

平成 13 年度

学士学位論文

DDP を用いた並列動画画像処理における 負荷分散方法の検討

An examination of the load distribution method
in the parallel video processing using DDP

1020326 松本 憲英

指導教員 島村 和典

2002 年 2 月 8 日

高知工科大学 情報システム工学科

要 旨

DDP を用いた並列動画像処理における 負荷分散方法の検討

松本 憲英

既存の DDP での負荷分散方法は静的に行われていた。しかし、この方法では動的なデータが送られてくる場合に現状の負荷分散方法では対応できない。本研究では、DDP においてネットワークデータのような動的な負荷に対しても効率の良い分散を行える方式を検討した。DDP では内部の負荷を観察することができない。よって LSP (仮想的な負荷分散プロセッサ) を置き、負荷を予測させ 2 つの DDP に分散させる。LSP において分散させる方式として方式 1 と方式 2 について実験を行った。この 2 つの負荷分散予測方式の特徴は次の通りである。

- 方式 1 : 入力毎に、負荷予測値を与える。その負荷予測値は一定時間で減少させる。
- 方式 2 : 入力毎に、漸増その後漸減する負荷予測値を与える。負荷予測値の変動属性として、増加率・最大負荷・減衰率を与える。

実験の結果から、方式 1 は予測値と実際の値との誤差が大きくなることがわかった。方式 2 は、方式 1 に比べ誤差も小さく非常に精度の高い結果が得られた。考察した結果、方式 1 は回路規模も小さく実装しやすいが誤差が大きくなることが考えられ、方式 2 は誤差は小さく精度は高いが条件が増えたために回路規模が増大することが考えられる。現段階では精度の高い方式 2 を提案する。今後の課題として回路規模に見合うだけの効果が得られるということを検証することが挙げられる。

キーワード DDP 負荷分散予測方式 ネットワークデータ 仮想負荷分散プロセッサ

Abstract

An examination of the load distribution method in the parallel video processing using DDP

The load distribution method in the existing DDP was performed statically. However, when dynamic data is sent by this method, it cannot respond by the present load distribution method. This research examined the system which can perform efficient distribution also to dynamic load like network data in DDP. In DDP, internal load is unobservable. Therefore LSP (Virtual load switching processor) is placed and load is predicted and it distributes to two DDP. I experimented about the load prediction method 1 and the method 2 in LSP. The feature of these two load prediction methods is as follows.

The method 1 gives a load prediction value for every input. The load prediction value is decreased in fixed rate.

The method 2 gives the load prediction value is given for every input. The value is gradual increased and then gradual decreased by LSP. As a change attribute of a load prediction value, the rate of increase, the maximum load, and the rate of decrease are given.

As for a method 1, the error of prediction value and actual value becomes large as a result of an experiment. Compared with the method 1, as for the method 2, the high result of accuracy with small error was obtained. Since the circuit scale is small, it is easy to mount a method 1. However, it is possible that an error is large. An error is

small and the accuracy of a method 2 is high. However,since conditions increased,it is possible that a circuit scale increases. The method 2 with high accuracy is proposed at a present stage. As a future subject,verifying that only the effect of balancing a circuit scale is acquired is mentioned.

key words DDP Load distribution prediction method Network data Virtual load switching processor

目次

第 1 章	研究の背景	1
第 2 章	研究の目的	2
2.1	目的	2
2.2	従来技術の問題点	2
2.3	DDP(Date Driven Processor) の利点	3
第 3 章	研究で扱う動画画像データ処理	6
3.1	DCT(離散コサイン変換)	6
3.2	本研究で扱う画像処理	8
第 4 章	本研究で取り組む課題	9
4.1	DDP	9
第 5 章	動的負荷分散方法	10
5.1	本研究における負荷とは	10
5.2	負荷分散方法を提案するにあたり	10
第 6 章	結果	18
6.1	方式 1 結果	18
6.2	方式 2 結果	21
第 7 章	考察	24
7.1	考察	24
7.2	方式 1 比較	24
7.3	方式 2 比較	26

7.4	考察の結果	27
第 8 章	まとめ	28
	謝辞	29
	参考文献	30

目次

2.1	パイプラインの破棄	3
2.2	DDP のデータの処理の流れ	3
2.3	データ駆動型	4
2.4	DDP のパイプライン構成	5
3.1	画像の 2 次元 DCT	6
3.2	4 × 4 画素値と DCT 係数	7
5.1	DCT フローグラフ	11
5.2	DCT 係数 8 個を繰り返し 1 回で演算した場合の負荷 (左)	12
5.3	DCT 係数 8 個を繰り返し 2 回で演算した場合の負荷 (右)	12
5.4	方式 1	12
5.5	方式 1-1	13
5.6	方式 2	13
5.7	方式 1	14
5.8	方式 1-1	14
5.9	方式 2	15
5.10	負荷分散機能付き DDMP の構成	15
5.11	LSP の内部構成	16
6.1	方式 1 による負荷分散予測	19
6.2	負荷分散決定ポイント	19
6.3	方式 1 による負荷分散	20
6.4	負荷分散決定ポイント	20
6.5	方式 2 による負荷分散予測	22

6.6	負荷分散決定ポイント	22
6.7	方式 2 による負荷分散予測	23
6.8	負荷分散決定ポイント	23
7.1	PE0 の比較	24
7.2	PE1 の比較	25
7.3	PE0 の比較	26
7.4	PE1 の比較	26

表目次

6.1 DCT を 3 分割した時に入力する際に与える負荷値	18
6.2 DCT を 3 分割した時に入力する際に与える負荷値	21

第 1 章

研究の背景

近年、情報通信の果たす役割があらゆる分野で大きくなりつつあり、また、その媒体である通信ネットワークは高速・広帯域の利用が進みつつある。この技術背景の下で、情報処理そのものも飛躍的な高速化が望まれている。現在のコンピュータは、ムーアの法則にしたがい 18 ~ 24ヶ月に 2 倍の速度向上が行われている。しかしネットワークの伝送速度はさらに進歩がはやく、このままではネットワークデータの処理にコンピュータが追いつかなくなる。また機器の高速化の他に、並列処理化で対応しようとする動きがある。しかし、既存のノイマン型プロセッサでの並列処理は効率が悪い点があり高速化するにあたり適しているとは言えない。そこで新たな情報処理体系が考えられ高速の情報処理を容易にする研究が進められている。

第 2 章

研究の目的

2.1 目的

DDP(Data Driven Processor) によるネットワークデータの高効率処理. ネットワークを通して多地点から送られてくる動画像データを同時に効率良く処理することを DDP を用いて検討していく. 動画像処理技術として DCT(離散コサイン変換) を用い, 動画像データを DDP に適応させた DCT に流し, 負荷の発生状況から DDP にかかる負荷を分散させる方式を検討する.

2.2 従来技術の問題点

プロセッサの利用効率が悪い

- パイプラインの破棄 (条件分岐やプロセスの切り替えによる破棄)

次ページの図 2.1 のようなパイプラインの処理で, jp まで処理がきた場合に条件分岐により sub に流れ飛んだ場合に jp の後に続く処理は破棄されてしまう.

プロセスの切り替えをする際もパイプラインの破棄が行われる.

- 同期パイプライン (処理に関わらず電力が必要)

パイプラインの全体で同期を取るために常に作動している. 処理を行わない間も作動する必要があるために余分な熱量が消費される.

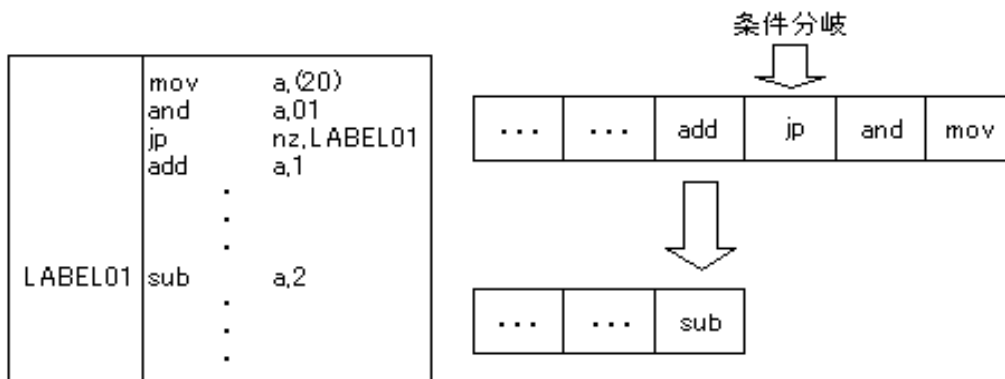


図 2.1 パイプラインの破棄

2.3 DDP(Date Driven Processor)の利点

プロセッサの利用効率が良い

- データ駆動型 (パイプラインの破棄が無い)

データ駆動型ではデータにデータの識別情報、命令コード等のタグ情報を付けデータ処理を行う。このためデータとデータに依存関係がないので、これらのデータを並列的に処理することができる。図 1 は DDP のデータの処理の流れを図示している。

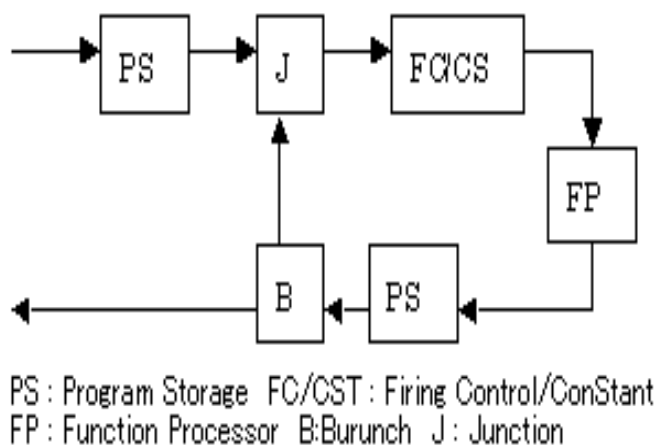


図 2.2 DDP のデータの処理の流れ

PS は命令コード・データ識別情報等の値を与える。J は PS と B から送られてくるデー

データを競合しないようにする,FC/CST はデータの待ち合わせを行う.FP は演算を行う.

データ駆動型では先に命令に入ったデータは対になる次のデータが命令に入ってくるまで待ち,二つのデータが揃って初めて命令を実行する.

図 2.3 は $c=a+b$ を出力する場合のデータ駆動型の処理の流れを図示している.

左側が $a=1$, 右側が $b=3$ のデータが命令に入ってくる場合には $1+3$ の演算が実行されるが, 左側が $a=1$, 右側が $b=3$ が入ってくる前に $b=4$ が入ってきても $1+4$ の演算は行われない. これはデータ自体に世代番号を持たせ同じ世代のデータ同士で演算するからである.

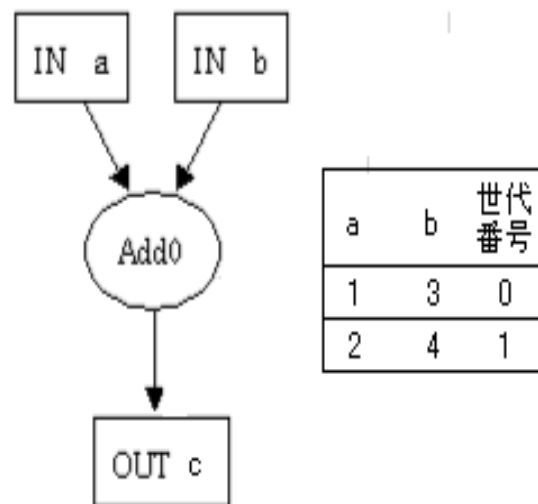


図 2.3 データ駆動型

- 非同期パイプライン (データ駆動を行う回路のみ動作)

データ処理時のみ動作するため消費電力が低い.

クロックではなく C 素子のハンドシェイクにより処理が行われる.

データが入力された時点で処理が始まるので遅延・ジッタなどに強い

図 2.4 は DDP のパイプラインを図示

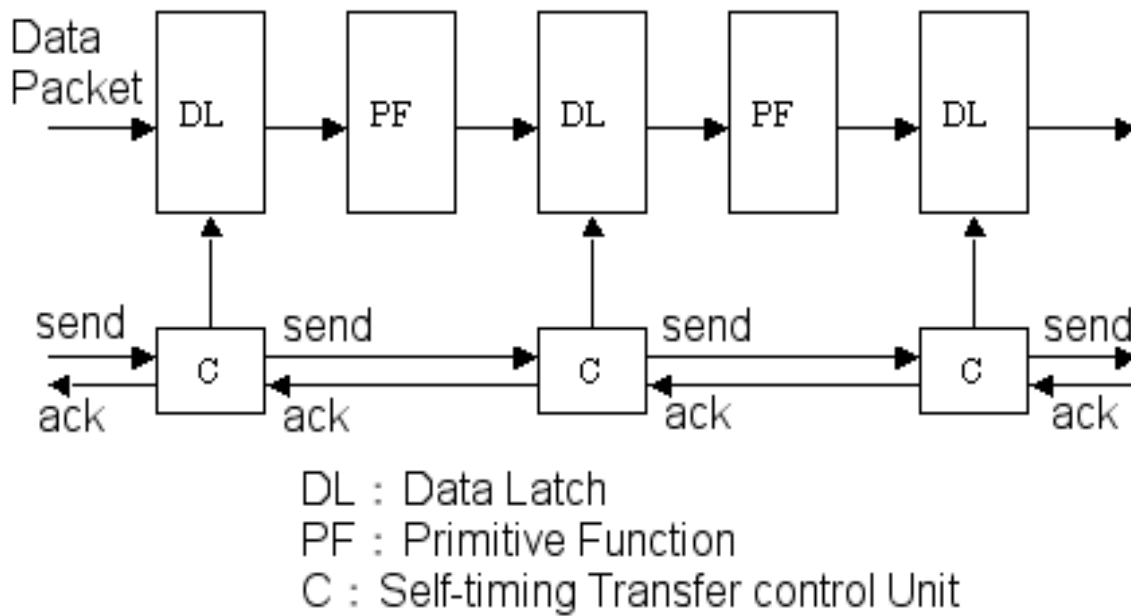


図 2.4 DDP のパイプライン構成

C素子は前段からの転送要求信号と後段からの転送許可信号の両方がアクティブな時に Data latch にクロック信号を送る。転送要求信号は前段で処理中のデータパケットを DL に入力する準備ができていることを示す。転送許可信号は後続ステージが空であることを示す。

第 3 章

研究で扱う動画画像データ処理

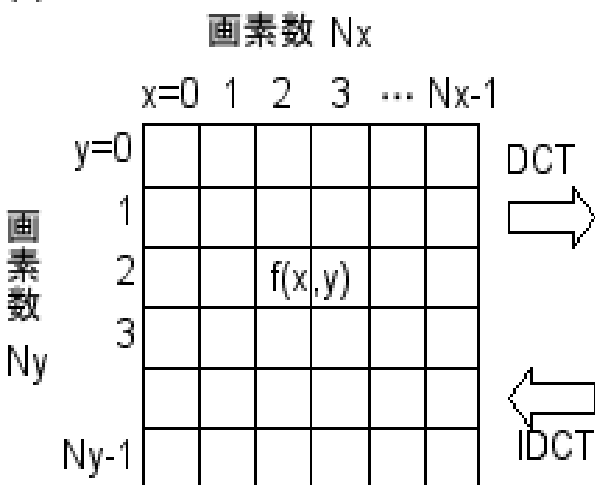
3.1 DCT(離散コサイン変換)

画素数 $N_x \times N_y$ のデジタル画像情報が図 3.1(a) のように与えられた時,DCT の演算式は下記式のようなになる. 横方向位置 x , 縦方向位置 y の画素値を $f(x,y)$ とする時, 周波数軸上で横方向位置 u ・縦方向位置 v の DCT 係数は $F(u,v)$ で与えられる. 図 3.1(b)

$$F(u, v) = \frac{2c(u)c(v)}{\sqrt{N_x N_y}} \sum_{x=0}^{N_x-1} \sum_{y=0}^{N_y-1} f(x, y) \cos\left\{\frac{(2x+1)u}{2N_x}\right\} \cos\left\{\frac{(2y+1)v}{2N_y}\right\}$$

ただし、 $c(i) = \frac{1}{\sqrt{2}} \dots i = 0, 1 \dots i \neq 0$

(a)原画像



(b)DCT変換後

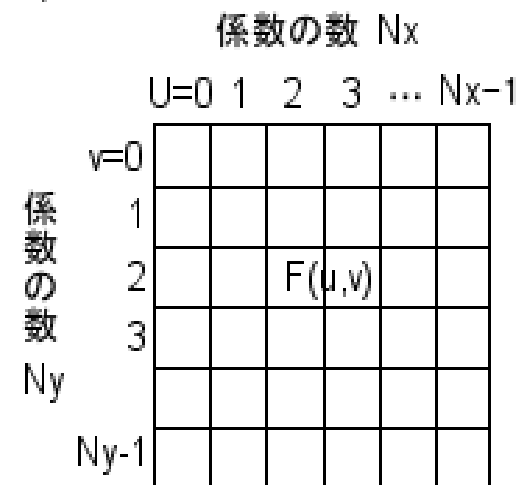


図 3.1 画像の 2 次元 DCT

下図 3.2(a) の 4×4 画素に対して DCT を行うと下図 3.2(b) のような係数群が得られる。DCT 係数は周波数成分を表し、この場合左から右、上から下に周波数が高くなっている。(b) の左上端の値 548 は全ての画素値の平均値 $(200+180+160+140+175+\dots+100)/16=137$ の 4 倍に一致する。つまりこの値は画像全体の画素値の平均値に比例する値になるので、この値を省略すると全体が明るくなったり暗くなったり画像が大きく変わることになる。それに対して右下の値は省略しても影響は大きくない。JPEG や MPEG の圧縮符号化では、低周波成分より高周波成分の省略を大きくするように DCT 係数の係数を減らして符号量を削減する。このような情報省略が行いやすいことから DCT を圧縮符号化に使用する。

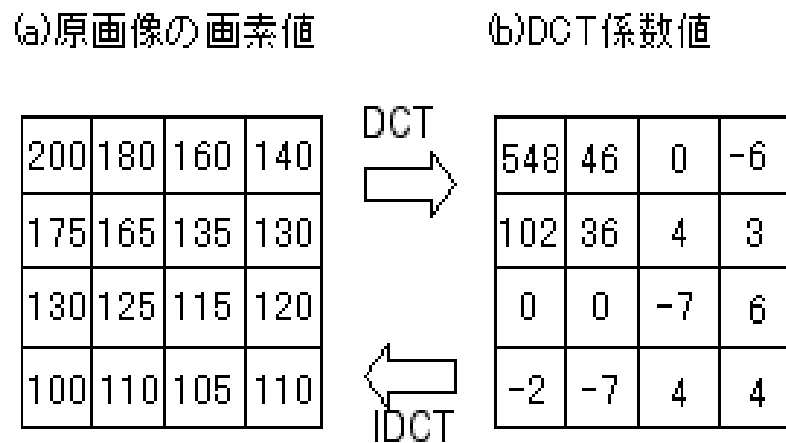


図 3.2 4×4 画素値と DCT 係数

3.2 本研究で扱う画像処理

本研究では

8 × 8 の 64 画素を単位とするブロックに分割し DCT を行う。

YUV(輝度 Y 信号と赤の色差 U と青の色差 V の 2 つの色度で色を表現する方法) のうち

Y のみで行う。

下に動画画像処理技術について表記する。

動画画像処理技術	
動き補償予測	演算時間より, メモリアクセス時間が大きな割合を占める DDP 利用効果が薄い
量子化	処理量が小さい
<i>DCT</i>	並列化に適當
可変長符号 (ハフマンランレングス符号)	逐次処理のため並列化が困難

第 4 章

本研究で取り組む課題

4.1 DDP

問題点

- DDP 内部のパケットを実際に観測することができない。

DDP は非同期パイプラインをとるため、ある時点でプロセッサを止めて内部状態を観測することができない。

パイプライン内のパケットの有無を把握しにくい。

- 動的なデータが流れてくる場合に現状の負荷分散方法では対応できない。

現在の DDP の静的な負荷分散方式ではプロセッサでの割り振りが決まっているので、データの種類、受信タイミングが動的に変化するネットワークデータを処理する場合に負荷を適切に割り振ることができない。

第 5 章

動的負荷分散方法

5.1 本研究における負荷とは

プロセッサ内のパケットの数を負荷と考える。

第 4 章の DDP の問題点より DDP 内部の負荷の観測は困難である。

5.2 負荷分散方法を提案するにあたり

- DCT における全体の負荷を算出

DDP を用いた DCT に NTSC(縦 720*横 480*30 フレーム/sec) のデータを入力し DCT 全体の負荷を算出する。使用する DDP のパイプラインの動作は 120MHz 周期であるために、DDP に入力される間隔は $120\text{MHz}/10368000 = 12\text{time}$ となる。8*8 の 64 画素単位のブロックで DCT を行うため、8 つの DCT 係数を 12time の間隔で入力し演算する。time とはパイプライン一段の処理時間である。扱う画像データは YUV(輝度 Y 信号と赤の色差 U と青の色差 V の 2 つの色度で色を表現する方法) のうち Y のみとする。これは YUV では Y が最も重要なデータであり、U・V は画像フォーマットによってデータ量が大きく異なるため今回の検討から除外した。

次のページに DCT フローグラフを図示する。

butter2 -s.fg

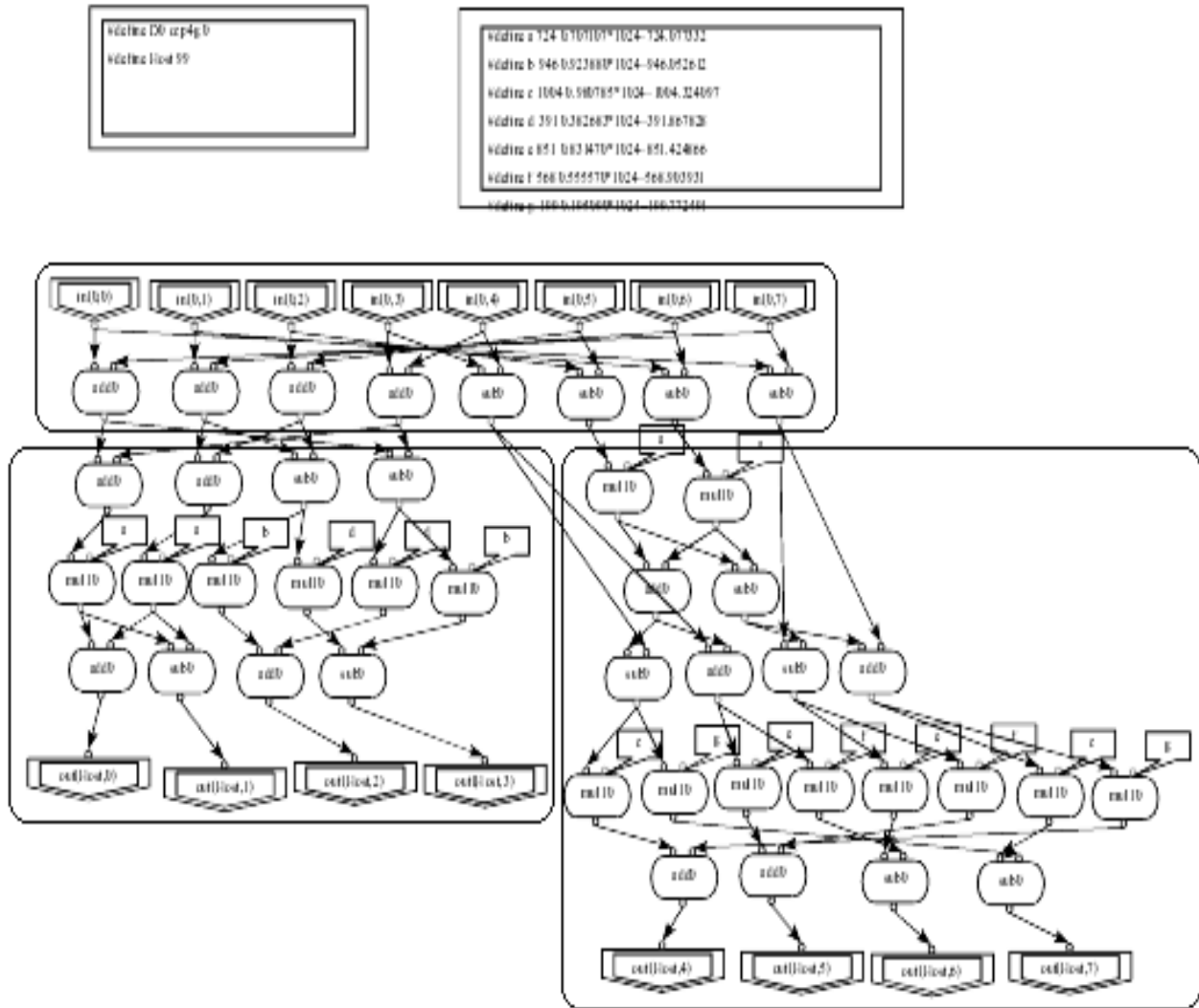


図 5.1 DCT フローグラフ

DCT の処理は大きいので 8×14 (最上部), 8×4 (左部分), 6×4 (右部分) の 3 つのブロックに分割して考える.

8 入力の場合の DCT 全体の負荷を下に図示する.

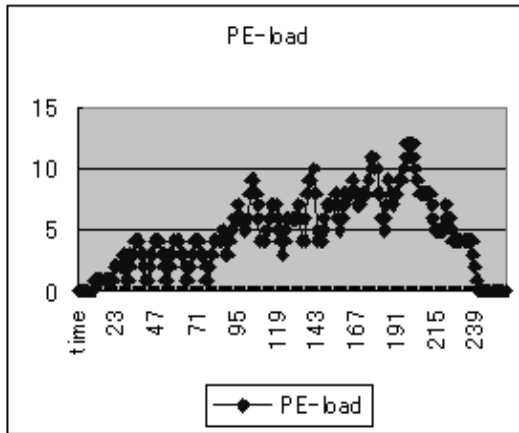


図 5.2 DCT 係数 8 個を繰り返し 1 回で演算した場合の負荷 (左)

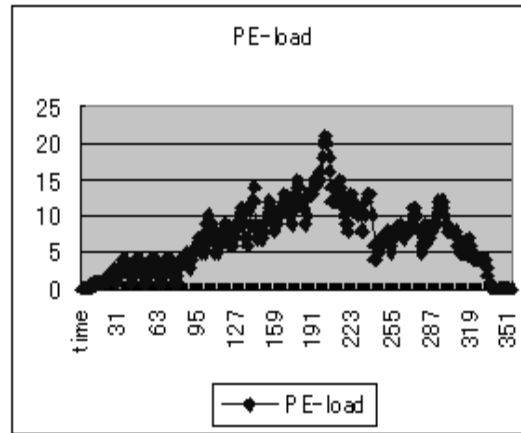


図 5.3 DCT 係数 8 個を繰り返し 2 回で演算した場合の負荷 (右)

● 負荷予測実験

－ 方式 1(図 a)

- * データを入力する際に, その入力毎の負荷を与える.
- * その負荷は一定の時間で減少させる.

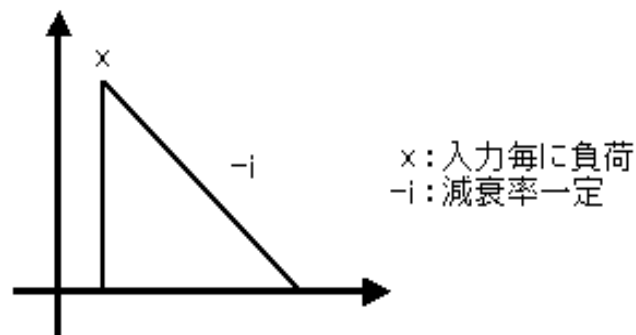


図 5.4 方式 1

－ 方式 1-1(図 b)(方式 1 の応用)

- * データを入力する際に, その入力毎の負荷を与える.
- * その負荷は入力毎に減少させる.

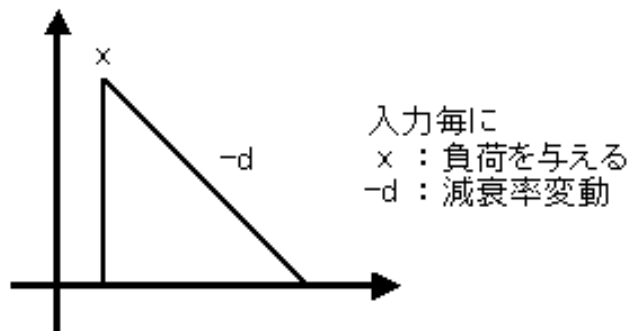


図 5.5 方式 1-1

－ 方式 2(図 c)

- * データを入力する際に, 漸増その後漸減する負荷を与える.
- * 負荷の変動属性として, 増加率, 最大負荷, 減衰率を与える.

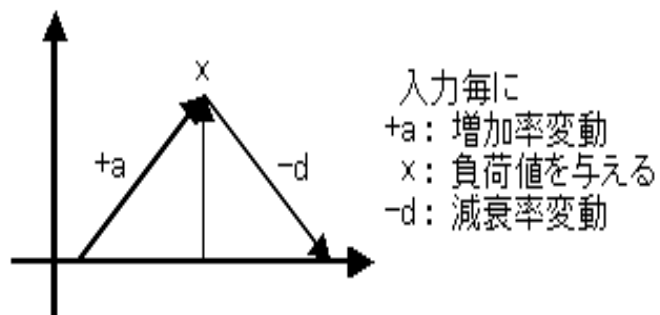


図 5.6 方式 2

方式 1, 方式 1-1, 方式 2 で DCT 係数 8 個を繰り返し 1 回の全体の負荷を表してみると
 下図になる. 図 5.7 は下記の条件によって方式 1 で表したグラフ

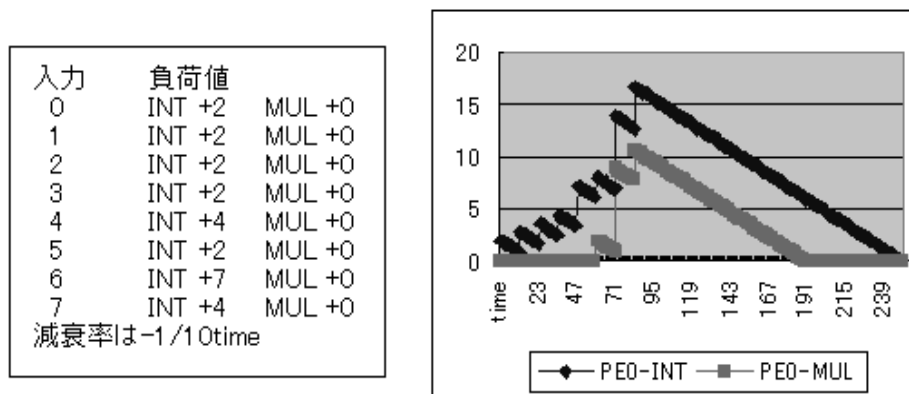


図 5.7 方式 1

図 5.8 は下記の条件によって方式 1-1 で表したグラフ

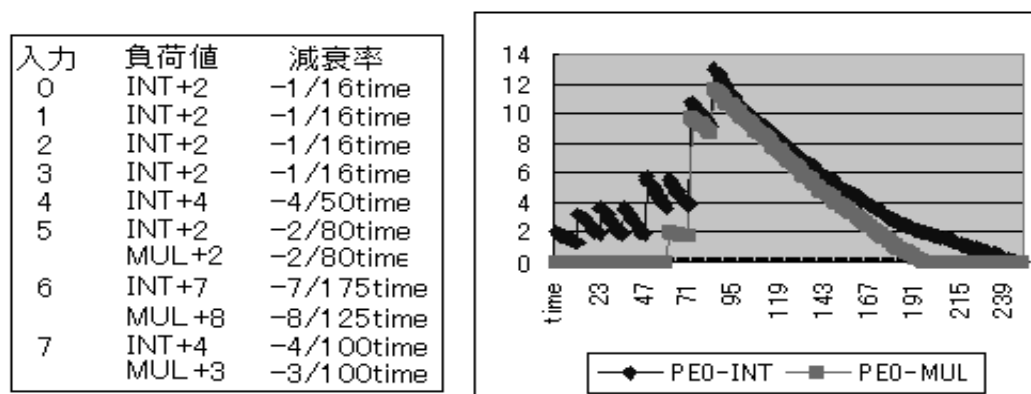


図 5.8 方式 1-1

図 5.8 から方式 1-1 は条件が増える割に, 方式 1 と比べそれほどたいした効果をもたらしていないと判断し, 方式 1-1 を用いた負荷分散の予測は行わないことにする.

図 5.9 は下記の条件のよって方式 2 で表したグラフ

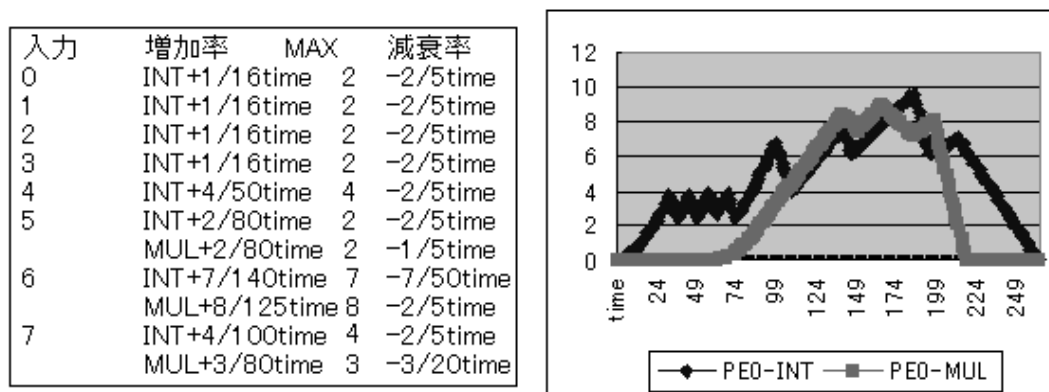


図 5.9 方式 2

● 負荷予測分散

シュミレータにより予測分散を算出する.

図 5.10 は負荷分散機能付き DDMP の構成を図示する.

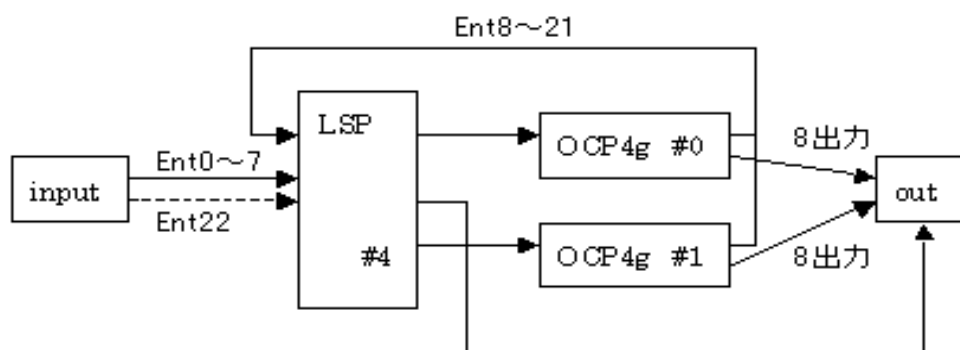


図 5.10 負荷分散機能付き DDMP の構成

input から Ent(入力)0 ~ 7 が入力され LSP に送られる。LSP は方式 1 または方式 2 を適用して後続の 2 つのプロセッサの負荷予測値の合計を出し、常に比べ、予測負荷値の小さい方にパケットを送る。Ent0 ~ 7 の演算結果は Ent8 ~ 21 として再度 LSP に入力し Ent8 ~ 15, Ent16 ~ 21 に負荷分散を行う。同じ世代の演算が終われば出力する。Ent22 は、LSP の負荷予測値観測用のダミー入力である。

方式 1 における LSP の内部の構成を下に図示する。図 5.11 は LSP の内部構成を図示

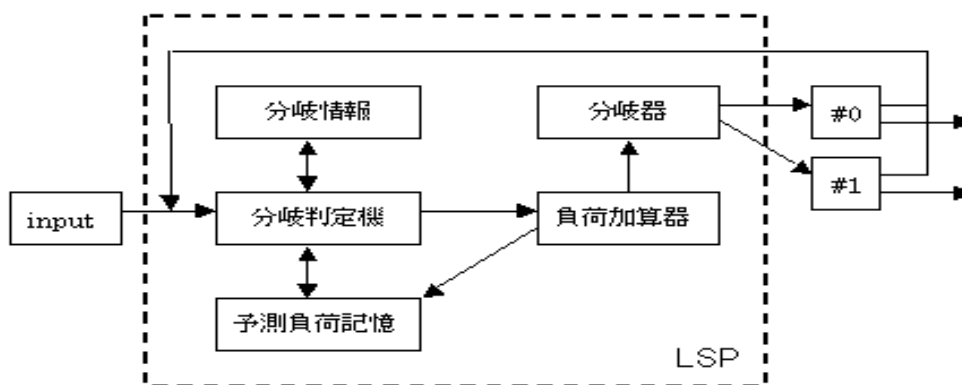


図 5.11 LSP の内部構成

input : データがエントリーされる。

分岐情報 : DDP では、例えば、世代番号が 0 の入力 0 ~ 7 があり、入力 0 がプロセッサ 0 に割り振られた場合に残りの入力 1 ~ 7 はプロセッサ 0 に自動的に割り振る必要がある。この世代番号が同じパケットがどちらのプロセッサに送られたかの情報を保持する場所である。

分岐判定器 : 分岐情報にすでに同じ分割内の入力を送られている場合には、そのパケットと同じプロセッサ番号をパケットに設定する。分岐情報から自動的にプロセッサが割

り振られるパケットでない場合には、予測負荷記憶からプロセッサ 0 かプロセッサ 1 の負荷の少ない方へ分岐するようにパケットのプロセッサ番号を変更する。プロセッサ番号が決定されたら、後続のパケットのため分岐情報にそのパケットの入力と世代番号を記憶させる。

予測負荷記憶：負荷加算器から送られてくるプロセッサ番号と負荷の offset から、プロセッサの負荷へ offset を追加する。

予測加算器：分岐判定器より分岐が決定されて送られてくるパケットの情報から、入力に対応する負荷の offset を、プロセッサの負荷に追加または減少するよう予測負荷記憶に送る。

分岐器：負荷加算器より送られてくるパケットを、プロセッサ番号を参照して適切なプロセッサへ送り出す。

第 6 章

結果

6.1 方式 1 結果

入力	負荷予測値	入力	予測負荷値	入力	予測負荷値	入力	予測負荷値
0	200	6	200	12	400	18	300
1	200	7	200	13	300	19	1300
2	200	8	100	14	400	20	100
3	200	9	100	15	300	21	100
4	200	10	100	16	100	-	-
5	200	11	100	17	100	-	-

表 6.1 DCT を 3 分割した時に入力する際に与える負荷値

DDP では小数の演算が行えないため、方式 1 の場合は負荷値は 100 倍して与える。

条件：3 ストリーム 繰り返し 5 回

減衰率は負荷予測値の合計×ストリーム数×繰り返しの数÷入力を開始してから全出力が得られるまでの時間となる。上記の条件から $5400 \times 3 \times 5 / 639 + 164 \times 104 / \text{time}$ とする。(time は DDP の処理の周期)

dec-PE0 と dec-PE1 は負荷を分散させるポイントを表している。

負荷分散決定ポイントグラフで表されている時間に負荷予測値が入力されている。

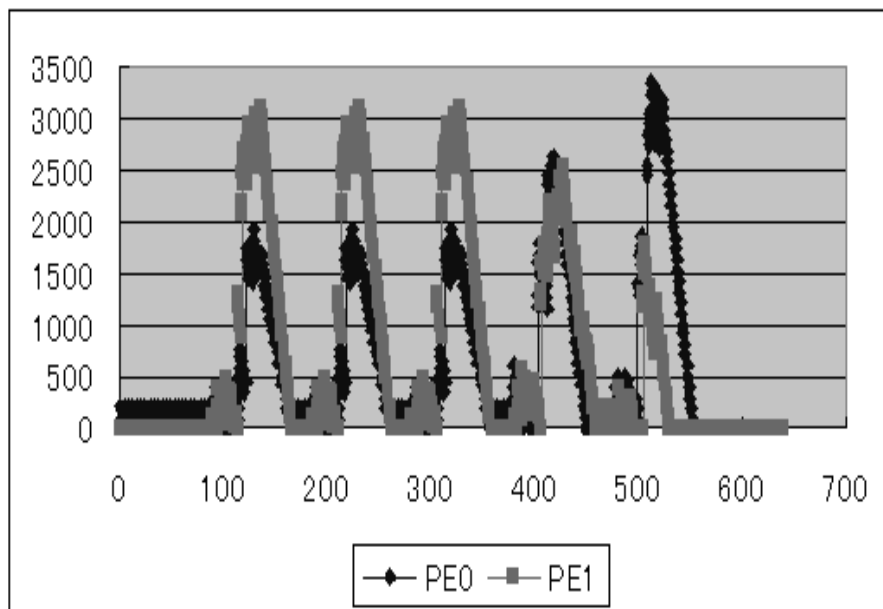


図 6.1 方式1 による負荷分散予測

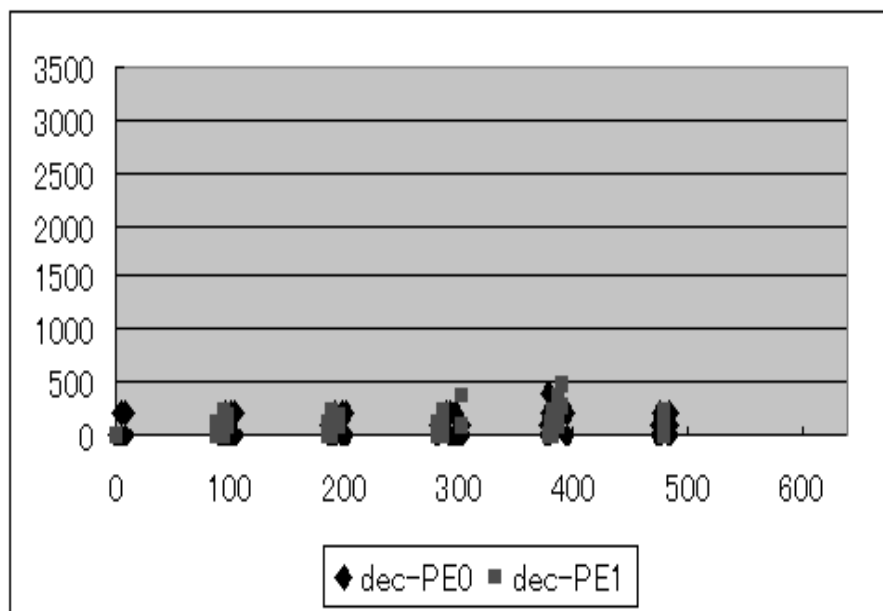


図 6.2 負荷分散決定ポイント

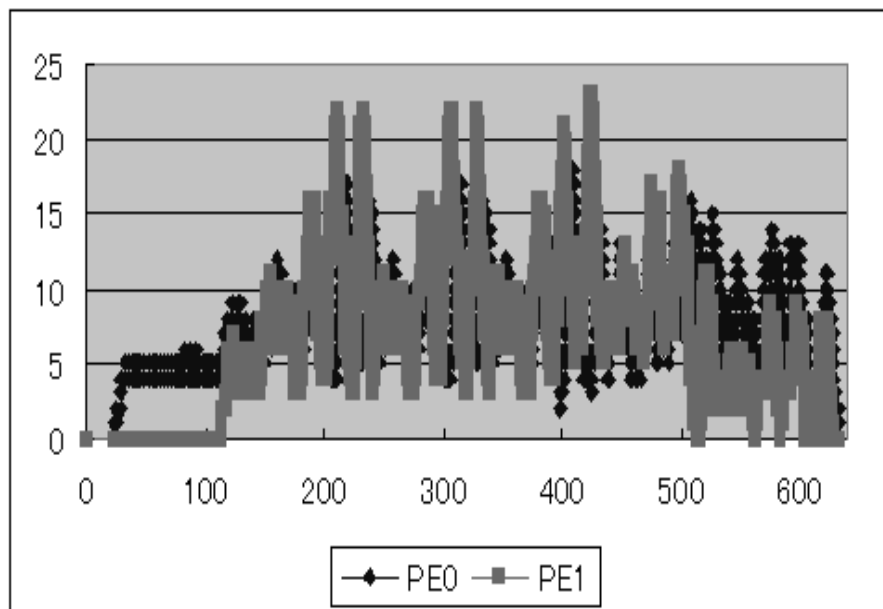


図 6.3 方式 1 による負荷分散

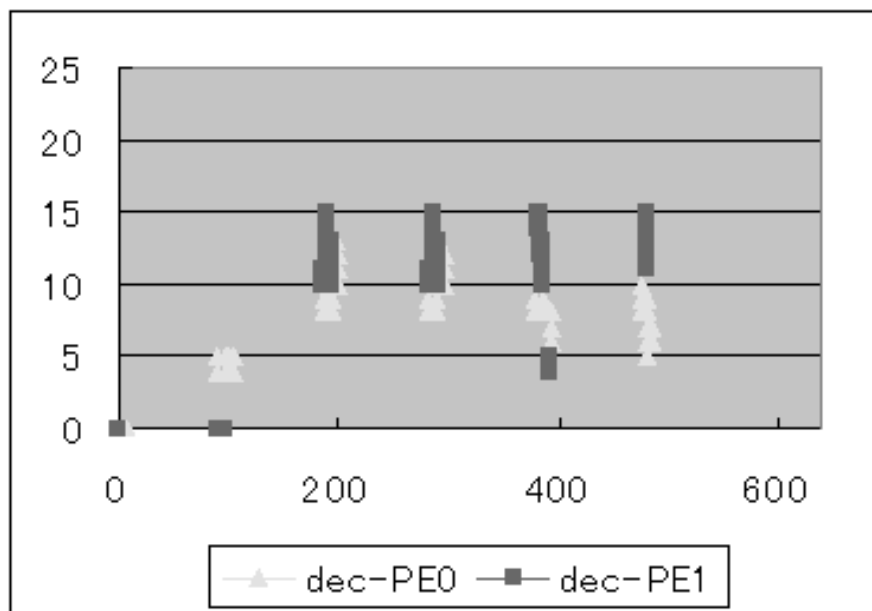


図 6.4 負荷分散決定ポイント

6.2 方式 2 結果

入力	増加率	最大負荷	減衰率	入力	増加率	最大負荷	減衰率
0	625	20000	4000	11	2000	10000	4000
1	625	20000	4000	12	1500	30000	750
2	625	20000	4000	13	1500	30000	750
3	625	20000	4000	14	1500	30000	600
4	625	20000	4000	15	1500	30000	600
5	625	20000	4000	16	2000	10000	2000
6	625	20000	4000	17	2000	10000	2000
7	625	20000	4000	18	625	20000	2500
8	2000	10000	4000	19	1500	120000	3000
9	2000	10000	4000	20	2000	10000	2000
10	2000	10000	4000	21	2000	10000	2000

表 6.2 DCT を 3 分割した時に入力する際に与える負荷値

DDP では小数の演算が行えないため、方式 2 の場合は負荷値は 10000 倍して与える。

条件：3 ストリーム 繰り返し 5 回

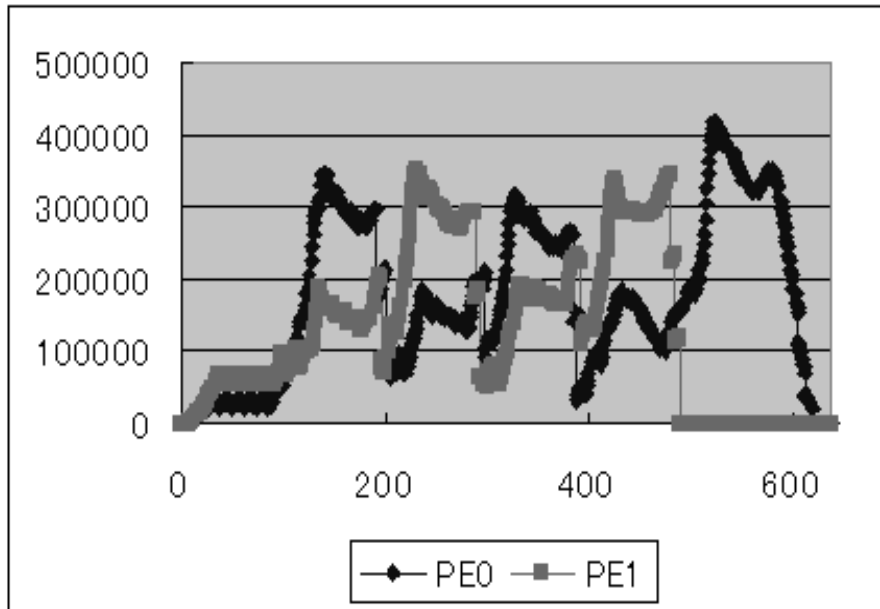


図 6.5 方式 2 による負荷分散予測

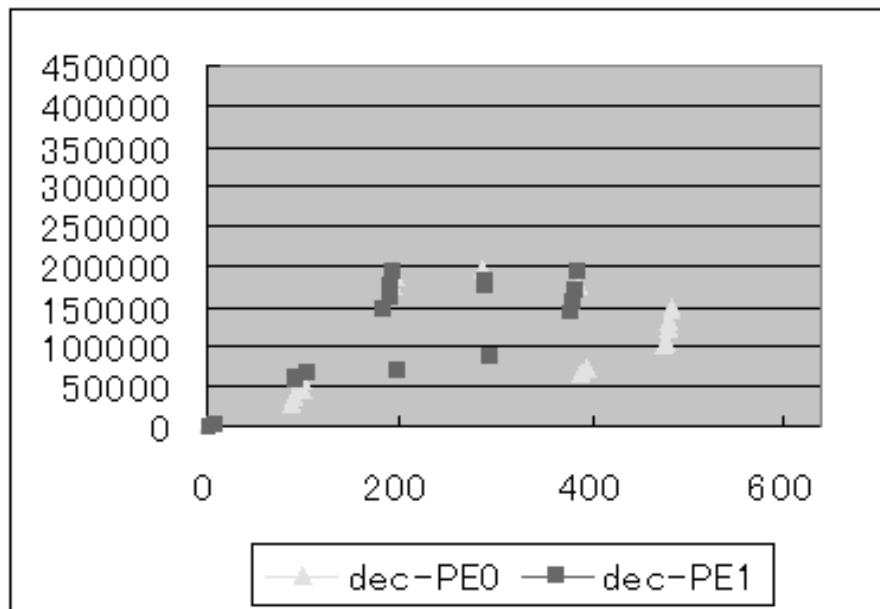


図 6.6 負荷分散決定ポイント

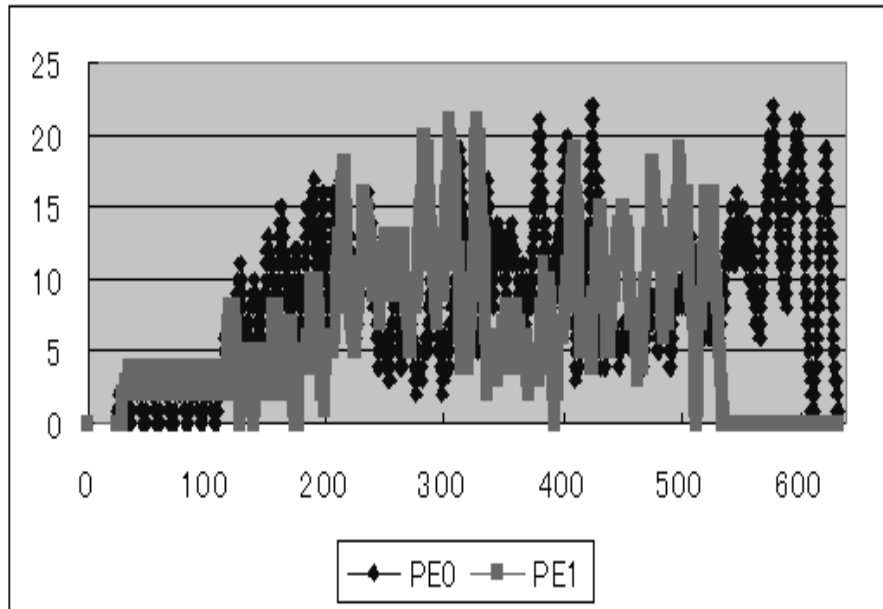


図 6.7 方式 2 による負荷分散予測

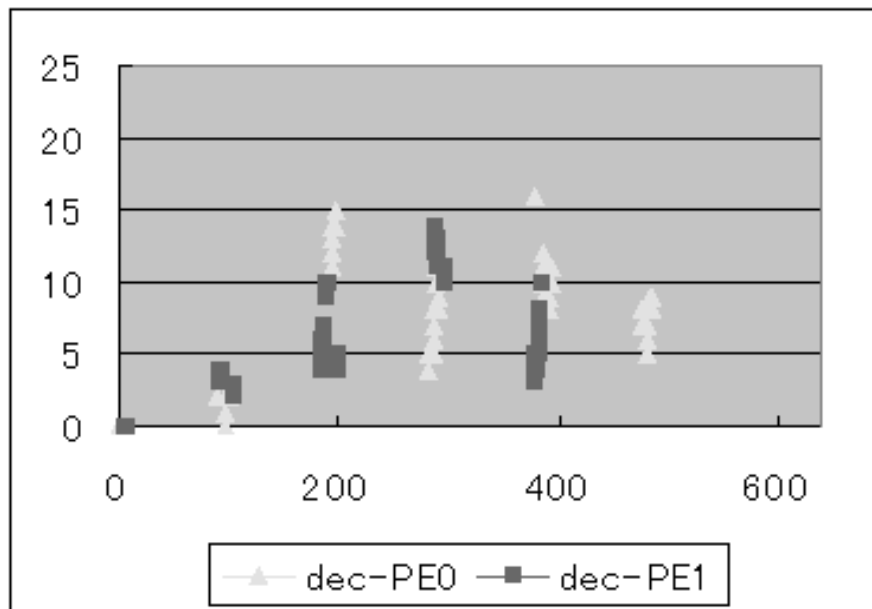


図 6.8 負荷分散決定ポイント

第7章

考察

7.1 考察

- ・第6章の結果から方式1と方式2の予測値と実際の値とを比較する.

7.2 方式1比較

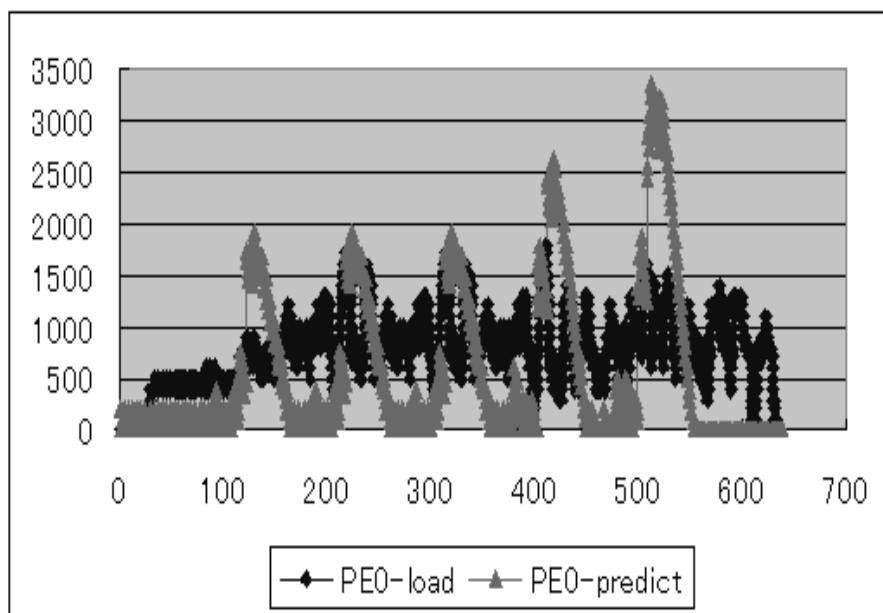


図 7.1 PE0 の比較

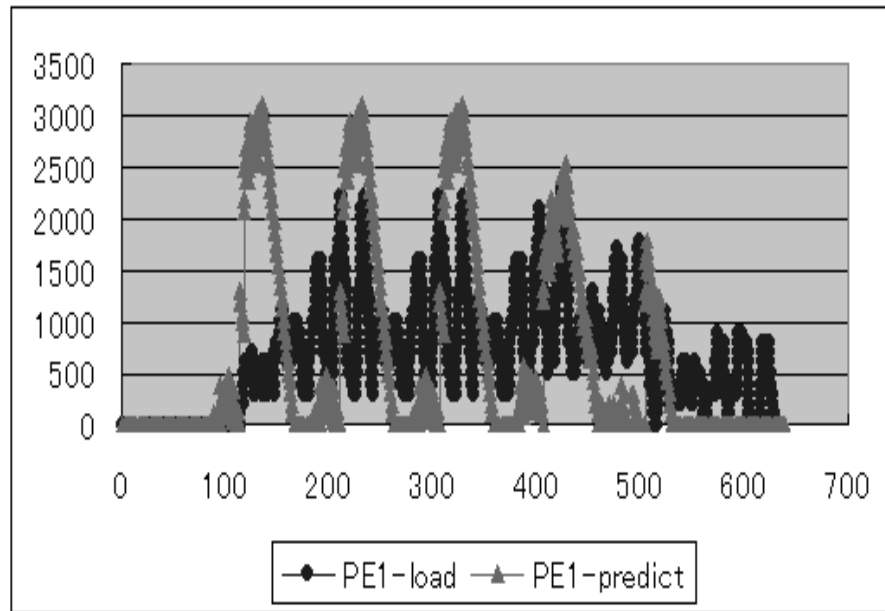


図 7.2 PE1 の比較

図 7.1 からは、序盤は誤差も少なく分散できているが、終盤で誤差が大きくなり、うまく分散できていないことがわかる。

図 7.2 からは序盤と終盤の誤差が大きいことがわかる。

また互いの図から予測値の収束が速すぎることがわかる、この原因としては減衰は一定であるが、負荷予測値にバラツキがあるため、負荷予測値が小さい部分は収束が速くなるからであると考えられる。

7.3 方式 2 比較

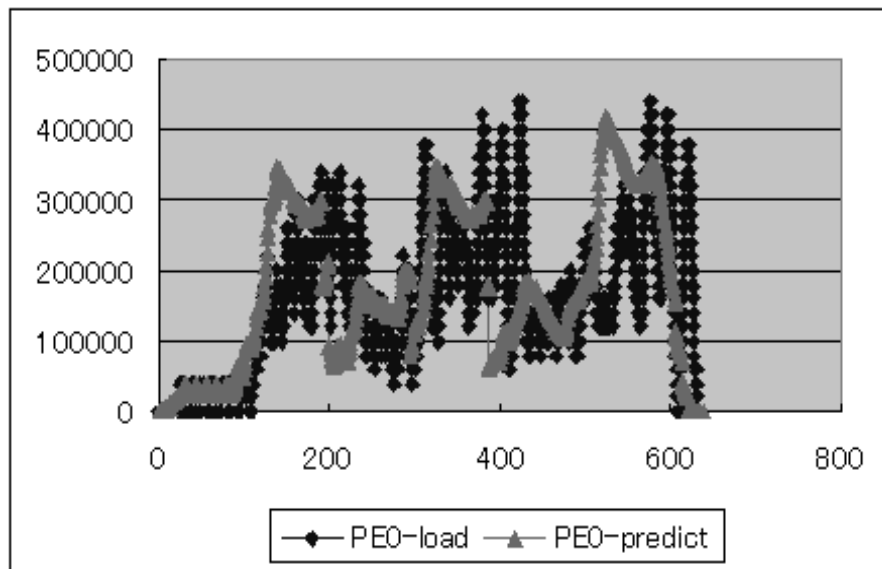


図 7.3 PE0 の比較

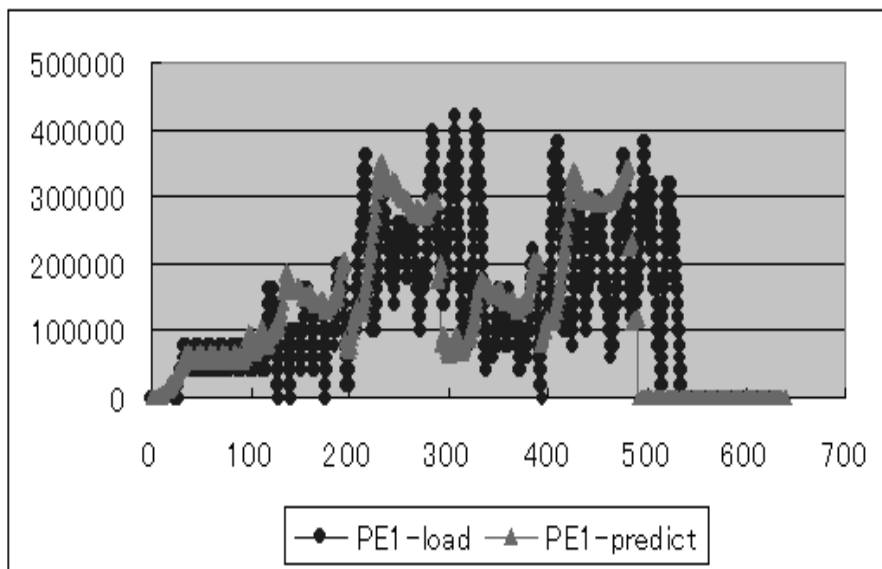


図 7.4 PE1 の比較

方式 1 での比較と比べると PE0,PE1 共に誤差も少なくより精度の高い分散が行えている.

7.4 考察の結果

これらの図から方式 1 より方式 2 の方が精度の高い分散が行えることがわかる。しかし、方式 2 では変動属性としての条件が増えたために回路規模が大きくなることが考えられ、それに見合うだけの効果が得られたかというところが問題となる。

第 8 章

まとめ

通信ネットワークが大幅に進歩し、それに対応すべく処理機能の向上が図られているなか、本研究では DDP の動的な負荷分散方法を検討した。負荷分散を行うに当たって予測負荷値から分散させるために方式を 2 つ提案した。方式 1 は回路規模が小さく実装しやすいが誤差が大きく適切な分散が行われなかった。方式 2 は方式 1 に比べ誤差も少なく精度の高い予測値を出せた。方式 1 と方式 2 では方式 2 の方がはるかに優れているので本研究では方式 2 を推奨する。今後の課題として回路規模の増大に見合うだけの効果を得られるかを検証することが挙げられる。

謝辞

本研究を行うにあたり、多くの御指導、御鞭撻を賜りました本情報システム工学科の島村和典教授に深く感謝いたします。また、常日頃より多くの御意見、御討論を頂きました通信・放送機構高知トラフィックセンターの高松希匠研究員に深く感謝いたします。また、貴重な意見を頂きました小林寛征さんをはじめとする島村研究室の皆様、通信・放送機構高知トラフィックセンターの皆様に深く感謝いたします。

参考文献

- [1] 弓場敏嗣, 山口喜教, ”データ駆動型並列処理計算機” オーム社
- [2] 越智宏, 黒田英夫, ”JPEG & MPEG 図解でわかる画像圧縮技術” 日本実業出版社
- [3] 小泉修, ”図解でわかる PC アーキテクチャのすべて” 日本実業出版社