(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第6885193号
(P6885193)

(45)発行日　令和3年6月9日(2021.6.9)　(24)登録日　令和3年5月17日(2021.5.17)

(51)Int.Cl.		F I		
G06F 9/50	(2006.01)	G06F	9/50	150Z
G06F 9/48	(2006.01)	G06F	9/48	300Z
G06F 15/80	(2006.01)	G06F	15/80	
G06F 15/177	(2006.01)	G06F	15/177	A

請求項の数 5　(全 20 頁)

(21)出願番号　特願2017-95200(P2017-95200)
(22)出願日　平成29年5月12日(2017.5.12)
(65)公開番号　特開2018-194875(P2018-194875A)
(43)公開日　平成30年12月6日(2018.12.6)
審査請求日　令和2年2月13日(2020.2.13)

(73)特許権者　000005223
富士通株式会社
神奈川県川崎市中原区上小田中４丁目１番１号
(74)代理人　100104190
弁理士　酒井　昭徳
(72)発明者　小久保　良輔
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　橋本　剛
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

審査官　井上　宏一

最終頁に続く

(54)【発明の名称】並列処理装置、ジョブ管理方法、およびジョブ管理プログラム

(57)【特許請求の範囲】
【請求項１】
実行待ちの各ジョブの実行に使用されるノード数と、前記各ジョブの実行にかかる実行予定時間とに基づいて、前記各ジョブの実行規模を算出し、
算出した前記実行規模が大きいジョブから順に、複数のノードが配置された領域を区分けして分割された複数のエリアのうち、故障可能性が高い問題ノードの数が少ないエリアを選択し、選択した前記エリア内のノード群にジョブを割り当てる、
制御部を有することを特徴とする並列処理装置。
【請求項２】
前記制御部は、
前記ジョブの割り当てを行う際に、前記複数のノードそれぞれの使用状態を示す情報を参照して、前記ジョブの実行に使用されるノード数に基づいて、選択した前記エリアにおいて、前記問題ノードを含まない、前記ジョブを割り当て可能なノード群を探索し、
前記ノード群が探索された場合に、当該ノード群を選択して前記ジョブの割り当てを行う、
ことを特徴とする請求項１に記載の並列処理装置。
【請求項３】
前記制御部は、
前記複数のエリアの全てについて前記問題ノードを含まないノード群を選択した前記ジョブの割り当てができないときは、選択した前記エリアにおいて、前記問題ノードの数が

最小となるように、前記ジョブを割り当て可能なノード群を探索し、
前記ノード群が探索された場合に、当該ノード群を選択して前記ジョブの割り当てを行う、
ことを特徴とする請求項２に記載の並列処理装置。
【請求項４】
実行待ちの各ジョブの実行に使用されるノード数と、前記各ジョブの実行にかかる実行予定時間とに基づいて、前記各ジョブの実行規模を算出し、
算出した前記実行規模が大きいジョブから順に、複数のノードが配置された領域を区分けして分割された複数のエリアのうち、故障可能性が高い問題ノードの数が少ないエリアを選択し、選択した前記エリア内のノード群にジョブを割り当てる、
処理をコンピュータが実行することを特徴とするジョブ管理方法。
【請求項５】
実行待ちの各ジョブの実行に使用されるノード数と、前記各ジョブの実行にかかる実行予定時間とに基づいて、前記各ジョブの実行規模を算出し、
算出した前記実行規模が大きいジョブから順に、複数のノードが配置された領域を区分けして分割された複数のエリアのうち、故障可能性が高い問題ノードの数が少ないエリアを選択し、選択した前記エリア内のノード群にジョブを割り当てる、
処理をコンピュータに実行させることを特徴とするジョブ管理プログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、並列処理装置、ジョブ管理方法、およびジョブ管理プログラムに関する。
【背景技術】
【０００２】
従来、コンピュータシステムを用いて科学技術計算などの大規模な計算を行う場合、複数の計算機を用いた並列計算が行われる。並列計算が可能なコンピュータシステムは、並列計算機システムと呼ばれる。大規模な並列計算機システムは、並列計算を行う多数の計算機と、管理用計算機とが含まれる。管理用計算機は、計算機に実行させるジョブを管理する。以下の説明では、並列計算を行う計算機を「計算ノード」と表記し、管理用計算機を「管理ノード」と表記する場合がある。
【０００３】
大規模な並列計算機システムにおいては、複数の計算ノードを管理し並列に動作させることで、システム全体の演算性能を高めている。システム全体の演算性能を向上するためには、大量の計算ノードが必要となる。また、大規模な並列計算機システムの管理ノードでは、ジョブスケジューラ機能が、計算ノード群にユーザのジョブを割り当てる制御を実施する。
【０００４】
先行技術としては、例えば、障害影響度の高いジョブをリスク度の低い実行サーバで実行し、障害影響度の高いジョブを実行中または実行が予定されている実行サーバの多重度を下げて、高負荷状態にせずに障害リスク度が低い状態に保つものがある。また、ジョブに関する情報を基にジョブの形状毎の影響度を求め、影響度が高い順に所定数のジョブの形状を計算対象形状として決定し、計算対象形状及び影響度を基に、計算ノードそれぞれへのジョブの割り当て方であるジョブの事前配置を決定する技術がある。投入されたジョブが計算対象形状のいずれかに一致する場合、事前配置にしたがい投入されたジョブが計算ノードへ割り当てられる。
【先行技術文献】
【特許文献】
【０００５】
【特許文献１】特開２０１１－２１５６６１号公報
【特許文献２】特開２０１５－６９５７７号公報

【発明の概要】
【発明が解決しようとする課題】
【０００６】
しかしながら、従来技術では、大規模な並列計算機システムの実行稼働率を低下させないように、ジョブを計算ノードに割り当てることが難しい場合がある。
【０００７】
一つの側面では、本発明は、複数のノードを含むシステムの実行稼働率を向上させることを目的とする。
【課題を解決するための手段】
【０００８】
１つの実施態様では、実行待ちの各ジョブの実行に使用されるノード数と、前記各ジョブの実行にかかる実行予定時間とに基づいて、前記各ジョブの実行規模を算出し、算出した前記実行規模が大きいジョブから順に、複数のノードが配置された領域を区分けして分割された複数のエリアのうち、故障可能性が高い問題ノードの数が少ないエリアからジョブを割り当てる、並列処理装置が提供される。
【発明の効果】
【０００９】
本発明の一側面によれば、複数のノードを含むシステムの実行稼働率を向上させることができる。
【図面の簡単な説明】
【００１０】
【図１Ａ】図１Ａは、実施の形態にかかるジョブ管理方法の一実施例を示す説明図（その１）である。
【図１Ｂ】図１Ｂは、実施の形態にかかるジョブ管理方法の一実施例を示す説明図（その２）である。
【図２】図２は、並列計算機システム２００のシステム構成例を示す説明図である。
【図３】図３は、並列処理装置１０１のハードウェア構成例を示すブロック図である。
【図４】図４は、ノード管理テーブル２２０の記憶内容の一例を示す説明図である。
【図５】図５は、ジョブ管理テーブル２３０の記憶内容の一例を示す説明図である。
【図６】図６は、問題ノード一覧情報６００の具体例を示す説明図である。
【図７】図７は、並列処理装置１０１の機能的構成例を示すブロック図である。
【図８】図８は、ノード管理テーブル２２０の記憶内容の更新例を示す説明図である。
【図９】図９は、ジョブ管理テーブル２３０の記憶内容の更新例を示す説明図である。
【図１０】図１０は、並列処理装置１０１のジョブ管理処理手順の一例を示すフローチャートである。
【図１１】図１１は、ジョブ割当処理の具体的処理手順の一例を示すフローチャート（その１）である。
【図１２】図１２は、ジョブ割当処理の具体的処理手順の一例を示すフローチャート（その２）である。
【発明を実施するための形態】
【００１１】
以下に図面を参照して、本発明にかかる並列処理装置、ジョブ管理方法、およびジョブ管理プログラムの実施の形態を詳細に説明する。
【００１２】
（実施の形態）
図１Ａおよび図１Ｂは、実施の形態にかかるジョブ管理方法の一実施例を示す説明図である。図１において、並列処理装置１０１は、複数のノードＮに実行させるジョブを管理するコンピュータ（いわゆる、管理ノード）である。ノードＮは、並列計算機システムの構成要素であり、並列計算を行うコンピュータ（いわゆる、計算ノード）である。ジョブは、ユーザがコンピュータに依頼する仕事の単位である。ジョブとしては、例えば、科学

技術計算などの大規模な計算を行うジョブが挙げられる。

【００１３】

ここで、大規模な並列計算機システムにおけるジョブは、特定１ノードに割り当てられるのではなく、同時に複数のノードを占有して実行する場合が多い。また、メッシュないしトーラスネットワークを持つシステムでは、１ジョブへの割り当て範囲の部分ネットワークが、サブメッシュないしサブトーラス（ｎ次元直方体状）であることが必要な場合が多い。例えば、トーラスネットワークを有する並列計算機システムのジョブスケジューラでは、計算ノードにジョブを「ｎ次元直方体の形に割り当てる」のように割り当てる。

【００１４】

一方で、大規模な並列計算機システムでは、計算ノードの台数増加に比例して計算ノードの故障率が高くなる傾向がある。例えば、ユーザのジョブを実行している計算ノードがハードウェア故障により停止してしまうと、該当ノード上で実行されているジョブは異常終了してしまう。

【００１５】

このため、大規模な並列計算機システムにおいては、各計算ノードのシステムログとしてハードウェア故障を予見するログが出力されたことを事前検知し、該当計算ノードをジョブ実行で利用しないよう運用から動的に切り離すシステム監視機能が知られている。システム監視機能により運用から切り離された計算ノードは、管理ノードのジョブスケジューラ機能により、当該計算ノードに新規にジョブを割り当てないよう制御される。

【００１６】

ところが、ハードウェアが故障する確率が高い計算ノードを特定できたとしても、必ず故障する計算ノードを特定するのは困難である。故障確率が高いとはいえ、まだ健全な計算ノードを運用から切り離してしまうと、並列計算機システムの実行稼働率（スループット）が低下してしまう。なお、並列計算機システムの実行稼働率は、例えば、下記式（１）によって表すことができる。

【００１７】

並列計算機システムの実行稼働率＝Σ（各計算ノードで正常終了したジョブが実行されていた時間）／Σ（各計算ノードの電源が投入されていた時間）・・・（１）

【００１８】

そこで、本実施の形態では、故障確率が高いノードＮを避けながら、できるだけ並列計算機システムの実行稼働率を向上させるジョブ管理方法について説明する。以下、並列処理装置１０１の処理例について説明する。ここでは、複数のノードＮとして、「ノードＮ１～Ｎ６０」を例に挙げて説明する。また、実行待ちのジョブとして、「ジョブＪ１～Ｊ３」を例に挙げて説明する。また、本実施の形態では、複数のノードＮが配置された領域として、２次元の領域を例に挙げて説明するが、３次元以上のｎ次元の領域にも適用可能である。

【００１９】

（１）並列処理装置１０１は、実行待ちの各ジョブＪの実行ノード数Ｃと実行予定時間Ｔとに基づいて、各ジョブＪの実行規模Ｓを算出する。ここで、実行ノード数Ｃは、実行待ちの各ジョブＪの実行に使用されるノード数である。実行予定時間Ｔは、各ジョブの実行にかかる予定時間である。ジョブＪの実行ノード数Ｃおよび実行予定時間Ｔは、例えば、ジョブＪを投入するユーザにより指定される。

【００２０】

また、実行規模Ｓは、ジョブＪが異常終了した際に並列計算機システムの実行稼働率に与える影響度合いが高いほど大きくなる指標である。例えば、実行ノード数Ｃが多いジョブＪほど、実行中に多くのノードＮを占有することになり、異常終了した際に実行稼働率に与える影響度合いは高いといえる。また、実行予定時間Ｔが長いジョブＪほど、実行中に長い時間ノードＮを占有することになり、異常終了した際に実行稼働率に与える影響度合いは高いといえる。

【００２１】

このため、並列処理装置１０１は、例えば、実行待ちの各ジョブＪの実行ノード数Ｃと実行予定時間Ｔとを乗算することにより、各ジョブＪの実行規模Ｓを算出してもよい。図１Ａの例では、各ジョブＪ１～Ｊ３の実行規模Ｓ１～Ｓ３が算出された結果、各ジョブＪ１～Ｊ３が実行規模Ｓ１～Ｓ３の大きい順にソートされている（Ｊ１⇒Ｊ２⇒Ｊ３）。

【００２２】

（２）並列処理装置１０１は、複数のノードＮが配置された領域を区分けして複数のエリアＡに分割する。ここで、領域は、複数のノードＮが配置された平面あるいは空間のことである。以下の説明では、複数のノードＮが配置された領域を「ノードエリアＡＲ」と表記する場合がある。

【００２３】

具体的には、例えば、並列処理装置１０１は、ノードエリアＡＲを四角形（あるいは、ｎ次元直方体形状）に均等に区分けして複数のエリアＡに分割する。分割数は、例えば、並列計算機システムのシステムサイズに応じて設定される。図１Ａの例では、ノードエリアＡＲがエリアＡ１～Ａ４に分割されている。また、エリアＡ１～Ａ４内に存在する問題ノードの数が少ない順にエリアＡ１～Ａ４がソートされている。

【００２４】

問題ノードとは、故障可能性が高いノードＮである。問題ノードは、例えば、ハードウェア故障を予見するログが出力されたノードＮであってもよく、また、複数のノードＮのうち使用年数等をもとに相対的に故障可能性が高いと判断されたノードＮであってもよい。なお、図１Ａおよび図１Ｂ中、問題ノードは、白抜きの四角で表す。

【００２５】

（３）並列処理装置１０１は、算出した実行規模Ｓが大きいジョブＪから順に、ノードエリアＡＲを区分けして分割された複数のエリアＡのうち、問題ノードの数が少ないエリアＡからジョブＪを割り当てる。具体的には、例えば、並列処理装置１０１は、ジョブＪの割り当てを行う際に、問題ノードを含まないノードＮ群を選択してジョブＪの割り当てを行う。

【００２６】

図１Ｂの例では、まず、ジョブＪ１～Ｊ３のうち実行規模Ｓが最大のジョブＪ１が、問題ノードの数が最小のエリアＡ２内の空き領域に割り当てられる。つぎに、２番目に実行規模Ｓが大きいジョブＪ２が、問題ノードの数が２番目に少ないエリアＡ１内の空き領域に割り当てられる。最後に、実行規模Ｓが最小のジョブＪ３が、問題ノードの数が３番目に少ないエリアＡ３内の空き領域に割り当てられる。

【００２７】

なお、各ジョブＪの割当先となる空き領域は、例えば、四角形（あるいは、ｎ次元直方体形状）のサブトーラスを形成するノードＮ群であって、他のジョブＪの実行に使用されていない未使用のノードＮを、少なくとも各ジョブＪの実行ノード数Ｃ分含むノードＮ群を含む領域である。

【００２８】

このように、並列処理装置１０１によれば、実行に使用されるノード数が多く、実行に時間がかかるジョブＪに、故障可能性が高い問題ノードができるだけ割り当てられないように、ジョブＪを実行するノードＮを効率よく選定することができる。このため、異常終了時の影響度合いが大きいジョブＪが問題ノードに割り当たる確率を下げて、並列計算機システムの実行稼働率（スループット）を向上させることができる。また、ノードエリアＡＲを分割したエリア単位でジョブＪの割当先となるノードＮ群を探索することができるため、ジョブＪの割当先を決める際の処理時間を短縮して、ジョブＪの開始時間が遅延するのを防ぐことができる。

【００２９】

（並列計算機システム２００のシステム構成例）

つぎに、図１に示した並列処理装置１０１を含む並列計算機システム２００のシステム構成例について説明する。

【００３０】
図２は、並列計算機システム２００のシステム構成例を示す説明図である。図２において、並列計算機システム２００は、並列処理装置１０１と、ノードＮ１～Ｎｎ（ｎ：２以上の自然数）と、クライアント装置２０１と、を含む。並列計算機システム２００において、並列処理装置１０１、ノードＮ１～Ｎｎおよびクライアント装置２０１は、有線または無線のネットワーク２１０を介して接続される。ネットワーク２１０は、例えば、ＬＡＮ（Ｌｏｃａｌ Ａｒｅａ Ｎｅｔｗｏｒｋ）、ＷＡＮ（Ｗｉｄｅ Ａｒｅａ Ｎｅｔｗｏｒｋ）、インターネットなどである。
【００３１】
並列処理装置１０１は、ノード管理テーブル２２０およびジョブ管理テーブル２３０を有し、ノードＮ１～Ｎｎに実行させるジョブを管理する。ノード管理テーブル２２０およびジョブ管理テーブル２３０の記憶内容については、図４および図５を用いて後述する。並列処理装置１０１は、例えば、サーバである。
【００３２】
ノードＮ１～Ｎｎは、並列計算を行うコンピュータである。各ノードＮ１～Ｎｎは、例えば、サーバである。ノードＮ１～Ｎｎは、例えば、ノード間の高速通信を可能にするトーラスネットワークを形成する。図１Ａに示したノードＮ１～Ｎ６０は、例えば、ノードＮ１～Ｎｎに相当する（ｎ＝６０）。
【００３３】
以下の説明では、ノードＮ１～Ｎｎのうちの任意のノードを「ノードＮ」と表記する場合がある。また、ノードＮ１～Ｎｎが配置された領域を「ノードエリアＡＲ」と表記する場合がある。
【００３４】
クライアント装置２０１は、並列計算機システム２００のユーザ（管理者を含む）が使用するコンピュータである。クライアント装置２０１は、例えば、ＰＣ（Ｐｅｒｓｏｎａｌ Ｃｏｍｐｕｔｅｒ）である。なお、図２の例では、クライアント装置２０１を１台のみ表記したが、これに限らない。例えば、クライアント装置２０１は、並列計算機システム２００のユーザごとに設けられる。
【００３５】
（並列処理装置１０１のハードウェア構成例）
図３は、並列処理装置１０１のハードウェア構成例を示すブロック図である。図３において、並列処理装置１０１は、ＣＰＵ（Ｃｅｎｔｒａｌ Ｐｒｏｃｅｓｓｉｎｇ Ｕｎｉｔ）３０１と、メモリ３０２と、Ｉ／Ｆ（Ｉｎｔｅｒｆａｃｅ）３０３と、ディスクドライブ３０４と、ディスク３０５と、を有する。また、各構成部は、バス３００によってそれぞれ接続される。
【００３６】
ここで、ＣＰＵ３０１は、並列処理装置１０１の全体の制御を司る。メモリ３０２は、例えば、ＲＯＭ（Ｒｅａｄ Ｏｎｌｙ Ｍｅｍｏｒｙ）、ＲＡＭ（Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙ）およびフラッシュＲＯＭなどを有する。具体的には、例えば、フラッシュＲＯＭやＲＯＭが各種プログラムを記憶し、ＲＡＭがＣＰＵ３０１のワークエリアとして使用される。メモリ３０２に記憶されるプログラムは、ＣＰＵ３０１にロードされることで、コーディングされている処理をＣＰＵ３０１に実行させる。
【００３７】
Ｉ／Ｆ３０３は、通信回線を通じてネットワーク２１０に接続され、ネットワーク２１０を介して外部のコンピュータ（例えば、図２に示したノードＮ１～Ｎｎ、クライアント装置２０１）に接続される。そして、Ｉ／Ｆ３０３は、ネットワーク２１０と装置内部とのインターフェースを司り、外部のコンピュータからのデータの入出力を制御する。Ｉ／Ｆ３０３には、例えば、モデムやＬＡＮアダプタなどを採用することができる。
【００３８】
ディスクドライブ３０４は、ＣＰＵ３０１の制御に従ってディスク３０５に対するデー

タのリード／ライトを制御する。ディスク３０５は、ディスクドライブ３０４の制御で書き込まれたデータを記憶する。ディスク３０５としては、例えば、磁気ディスク、光ディスクなどが挙げられる。

【００３９】

なお、並列処理装置１０１は、上述した構成部のほかに、例えば、ＳＳＤ（Ｓｏｌｉｄ Ｓｔａｔｅ Ｄｒｉｖｅ）、入力装置、ディスプレイ等を有することにしてもよい。また、図２に示したノードＮ１～Ｎｎおよびクライアント装置２０１についても、並列処理装置１０１と同様のハードウェア構成により実現することができる。ただし、クライアント装置２０１は、上述した構成部のほかに、入力装置、ディスプレイを有する。

【００４０】

（ノード管理テーブル２２０の記憶内容）

つぎに、並列処理装置１０１が有するノード管理テーブル２２０の記憶内容について説明する。ノード管理テーブル２２０は、例えば、図３に示したメモリ３０２、ディスク３０５などの記憶装置により実現される。

【００４１】

図４は、ノード管理テーブル２２０の記憶内容の一例を示す説明図である。図４において、ノード管理テーブル２２０は、ノードＩＤ、位置（ｘ、ｙ）、エリアＩＤ、故障可能性フラグおよび使用中フラグのフィールドを有する。各フィールドに情報を設定することで、ノード管理情報（例えば、ノード管理情報４００－１～４００－ｎ）がレコードとして記憶される。

【００４２】

ここで、ノードＩＤは、並列計算機システム２００に含まれるノードＮを一意に識別する識別子である。位置（ｘ、ｙ）は、ノードＮの位置を示す座標である。なお、ここではノードエリアＡＲとして２次元の領域を例に挙げて説明するが、ノードエリアＡＲが３次元以上のｎ次元の空間の場合には、位置フィールドには、ｎ次元座標系におけるノードＮの位置を示す座標が設定される。

【００４３】

エリアＩＤは、ノードＮが属するエリアＡを一意に識別する識別子である。エリアＡは、ノードＮ１～Ｎｎが配置されたノードエリアＡＲを区分けして分割されたエリアである。故障可能性フラグは、ノードＮが、故障可能性が高い問題ノードであるか否かを示すフラグである。故障可能性フラグ「０」は、ノードＮが問題ノードではないことを示す。故障可能性フラグ「１」は、ノードＮが問題ノードであることを示す。

【００４４】

使用中フラグは、ノードＮが、ジョブＪの実行に使用されているか否かを示すフラグである。使用中フラグ「０」は、ノードＮがジョブＪの実行に使用されていない空きノードであることを示す。使用中フラグ「１」は、ノードＮがジョブＪの実行に使用されている使用中ノードであることを示す。

【００４５】

（ジョブ管理テーブル２３０の記憶内容）

つぎに、並列処理装置１０１が有するジョブ管理テーブル２３０の記憶内容について説明する。ジョブ管理テーブル２３０は、例えば、図３に示したメモリ３０２、ディスク３０５などの記憶装置により実現される。

【００４６】

図５は、ジョブ管理テーブル２３０の記憶内容の一例を示す説明図である。図５において、ジョブ管理テーブル２３０は、ジョブＩＤ、実行ノード数、実行予定時間および実行規模のフィールドを有し、各フィールドに情報を設定することで、ジョブ管理情報（例えば、ジョブ管理情報５００－１～５００－３）をレコードとして記憶する。

【００４７】

ここで、ジョブＩＤは、実行待ちのジョブＪを一意に識別する識別子である。実行ノード数は、ジョブＪの実行に使用されるノード数である。実行予定時間は、ジョブＪの実行

にかかる予定時間である。実行規模は、ジョブＪが異常終了した際に並列計算機システム２００の実行稼働率に与える影響度合いを表す指標である。

【００４８】

（問題ノード一覧情報６００の具体例）

つぎに、並列処理装置１０１が用いる問題ノード一覧情報６００の具体例について説明する。

【００４９】

図６は、問題ノード一覧情報６００の具体例を示す説明図である。図６において、問題ノード一覧情報６００は、ノードＮ１～Ｎｎのうちの故障可能性が高い問題ノードを識別するノードＩＤを示す情報である。問題ノード一覧情報６００は、例えば、並列処理装置１０１において作成されてもよく、また、並列処理装置１０１とは異なる他のコンピュータにおいて作成されることにしてもよい。

【００５０】

（並列処理装置１０１の機能的構成例）

図７は、並列処理装置１０１の機能的構成例を示すブロック図である。図７において、並列処理装置１０１は、取得部７０１と、受付部７０２と、算出部７０３と、分割部７０４と、割当制御部７０５と、を含む構成である。取得部７０１～割当制御部７０５は制御部となる機能であり、具体的には、例えば、図３に示したメモリ３０２、ディスク３０５などの記憶装置に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、Ｉ／Ｆ３０３により、その機能を実現する。各機能部の処理結果は、例えば、メモリ３０２、ディスク３０５などの記憶装置に記憶される。より具体的には、各機能部は、例えば、並列処理装置１０１のジョブスケジューラにより実現することができる。

【００５１】

取得部７０１は、ノードＮの位置情報を取得する。ここで、ノードの位置情報は、ノードＮの位置を示す情報であり、例えば、ノードエリアＡＲにおけるノードＮの位置を示す座標である。ノードＮの位置情報には、例えば、ノードＮを識別するノードＩＤが含まれる。ノードＩＤとしては、例えば、ノードＮのＭＡＣ（Ｍｅｄｉａ　Ａｃｃｅｓｓ　Ｃｏｎｔｒｏｌ）アドレスを用いることができる。

【００５２】

具体的には、例えば、取得部７０１は、ネットワーク２１０（図２参照）を介して、他のコンピュータ（例えば、クライアント装置２０１）からノードＮの位置情報を受信することにより、ノードＮの位置情報を取得することにしてもよい。また、取得部７０１は、例えば、不図示の入力装置を用いたユーザの操作入力により、ノードＮの位置情報を取得することにしてもよい。

【００５３】

取得されたノードＮの位置情報は、例えば、図４に示したノード管理テーブル２２０に記憶される。ここで、ノード管理テーブル２２０の記憶内容の更新例について説明する。

【００５４】

図８は、ノード管理テーブル２２０の記憶内容の更新例を示す説明図である。図８の（８－１）において、ノード管理テーブル２２０のノードＩＤおよび位置（ｘ、ｙ）の各フィールドに情報が設定された結果、ノード管理情報（例えば、ノード管理情報４００－１～４００－３）がレコードとして記憶される。ただし、この時点では、各ノード管理情報のエリアＩＤフィールドは「－（Ｎｕｌｌ）」である。また、各ノード管理情報の故障可能性フラグおよび使用中フラグの各フィールドは初期状態「０」である。

【００５５】

図７の説明に戻り、取得部７０１は、問題ノードを示す情報を取得する。ここで、問題ノードは、故障可能性が高いノードＮである。具体的には、例えば、取得部７０１は、ネットワーク２１０を介して、他のコンピュータ（例えば、クライアント装置２０１）から問題ノード一覧情報６００を受信することにより、問題ノードを示す情報を取得することにしてもよい。また、取得部７０１は、例えば、不図示の入力装置を用いたユーザの操作

入力により、問題ノード一覧情報６００を取得することにしてもよい。
【００５６】
また、取得部７０１は、各ノードＮのシステムログを監視して、問題ノード一覧情報６００を作成することにしてもよい。より詳細に説明すると、取得部７０１は、例えば、ノードＮのシステムログとしてハードウェア故障を予見するログを検出すると、当該ノードＮを問題ノードとして問題ノード一覧情報６００に登録する。
【００５７】
問題ノードを示す情報が取得されると、例えば、ノード管理テーブル２２０内の対応するノード管理情報の故障可能性フラグが「１」に更新される。例えば、問題ノード一覧情報６００が示すノードＩＤ「Ｎ１５」を例に挙げると、図８の（８－２）に示すように、ノード管理情報４００－１５の故障可能性フラグが「１」に更新される。
【００５８】
受付部７０２は、ジョブＪの実行ノード数Ｃと実行予定時間Ｔとを受け付ける。ここで、実行ノード数Ｃは、ジョブＪの実行に使用されるノード数である。実行予定時間Ｔは、ジョブの実行にかかる予定時間である。実行予定時間Ｔの単位は、任意に設定可能であり、例えば、「分」や「時間」に設定される。
【００５９】
具体的には、例えば、並列計算機システム２００のユーザが、クライアント装置２０１において、ジョブＪを投入する際に、ジョブＪの実行ノード数Ｃと実行予定時間Ｔとを指定する。この場合、受付部７０２は、クライアント装置２０１において指定されたジョブＪの実行ノード数Ｃと実行予定時間Ｔとを受け付ける。また、受付部７０２は、例えば、不図示の入力装置を用いたユーザの操作入力により、ジョブＪの実行ノード数Ｃと実行予定時間Ｔとを受け付けることにしてもよい。
【００６０】
受け付けられたジョブＪの実行ノード数Ｃと実行予定時間Ｔは、例えば、図５に示したジョブ管理テーブル２３０に記憶される。ここで、ジョブ管理テーブル２３０の記憶内容の更新例について説明する。
【００６１】
図９は、ジョブ管理テーブル２３０の記憶内容の更新例を示す説明図である。図９の（９－１）において、ジョブ管理テーブル２３０のジョブＩＤ、実行ノード数および実行予定時間の各フィールドに情報が設定された結果、ジョブ管理情報（例えば、ジョブ管理情報５００－１～５００－３）がレコードとして記憶される。ただし、この時点では、各ジョブ管理情報の実行規模フィールドは「－」である。
【００６２】
図７の説明に戻り、算出部７０３は、実行待ちの各ジョブＪの実行ノード数Ｃと実行予定時間Ｔとに基づいて、各ジョブＪの実行規模Ｓを算出する。ここで、実行規模Ｓは、ジョブＪが異常終了した際に並列計算機システム２００の実行稼働率に与える影響度合いを表す指標である。
【００６３】
具体的には、例えば、算出部７０３は、ジョブ管理テーブル２３０を参照して、実行待ちの各ジョブＪの実行ノード数Ｃと実行予定時間Ｔとを乗算することにより、各ジョブＪの実行規模Ｓを算出する。算出された各ジョブＪの実行規模Ｓは、例えば、図９の（９－２）に示すように、各ジョブＪのジョブＩＤと対応付けて、ジョブ管理テーブル２３０の実行規模フィールドに記憶される。
【００６４】
分割部７０４は、ノードＮ１～Ｎｎが配置されたノードエリアＡＲを区分けして複数のエリアＡに分割する。例えば、ノードエリアＡＲが２次元の平面の場合、各エリアＡは、四角形の領域となる。例えば、ノードエリアＡＲがｎ次元の空間の場合、各エリアＡは、ｎ次元直方体の領域となる。具体的には、例えば、分割部７０４は、ノードエリアＡＲを四角形（あるいは、ｎ次元直方体形状）に均等に区分けして複数のエリアＡに分割する。

分割数は、例えば、並列計算機システム２００のシステムサイズに応じて適宜設定される。

【００６５】

また、分割部７０４は、各エリアＡの探索開始位置を設定する。ここで、探索開始位置とは、各エリアＡにおいて、ジョブＪを割り当てる空き領域を探索する際の開始位置となる位置である。空き領域とは、ジョブＪの実行に使用されていない未使用のノードＮ群を含む領域である。各エリア内のどの位置を探索開始位置とするかは任意に設定可能である。具体的には、例えば、分割部７０４は、ノードエリアＡＲを四角形に区分けした各エリアＡの左下の位置を探索開始位置に設定することにしてもよい。

【００６６】

一例として、図１Ａに示したようにノードエリアＡＲを４分割する場合、ノードエリアＡＲの左下を原点とすると、左下のエリアＡ１の探索開始位置は、「（ｘ、ｙ）＝（０、０）」となる。また、右下のエリアＡ２の探索開始位置は、「（ｘ、ｙ）＝（ｘ軸最大値÷２、０）」となる。また、左上のエリアＡ３の探索開始位置は、「（ｘ、ｙ）＝（０、ｙ軸最大値÷２）」となる。また、右上のエリアＡ４の探索開始位置は、「（ｘ、ｙ）＝（ｘ軸最大値÷２、ｙ軸最大値÷２）」となる。

【００６７】

また、分割部７０４は、ノードＮが属するエリアＡを特定する。具体的には、例えば、分割部７０４は、ノード管理テーブル２２０を参照して、各ノードＮが属するエリアＡを特定する。特定された結果（エリアＡのエリアＩＤ）は、例えば、図８の（８－３）に示すように、各ノードＮのノードＩＤと対応付けて、ノード管理テーブル２２０のエリアＩＤフィールドに記憶される。

【００６８】

割当制御部７０５は、実行待ちのジョブＪを割り当てる制御を行う。具体的には、例えば、割当制御部７０５は、ノード管理テーブル２２０を参照して、各エリアＡの問題ノード数ｐを算出する。問題ノード数ｐは、各エリアＡに属する問題ノードの数である。一例として、エリアＡ１の問題ノード数ｐ１を算出するとする。この場合、割当制御部７０５は、エリアＩＤフィールドに「Ａ１」が設定されたノード管理情報のうち、故障可能性フラグに「１」が設定されたノード管理情報の数を、エリアＡ１の問題ノード数ｐ１として算出する。

【００６９】

そして、割当制御部７０５は、ジョブ管理テーブル２３０を参照して、算出された実行規模Ｓが大きいジョブＪから順に、複数のエリアＡのうち、算出した問題ノード数ｐが少ないエリアＡからジョブＪを割り当てる。この際、割当制御部７０５は、例えば、問題ノードを含まないノードＮ群を選択してジョブＪの割り当てを行う。

【００７０】

より詳細に説明すると、例えば、まず、割当制御部７０５は、ジョブ管理テーブル２３０を参照して、実行待ちのジョブＪ１～Ｊ３を実行規模Ｓが大きい順にソートする。ここで、実行規模Ｓ１～Ｓ３の大小関係を「Ｓ１＞Ｓ２＞Ｓ３」とする。この場合、ジョブＪ１～Ｊ３を実行規模Ｓが大きい順にソートすると、｛Ｊ１，Ｊ２，Ｊ３｝となる。なお、実行規模Ｓが同一のジョブＪが存在する場合には、割当制御部７０５は、例えば、それらジョブＪについて、キューに入れられた順にソートすることにしてもよい。

【００７１】

また、割当制御部７０５は、複数のエリアＡを問題ノード数ｐが少ない順にソートする。ここで、複数のエリアＡを「エリアＡ１～Ａ４」とし、エリアＡ１～Ａ４の問題ノード数ｐ１～ｐ４の大小関係を「ｐ４＞ｐ３＞ｐ１＞ｐ２」とする。この場合、エリアＡ１～Ａ４を問題ノード数ｐが少ない順にソートすると、｛Ａ２，Ａ１，Ａ３，Ａ４｝となる。なお、問題ノード数ｐが同一のエリアＡが存在する場合には、割当制御部７０５は、例えば、それらエリアＡについて、探索開始位置に近い問題ノードの数が少ないエリアを上位にソートすることにしてもよい。

【００７２】
つぎに、割当制御部７０５は、｛Ｊ１，Ｊ２，Ｊ３｝から、実行規模Ｓが最大のジョブＪ１を選択する。また、割当制御部７０５は、｛Ａ２，Ａ１，Ａ３，Ａ４｝から、問題ノード数ｐが最小のエリアＡ２を選択する。そして、割当制御部７０５は、ノード管理テーブル２２０を参照して、選択したエリアＡ２から、問題ノードを含まない、ジョブＪ１を割り当て可能なノードＮ群を探索する。
【００７３】
ここで、ジョブＪ１を割り当て可能なノードＮ群は、例えば、サブトーラスを形成するノードＮの集合であって、他のジョブＪの実行に使用されていない未使用のノードＮを、少なくともジョブＪ１の実行ノード数Ｃ１分含むノードＮの集合である。
【００７４】
具体的には、例えば、割当制御部７０５は、エリアＡ２の探索開始位置から徐々に範囲を広げながら、問題ノードを含まない、ジョブＪ１を割り当て可能なノードＮ群を探索する。この際、割当制御部７０５は、例えば、ノード単位、あるいは、シャーシ単位で範囲を広げることにしてもよい。シャーシとは、サブトーラスを形成するノードＮの集合である。そして、ノードＮ群の探索に成功すると、割当制御部７０５は、探索したノードＮ群を選択してジョブＪ１を割り当てる。
【００７５】
一方、ノードＮ群の探索に失敗すると、割当制御部７０５は、｛Ａ２，Ａ１，Ａ３，Ａ４｝から、問題ノード数ｐが次に少ないエリアＡ１を選択する。そして、割当制御部７０５は、選択したエリアＡ２から、問題ノードを含まない、ジョブＪ１を割り当て可能なノードＮ群を探索する。割当制御部７０５は、ノードＮ群の探索に成功する、あるいは、未選択のエリアＡがなくなるまで、上述した一連の処理を繰り返す。
【００７６】
また、ジョブＪ１の割り当てが完了すると、割当制御部７０５は、｛Ｊ１，Ｊ２，Ｊ３｝から、実行規模Ｓが次に大きいジョブＪ２を選択して、ジョブＪ１と同様の処理を行う。そして、ジョブＪ２の割り当てが完了すると、割当制御部７０５は、｛Ｊ１，Ｊ２，Ｊ３｝から、実行規模Ｓが次に大きいジョブＪ３を選択して、ジョブＪ１，Ｊ２と同様の処理を行う。
【００７７】
ただし、複数のエリアＡの全てについて、問題ノードを含まないノードＮ群を選択したジョブＪの割り当てができないときがある。この場合、割当制御部７０５は、問題ノード数ｐが少ないエリアＡから、問題ノードの数が最小となるようにノードＮ群を選択してジョブＪの割り当てを行うことにしてもよい。
【００７８】
すなわち、割当制御部７０５は、問題ノードを含むことを許容して、ジョブＪを割り当て可能なノードＮ群を探索する。この際、割当制御部７０５は、例えば、問題ノードの数が最小となるように、ジョブＪを割り当て可能なノードＮ群をエリアＡから探索する。そして、ノードＮ群の探索に成功すると、割当制御部７０５は、探索したノードＮ群を選択してジョブＪを割り当てる。ただし、複数のエリアＡの全てについて、問題ノードを含むことを許容してもノードＮ群の探索に失敗した場合、割当制御部７０５は、ジョブＪをキューに戻すことにしてもよい。
【００７９】
なお、ジョブＪの割り当てが完了すると、割当制御部７０５は、ジョブＪを割り当てたノードＮに対応する、ノード管理テーブル２２０内の使用中フラグを「１」に変更する。また、ジョブＪの実行が終了すると、割当制御部７０５は、ジョブＪが割り当てられていたノードＮに対応する、ノード管理テーブル２２０内の使用中フラグを「０」に変更する。
【００８０】
（並列処理装置１０１のジョブ管理処理手順）

つぎに、並列処理装置１０１のジョブ管理処理手順について説明する。ジョブ管理処理は、例えば、定期的に実行されることにしてもよく、新たなジョブＪが投入される、あるいは、投入済みのいずれかのジョブＪの実行が完了したことに応じて実行されることにしてもよい。また、ノードＮの位置情報は、ノード管理テーブル２２０に記憶されているとする。

【００８１】

図１０は、並列処理装置１０１のジョブ管理処理手順の一例を示すフローチャートである。図１０のフローチャートにおいて、まず、並列処理装置１０１は、問題ノード一覧情報６００を取得する（ステップＳ１００１）。そして、並列処理装置１０１は、取得した問題ノード一覧情報６００に基づいて、ノード管理テーブル２２０内の故障可能性フラグを更新する（ステップＳ１００２）。

【００８２】

つぎに、並列処理装置１０１は、ジョブＪの実行ノード数Ｃと実行予定時間Ｔとを受け付ける（ステップＳ１００３）。受け付けられたジョブＪの実行ノード数Ｃと実行予定時間Ｔは、ジョブ管理テーブル２３０に記憶される。

【００８３】

そして、並列処理装置１０１は、ジョブ管理テーブル２３０を参照して、実行待ちの各ジョブＪの実行ノード数Ｃと実行予定時間Ｔとを乗算することにより、各ジョブＪの実行規模Ｓを算出する（ステップＳ１００４）。算出された各ジョブＪの実行規模Ｓは、ジョブ管理テーブル２３０に記憶される。

【００８４】

つぎに、並列処理装置１０１は、ジョブ管理テーブル２３０を参照して、実行待ちのジョブＪを実行規模Ｓが大きい順にソートする（ステップＳ１００５）。そして、並列処理装置１０１は、ノードＮ１～Ｎｎが配置されたノードエリアＡＲを区分けして複数のエリアＡに分割する（ステップＳ１００６）。この際、並列処理装置１０１は、各エリアＡの探索開始位置を設定する。

【００８５】

つぎに、並列処理装置１０１は、ノード管理テーブル２２０を参照して、各ノードＮが属するエリアＡを特定する（ステップＳ１００７）。特定された結果（エリアＡのエリアＩＤ）は、ノード管理テーブル２２０に記憶される。そして、並列処理装置１０１は、ノード管理テーブル２２０を参照して、各エリアＡの問題ノード数ｐを算出する（ステップＳ１００８）。

【００８６】

つぎに、並列処理装置１０１は、複数のエリアＡを問題ノード数ｐが少ない順にソートする（ステップＳ１００９）。つぎに、並列処理装置１０１は、実行規模Ｓが大きい順にソートした実行待ちのジョブＪの先頭から未選択のジョブＪを選択する（ステップＳ１０１０）。

【００８７】

そして、並列処理装置１０１は、選択したジョブＪを割り当てるジョブ割当処理を実行する（ステップＳ１０１１）。ジョブ割当処理の具体的な処理手順については、図１１および図１２を用いて後述する。つぎに、並列処理装置１０１は、実行規模Ｓが大きい順にソートした実行待ちのジョブＪのうち選択されていない未選択のジョブＪがあるか否かを判断する（ステップＳ１０１２）。

【００８８】

ここで、未選択のジョブＪがある場合（ステップＳ１０１２：Ｙｅｓ）、並列処理装置１０１は、ステップＳ１０１０に戻る。一方、未選択のジョブＪがない場合（ステップＳ１０１２：Ｎｏ）、並列処理装置１０１は、本フローチャートによる一連の処理を終了する。これにより、実行待ちのジョブＪの割り当てを行うことができる。

【００８９】

つぎに、ステップＳ１０１１のジョブ割当処理の具体的な処理手順について説明する。

【００９０】
図１１および図１２は、ジョブ割当処理の具体的処理手順の一例を示すフローチャートである。図１１のフローチャートにおいて、まず、並列処理装置１０１は、問題ノード数ｐが少ない順にソートした複数のエリアＡの先頭から未選択のエリアＡを選択する（ステップＳ１１０１）。
【００９１】
つぎに、並列処理装置１０１は、選択したエリアＡから、問題ノードを含まない、選択したジョブＪを割り当て可能なノードＮ群を探索する（ステップＳ１１０２）。なお、ジョブＪを割り当て可能なノードＮ群は、例えば、サブトーラスを形成するノードＮの集合であって、未使用のノードＮを実行ノード数Ｃ分含むノードＮの集合である。
【００９２】
そして、並列処理装置１０１は、ノードＮ群が探索されたか否かを判断する（ステップＳ１１０３）。ここで、ノードＮ群が探索された場合（ステップＳ１１０３：Ｙｅｓ）、並列処理装置１０１は、探索したノードＮ群を選択してジョブＪを割り当てて（ステップＳ１１０４）、ジョブ割当処理を呼び出したステップに戻る。
【００９３】
一方、ノードＮ群が探索されなかった場合（ステップＳ１１０３：Ｎｏ）、並列処理装置１０１は、問題ノード数ｐが少ない順にソートした複数のエリアＡのうち、ステップＳ１１０１において選択されていない未選択のエリアＡがあるか否かを判断する（ステップＳ１１０５）。
【００９４】
ここで、未選択のエリアＡがある場合（ステップＳ１１０５：Ｙｅｓ）、並列処理装置１０１は、ステップＳ１１０１に戻る。一方、未選択のエリアＡがない場合には（ステップＳ１１０５：Ｎｏ）、並列処理装置１０１は、図１２に示すステップＳ１２０１に移行する。
【００９５】
図１２のフローチャートにおいて、まず、並列処理装置１０１は、問題ノード数ｐが少ない順にソートした複数のエリアＡの先頭から未選択のエリアＡを選択する（ステップＳ１２０１）。そして、並列処理装置１０１は、選択したエリアＡから、問題ノードを含むことを許容して、問題ノードの数が最小となるように、ジョブＪを割り当て可能なノードＮ群を探索する（ステップＳ１２０２）。
【００９６】
つぎに、並列処理装置１０１は、ノードＮ群が探索されたか否かを判断する（ステップＳ１２０３）。ここで、ノードＮ群が探索された場合（ステップＳ１２０３：Ｙｅｓ）、並列処理装置１０１は、探索したノードＮ群を選択してジョブＪを割り当てて（ステップＳ１２０４）、ジョブ割当処理を呼び出したステップに戻る。
【００９７】
一方、ノードＮ群が探索されなかった場合（ステップＳ１２０３：Ｎｏ）、並列処理装置１０１は、問題ノード数ｐが少ない順にソートした複数のエリアＡのうち、ステップＳ１２０１において選択されていない未選択のエリアＡがあるか否かを判断する（ステップＳ１２０５）。
【００９８】
ここで、未選択のエリアＡがある場合（ステップＳ１２０５：Ｙｅｓ）、並列処理装置１０１は、ステップＳ１２０１に戻る。一方、未選択のエリアＡがない場合には（ステップＳ１２０５：Ｎｏ）、並列処理装置１０１は、選択したジョブＪをキューに入れて（ステップＳ１２０６）、ジョブ割当処理を呼び出したステップに戻る。
【００９９】
これにより、実行規模Ｓが大きいジョブＪに、故障可能性が高い問題ノードができるだけ割り当てられないように制御することができる。
【０１００】

以上説明したように、実施の形態にかかる並列処理装置１０１によれば、実行待ちの各ジョブＪの実行ノード数Ｃと実行予定時間Ｔとに基づいて、各ジョブＪの実行規模Ｓを算出することができる。そして、並列処理装置１０１によれば、実行規模Ｓが大きいジョブＪから順に、ノードＮ１～Ｎｎが配置されたノードエリアＡＲを区分けして分割された複数のエリアＡのうち、問題ノード数ｐが少ないエリアＡからジョブＪを割り当てることができる。

【０１０１】

これにより、実行に使用されるノード数が多く、実行に時間がかかるジョブＪに、故障可能性が高い問題ノードができるだけ割り当てられないように、ジョブＪを実行するノードＮを効率よく選定することができる。このため、異常終了時の影響度合いが大きいジョブＪが問題ノードに割り当たる確率を下げて、並列計算機システム２００の実行稼働率（スループット）を向上させることができる。また、ジョブＪの割当先を決める際の処理時間を短縮することができ、ジョブＪの開始時間が遅延するのを防ぐことができる。

【０１０２】

また、並列処理装置１０１によれば、ジョブＪの割り当てを行う際に、問題ノードを含まないノードＮ群を選択してジョブＪの割り当てを行うことができる。これにより、実行中のジョブＪが異常終了してしまうのを防ぐことができ、無駄な処理が生じて並列計算機システム２００の実行稼働率が低下するのを抑制することができる。

【０１０３】

また、並列処理装置１０１によれば、複数のエリアＡの全てについて問題ノードを含まないノードＮ群を選択したジョブＪの割り当てができないときは、問題ノードの数が最小となるようにノードＮ群を選択してジョブＪの割り当てを行うことができる。これにより、問題ノードを避けたジョブＪの割り当てができない場合は、問題ノード数を最小化することで、実行中のジョブＪが異常終了する確率を下げることができる。

【０１０４】

また、並列処理装置１０１によれば、問題ノード数ｐとして、各ノードＮで記録されるシステムログからハードウェア故障が予見されたノードＮの数を計数することができる。これにより、ハードウェア故障の可能性が高い問題ノードにジョブＪができるだけ割り当てられないように、ジョブＪを実行するノードＮを効率よく選定することができる。

【０１０５】

これらのことから、並列処理装置１０１によれば、トーラスネットワークを有するような大規模な並列計算機システム２００において、実行稼働率を下げないように、部分ネットワーク（例えば、２次元平面状やｎ次元直方体状のサブトーラス）にジョブＪを割り当てることが可能となる。

【０１０６】

なお、本実施の形態で説明したジョブ管理方法は、予め用意されたプログラムをパーソナル・コンピュータやワークステーション等のコンピュータで実行することにより実現することができる。本ジョブ管理プログラムは、ハードディスク、フレキシブルディスク、ＣＤ（Ｃｏｍｐａｃｔ　Ｄｉｓｃ）－ＲＯＭ、ＭＯ（Ｍａｇｎｅｔｏ－Ｏｐｔｉｃａｌ　ｄｉｓｋ）、ＤＶＤ（Ｄｉｇｉｔａｌ　Ｖｅｒｓａｔｉｌｅ　Ｄｉｓｋ）、ＵＳＢ（Ｕｎｉｖｅｒｓａｌ　Ｓｅｒｉａｌ　Ｂｕｓ）メモリ等のコンピュータで読み取り可能な記録媒体に記録され、コンピュータによって記録媒体から読み出されることによって実行される。また、本ジョブ管理プログラムは、インターネット等のネットワークを介して配布してもよい。

【０１０７】

上述した実施の形態に関し、さらに以下の付記を開示する。

【０１０８】

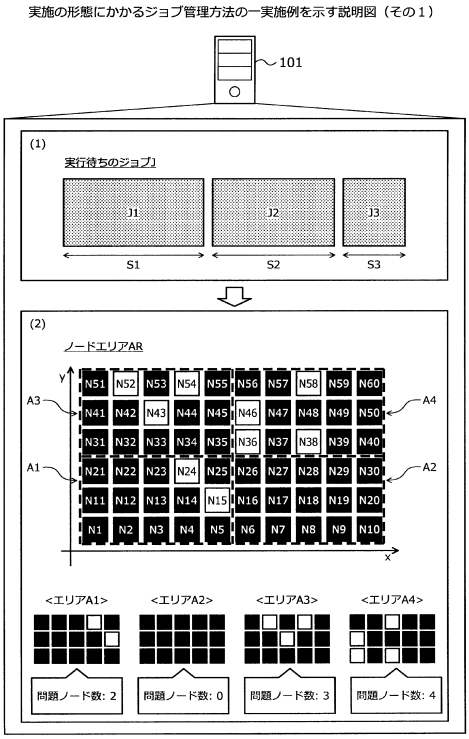
（付記１）実行待ちの各ジョブの実行に使用されるノード数と、前記各ジョブの実行にかかる実行予定時間とに基づいて、前記各ジョブの実行規模を算出し、

算出した前記実行規模が大きいジョブから順に、複数のノードが配置された領域を区分

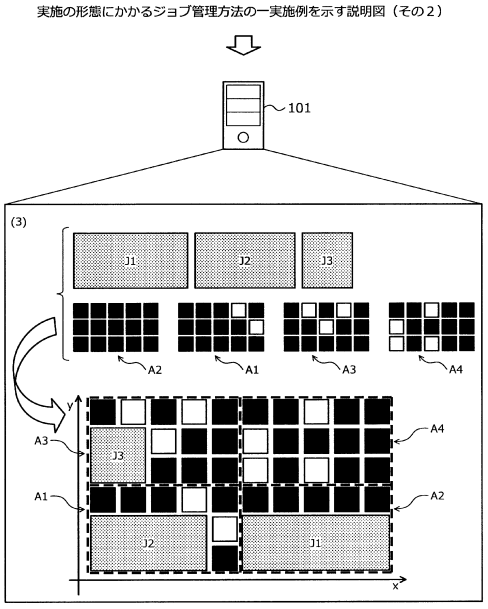
けして分割された複数のエリアのうち、故障可能性が高い問題ノードの数が少ないエリアからジョブを割り当てる、
制御部を有することを特徴とする並列処理装置。
【０１０９】
（付記２）前記制御部は、
前記ジョブの割り当てを行う際に、前記問題ノードを含まないノード群を選択して前記ジョブの割り当てを行う、
ことを特徴とする付記１に記載の並列処理装置。
【０１１０】
（付記３）前記制御部は、
前記複数のエリアの全てについて前記問題ノードを含まないノード群を選択した前記ジョブの割り当てができないときは、前記問題ノードの数が最小となるようにノード群を選択して前記ジョブの割り当てを行う、
ことを特徴とする付記２に記載の並列処理装置。
【０１１１】
（付記４）前記複数のノードは、トーラスネットワークを形成するノードである、ことを特徴とする付記１～３のいずれか一つに記載の並列処理装置。
【０１１２】
（付記５）前記問題ノードは、前記複数のノードそれぞれで記録されるシステムログからハードウェア故障が予見されたノードである、ことを特徴とする付記１～４のいずれか一つに記載の並列処理装置。
【０１１３】
（付記６）実行待ちの各ジョブの実行に使用されるノード数と、前記各ジョブの実行にかかる実行予定時間とに基づいて、前記各ジョブの実行規模を算出し、
算出した前記実行規模が大きいジョブから順に、複数のノードが配置された領域を区分けして分割された複数のエリアのうち、故障可能性が高い問題ノードの数が少ないエリアからジョブを割り当てる、
処理をコンピュータが実行することを特徴とするジョブ管理方法。
【０１１４】
（付記７）実行待ちの各ジョブの実行に使用されるノード数と、前記各ジョブの実行にかかる実行予定時間とに基づいて、前記各ジョブの実行規模を算出し、
算出した前記実行規模が大きいジョブから順に、複数のノードが配置された領域を区分けして分割された複数のエリアのうち、故障可能性が高い問題ノードの数が少ないエリアからジョブを割り当てる、
処理をコンピュータに実行させることを特徴とするジョブ管理プログラム。
【符号の説明】
【０１１５】
１０１　並列処理装置
２００　並列計算機システム
２０１　クライアント装置
２１０　ネットワーク
２２０　ノード管理テーブル
２３０　ジョブ管理テーブル
３００　バス
３０１　ＣＰＵ
３０２　メモリ
３０３　Ｉ／Ｆ
３０４　ディスクドライブ
３０５　ディスク
６００　問題ノード一覧情報

７０１　取得部
７０２　受付部
７０３　算出部
７０４　分割部
７０５　割当制御部
Ａ　エリア
ＡＲ　ノードエリア
Ｎ　ノード

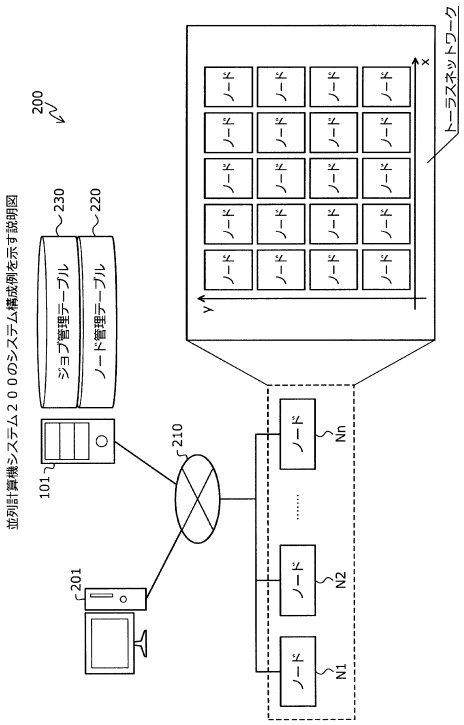
【図１Ａ】



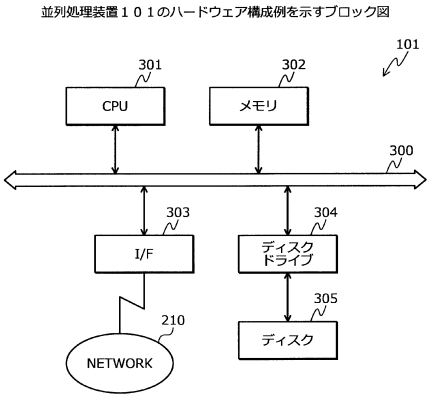
【図１Ｂ】



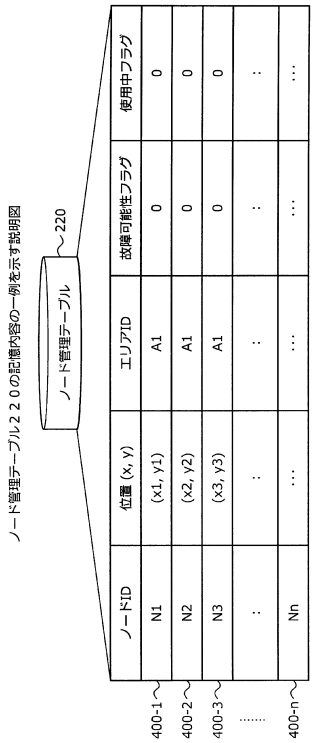
【図２】


並列計算機システム２００のシステム構成例を示す説明図
200
230
220
ジョブ管理テーブル
ノード管理テーブル
トーラスネットワーク
ノード
x
y
101
210
201
N1
N2
Nn

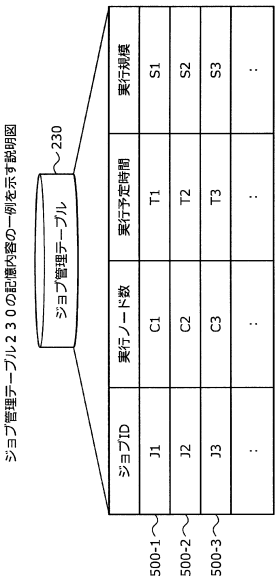
【図３】


並列処理装置１０１のハードウェア構成例を示すブロック図
101
301
CPU
302
メモリ
300
303
I/F
304
ディスクドライブ
305
ディスク
210
NETWORK

【図４】

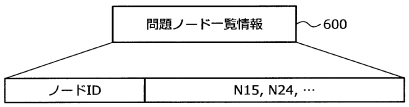

ノード管理テーブル２２０の記憶内容の一例を示す説明図
220
ノード管理テーブル
ノードID
位置 (x, y)
エリアID
故障可能性フラグ
使用中フラグ
N1
(x1, y1)
A1
0
0
N2
(x2, y2)
A1
0
0
N3
(x3, y3)
A1
0
0
Nn
400-1
400-2
400-3
400-n

【図５】


ジョブ管理テーブル２３０の記憶内容の一例を示す説明図
230
ジョブ管理テーブル
ジョブID
実行ノード数
実行予定時間
実行規模
J1
C1
T1
S1
J2
C2
T2
S2
J3
C3
T3
S3
500-1
500-2
500-3

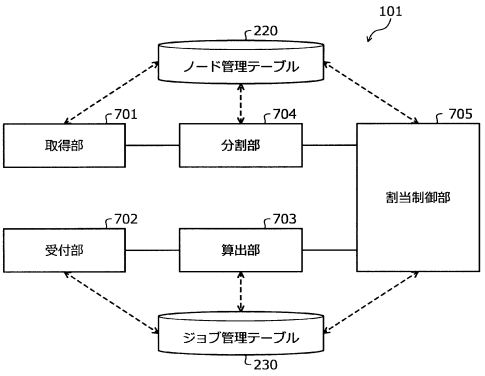
【図６】

問題ノード一覧情報６００の具体例を示す説明図



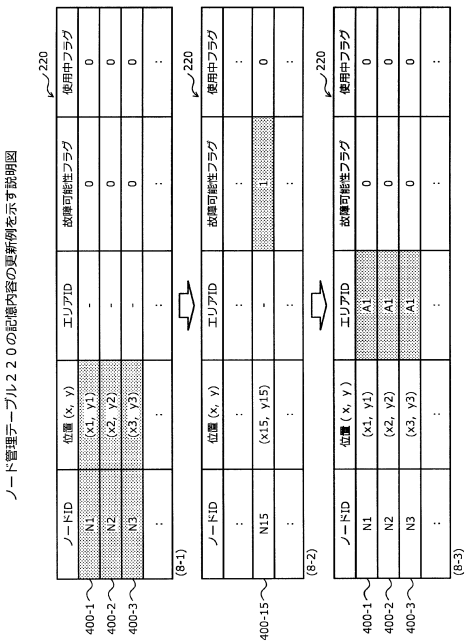
【図７】

並列処理装置１０１の機能的構成例を示すブロック図



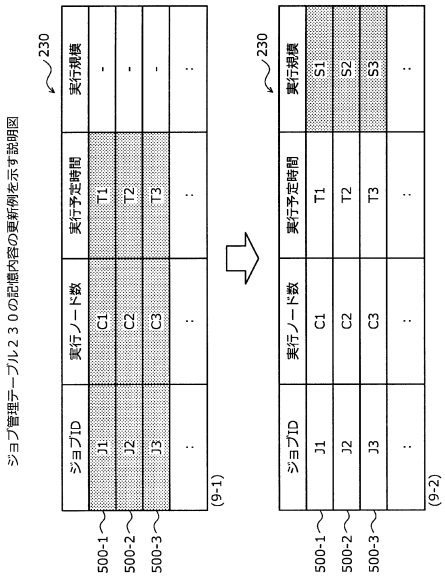
【図８】

ノード管理テーブル２２０の記憶内容の更新例を示す説明図



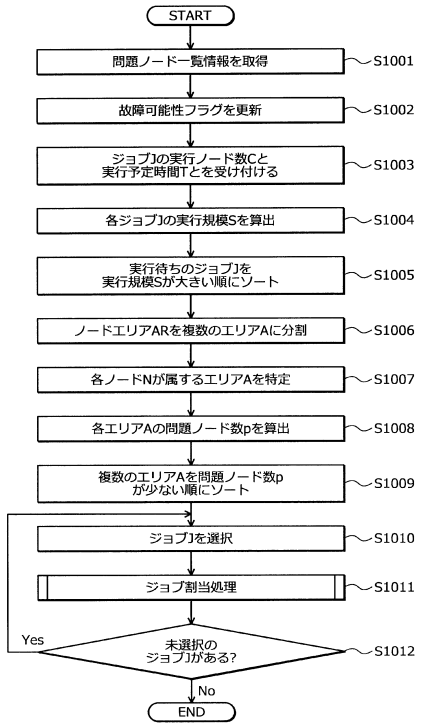
【図９】

ジョブ管理テーブル２３０の記憶内容の更新例を示す説明図

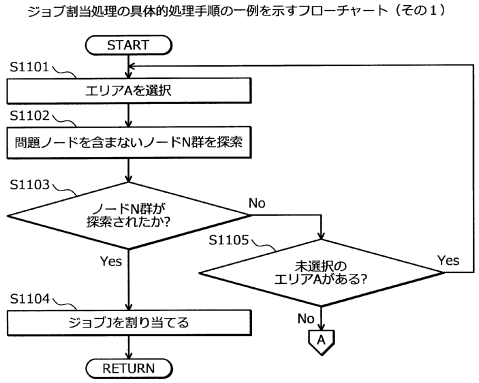


【図１０】

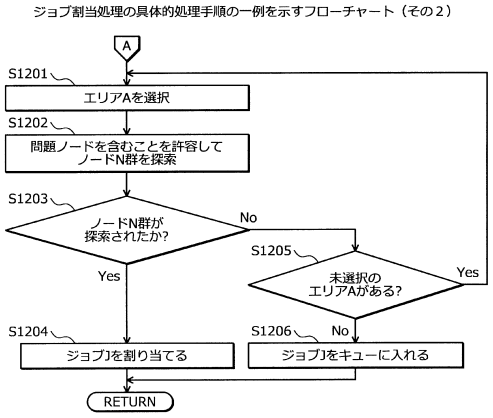
並列処理装置１０１のジョブ管理処理手順の一例を示すフローチャート



【図１１】


ジョブ割当処理の具体的処理手順の一例を示すフローチャート（その１）
START
S1101
エリアAを選択
S1102
問題ノードを含まないノードN群を探索
S1103
ノードN群が
探索されたか?
No
Yes
S1105
未選択の
エリアAがある?
Yes
No
A
S1104
ジョブJを割り当てる
RETURN

【図１２】


ジョブ割当処理の具体的処理手順の一例を示すフローチャート（その２）
A
S1201
エリアAを選択
S1202
問題ノードを含むことを許容して
ノードN群を探索
S1203
ノードN群が
探索されたか?
No
Yes
S1205
未選択の
エリアAがある?
Yes
S1204
ジョブJを割り当てる
S1206
No
ジョブJをキューに入れる
RETURN

フロントページの続き

(56)参考文献 特開２００３－ ５８５１８（ＪＰ，Ａ）
特開２０１５－ ６９５７７（ＪＰ，Ａ）
特表２００７－５３３０３１（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ 9／４５５－9／５４
Ｇ０６Ｆ １５／１７７
Ｇ０６Ｆ １５／８０