分散計算システムにおける漸次縮退の一方式

7L−1

布留川 哲也　南谷 崇

東京工業大学 工学部

1 まえがき

ある種の大規模計算を実行するには、多数の計算機要素間を通信線要素で規則的に接続した大規模な分散計算システム [1] が有効である。分散計算システムの性能を向上させるために構成要素数の増加が望まれることがしばしばあるが、要素数の増大に伴ってシステム内に故障が生じる可能性は高まる。その結果、システムに障害が生じ、計算に支障を来たす可能性が高まるという問題が生じる。このような障害を回避するために必要なフォールトトレランス [2] の一形態として、システムの構成要素に故障が生じた場合、要素数の減少によるある程度の機能縮小または品質低下があっても、部分的なサービスを提供し続ける漸次縮退 (graceful degradation)[3] が、分散計算システムにとってコストパフォーマンスの点で優れている。特に、計算システムとしての実用性を重視する場合、システムの構成要素が失われた際、システムの機能の縮小を行なわず残された正常な要素数に応じたある水準の性能を発揮しつつ、実行中の計算を正しく続行するような漸次縮退特性が望ましい。

一般に、分散計算システムに漸次縮退機能を付加するにはコストがかかるので、故障が全く発生していない状態では、漸次縮退システムは漸次縮退導入前のシステムと比較して性能が劣る。ここでは、多次元メッシュ状アプリケーションを多次元トーラス状分散計算システム上で実行する場合に限定して、無故障時の性能のオーバーヘッドが極めて小さく故障時の性能の低下が比較的小さい漸次縮退システムの実現法の一方式を提案し、この方式の有効性を実験により確かめる。

2 分散計算のモデル

2.1 分散計算システムのモデル

n 次元のトーラスを複数個用意しトーラス状に結合すると、$n+1$ 次元のトーラスが得られる。(図 1) ここで対象とするのは、計算機要素 (ノード) と、それらを多次元トーラス状に接続する通信線要素 (リンク) からなる並列計算に向いた分散計算システムである。この種の分散計算システムは並列計算システムと呼ばれることもある。

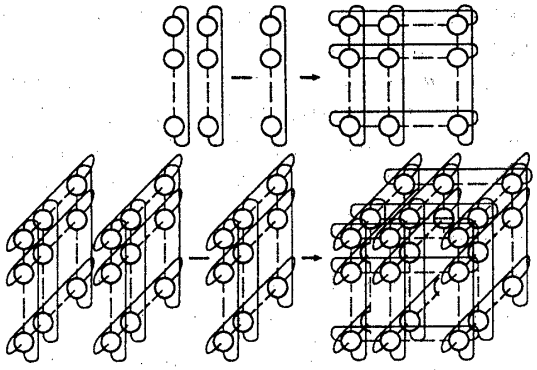

図 1: 多次元トーラス

A Method of Graceful Degradation in Distributed Systems
Tetsuya FURUKAWA, Takashi NANYA
Faculty of Engineering, Tokyo Institute of Technology

2.2 アプリケーションのモデル

ここで対象とするのは、多次元直方領域を各辺に垂直な超平面で多次元メッシュに分割し (図 2)、すべてのメッシュ点の値を隣接するメッシュ点の値によって更新する計算を決められた回数だけ繰り返すアプリケーションである。このアプリケーションを分散計算システムで実行するためには、システムの形状に適合するように多次元直方領域を分割し、各分割領域に属する全てのメッシュ点を一つのノードに割り当てる。

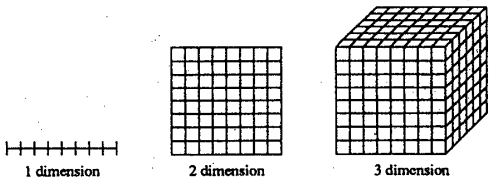

図 2: 多次元メッシュ

3 故障のモデル

ここで対象とする故障はノードのハードウェアフォールトである。リンクの故障は無視する。また、リンクは故障したノードを飛び越えて使用可能であるとする。更に次の仮定を設ける。

仮定 1 (フェイルストップノード) ノードで故障が発生すると、そのノードは誤ったメッセージを送出する前に動作を停止する。

仮定 2 故障は単独で発生し、故障が発生してからアプリケーションの実行が再開されるまでの間に新たな故障は発生しない。

4 漸次縮退の実現法

漸次縮退システムでは、故障が発生すると、故障の検出、失われた情報の回復、システムの再構成という三つの回復手続きを実行してから、アプリケーションの実行を再開する。一般に、分散計算システムに漸次縮退機能を付加する際の問題点は、これらの回復手続きを如何に低コストに実現するかにある。但し、故障の検出については、仮定1に基づき、受信待ちのタイムアウトを利用して容易に実現できる。本方式では、以下のような方法で失われた情報の回復、システムの再構成を実現する。

4.1 失われた情報の回復

故障によって失われたノードが保持していた情報をシステムに残された正常なノードだけで回復する。

一般のアプリケーションに対してこれを常に可能にするには、一つのシステム内に独立した二つの部分システムを構成してそれぞれで同一のアプリケーションを実行する方法 (計算の二重化) や、ノードの保持している全情報をチェックポイントごとに別のノードに送って保存する方法 (通信による情報の二重化) があるが、どちらも計算量や通信量の増加を招き、システムの性能を低下させる直接的な要因になる。そこで、多次元メッシュアプリケーションに依存した、計算量、通信量を増加させない次の回復法を提案する。

通信履歴による回復 メッシュ点の値を隣接するメッシュ点の値を使って更新する計算式を変形して、メッシュ点の現在値、更新値と隣接するメッシュ点の内一個だけ除いたものの値から隣接するメッシュ点一個の値を求める計算式が得られる場合もあ

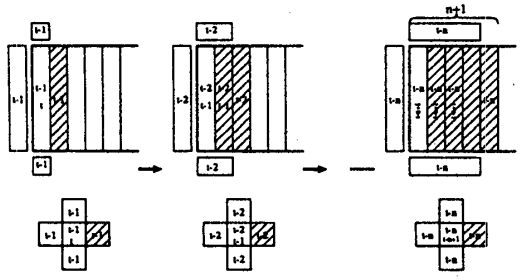

図 3: 通信履歴による回復

	計算量	通信量	記憶領域
計算の二重化 $(2w \times h)$	wh	$2w$	$2wh$
通信による状態の二重化	0	$f(wh-2w-2h+4)$	wh
通信履歴による回復	0	0	$w^2+2wh+2w+2h$

表 1: 回復法のコスト

る。このような逆計算が常に可能ならば、これを繰り返すことで徐々に時点を遡り、最終的には、ノードに割り当てられていた全てのメッシュ点の値を回復することができる。2 次元アプリケーションの場合に、ノードに割り当てられた領域の左半分について回復の様子を図示したのが図 3である。図中の斜線はその時点で回復される領域を示す。

2 次元アプリケーションの場合、ノードに割り当てられた領域のサイズが $w \times h(w \leq h)$ であるとして、これらの回復法のノードあたりのコストをまとめた結果を表 1に示す。但し、$f(<1)$ は通信による状態の二重化の場合のチェックポイントの頻度を表す。通信履歴による回復法には、回復に時間がかかる、逆計算による丸め誤差が存在するという欠点が存在するが、記憶領域だけがコストとなるので性能のオーバーヘッドが小さく抑えられることが期待できる。

4.2 システムの再構成

計算の分担を再決定する。計算の分担の最適化を高い水準まで要求すると、計算の分担を決定する問題は非常に困難になる。本方式では、正常なノードをすべて使用して、ノード毎に割り当てられる変数の数を均一化するように再構成を行なう。これに対して、故障ノードを含む行または列を切り捨てる方法もよく知られている。

5 シミュレーション実験

8×8 の 2 次元トーラス状並列計算機上で故障をシミュレートし、アプリケーションのメッシュサイズを 256 × 256 に固定して本方式の実験を行なった。全領域のメッシュ点の値を更新する計算を 5000 回繰り返すアプリケーションの実行中に 4 つのノードに順次故障を発生させた結果 (破線) を図 4に示す。横軸は時刻 (秒)、縦軸はそれまでの反復回数を表す。従って線分の傾きがその時点での性能を表す。故障を発生させなかった場合の実験結果 (実線) と漸次縮退導入前のシステムの実験結果 (点線) を比較すると、わずかな差しかない。実際の測定値からは、漸次縮退導入前と比べての無故障時の性能低下が約 1.1% にとどまっていることがわかった。

故障ノード数の増加に伴う性能の変化を測定した結果を図 5に示す。横軸は故障ノード数、縦軸はその時点での性能 (1 秒間に全メッシュ点の値を更新する回数) を表す。故障ノード数が等しくてもシステム内の故障の位置によって性能は異なるので、故障が順次発生していく例を五通り示す。実線は性能変化の理想的な特性である。

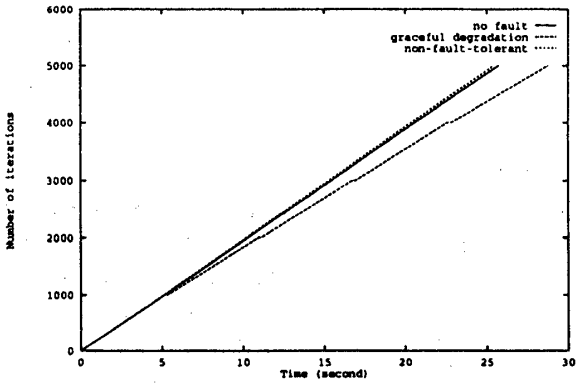

図 4: 漸次縮退の実験結果

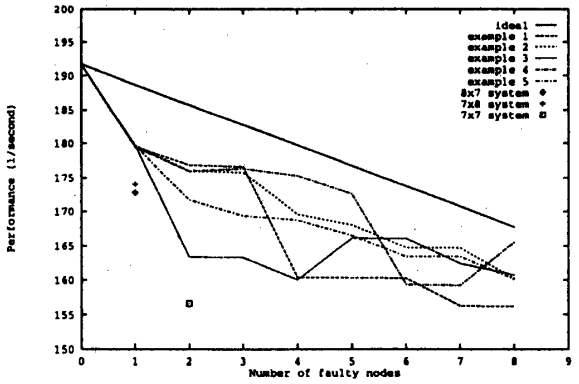

図 5: 縮退に伴う性能の変化

故障ノード数の増加に伴いかえって性能が向上している例があるのは、正常ノードをすべて利用する再構成法が必ずしも最善ではないためである。また、三つの点は行、列毎に縮退させた場合の性能である。これにより、行や列毎の縮退に比べると本方式の再構成法が優れていることがわかる。

6 まとめ

多次元メッシュ状アプリケーションを実行する多次元トーラス状の分散計算システム上で性能低下の極めて少ない漸次縮退の実現法を提案した。実際の計算機上でのシミュレーションにより、本方式の回復法では漸次縮退導入前と比べて性能低下が 1.1% と極めて少ないことと、本方式の再構成法では行や列毎に縮退させる方式と比べて性能が高いことを確かめた。尚、実験には (株) 富士通研究所並列処理研究センターの並列計算機 AP1000 を使用させて頂いた。

参考文献

[1] B. W. Lampson, M. Paul and H. J. Siegert (Eds.) : "Distributed Systems", Springer Verlag (1981).

[2] A. Avizienis : "Fault tolerance, the survival attribute of digital systems", Proc. IEEE, Vol. 66, No. 10, pp. 1109-1125 (Oct. 1978).

[3] B. R. Borgerson and R. F. Freitas : "A reliability model for gracefully degrading and stand-by sparing systems", IEEE Trans. Comput., Vol. C-24, No. 5, pp. 517-525 (May 1975).