Convergence of the Discrete FGDLS Algorithm

Sabin Tabirca[1], Tatiana Tabirca[1,⋆], and Laurence T. Yang[2]

[1] University College Cork, Computer Science Department,
College Road, Cork, Ireland
s.tabirca@cs.man.ac.uk

[2] St. Francis Xavier University, Computer Science Department,
Antigonish, B2G 2W5, Canada
lyang@stfx.ca

Abstract. The Feedback-Guided Dynamic Loop Scheduling (FGDLS) algorithm [1] is a recent dynamic approach to the scheduling of a parallel loop within a sequential outer loop. Earlier papers have analysed convergence under the assumption that the workload is a positive, continuous, function of a continuous argument (the iteration number). However, this assumption is unrealistic since it is known that the iteration number is a discrete variable. In this paper we extend the proof of convergence of the algorithm to the case where the iteration number is treated as a discrete variable. We are able to establish convergence of the FGDLS algorithm for the case when the workload is monotonically decreasing.

1 Introduction

It is widely recognised that loops are a very important source of parallelism in many practical applications. Since a significant overhead in many parallel implementations is represented by load imbalance, a number of algorithms have been designed to schedule loop iterations to processors of a shared-memory machine in an optimal way (so-called loop scheduling algorithms).

An important class of loop scheduling algorithms is based on *Guided Self-Scheduling* (Polychronopoulos and Kuck [8]) or some variant of *Guided Self-Scheduling* (see for example Eager and Zahorjan [4], Hummel et al. [5], Lucco [6], Tzen and Ni [14]). These algorithms divide the loop iterations into a relatively large number of chunks which are assigned to processors from a central queue. One of the motivations for this approach is the assumption that each execution of a loop is independent of any previous executions of the same loop, and therefore has to be rescheduled 'from scratch'. Important overheads such as additional synchronisation, loss of data locality, and reductions in the efficiency of loop unrolling and pipelining can be caused by this approach.

The class of *Affinity Scheduling* algorithms (Markatos and LeBlanc [7], see also Subramanian and Eager [9] for variants of *Affinity Scheduling*) is an attempt to ameliorate some of this loss of performance. Rather than maintaining a single central queue these algorithms are based on per-processor work queues with exchange of work

⋆ This work was supported by Boole Centre for Research in Informatics at National University of Ireland, Cork.

L.T. Yang et al. (Eds.): HPCC 2005, LNCS 3726, pp. 233–244, 2005.
© Springer-Verlag Berlin Heidelberg 2005

(chunks of the loop iteration) if required. The underlying assumption of affinity scheduling algorithms remains that each execution of a parallel loop is independent of previous executions.

Feedback Guided Dynamic Loop Scheduling (FGDLS) is a relatively recent scheduling method ([1], [2]), which deals directly with a sequence of similar or identical parallel loops (see Figure 1). This loop structure is very important since it frequently occurs in a number of theoretical [10], [11] and practical applications [2]. The convergence of the FGDLS method has been studied in [3] and [12]. In these papers the workload is assumed to be a continuous positive function of a continuous argument (iteration number). However, the approach is artificial since in reality the workload is a positive function of the discrete argument (iteration number).

1.1 The FGDLS Algorithm

The FGDLS algorithm aims to determine an optimal schedule, across p processors $P_1, P_2, ..., P_p$, for the sequence of parallel loops given in Figure 1. The (unknown) workloads of the parallel loop are assumed to be given by the values $\{w_i,\ i = 1,2,...,n\}$ (so that w_i is the workload of the call to the routine `loop body(i)`). The FGDLS algorithm calculates a block partitioning of the parallel (`i`) loop, where l_j^t and h_j^t are the lower and upper bounds of the loop block assigned to Processor j on outer iteration t. These bounds clearly should satisfy the simple equations

$$l_1^t = 1;\ \ h_p^t = n;\ \ l_{j+1}^t = h_j^t + 1, j = 1,2,...,p-1. \tag{1}$$

FGDLS starts with some initial loop bounds $\{(l_j^1, h_j^1),\ j = 1,2,...,p\}$ that are chosen arbitrarily. At the end of the outer iteration t, the new bounds $\{(l_j^{t+1}, h_j^{t+1}),\ j = 1,2,...,p\}$ are calculated from the bounds $\{(l_j^t, h_j^t),\ j = 1,2,...,p\}$ by approximately balancing the observed execution times. Assuming that the observed execution times $\{T_j^t, j = 1,2,...,p\}$ are given by

$$T_j^t = \sum_{i=l_j^t}^{h_j^t} w_i,\ j = 1,2,\ldots,p, \tag{2}$$

a piecewise constant approximation of the workload at the iteration t can be formed as

$$\hat{w}_i^t = \frac{T_j^t}{h_j^t - l_j^t + 1},\ l_j^t \le i \le h_j^t,\ j = 1,2,\ldots,p. \tag{3}$$

```
do sequential t = 1, nsteps
   do parallel i=1,n
      call loop_body(i)
   end do
end do
```

Fig. 1. The FGDLS loop structure

The piecewise constant workloads $\hat{w}_i^t$ can be interpreted as the mean observed workload per loop iteration index on the outer iteration t. It is this piecewise constant function that is approximately equidistributed amongst the p processors to define the new loop bounds $\{(l_j^{t+1}, h_j^{t+1}),\ j = 1,2,...,p\}$:

$$\sum_{i=l_j^{t+1}}^{h_j^{t+1}} \hat{w}_i^t \approx \frac{1}{p}\sum_{i=1}^{n} \hat{w}_i^t = \frac{1}{p}\sum_{k=1}^{p} T_k^t,\ \ j = 1,2,\ldots,p. \tag{4}$$

These new bounds also satisfy

$$l_1^{t+1} = 1; h_p^{t+1} = n : l_{i+1}^{t+1} = h_i^{t+1} + 1, i = 1,2,\ldots,p-1.$$

In order to find expressions for these new bounds two new functions are introduced [13]. Firstly, the function f^t gives the partial sums of the piecewise constant workloads

$$f^t(i) = \sum_{k=1}^{i} \hat{w}_k^t,\ i = 1,2,\ldots,n, \tag{5}$$

($f^t(i)$ is a piecewise linear function that approximates the cumulative workload), and, secondly, the corresponding f^t–inferior part function is given by [13]

$$f_{[]}^t(x) = i \Leftrightarrow f^t(i) \le x < f^t(i+1). \tag{6}$$

Using these functions the upper bounds $\{h_j^{t+1},\ j = 1,2,...,p\}$ are given by

$$h_0^{t+1} = 0,\ h_j^{t+1} = f_{[]}^t\left(f^t(h_{j-1}^{t+1}) + \overline{W}\right),\ j = 1,2,...,p, \tag{7}$$

where $\overline{W} = \frac{1}{p}\sum_{i=1}^{n} \hat{w}_i$ is the target (balanced) workload for each of the p processors.

It can be shown (see [13]) that the functions f^t and $f_{[]}^t$ satisfy the following lemmas.

Lemma 1. *If $l_j^t \le i \le h_j^t$ then*

$$f^t(i) = \sum_{q=1}^{j-1} T_q^t + \frac{T_j^t}{h_j^t - l_j^t + 1}(i - l_j^t + 1). \tag{8}$$

Lemma 2. *If $f^t\left(h_{j-1}^t\right) < x \le f^t\left(h_j^t\right)$ then*

$$f_{[]}^t(x) = h_{j-1}^t + \left[\left(x - \sum_{q=1}^{j-1} T_q^t\right)\ \frac{h_j^t - l_j^t + 1}{T_j^t}\right], \tag{9}$$

where $f_{[\ ]}$ represents the inferior part function.

2 Convergence of the FGDLS Algorithm

In this section the convergence of the FGDLS algorithm is considered. For the fixed workloads $\{w_i,\ i = 1,2,...,n\}$ we can find the optimal bounds $\{(l_j^*, h_j^*),\ j = 1,2,...,p\}$ so that

$$\sum_{i=l_j^*}^{h_j^*} w_i \approx \frac{1}{p}\sum_{i=1}^{n} w_i, \quad j=1,2,\dots,p, \tag{10}$$

where $l_1^* = 1$; $h_p^* = n$; $l_{i+1}^* = h_i^* + 1, i = 1,2,\dots,p-1$.

It can be shown that the optimal bounds also satisfy the equations:

$$h_0^* = 0, \;\; h_j^* = f_{[]}\left(f(h_{j-1}^*) + \overline{W}\right), \; j = 1,2,...,p, \tag{11}$$

where $f(i) = \sum_{k=1}^{i} w_k, i = 1,2,\dots,n$, represent the partial sums of the workloads (the cumulative workload) and the corresponding inferior part function is given by

$$f_{[]}(x) = i \Leftrightarrow f(i) \le x < f(i+1).$$

The problem of convergence of the FGDLS algorithm can be stated as follows:

Convergence of Discrete FGDLS Algorithm: *Given the fixed, strictly positive, workloads* $\{w_i,\ i = 1,2,...,n\}$ *and the initial upper bounds* $\{h_j^1, j = 0,1,\dots,p\}$*, find conditions such that the upper bound sequences* $\{h_j^t\}, t > 0$ *are convergent and* $\lim_{t\to\infty} h_j^t = h_j^*,\ j = 0,1,\dots,p$.

Since $h_0^t = h_0^* = 0$, and $h_p^t = h_p^* = n, \forall\ t > 0$, we find that the convergence holds trivially for the cases $j = 0$ and $j = p$. Recall that the upper bounds are integers so that the sequence $\{h_j^t,\ t = 1,2,...\}$ is convergent to h_j^* whenever, $\exists t_0 > 0$ such that

$$h_j^t = h_j^*, \forall t \ge t_0. \tag{12}$$

Thus, we have to establish that the upper bounds h_j^t are equal to the optimal bound h_j^* from some index t_0 onwards.

In the following we analyse the convergence of the FGDLS scheduling algorithm for the case when the workloads are monotonically decreasing; we assume that the workloads satisfy the inequalities

$$w_1 \ge w_2 \ge ... \ge w_n. \tag{13}$$

We prove by induction that Equation (12) holds whenever Equation (13) is satisfied. Firstly we show that the sequence of bounds $\{h_1^t\}, t > 0$, is convergent to h_1^*. Inductively, we assume that the sequences $\{h_1^t\}, t > 0$, $\{h_2^t\}, t > 0$, ..., $\{h_j^t\}, t > 0$ are convergent (to $h_1^*, h_2^*, \dots, h_j^*$, respectively) and prove that the sequence $\{h_{j+1}^t\}, t > 0$ is convergent to h_{j+1}^*.

2.1 The Convergence of $\{h_1^t\}, t > 0$.

Recall that the upper bound h_j^{t+1} satisfies the equation $h_j^{t+1} = f_{[]}^t\left(f^t(h_{j-1}^{t+1}) + \overline{W}\right)$. Since, $h_0^t = 0$ and $f^t(0) = 0$ we find that the upper bound h_1^{t+1} satisfies

$$h_1^{t+1} = f_{[]}^t\left(f^t(h_0^{t+1}) + \overline{W}\right) = f_{[]}^t\left(\overline{W}\right) = h_{j-1}^t + \left\lfloor \left(\overline{W} - \sum_{q=1}^{j-1} T_q^t\right) \frac{h_j^t - l_j^t + 1}{T_j^t} \right\rfloor, \tag{14}$$

where j is the index that satisfies $f^t(h_{j-1}^t) < \overline{W} \le f^t(h_j^t)$. Some simple properties of the bounds $\{h_1^t,\ t = 1,2,...\}$ are given in following lemma.

Lemma 3. *The upper bounds* $\{h_1^t,\ t=1,2,\ldots\}$ *satisfy the following inequalities:*

1.
$$f^t(h_{j-1}^t) < \overline{W} \le f^t(h_j^t) \ \Rightarrow\ h_{j-1}^t < h_1^{t+1} \le h_j^t. \tag{15}$$

If the workloads decrease then

2.
$$h_1^{t+1} = h_1^t \ \Rightarrow\ h_1^{t+1} = h_1^t = h_1^*. \tag{16}$$

3.
$$h_1^t \le h_1^* \ \Rightarrow\ h_1^t \le h_1^{t+1} \tag{17}$$

4.
$$h_1^t \ge h_1^* \ \Rightarrow\ h_1^t \ge h_1^{t+1} \tag{18}$$

Proof. From (14) we know that h_1^{t+1} satisfies

$$f^t(h_1^{t+1}) \le \overline{W} < f^t(h_1^{t+1}+1), \tag{19}$$

and from (11) h_1^* satisfies

$$f(h_1^*) \le \overline{W} < f(h_1^*+1).$$

1. When $f^t(h_{j-1}^t) < \overline{W} \le f^t(h_j^t)$ the definition of h_1^{t+1}, together with Equations (19) and (15), directly gives

$$h_{j-1}^t \le h_1^{t+1} \le h_j^t.$$

2. When $h_1^{t+1} = h_1^t$ we have, from Equation (19), that

$$f^t(h_1^{t+1}) \le \overline{W} < f^t(h_1^{t+1}+1) \Rightarrow f^t(h_1^t) \le \overline{W} < f^t(h_1^t+1) \Rightarrow \tag{20}$$

$$f(h_1^t) \le \overline{W} < f(h_1^t) + \hat{w}_{h_1^t+1}^t \Rightarrow f(h_1^t) \le \overline{W} < f(h_1^t) + \frac{\sum_{i=l_2^t}^{h_2^t} w_i}{h_2^t - l_2^t + 1}. \tag{21}$$

Since, the workloads are monotonically decreasing we find that

$$\frac{\sum_{i=l_2^t}^{h_2^t} w_i}{h_2^t - l_2^t + 1} \le \frac{\sum_{i=l_2^t}^{h_2^t} w_{h_1^t+1}}{h_2^t - l_2^t + 1} = w_{h_1^t+1},$$

and thus

$$f(h_1^t) \le \overline{W} < f(h_1^t) + w_{h_1^t+1} = f(h_1^t+1),$$

which implies that

$$h_1^t = h_1^*.$$

3. If $h_1^t \le h_1^*$ it follows that $f^t(h_1^t) = f(h_1^t) \le f(h_1^*) \le \overline{W}$. If $f^t(h_1^t) = \overline{W}$ then $f^t(h_1^t) = \overline{W} < f^t(h_1^t+1)$ and thus $h_1^{t+1} = h_1^t$. When $f^t(h_1^t) < \overline{W}$, let j be the index such that $f^t(h_{j-1}^t) < \overline{W} \le f^t(h_j^t)$, then $h_1^t \le h_{j-1}^t$. Thus we find $h_1^t \le h_{j-1}^t \le h_1^{t+1}$.
4. The case $h_1^t \ge h_1^*$ is similar to case 3. ♠

Equations (17, 18) establish that the upper bounds $\{h_1^t,\ t>0\}$ behave monotonically. For example, when the upper bound h_1^t is less than, or equal to, the upper bound h_1^*, we find that the new upper bound h_1^{t+1} is greater than, or equal to, h_1^t.

Lemma 4. *If the workloads* $\{w_i, i = 1,2,...,n\}$ *are monotonically decreasing then the sequence* $\left\{\frac{h}{\sum_{i=1}^{h} w_i},\ h = 1,2,...,n\right\}$ *is monotonically increasing.*

Proof. The difference between two consecutive terms of the sequence is given by:

$$\frac{h+1}{\sum_{i=1}^{h+1} w_i} - \frac{h}{\sum_{i=1}^{h} w_i} = \frac{(h+1)\sum_{i=1}^{h} w_i - h\sum_{i=1}^{h+1} w_i}{(\sum_{i=1}^{h} w_i)(\sum_{i=1}^{h+1} w_i)} = \quad (22)$$

$$= \frac{(h+1)\sum_{i=1}^{h} w_i - h\sum_{i=1}^{h} w_i - h\ w_{h+1}}{(\sum_{i=1}^{h} w_i)(\sum_{i=1}^{h+1} w_i)} = \frac{\sum_{i=1}^{h} w_i - h\ w_{h+1}}{(\sum_{i=1}^{h} w_i)\ (\sum_{i=1}^{h+1} w_i)}. \quad (23)$$

Since the workloads are monotonically decreasing we find that

$$\sum_{i=1}^{h} w_i - h \cdot w_{h+1} \geq 0$$

and therefore

$$\frac{h+1}{\sum_{i=1}^{h+1} w_i} - \frac{h}{\sum_{i=1}^{h} w_i} \geq 0.$$

Thus the sequence increases. ♠

Theorem 1. *If the workloads* $\{w_i,\ i = 1,2,...,n\}$ *decrease then the upper bounds* $\{h_1^t,\ t > 0\}$ *converge to* h_1^*.

Proof. Two cases are analysed in the following.

– **Case 1.** $h_1^t \leq h_1^*, \forall t > 0$.
  Equation (17) gives that $h_1^t \leq h_1^{t+1}, \forall t > 0$. Thus the sequence of bounds $\{h_1^t,\ t > 0\}$ increases and is bounded above by h_1^*, and therefore it converges.
– **Case 2.** $\exists t_0 > 0$, **such that** $h_1^{t_0} \geq h_1^*$.
  By induction, we prove that $h_1^t \geq h_1^*, \forall t \geq t_0$. Let us suppose that this holds for t so that the upper bound h_1^t satisfy $h_1^t \geq h_1^*$. Since $h_1^t \geq h_1^*$ we find $f^t(h_0^t) = 0 < \overline{W} \leq f^t(h_1^t)$, therefore the index j from Equation (23) is 0. Equation (23) can be re-written as:

$$h_1^{t+1} = \left\lceil \overline{W}\ \frac{h_1^t}{\sum_{i=1}^{h_1^t} w_i} \right\rceil. \quad (24)$$

  Since, the sequence $\left\{\frac{h}{\sum_{i=1}^{h} w_i},\ h = 1,2,...,n\right\}$ increases and $h_1^t \geq h_1^*$, we find that

$$\frac{h_1^t}{\sum_{i=1}^{h_1^t} w_i} \geq \frac{h_1^*}{\sum_{i=1}^{h_1^*} w_i} \Rightarrow h_1^{t+1} = \left\lceil \overline{W}\ \frac{h_1^t}{\sum_{i=1}^{h_1^t} w_i} \right\rceil \geq \left\lceil \overline{W}\ \frac{h_1^*}{\sum_{i=1}^{h_1^*} w_i} \right\rceil.$$

  Since $\sum_{i=1}^{h_1^*} w_i \leq \overline{W}$, we find that $\frac{\overline{W}}{\sum_{i=1}^{h_1^*} w_i} \geq 1$ and therefore $h_1^{t+1} \geq h_1^*$. Therefore, $h_1^t \geq h_1^*, \forall t \geq t_0$ holds.
  From Equation (18) we find that $h_1^t \geq h_1^{t+1},\ \forall t \geq t_0$. Hence, the sequence of upper bounds $\{h_1^t,\ t > 0\}$ is monotonically decreasing and is bounded below by h_1^* so that it converges.

In both of the above cases we find that the sequence $\{h_1^t,\ t > 0\}$ converges. Therefore, we find that there exists an index $t_0 > 0$ such that the sequence is constant $h_1^t = h_1^{t+1},\ \forall t > t_0$. Finally, we apply Equation (16) to obtain that $h_1^t = h_1^*,\ \forall t > t_0$. ♠

2.2 The Induction Step

In this subsection we present the induction step which proves that if the sequences $\{h_k^t,\ t > 0\}$ are convergent to h_k^* for $k = 1,2,...,j-1$ then the sequence $\{h_j^t,\ t > 0\}$ is convergent to h_j^*. Given that the sequences $\{h_k^t,\ t > 0\}$ are convergent we know that $\exists t_0 > 0$ such that

$$h_k^t = h_k^*,\ \forall t \geq t_0,\ \ k = 1,2,...,j-1. \tag{25}$$

Thus, for $t \geq t_0$ the upper bound satisfies

$$h_j^{t+1} = f_{[]}^t\left(f^t(h_{j-1}^{t+1}) + \overline{W}\right) = f_{[]}^t\left(f^t(h_{j-1}^t) + \overline{W}\right) = f_{[]}^t\left(f(h_{j-1}^*) + \overline{W}\right). \tag{26}$$

Let $u(j)$ be the index such that

$$f^t(h_{u(j)-1}^t) < f(h_{j-1}^*) + \overline{W} \leq f^t(h_{u(j)}^t).$$

Then the upper bounds $\{h_1^t,\ t = 1,2,...\}$ satisfy

$$h_1^{t+1} = h_{u(j)-1}^t + \left\lfloor \left(f(h_{j-1}^*) + \overline{W} - f(h_{u(j)-1}^t)\right) \frac{h_{u(j)}^t - l_{u(j)}^t + 1}{\sum_{i=l_{u(j)}^t}^{h_{u(j)}^t} w_i} \right\rfloor. \tag{27}$$

Lemma 5. *The upper bounds $\{h_j^t,\ t = 1,2,...\}$ satisfy:*

1.

$$f^t(h_{u(j)-1}^t) < f(h_{j-1}^*) + \overline{W} \leq f^t(h_{u(j)}^t) \Rightarrow h_{u(j)-1}^t < h_j^{t+1} \leq h_{u(j)}^t. \tag{28}$$

If the workloads decrease then

2.

$$h_j^{t+1} = h_j^t \Rightarrow h_j^{t+1} = h_j^t = h_j^*. \tag{29}$$

3.

$$h_j^t \leq h_j^* \Rightarrow h_j^t \leq h_j^{t+1}. \tag{30}$$

4.

$$h_j^t \geq h_j^* \Rightarrow h_j^t \geq h_j^{t+1}. \tag{31}$$

Proof. The proof is similar to the proof of Lemma 3.

Thus, we find the same monotonic behaviour for the upper bounds $\{h_j^t,\ t > 0\}$ as for $\{h_1^t,\ t > 0\}$.

Lemma 6. *If the workloads $\{w_1, w_2, ..., w_n\}$ decrease then the sequence $\left\{\frac{h - h_{j-1}^*}{\sum_{i=h_{j-1}^*+1}^{h} w_i},\ h = h_{j-1}^*+1, ..., n\right\}$ is monotonically increasing.*

Proof. The result follows directly by applying Lemma 4 for the workloads $\{w_j,\ j = h^*_{j-1}+1,...,n\}$.

Theorem 2. *If the workloads* $\{w_i,\ i = 1,2,...,n\}$ *decrease monotonically then the sequence of upper bounds* $\{h^t_j,\ t > 0\}$ *converges to* h^*_j.

Proof. We again analyse two cases.

- **Case 1.** $h^t_j \le h^*_j,\ \forall t > t_0$**.**
  Based on Equation (30) we find $h^t_1 \le h^{t+1}_1,\ \forall t > t_0$, which means that the sequence of bounds $\{h^t_j,\ t > 0\}$ is monotonically increasing and is bounded above by h^*_j, and therefore it converges.
- **Case 2.** $\exists t_1 \ge t_0$ **such that** $h^{t_1}_j \ge h^*_j$.
  By induction we prove that $h^t_1 \ge h^*_j,\ \ \forall t \ge t_1$. Let us suppose that this holds for t so that $h^t_j \ge h^*_j$. Since $h^t_j \ge h^*_j$ we find

$$f^t(h^t_{j-1}) = f(h^*_{j-1}) \le f(h^*_{j-1}) + \overline{W} \le f^t(h^t_j),$$

therefore the index $u(j)$ is j so that two terms in Equation (27) reduce. Equation (27) becomes

$$h^{t+1}_j = h^t_{j-1} + \left\lceil \overline{W}\ \frac{h^t_j - l^t_j + 1}{\sum_{i=l^t_j}^{h^t_j} w_i} \right\rceil. \tag{32}$$

Since, the sequence $\left\{ \frac{h}{\sum_{i=h^*_{j-1}+1}^{h} w_i},\ h = h^*_{j-1}+1,...,n \right\}$ is monotonically increasing and $h^t_j \ge h^*_j$ we find that

$$\frac{h^t_j - l^t_j + 1}{\sum_{i=l^t_j}^{h^t_j} w_i} = \frac{h^t_j - h^t_{j-1}}{\sum_{i=h^t_{j-1}+1}^{h^t_j} w_i} \ge \frac{h^*_j - h^*_{j-1}}{\sum_{i=h^*_{j-1}+1}^{h^*_j} w_i} \Rightarrow$$

$$h^{t+1}_j = h^t_{j-1} + \left\lceil \overline{W}\ \frac{h^t_j - h^t_{j-1}}{\sum_{i=h^t_{j-1}+1}^{h^t_j} w_i} \right\rceil \ge h^*_{j-1} + \left\lceil \overline{W}\ \frac{h^*_j - h^*_{j-1}}{\sum_{i=h^*_{j-1}+1}^{h^*_j} w_i} \right\rceil.$$

Based on $\sum_{i=h^*_{j-1}+1}^{h^*_j} w_i \le \overline{W}$ we have that $\frac{\overline{W}}{\sum_{i=h^*_{j-1}+1}^{h^*_j} w_i} \ge 1$ so that $h^{t+1}_j \ge h^*_{j-1} + \left\lceil h^*_j - h^*_{j-1} \right\rceil = h^*_j$. Therefore, $h^t_j \ge h^*_j, \forall t \ge t_1$.

Based on Equation (18) we find that $h^t_j \ge h^{t+1}_j,\ \ \forall t \ge t_0$. Hence, the sequence of upper bounds $\{h^t_j,\ t > 0\}$ decreases and is bounded below by h^*_j so that it converges.

Equation (29) finally gives that the upper bounds are constant, $h^t_1 = h^*_1,\ \forall t > t_2$. ♠

In conclusion we have proved that

- The sequence $\{h_1^t,\ t > 0\}$ converges to h_1^*.
- If the sequences $\{h_k^t,\ t > 0\}$ converge to h_k^* for all $k < j$ then the sequence $\{h_j^t,\ t > 0\}$ converges to h_j^*.

Therefore the sequences of upper bounds $\{h_j^t,\ t > 0\}$ converge to h_j^* for all $j = 1, 2, ..., p$.

One might reasonably expect also to prove the convergence of the FGDLS algorithm in the case when the workload is monotonically increasing. Unfortunately, it has not been possible to establish convergence in this case and moreover we give a counter example that demonstrates convergence to a periodic solution in this case.

2.3 Numerical Results

In this section some numerical results are presented to illustrate the convergence of the FGDLS algorithm. Firstly, the workloads $\{w_i = 1001 - i,\ i = 1, ..., 1000\}$ are considered. Note that the workload decreases so that the FGDLS algorithm converges. The initial upper bounds are $h^1 = (250,\ 500,\ 750,\ 1000)$ with the corresponding lower bounds are $l^1 = (1, 251,\ 501,\ 751)$. The sequence of upper bounds $\{h_j^t,\ j = 1, 2, 3, 4\}$ for the first 5 iterations is given below.

```
t=1   250  500  750  1000
t=2   142  298  497  1000
t=3   134  292  498  1000
t=4   133  291  497  1000
t=5   133  291  497  1000
```

The corresponding execution times $T_j^t = \sum_{i=l_j^t}^{h_j^t} (1000 - i), j = 1, 2, 3, 4$, are given by

```
t=1 218,625  156,125   93,625   31,125
t=2 131,847  121,602  119,798  126,253
t=3 124,955  124,267  124,527  125,751
t=4 124,089  124,425  124,733  126,253
t=5 124,089  124,425  124,733  126,253
```

and are displayed in Figure 2. In this case convergence is achieved in only 5 steps.

Secondly, we investigate the case when the workloads $\{w_i = i, i = 1, 2, ..., 1000\}$ are monotonically increasing. The initial upper bounds are $h^1 = (250,\ 500,\ 750,\ 1000)$ with corresponding lower bounds $l^1 = (1, 251,\ 501,\ 751)$ The sequence of upper bounds $\{h_j^t,\ j = 1, 2, 3, 4\}$ for the first 6 iterations are

```
t=1   250  500  750  1000
t=2   499  699  856  1000
t=3   499  705  864  1000
t=4   499  706  865  1000
t=5   499  705  864  1000
t=6   499  706  865  1000
```

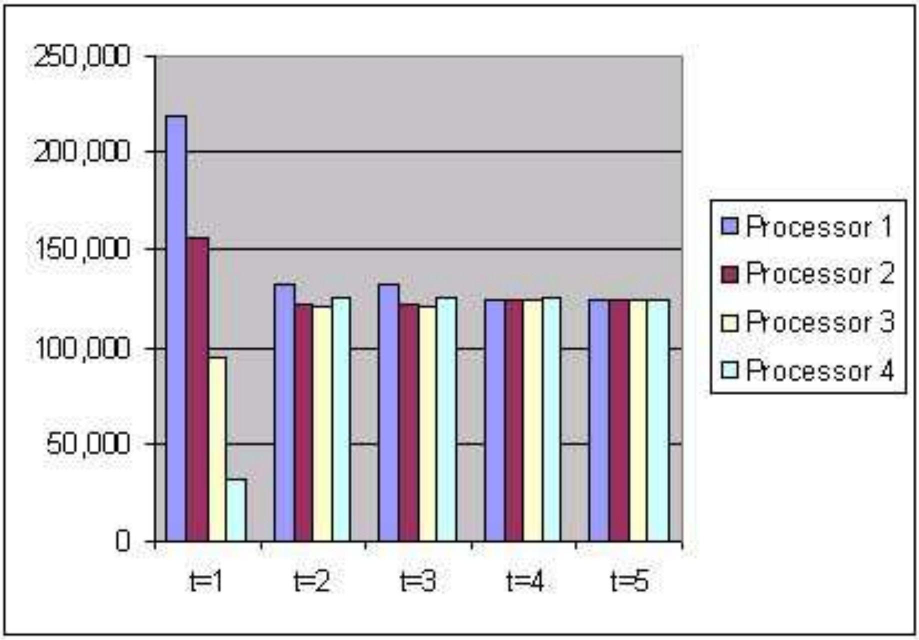

Fig. 2. The Running Times for the Workloads $w_i = 1001 - i,\ i = 1, \ldots, 1000$

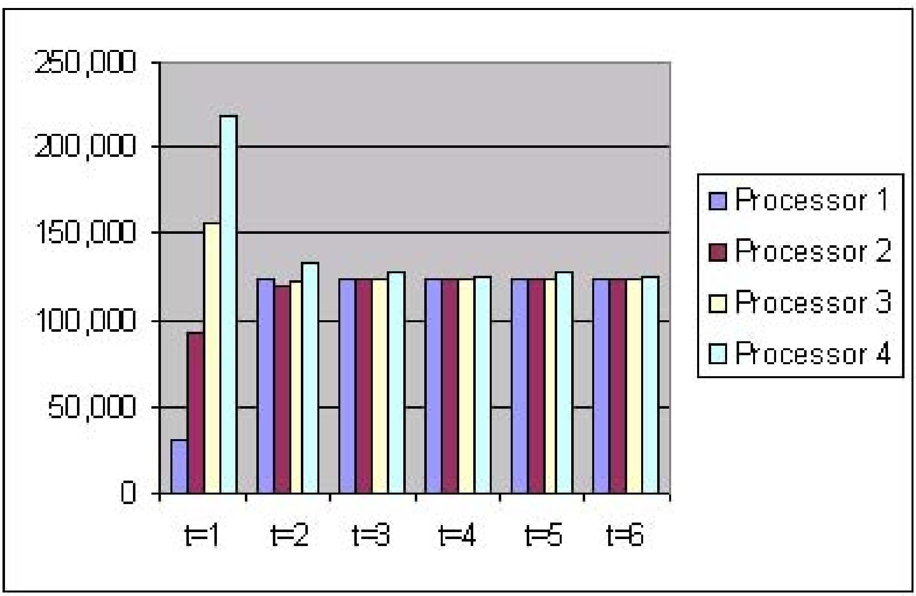

Fig. 3. The Running Times for the Workloads $w_i = i,\ i = 1, \ldots, 1000$

and the corresponding execution times $T_j^t = \sum_{i=l_j^t}^{h_j^t} i, j = 1,2,3,4$, are given by

t=1	31,375	93,875	156,375	218,875
t=2	124,750	119,900	122,146	133,704
t=3	124,750	124,115	124,815	126,820
t=4	124,750	124,821	124,974	125,955
t=5	124,750	124,115	124,815	126,820
t=6	124,750	124,821	124,974	125,955

and are displayed in Figure 3. In this case the convergence is not achieved since the second and third upper bounds h_2^t, h_3^t are periodic. Although the algorithm does not strictly converge, one can see that an acceptable load balance is achieved.

3 Conclusions

This paper has developed a convergence study for the FGDLS algorithm under the realistic assumption that the workloads $\{w_i,\ i = 1,2,...,n\}$ are discrete. The convergence of the algorithm has been established in the case when the workloads are monotonically decreasing. Two numerical examples are presented; one demonstrates convergence in the case of a monotonically decreasing workload, the second illustrates failure to converge in the case of a monotonically increasing workload.

References

1. J.M. Bull, "Feedback Guided Loop Scheduling: Algorithms and Experiments", *Proceedings of Euro-Par'98*, Lecture Notes in Computer Science, Springer-Verlag, 1998.
2. J.M. Bull, R.W. Ford, and A.,Dickinson, "A Feedback Based Load Balance Algorithm for Physics Routines in NWP", *Proceedings of Seventh ECMWF Workshop on the Use of Parallel Processors in Meteorology*, World Scientific, 1996.
3. J.M. Bull, R.W.Ford, T.L.Freeman, and A. Hancock, "A Theoretical Investigation of Feedback Based Load Balance Algorithm", *Proceedings of Ninth SIAM Conference on Parallel Processing for Scientific Computing*, SIAM Press, 1999.
4. D.L. Eager and J. Zahorjan, "Adaptive Guided Self-Scheduling", *Technical Report 92-01-01*, Department of Computer Science and Engineering, University of Washington 1992.
5. S.F. Hummel, E. Schonberg, and L.E. Flynn, "Factoring: A Practical and Robust Method for Scheduling Parallel Loops", *Communications of the ACM*, vol. 35, no. **8**, pp.90-101, 1992.
6. S. Lucco, "A Dynamic Scheduling Method for Irregular Parallel Programs", *Proceedings of ACM SIGPLAN '92 Conference on Programming Language Design and Implementation*, pp.200-211, San Francisco, CA, 1992.
7. E.P. Markatos and T.J.LeBlanc, "Using Processor Affinity in in Loop Scheduling on Shared Memory Multiprocessors", *IEEE Transactions on Parallel and Distributed Systems*, vol. 5, no. **4**, pp. 379–400, 1994.
8. C.D.Polychronopoulos and D.J. Kuck, "Guided Self-Scheduling: A Practical Scheduling Scheme for Parallel Supercomputers", *IEEE Transactions on Computers*, vol. C-36, no. **12**, pp.1425-1439, 1987.
9. S. Subramanian and D.L. Eager, "Affinity Scheduling of Unbalanced Workloads", *Proceedings of Supercomputing'94*, IEEE Comp. Soc. Press, pp.214-226, 1994.

10. T.Tabirca, L.T.Freeman and S. Tabirca, ”A Theoretical Application of Feedback Guided Dynamic Loop Scheduling”, *Proceedings of the NATO Advanced Research Workshop on Advanced Environments, Tools and Applications for Cluster Computing*, Lecture Notes in Computer Science, Springer-Verlag, Vol. **2325**, pp.287-292, 2001.
11. T.Tabirca, L.T.Freeman, S. Tabirca and T.L.Yang, ”An Application of Feedback Guided Dynamic Loop Scheduling to the Shortest Path Problem”, *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'02)*, CSREA Press, Bogart, Georgia, pp.1786-1789, 2001.
12. T.Tabirca, L.T.Freeman, S. Tabirca and T.L.Yang, ”Feedback Guided Dynamic Loop Scheduling; A Theoretical Approach”, *T M Pinkston Proceedings of the 2001 ICPP Workshops*, IEEE Computer Society Press, pp.115-121, 2001.
13. T.Tabirca, S.Tabirca, L.T. Freeman, T. Yang, ”An O(p+logp) Algorithm for the Discrete FGDLS”, Proceedings of The 2003 International Conference on Parallel Processing, ICPP-HPSECA 2003, Taiwan, pp. 164-170, 2003.
14. T.H.Tzen and L.M.Ni, ”Trapezoid Self-Scheduling Scheme for Parallel Computers”, *IEEE Trans. on Parallel and Distributed Systems*, vol. 4, no. **1**,pp. 87–98, 1993.