8 4 5 の収縮の位置を求める処理の手順を示す P A D である。ここでは、簡単のためにパラメータセット群が(a1,b1,c1) (a2,b2,c2) (a3,b3,c3) (a4,b4,c4)で、この中の最良点が(a1,b1,c1)であった場合について説明する。引数として、facを取る。「収縮」の場合は、facは0.5となる。

【 0 0 3 0 】

```

ar1=fac×(a2+a1)            ( 数 9 )
br1=fac×(b2+b1)            ( 数 1 0 )
cr1=fac×(c2+c1)            ( 数 1 1 )
ar2=fac×(a3+a1)            ( 数 1 2 )
br2=fac×(b3+b1)            ( 数 1 3 )
cr2=fac×(c3+c1)            ( 数 1 4 )
ar3=fac×(a4+a1)            ( 数 1 5 )
br3=fac×(b4+b1)            ( 数 1 6 )
cr3=fac×(c4+c1)            ( 数 1 7 )

```

以上から求まった(ar1,br1,cr1) (ar2,br2,cr2) (ar3,br3,cr3)の3点が収縮の位置となる。

【 実施例 2 】

【 0 0 3 1 】

実施例 1 において、複数回の実験に亘って最良点が同じ点であった場合、最良点が本当の最良点か、実験の結果として偶然良い結果が出てしまったのが不明というケースがある。実施例 2 は、その最良点のパラメータセットを用いてもう一度実験し、計測した性能情報と前回の最良点の性能情報の平均を、その点の最良点とするものである。

【 0 0 3 2 】

図 1 1 は、実施例 2 のチューニング制御プログラム 4 0 2 の動作を設定する設定ファイルのデータ例を示す図である。実施例 1 の設定ファイルの例と比べて最良点であり続けた回数(top_times)という設定項目が加わっている。

【 0 0 3 3 】

図 1 2 は、実施例 2 のチューニング制御プログラム 4 0 2 の処理手順を示す P A D である。ステップ 6 0 1 ~ 6 0 3 の処理の後に、チューニング制御部 1 0 1 は、前回最良点のパラメータセット及び前回性能情報を格納するバッファを用意する(ステップ 1 2 0 1)。次にチューニング制御部 1 0 1 は、最良点であり続けた回数を計数するカウンタを用意する(ステップ 1 2 0 2)。カウンタの初期値は 0 である。次にチューニング制御部 1 0 1 は、終了条件に達するまでステップ 1 2 0 3 ~ 1 2 0 7 (ステップ 6 0 6 も含む)の処理を繰り返し実行するよう制御する(ステップ 6 0 4)。

【 0 0 3 4 】

前回の最良点がある場合に、前回の最良点と今回の最良点が同じであれば(ステップ 1 2 0 3 Y E S)、チューニング制御部 1 0 1 は、カウンタに 1 を加算する(ステップ 1 2 0 4)。次にカウンタが回数(top_times)より大きければ(ステップ 1 2 0 5)、次パラメータセットとして最良点のパラメータセットを再び次の実験に採用する(ステップ 1 2 0 6)。ステップ 6 0 6 では、チューニング制御部 1 0 1 は、次パラメータセット決定部 1 0 3 を起動し、同一パラメータセットの実験をもう一度行う。実験の結果として性能情報が前回と同じとは限らない。次にチューニング制御部 1 0 1 は、前回の性能情報と今回の性能情報の平均を取り、今回実験したパラメータセットの性能情報とする(ステップ 1 2 0 7)。前回の最良点がないか、前回の最良点と今回の最良点が異なれば(ステップ 1 2 0 3 N O)、チューニング制御部 1 0 1 は、バッファ中の前回の最良点のパラメータ

10

20

30

40

50

セット及び前回の性能情報を今回の最良点のパラメータセット及び性能情報に置き換える（ステップ1208）。前回の最良点がなければ、単にバッファに今回の最良点のパラメータセットとその性能情報を格納するだけである。次にチューニング制御部101は、カウンタを0に初期化する（ステップ1209）。

【0035】

実施例2によって滞留していた一時的な最良点から脱出し、本当の最良点に向かうことができる場合がある。

【実施例3】

【0036】

実施例3は、初期実験をパラメータ数+1回より多く行うことによって、最適解への収束を速くすることを目指すチューニング方法の実施例である。

10

【0037】

図13は、実施例3のチューニング制御プログラム402の動作を設定する設定ファイルのデータ例を示す図である。この例は、初期パラメータセットとして、init_param.1~init_param.10までの10個のパラメータセットを設定する。

【0038】

図14は、実施例3のチューニング制御プログラム402の処理手順を示すPADである。チューニング制御部101は、ステップ603の処理をすべての初期パラメータセットについて繰り返し、性能情報を取得する（ステップ602）。ここでは10個の初期パラメータセットがあるからステップ603を10回繰り返す。次にチューニング制御部101は、性能情報が良い順にパラメータ数+1個のパラメータセットを抜き出し、これらをパラメータセット群とする。以下、ステップ604~607は、実施例1と同様である。

20

【実施例4】

【0039】

JavaVMのような実行基盤の下で多階層システム110のプログラム112が実行される場合に、不要部分の整理（Garbage Collection（GC））のような不定期の処理が実行されることがある。実施例4は、このような不定期の処理が実行される時間帯にかかる性能計測結果をサンプリング対象から除外する実施例である。（注）Javaは、米国Sun Microsystems社の登録商標である。

30

【0040】

実施例4は、チューニング対象のWebシステム440のうち、APサーバ461がJavaVMを利用し、不定期にGCが発生する場合を例にとる。Webシステム440は、GC監視機能をもつプログラムを設け、JavaVMからGCが発生した時刻、終了した時刻を計測し、その情報を記憶装置に保持する。

【0041】

図15は、実施例4のステップ603およびステップ606の処理のシーケンスを示す図である。GC監視プログラム1500は、GCの発生時刻および終了時刻を計測し、その情報を保持する。負荷かけツール431は、リクエストの送出しの繰り返し実行が終了するとき、実験制御部102に各リクエストの実行情報を返す。この実行情報は、各リクエストの発生時刻および返ってきたレスポンスの時刻を含んでいる。次に実験制御部102は、各リクエストの実行情報とGCの発生時刻、終了時刻からGCの発生していない時間帯のリクエストの実行情報のみから性能情報を作成する。

40

【実施例5】

【0042】

実施例5は、次パラメータセット決定部103が導き出すパラメータの値を2の倍数、3の倍数というように特定の数の倍数となるように補正する実施例である。これによってパラメータセットは早く最適解付近の解に収束することが可能となる。

【0043】

図16は、実施例5のチューニング制御プログラム402の動作を設定する設定ファイ

50

ルのデータ例を示す図である。この例は、実施例 1 のデータ例にパラメータの取り得る値の粒度(param_step)を追加する。この例は粒度を 3 とするので、パラメータは 3 の倍数のみを取り得る。ただしこの例では初期パラメータは、実施例 1 と同じデータを用いており、3 の倍数に限定していない。

【 0 0 4 4 】

次パラメータセット決定部 1 0 3 は、チューニング制御部 1 0 1 に次のパラメータセットを返す前に、パラメータがparam_stepで指定された値の倍数となるように補正する。その補正方法は、例えば算出されたパラメータセットの各パラメータの値をparam_stepで割った後に四捨五入又は小数点以下切捨てなどをして整数にした後、その値にparam_stepを掛けることによって次パラメータの値を求めるものである。あるいはその他の例として、算出されたパラメータセットの各パラメータの値をparam_stepで割って得られた余りを元の値から引くという方法もある。

10

【実施例 6】

【 0 0 4 5 】

実施例 6 は、次パラメータセット決定部 1 0 3 が導き出すパラメータセットが最良点のパラメータセットと同じであったとき、そのパラメータセットの各パラメータの値をランダムに微小量だけ増減するように補正する実施例である。微小量は例えば + 1 もしくは - 1 などである。これによって最良点付近を詳細に探索することが可能となる。

【 0 0 4 6 】

図 1 7 は、実施例 6 の次パラメータセットを決定する処理 (ステップ 6 0 5) の手順を示す P A D である。次パラメータセット決定部 1 0 3 は、ステップ 8 6 0 の処理の前に、次パラメータセットと最良点のパラメータセットが同じ場合 (ステップ 1 7 0 1)、次パラメータセットのすべての点のパラメータ値をランダムに + 1 もしくは - 1 増減する処理を行う (ステップ 1 7 0 2)。

20

【 0 0 4 7 】

本実施形態によれば、多階層システムの運用を開始する前、ハードウェアのリブレースなどの構成変更時、稼動するアプリケーションの変更時などに、誰でも容易に最適なチューニングを行うことが可能となる。本発明は、最近、重要度が増してきている W e b システムのチューニングにも適用できる。チューニングの対象とするパラメータとして、W e b サーバ、A P サーバ、D B サーバの最大同時処理数のほかに、各サーバの保持するキューに関して最大キュー数をチューニングすることも可能である。

30

【図面の簡単な説明】

【 0 0 4 8 】

【図 1】実施形態のチューニングシステムの構成図である。

【図 2】計算機のハードウェア構成を示す図である。

【図 3】チューニングのシーケンスを示す図である。

【図 4】W e b システムをチューニングする場合のシステム構成図である。

【図 5】チューニング制御プログラムが用いる設定ファイルの例を示す図である。

【図 6】実施例 1 のチューニング制御プログラムの処理手順を示す P A D である。

【図 7】実施例 1 の実験制御部による処理のシーケンスを示す図である。

40

【図 8】実施例 1 の次パラメータセットを決定する処理の手順を示す P A D である。

【図 9】反射の位置を求める処理の手順を示す P A D である。

【図 1 0】収縮の位置を求める処理の手順を示す P A D である。

【図 1 1】実施例 2 のチューニング制御プログラムの設定ファイルの例を示す図である。

【図 1 2】実施例 2 のチューニング制御プログラムの処理手順を示す P A D である。

【図 1 3】実施例 3 のチューニング制御プログラムの設定ファイルの例を示す図である。

【図 1 4】実施例 3 のチューニング制御プログラムの処理手順を示す P A D である。

【図 1 5】実施例 4 の実験制御部による処理のシーケンスを示す図である。

【図 1 6】実施例 5 のチューニング制御プログラムの設定ファイルの例を示す図である。

【図 1 7】実施例 6 の次パラメータセットを決定する処理の手順を示す P A D である。

50

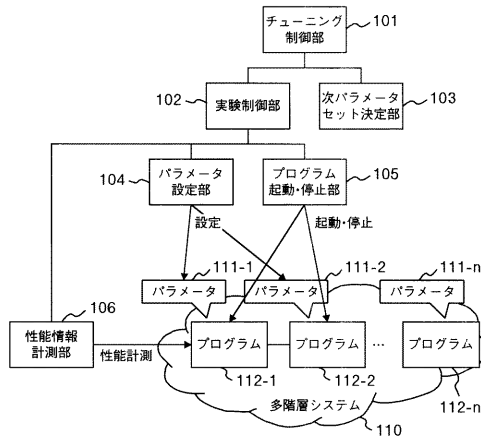
【符号の説明】

【0049】

101：チューニング制御部、102：実験制御部、103：次パラメータセット決定部、104：パラメータ設定部、105：プログラム起動・停止部、106：性能情報計測部、110：多階層システム、111-1～111-n：パラメータ、112-1～112-n：プログラム、402：チューニング制御プログラム、440：Webシステム、431-1～431-3：負荷かけツール。

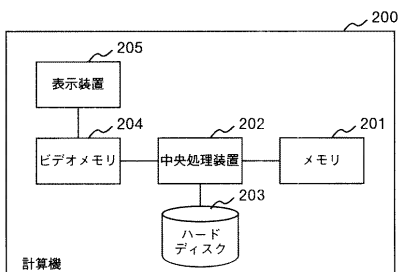
【図1】

図 1



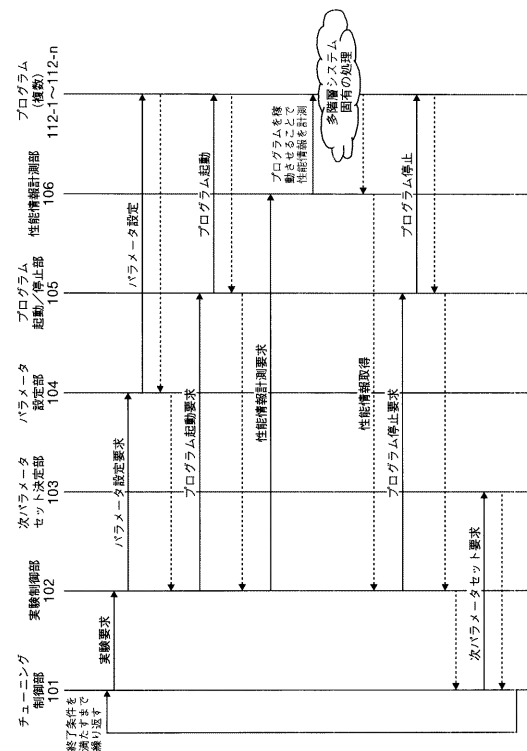
【図2】

図 2



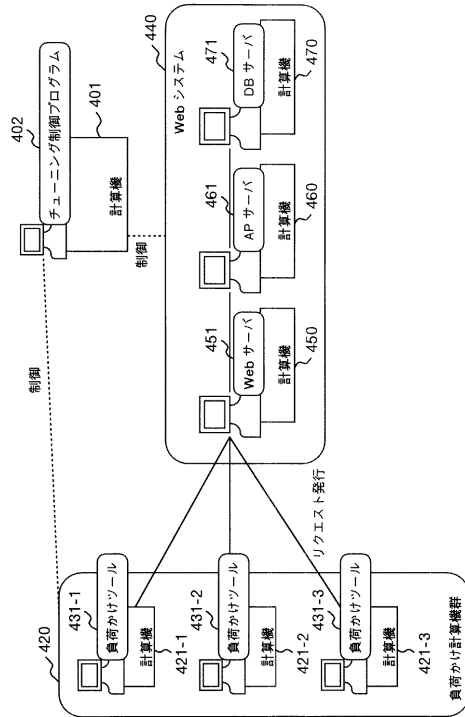
【図3】

図 3



【図 4】

図 4



【図 5】

図 5

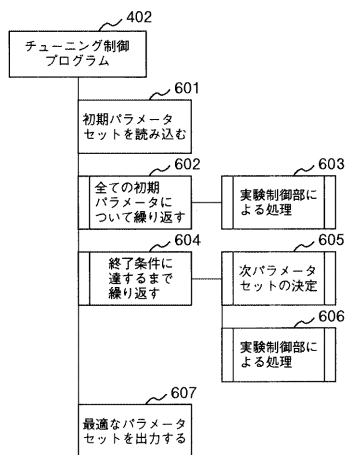
```

max_experiment_times=50
distance=3
param_size=3
init_param.1=10,8,5
init_param.2=25,20,10
init_param.3=50,30,20
init_param.4=75,50,35

```

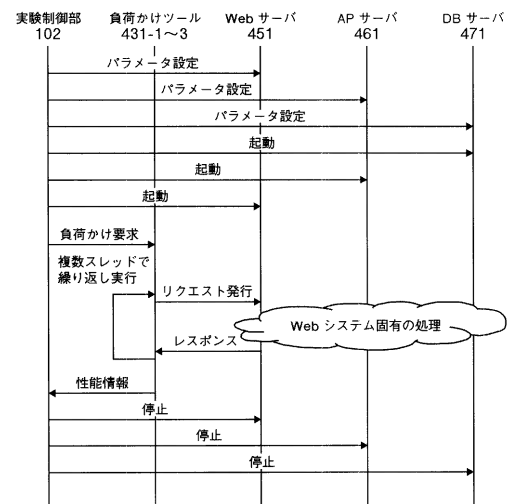
【図 6】

図 6

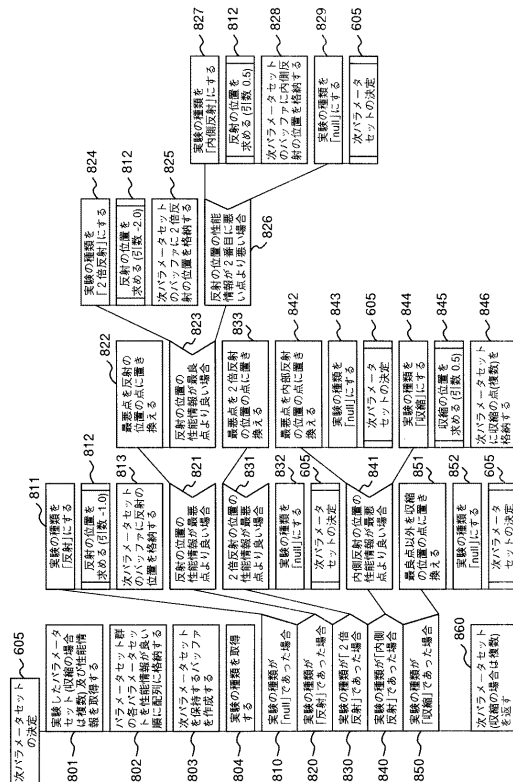


【図 7】

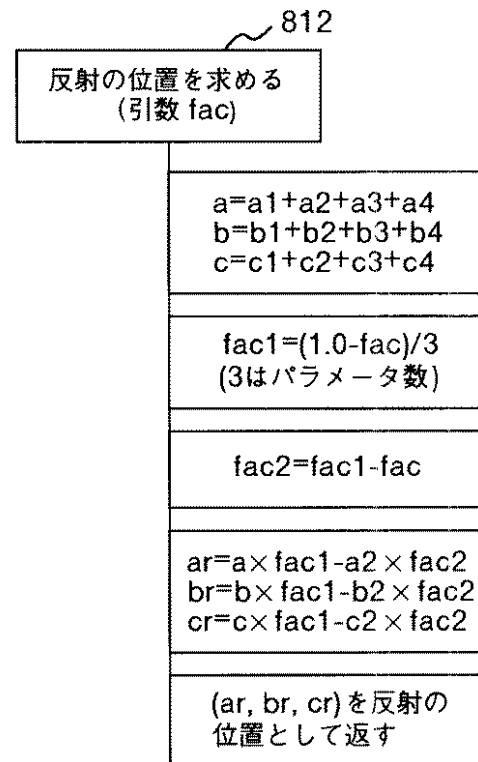
図 7



【 図 8 】

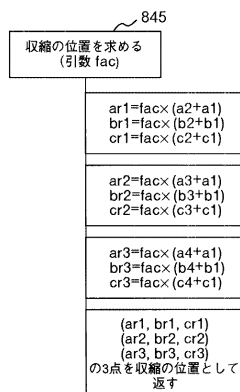


【 図 9 】



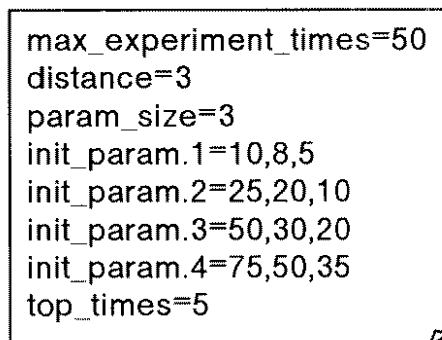
【 図 1 0 】

☒ 1 0



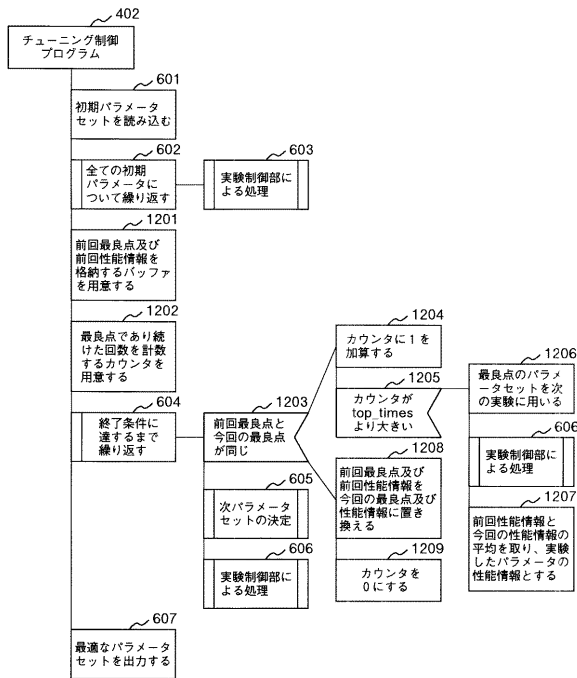
【 圖 1 1 】

図 11



【図 12】

図 12



【図 13】

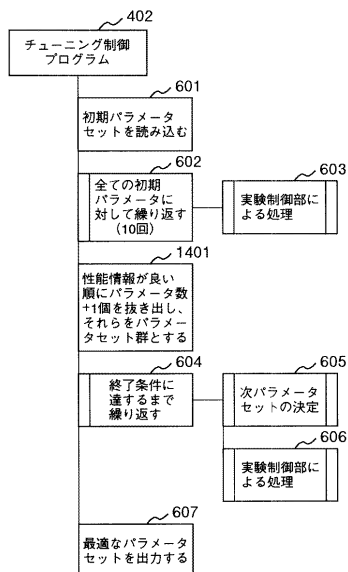
図 13

```

max_experiment_times=50
distance=3
param_size=3
init_param.1=10,8,5
init_param.2=25,20,10
init_param.3=50,30,20
init_param.4=75,50,35
init_param.5=15,10,8
init_param.6=20,17,11
init_param.7=30,28,27
init_param.8=40,38,15
init_param.9=60,55,45
init_param.10=65,59,58
  
```

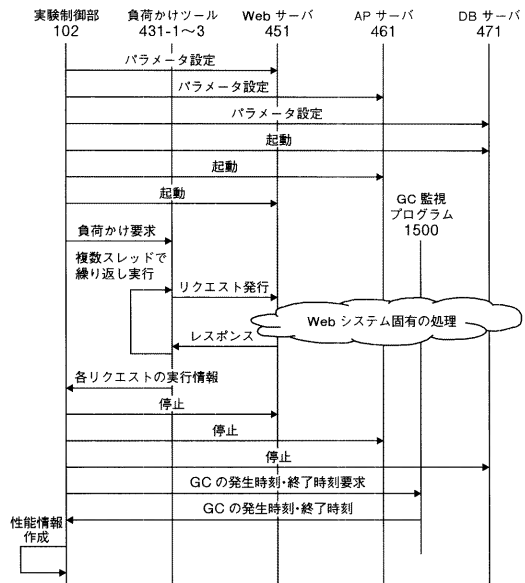
【図 14】

図 14



【図 15】

図 15



フロントページの続き

- (72)発明者 三宅 滋
神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所 システム開発研究所内
- (72)発明者 石崎 健史
神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所 システム開発研究所内

審査官 相崎 裕恒

- (56)参考文献 特開 2 0 0 2 - 2 4 4 8 9 1 (J P , A)
特開平 0 8 - 2 1 2 0 9 6 (J P , A)
西岡 他, システム性能シミュレーションのリモート実行環境の開発, コンピュータシステム・シンポジウム論文集, 日本, 社団法人情報処理学会, 2 0 0 1 年 1 1 月 1 9 日, 第2001巻, 第16号, p.129-135

- (58)調査した分野(Int.Cl. , D B 名)
G 0 6 F 1 1 / 3 4