Using solvable classes of systems for scheduling of real-time computations with minimized jitter

A.V. Gruzlikov, N.V. Kolesov and M.V. Tolmacheva

1 Candidate of Technical Sciences, Head of Department, Concern CSRI Elektropribor, Saint Petersburg, Russia
2 Doctor of Technical Sciences, Associate Professor, Chief Researcher, Concern CSRI Elektropribor, Saint Petersburg, Russia
3 Candidate of Technical Sciences, Senior Researcher, Concern CSRI Elektropribor, Saint Petersburg, Russia

E-mail: agruzlikov@yandex.ru

Abstract. An approach to scheduling a computational process in distributed real-time systems with jitter minimization is considered. The approach is based on the concept of a solvable class of systems for which there exist optimal scheduling algorithms of polynomial complexity.

Introduction

Modern integrated navigation systems process information in real time and are characterized by a high level of complexity. They may include more than a dozen processors that solve more than a hundred tasks. In this regard, the problems of organizing computing and scheduling in these systems play an important role. In modern scientific literature scheduling problems issue receive much attention. Surely, the considered problems include traditional applications in production planning, for example, flow shop scheduling. The scheduling of computations in real-time systems (RTS) is associated with this basic direction. It is discussed in this paper.

The optimal solution to the scheduling problem can be obtained by brute force algorithms, but all of them are characterized by exponential computational complexity, and therefore their application in a number of applications is impossible. For this reason, suboptimal algorithms are widely used in practice. In this area, the direction devoted to scheduling in multichannel computing systems, identified with flow shop scheduling, is widely represented. A significant number of works have been published on this topic, differing, first of all, in the features of the proposed algorithms, but also in the type of criterion according to which the schedule is formed. In this case, the most often used criteria are, for example, the minimum makespan or the minimum of the maximum deviation from the deadline, etc. For RTS systems, other criteria characteristic only for them can be used, for example, the minimum energy consumption or the minimum jitter (the deviation of start or completion time of a task from period to period). For many RTS, this parameter is important when some system tasks must be "tied" to specified points in time. The main content of this report is devoted to this topic, unlike [4], where the problem is considered in relation to distributed flow shop systems.

This paper focuses on the approach based on the use of the so-called solvable classes of systems – the SCS algorithm.
1. Problem Statement

Describe the problem statement in flow shop scheduling. A distributed computing system is considered. Assume that the considered set of operations is divided into independent groups of operations with precedence relationships (further referred to as jobs). As a result, $n$ independent equal priority jobs $\tau = \{\tau_j \mid j = 1, n\}$ processing input data arriving with period $T$ are scheduled. Each $j$-th job consists of $m$ operations $\tau_{j,i}$ with execution times $e_{j,i}$, $i = 1, m$. The durations are assumed exactly known. All processors used from the set $P$ have the same performance. The task assignment corresponds to the case of the flow shop system. Task graphs are directed and acyclic, and generally contain more than one path between any selected vertices. Denote the considered system by $C(F, \tau)$. The flow shop scheduling SCS algorithm discussed below is based on the concept of a solvable class of systems. An important consequence of the system belonging to a solvable class is the existence of an optimal linear complexity scheduling algorithm [3].

To determine the solvable classes, let us first introduce the domination relation $«\gg$ on the set of machines.

**Definition 1.** Machine $P_q$ dominates machine $P_r$ ($P_q > P_r$) if $\min e_{j,q} \geq \max e_{j,r}, (j = 1, m)$.

**Definition 2 (class 1).** A set of machines within the critical path forms decreasing sequence $P_1 > P_2 > ... > P_{m*}$ defined by relation of domination.

**Definition 3 (class 2).** A set of machines within the critical path forms increasing sequence $P_1 < P_2 < ... < P_{m*}$ defined by relation of domination.

**Definition 4 (class 3).** A set of machines within the critical path forms a pair of connected sequences $P_1 < P_2 < ... < P_{h*} > ... > P_{m*}$, $1 \leq h* \leq m*$, the first sequence increasing and the second one decreasing defined by the relation of domination ($h*$ is the number of machine where two sequences are joined).

**Definition 5 (class 4).** A set of machines within the critical path forms a pair of connected sequences $P_1 > P_2 > ... > P_{h*} < ... < P_{m*}$, $1 \leq h* \leq m*$, the first sequence decreasing and the second one increasing defined by the relation of domination.

The fourth class is not completely solvable since there is no known optimal scheduling algorithm of linear complexity for it, and the known algorithm is suboptimal.

2. Scheduling algorithms optimal by the criterion of minimum jitter in flow shop systems from solvable classes

This section offers optimal flow shop scheduling algorithms when using the jitter minimum characteristic of RTS as a criterion. We will say that task $j$ is tied to a time point $t_j$ with precision $\delta_j$ if the start time of executing this task lies in the interval $[t_j - \delta_j, t_j + \delta_j]$. The uncertainty of the task execution duration is defined by the value $\Delta_j = \frac{1}{2}(e_j - e_j)$. Let us consider the solution to scheduling problem using the minimum jitter criterion (inaccuracy of task completion timing) for flow shop scheduling.

**Statement 1.** The minimum average output jitter $\bar{\Delta}_{\tau}(\pi)$ for the system from Class 1 is reached in schedule $\pi$, where the jobs are sequenced in non-decreasing order of jitter durations of the first operations of the critical path, i.e., $\Delta(e_{1,1}) \leq \Delta(e_{2,1}) \leq ... \leq \Delta(e_{m,1})$. 


Statement 2. The minimum average output jitter $\bar{\Delta}_2(\pi)$ for the system from Class 2 is reached in schedule $\pi$, for which the following is fulfilled:

1) the jobs are sequenced in non-decreasing order of jitter durations of the last operations of the critical path, i.e.

$$\Delta(e^*_{m^*+1}) \leq \Delta(e^*_{m^*+2}) \leq \ldots \leq \Delta(e^*_{m^*})$$

2) the first job in the schedule $\pi$ meets the condition

$$j^* = \arg \min_j \sum_{i=1}^{m^*-1} \Delta(e^*_{j,i}).$$

Statement 3. The minimum average output jitter $\bar{\Delta}_3(\pi)$ for the system from Class 3 is reached in schedule $\pi$, for which the following is fulfilled:

1) the jobs are sequenced in non-decreasing order of jitter durations of the critical path joining operations, i.e.

$$\Delta(e^*_{1,1}) \leq \Delta(e^*_{2,1}) \leq \ldots \leq \Delta(e^*_{n,1})$$

2) the first job in the schedule $\pi$ meets the condition

$$j^* = \arg \min_j \sum_{i=1}^{n-1} e^*_{j,i}.$$}

Statement 4. The minimum estimated average output jitter $\bar{\Delta}_4(\pi)$ for the system from Class 4 is reached in schedule $\pi$, for which the following is fulfilled:

1) the jobs are sequenced in non-decreasing order of sum jitter durations of the first and last operations in the critical path, i.e.

$$((\Delta(e^*_{1,1}) + \Delta(e^*_{m^*+1})) \leq (\Delta(e^*_{2,1}) + \Delta(e^*_{m^*+2})) \leq \ldots \leq (\Delta(e^*_{1,n}) + \Delta(e^*_{m^*})))$$

2) the first job in the schedule $\pi$ meets the condition

$$j^* = \arg \min_j \sum_{i=1}^{n-1} \Delta(e^*_{j,i}).$$

If in some cases the conditions in statements 2, 3, or 4 contradict each other, the best option can be found by exhaustive search.

Obviously, for a general flow shop system described in the problem statement the conditions for its belonging to some solvable class are most often not fulfilled in practice. As a result, the optimality of algorithms described above is not guaranteed. In this regard, we propose a recursive scheduling algorithm that is approximate, but valid for any system under consideration, and is performed in a number of steps that is not greater than the number of tasks. At each step of the recursion, some analog of the critical path is defined, called a pseudo-critical path. Further the scheduling algorithm is applied (Statement 1–4), relevant to the solvable class closest to the system $\mathcal{C}' = \{P', \tau'\}$, where $\tau'$ is the set of unallocated jobs ($\tau' \subseteq \tau$). In this case, the selected task takes the first position from the range of free positions of the generated schedule. After placement, this task is excluded from the original set, and the next step of recursion is performed, which is implemented for the remaining set of tasks on the set of free schedule positions. As a result, the algorithm consistently allocates all the tasks under consideration in the schedule in the direction from the beginning to the end of the schedule.
3. Conclusion
The paper focuses on scheduling problems in flow shop systems. Minimum jitter is used as a criterion in schedule generation. The proposed SCS algorithm is characterized by its simplicity. Simulation has demonstrated the sufficient efficiency of the proposed algorithm.

This work was supported by the Russian Foundation for Basic Research, project no. 19-08-00052.

References
[1] Liu J.W.S. Real-Time Systems, Prentice Hall, Englewood Cliffs, NJ, 2000. 600p.
[2] Nawaz M, Enscore Jr. EE, Ham I. A Heuristic Algorithm for the m-Machine, n-Job Flow-shop Sequencing Problem // Omega – International Journal of Management Science, 1983. №11: 91-95.
[3] Gruzlikov A.M., Kolesov N.V., Skorodumov Yu.M., Tolmacheva M.V., Using solvable classes in flowshop scheduling // Int J Advanced Manufacturing Technology, 2016.
[4] Kolesov, N.V., Tolmacheva, M.V., and Yukhta, P.V., Sistemy real’nogo vremeni. Planirovanie, analiz, diagnostirovanie (Real Time Systems. Planning, Analysis, Diagnostics), St. Petersburg: Concern CSRI Elektropribor, 2014. (In Russian)