Investigation of scheduling of circular request arrays by angle and level algorithms in grid systems

Vladimir Kureichik and Andrey Saak
Southern Federal University, 105/42 Bolshaya Sadovaya Str., Rostov-on-Don, 344006, Russia

E-mail: vkur@sfedu.ru

Abstract. The paper studies polynomial angle and level algorithms in terms of scheduling circular request arrays in the GRID systems. The suggested algorithms are analyzed on test arrays obtained from lining a square with stripes of smaller squares. The resulting heuristic measures of the resource wrappers of the level algorithm with the lack are not greater than 0.61, and the amount of error is not greater than 22%. In the GRID systems of centralized architecture, we can recommend the polynomial level algorithm with the lack for scheduling of circular request arrays.

1. Problem statement
The modern development of computer technology is characterized by an increasingly complete implementation of the shift from a product to a service, i.e. from the computer itself to computing as a service [1-3]. Users of large computers knew where, when and on which processors of which characteristics their tasks will be solved. With the exponential growth in the number and power of processors, the bandwidth of communication channels, and the appearance of the GRID systems containing many parallel computing systems, users are abstracting from specific computing resources intended for handling requests.

Thus, the model of multiprocessor computing systems with one important, valuable, critical processor resource was a horizontal strip. A set of rectangles should be tiled into the strip while minimizing the occupied part and time spent for the maintenance of a set of tasks. For the GRID systems, with the equivalence of both processor and time resources, the model is represented by the first coordinate quadrant. The user's request model is represented by a resource rectangle which dimensions correspond to the number of processors and units of time required for handling the request.

The optimal tiling of the rectangles into the first coordinate quadrant is defined by the enclosing rectangle containing the entire set of initial rectangles without overlays. The resource rectangles are oriented, they are placed in parallel to the axes without rotations. The value of the optimality criteria is better when the shape of the enclosing rectangle is closer to a square and has fewer voids inside. The minimal heuristic measure is obtained in terms of tiling into a square with no voids and equals 0.5. The quality of scheduling by heuristic algorithms is assessed with the no-Euclidean heuristic measure that considers the area and shape of the enclosing resource rectangle [4].

In paper [5] the author introduced operations of horizontal and vertical dynamical integration with the lack, which account for the angle algorithm. The family of level algorithms is based on the operation of vertical dynamical integration with the lack [1] and operations of vertical dynamical integration with
the excess and minimal deviation [6]. Both analyzed algorithms require information on a total measure of the request array.

According to the classification of users’ requests introduced in [7], the requests that require the same number of processors and units of time are represented by squares and referred to as the circular type.

The paper investigates angle and level algorithms for scheduling the circular request arrays using the example of lining a square with the strips of smaller squares [8]. Tiling of the request arrays into the squared resource wrapper without voids is considered optimal.

2. Quality of scheduling of circular request arrays by the angle algorithm

To calculate the heuristic measures of the resource wrappers obtained from scheduling of circular request arrays with the angle and level algorithms, we sort the arrays or resource squares obtained from lining a square with stripes of smaller squares in descending order [9-12].

Let us introduce the following denotations: array I is an array with sides of smaller squares from 1 to 10, array II is an array with sides of smaller squares from 1 to 11, array III is an array with sides of smaller squares from 1 to 12, array IV is an array with sides of smaller squares from 1 to 13, array V is an array with sides of smaller squares from 1 to 14, array VI is an array with sides of smaller squares from 1 to 15.

Tiling of arrays IV-VI by the angle algorithm is demonstrated in figure 1-3.

![Figure 1](image1.png)

**Figure 1.** Tiling of array IV by the angle algorithm.

![Figure 2](image2.png)

**Figure 2.** Tiling of array V by the angle algorithm.
Heuristic measures of the resource wrappers of the angle algorithm and the amount of error $\Delta$ in % relation to the optimal value $\frac{1}{2}$ are demonstrated in table 1.

Table 1. Heuristic measures of resource wrappers of the angle algorithm.

| Array number | Heuristic measure | $\Delta$, % |
|--------------|-------------------|-------------|
| I            | 0.61              | 22          |
| II           | 0.60              | 20          |
| III          | 0.57              | 14          |
| IV           | 0.58              | 16          |
| V            | 0.64              | 28          |
| VI           | 0.66              | 32          |

Let us note that the heuristic measure of the resource wrappers of the angle algorithm is not greater than 0.5+0.16. The amount of error of the angle algorithm is not greater than 32%.

3. Quality of scheduling of circular request arrays by the level algorithm

Tiling of arrays IV-VI by the level algorithm with the lack is demonstrated in figures 4-6.

Figure 3. Tiling of array VI by the level algorithm.

Figure 4. Tiling of array IV by the level algorithm with the lack.
Heuristic measures of the resource wrappers of the level algorithm with the lack and the amount of error $\Delta$ in % relation to the optimal value $\frac{1}{2}$ are demonstrated in table 2.

| Array number | Heuristic measure | $\Delta$ , % |
|--------------|-------------------|--------------|
| I            | 0.56              | 12           |
| II           | 0.54              | 8            |
| III          | 0.53              | 6            |
| IV           | 0.54              | 8            |
| V            | 0.61              | 22           |
| VI           | 0.60              | 20           |

Let us note that the heuristic measure of the resource wrappers of the level algorithm with the lack is not greater than 0.5+0.11. The amount of error of the level algorithm with the lack is not greater than 22%.

Tiling of arrays IV-VI by the level algorithm with the excess is demonstrated in figures 7-9.
Heuristic measures of the resource wrappers of the level algorithm with the excess and the amount of error $\Delta$ in % relation to the optimal value $\frac{1}{2}$ are demonstrated in table 3.
Table 3. Heuristic measures of the resource wrappers of the level algorithm with the excess.

| Array number | Heuristic measure | $\Delta$, % |
|--------------|-------------------|------------|
| I            | 0.61              | 22         |
| II           | 0.59              | 18         |
| III          | 0.56              | 12         |
| IV           | 0.58              | 16         |
| V            | 0.66              | 32         |
| VI           | 0.60              | 20         |

Let us note that the heuristic measure of the resource wrappers of the level algorithm with the excess is not greater than 0.5+0.16. The amount of error of the level algorithm with the excess is not greater than 32%.

Tiling of arrays IV-VI by the level algorithm with the minimal deviation is demonstrated in figures 10-12.

Figure 10. Tiling of array IV by the level algorithm with minimal deviation.

Figure 11. Tiling of array V by the level algorithm with minimal deviation.
Figure 12. Tiling of array VI by the level algorithm with minimal deviation.

Heuristic measures of the resource wrappers of the level algorithm with minimal deviation and the amount of error $\Delta$ in % relation to the optimal value $\frac{1}{2}$ are demonstrated in table 4.

Table 4. Heuristic measures of the resource wrappers of the level algorithm with minimal deviation.

| Array number | Heuristic measure | $\Delta$, % |
|--------------|-------------------|-------------|
| I            | 0.58              | 16          |
| II           | 0.57              | 14          |
| III          | 0.54              | 8           |
| IV           | 0.56              | 12          |
| V            | 0.61              | 22          |
| VI           | 0.61              | 22          |

Let us note that the heuristic measure of the resource wrappers of the level algorithm with the minimal deviation is not greater than $0.5+0.11$. The amount of error of the level algorithm with the minimal deviation is not greater than 22%.

Graphs of the heuristic measures of the resource wrappers of the angle and level algorithms in terms of scheduling of circular request arrays I-VI, obtained from lining a square with the stripes of smaller squares are demonstrated in figure 13.

Figure 13. Heuristic measures of resource wrappers of the angle and level algorithms.
Figure 13 shows that the level algorithm with the lack has a smaller heuristic measure of the resource wrappers. The amount of error is less than 22% and allows us to recommend the level algorithm with the lack for scheduling circular request arrays.

4. Conclusion
The investigation of the angle algorithm and the level algorithms with the lack, excess, and minimal deviation using the arrays obtained from lining a square with the strips of smaller squares has given the following results. The heuristic measure of the resource wrappers of the level algorithm with the lack is not greater than 0.61, and the amount of error is not greater than 22% of the optimal value. We can recommend the polynomial level algorithm with the lack for scheduling the circular request arrays in the GRID systems of centralized architecture.

Acknowledgments
The reported study was funded by the Russian Foundation for Basic Research according to research project No. 20-01-00148.

References
[1] Baker B, Coffman E and Rivest R 1980 *SIAM J. on Comp* 9(4) 846-55
[2] Makarevich O 1980 *Avtom. i vych. tekh.* 5 24-30
[3] Blažewicz J, Weglarz J and Drabowski M 1984 *Inform. Process. Lett.* 18 267-73
[4] Saak A 2014 *Trudy KhII Vserossiyskogo soveschaniya po problemam upravleniya VSPU-2014* Moskva Institut problem upravleniya im. V.A. Trapeznikova RAN 7489-98
[5] Martello S and Monaci M 2015 *Comp. & Oper. Res.* 63 161-71
[6] Ortman F, Ntene N and Vuuren J 2010 *Europ. J. of Oper. Res.* 203(2) 306-15
[7] Jansen K and Zhang G 2007 *Algor* 47 323-42
[8] Srinivasan S, Kettimuthu R, Subramani V and Sadayappan P 2002 Characterization of Backfilling Strategies for Parallel Job Scheduling *ICPP Workshops* 514-22
[9] Jansen K, Solis-Oba R 2009 *Discr. Opt.* 6 310-23
[10] Saak A 2017 *Inform. tec.* 23(6) 469-75
[11] Kettimuthu R, Subramani V, Srinivasan S, Gopalasamy T, Panda D and Sadayappan P 2005 *Int J. of High Perf. Comp. and Net.* 3(2/3) 122-52
[12] Jansen K and Prädel L 2013 *Algor.* 74(1) 919-36