





図 2: 階層挟み撃ち探索

3.2.2 探索領域の割り当て

PDF/IHSにおける探索領域の割り当ては、 $PE_1$ が $PE_1$ の探索経路上の深さの浅いノードを根として持つ部分木から順に、割り当て待ち状態にある $PE$ に割り当てていくという方法を用いる。このため、割り当て領域と割り当て深さが1対1に対応するので、次割り当て領域の指定が割り当て深さを指定するだけで簡単に行なえるので、スケジューラのオーバーヘッドを小さく抑えることができる。また、スレーブ $PE$ が割り当てられた領域の探索を開始する際には、 $PE_1$ の探索経路から探索に必要なデータをほぼ独立に各スレーブ $PE$ で再生できるため、領域割当て後は $PE$ 間のデータ転送を行なうことなく並列処理が可能である。

4 性能評価

本章では、PDF/IHSを分散共有キャッシュ型マルチプロセッサシステム Kendall Square Research社のKSR1上にインプリメントし、スケジューリングを行なった際の処理時間の計測結果について述べる。KSR1では32MBのLocal Cacheが階層的なALLCACHE ENGINEにより共有されており、ここでは32プロセッサまでの1クラスタを用いて計測を行なった。

4.1 評価方法

以下では用語の混乱を避けるため、スケジューリング問題の中で使われるプロセッサを単にプロセッサと表現し、PDF/IHSの並列処理のために使用するKSR1のプロセッサを $PE$ と表現する。

評価にあたっては、タスク間の先行制約が一般形状であり、タスク数50~500、平均タスク処理時間は20[u.t.]で標準偏差4[u.t.]の正規分布、各タスクの平均先行タスク数1~6個(ランダム)となるように、乱数を用いてタスクグラフを多数作成し、それぞれ2、3、4台のプロセッサにスケジューリングを行なった際に、いずれかのプロセッサ数においては下限値がCP/MISF解と一致せず探索を必要とするタスクグラフを100例選び、この合計300例のスケジューリング問題に対し、KSR1の $PE$ を2、4、8台用いてのPDF/IHSを適用し、その計算時間について以下の計算式より加速率を計算した。

$$\text{加速率} = \frac{\text{PDF/IHSを用いた場合の探索時間}}{\text{問題が強 NP 困難であることから、探索時間は600秒を上限とし、DF/IHSあるいは少数のPEを用いてのPDF/IHSでは時間内に最適解が得られなかったが、多数のPEを用いれば時間内に最適解が見つかったという場合には、探索を打ち切った時間で最適解が求まったものとして、加速$$

率を計算した。なお、8 $PE$ を用いても時間内に最適解が求まらない問題は存在しなかった。

また、より大規模な例として、タスク数1000、平均先行タスク数10個、その他の条件は同じとしたタスクグラフを5例作り、プロセッサ数4、5、6台(合計15通り)にKSR1の $PE$ を4、8、16台用いてスケジューリングした。

4.2 評価結果

前述の300例の問題では、表1の通り、加速率が $PE$ 数以上となる現象(スーパーリニアスピードアップ)がよく引き出されている。また、平均加速率も2 $PE$ で6.93倍、4 $PE$ で26.41倍、8 $PE$ で57.61倍と高い値になっており、PDF/IHSがDF/IHSに比較しても非常に高速な探索を可能としていることがわかる。

またタスク数1000の大規模問題に対しては、表2のようにDF/IHSでは制限時間内に最適解を得られない問題でも、 $PE$ を増やせば求解確率が上がることが確認できる。

表 1: 平均探索時間と加速率(タスク数500以下)

PE台数	1	2	4	8
平均探索時間(秒)	275.4	39.7	10.4	4.8
平均加速率(倍)	1.0	6.9	26.4	57.6
最大加速率(倍)	1.0	1473.5	2876.3	8968.6

表 2: 制限時間内最適解求解率(タスク数1000)

PE台数	1	4	8	16
最適解求解率(%)	26.7	80.0	93.3	100.0

5 まとめ

本稿では、初期解がCP/MISF解と一致せず、探索を必要とするようなタスクグラフに対して並列最適化マルチプロセッサスケジューリングアルゴリズムPDF/IHSを適用した場合の求解時間をKSR1上で評価した。その結果、DF/IHSでは求解に長時間を要した問題に対して平均で2 $PE$ で6.9倍以上のスピードアップが得られただけでなく、大規模問題に対しても有効であることが確認された。

今後は、データ転送を考慮したスケジューリング問題に対する実用的な最適化スケジューリングアルゴリズムを開発していく予定である。

謝辞

本研究の一部は文部省科学研究費補助金(一般研究(C) No.07680372)の補助で行なわれました。また本研究にあたりKSR1を使用させて頂いたキャノンスーパーコンピュータに感謝致します。

参考文献

- [1] H. Kasahara, S. Narita, "Practical Multiprocessor Scheduling Algorithms for Efficient Parallel Processing", IEEE Trans. on Computers, Vol.C-33, No.11, Nov.1984.
- [2] H. Kasahara, S. Narita, "Parallel Processing of Robot Arm Control Computation on a Multimicroprocessor System", IEEE Journal of Robotics and Automation, Vol.RA-1, No.2, Jun.1985.
- [3] 笠原, 伊藤(教), 田中, 伊藤(敬), "実行時間最小マルチプロセッサスケジューリング問題に対する並列最適化アルゴリズム", 信学論, Vol.J74-D-I, No.11, Nov.1991
- [4] E. B. Fernandez, B. Bussell, "Bounds on the Number of Processors and Time for Multiprocessor Optimal Schedules", IEEE Trans. Comp., Vol.C-22, No.8, Aug.1973.
- [5] Li G. J., Wah B. W., "Coping with anomalies in parallel branch-and-bound algorithms", IEEE Trans. Comput., Vol.C-35, No.6, Jun.1986.