THE JOURNAL OF NAVIGATION (2011), **64**, S83–S90. © The Royal Institute of Navigation
doi:10.1017/S0373463311000464

A Method for Scheduling Receiver Tasks Based on Maximum Allowed Execution Time Dichotomy Search

Wei Wu, Shao-jie Ni and Fei-xue Wang

(*Satellite Navigation R&D Centre, School of Electronic Science and Engineering, National Univ. of Defense Technology, Changsha, China, 410073*)
(Email: wuwei198106@163.com)

In designing and developing navigation receiver embedded software, the correct scheduling of navigation tasks is critical for ensuring that a receiver works in the right way. If a receiver's timing scheduling becomes abnormal, it may lead to timing conflicts of the multi-tasking navigation receiver, which may seriously influence the system's performance of real-time features and functionalities. This paper models the timing scheduling of navigation receivers by extracting all important timing parameters including the execution time, period, deadline and priority etc., and focusing on the execution time. The maximum execution time of a navigation task relies on the design and implementation algorithms and the computing capability of the processor. The execution time parameter can be properly adjusted in receiver software design or even in the phase of the receiver requirement changes. On the other hand, the other parameters (i.e., the period, deadline and priority) mainly depend on the receiver's technical requirements, and usually these parameters are set to constant values. In this research, a novel off-line method based on the maximum allowed execution time dichotomy optimization is proposed, and the equilibrium rule is proposed to apply to the multi-task maximum allowed execution time dichotomy search and an analysis of time series of those parameters obtained from real observations of receiver tasks was also conducted. Results indicate that the relative equilibrium is more suitable to the multi-task scheduling of receiver navigation, compared to the absolute equilibrium. The proposed method provides a fast off-line scheduling approach for the top level task design of navigation receivers, and remarkably improves the development efficiency of receiver embedded software.

KEY WORDS

1. Navigation Receiver. 2. Real-time Scheduling. 3. Execution Time. 4. Dichotomy Search. 5. Equilibrium Rule.

1. INTRODUCTION. In the navigation and communication types of receivers, tasks such as carrier navigation, signal measurement and data communication need to be processed in real time. Taking a navigation receiver as an example, it is required to receive, process and decode satellite-based navigation signals so that it provides services like round-the-clock real-time navigation and distance/velocity telemetering for a variety of users. The navigation receiver plays a very important role

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press

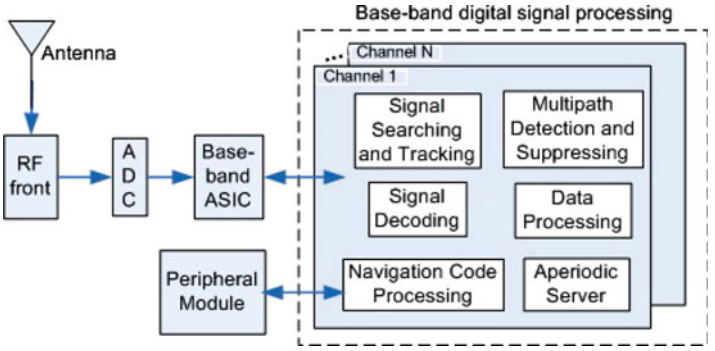


Figure 1. Typical architectural block of navigation receiver.

in military/civil navigation and communication domains, such as in high dynamic applications (vehicle-based, ship-based, aero-based, missile-based, etc.), weak signal detection, anti-jamming etc (Kai, 2006; Elliott, 2007). In a navigation receiver, a satellite signal is received by its antenna first, then passes through its RF (Radio Frequency) module, after the IF (Intermediate Frequency) sampling of ADC (Analog-Digital Converter) and frequency down-conversion, the baseband signal is obtained. In the baseband processing, some important tasks such as multi-frequency multi-channel satellite signal (P and C code) searching and tracking, signal decoding, navigation code processing, multipath detection and suppressing (Jovancevic, 2001; Jovancevic, 2003) need to be performed. The typical architecture of a navigation receiver is shown in Figure 1.

Nowadays, with the improvement of the Integrated Circuit and Digital Signal Processing technologies, microprocessor chips of receivers are being developed to feature smaller size, higher speed and lower power consumption. The receivers also begin to be able to combine multipath suppressing, digital beam control, self-adaptive algorithm in both space-time and space-frequency domains, and real-time high dynamic telemetering and positioning algorithms (Gold, 2003; SPIRIT Company, 2008). However, the increase of the new task modules makes the processing more time consuming and needs more complicated calculations. Thus, sometimes it is not easy to meet the system's real-time processing requirement. Consequently, the task scheduling of a receiver becomes one of the main bottlenecks for quick response to changing task situations.

The period and deadline of the receiver task are determined by the system requirements and so they can be treated as fixed/constant parameters. Using priority assignment algorithms such as LNPA and APG (Xing, 2007), an optimal priority assignment scheme can be obtained. In addition, using the baseband digital signal processing optimizing method (Wu, 2005), the task's execution time can be decreased. According to these characteristics of the receiver tasks, in this paper, a novel dichotomy optimization method based on the equilibrium rule is proposed for handling the situation of task changing. This method can search for the maximum allowed execution time of receiver tasks, and complete the receiver scheduling scheme off-line.

In Section 2, receiver tasks are analysed, a receiver-scheduling model is established, and schedulable condition for the receiver tasks is given. In Section 3.1, the maximum allowed execution time dichotomy optimization method is presented and analysed.

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press

Furthermore, the maximum allowed execution time optimization for multi-task situations using the equilibrium rule is illustrated in Section 3.2. In Section 4, simulations and experiment results are presented for the validation of the proposed method.

2. SCHEDULING MODEL OF RECEIVER TASKS.

In this section, commonly used navigation tasks are analysed and the commonly used scheduling model for the tasks is explained.

2.1. *Analysis of Navigation Tasks.*

The receipt and processing of navigation signals are achieved by the following tasks (NB: those tasks that are triggered by hardware/software interrupts and the execution time required by hardware/software interrupts are counted in these tasks):

- *Signal Searching and Tracking (SST).* After the frequency down-conversion is performed, multi-frequency multi-channel related data are transferred to the receiver's microprocessor at the rate of a few kHz, and the searching and tracking of the satellite signal are carried through. This task is the foundation of all of the other tasks and it consumes a great amount of the processor time, and it also executes more frequently than any other tasks.
- *Signal Decoding (SD).* When a satellite signal is captured, the related data need to be used for decoding and the determination of its sign. This task realizes bit and frame synchronizations and completes the decoding process of the navigation data. Its execution frequency is the second highest among all tasks – second only to the SST task.
- *Data Processing (DP).* Data processing rates of 50 pps DP and 1 pps DP, i.e., performing data processing at refresh rates of 50 Hz and 1 Hz respectively. This signal processing task includes many subtasks such as updating carrier wave phases, estimating carrier noise ratios, assisting navigation, calculating positions, outputting navigation information etc.
- *Navigation Code Processing (NCP).* This task implements the processing and format conversion of navigation code information.
- *Peripheral Module Processing (PMP).* For security and safety of communications, navigation receivers usually need to schedule data frames into the peripheral module so that they can perform tasks such as P code reception, encryption and decryption processing etc.
- *Aperiodic Server (AS).* This task processes aperiodic tasks, such as user query events, aperiodic event processing etc.

Apart from these listed tasks, navigation receivers that have some special auxiliary features may need more modules such as:

- *Multipath Detection and Suppressing (MDS).* This task cumulates related data in a relatively long time, separates the direct signal from multipath signals for estimating and mitigating the pseudo-range error inducted by multipath effects. The higher the precision of the multipath estimation, the more the number of iterations needed by the calculation of the task, and the longer the execution time takes.

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press

2.2. *Scheduling Model*. If the scheduling of N periodic tasks uses static priorities and the task set is $\mathbf{T} = \{\tau_1, \tau_2, \ldots, \tau_N\}$, then each task τ_i in the task set can be modelled using the four elements (Xing, 2007; Wu, 2009):

$$\tau_i = \{p_i, C_i, T_i, D_i\} \quad i = 1, 2, \ldots, N \tag{1}$$

where p_i denotes the priority of τ_i; C_i denotes the worst-case execution time of τ_i; T_i denotes the period of τ_i and D_i denotes the relative deadline of τ_i with $D_i \leqslant T_i$.

The processor utilization of task τ_i is defined as:

$$U_i = C_i/T_i \quad i = 1, 2, \ldots, N \tag{2}$$

The task with higher priority can pre-empt the task with lower priority. Those tasks with the same priority follow the FIFO (First In First Out) queue, i.e., the earlier the task comes to the FIFO queue, the earlier the task will be executed. In order to schedule receiver tasks in the right way, task parameters p_i, C_i, T_i and D_i need to be rationally designed. If all of the tasks can be completed before the deadline, then the task set is schedulable.

2.3. *Schedulability Condition*. In the task scheduling with static priorities, the classical 'level-i busy period' method gives a schedulability condition for the arbitrary deadline circumstance (Tindell, 1994). Reference (Wu, 2009) proposes a schedulability condition suitable for the FIFO queue. In the scheduling scheme of receiver tasks, usually, all tasks are required to complete in time, i.e., the relative deadline is not longer than its period ($D_i \leqslant T_i$). In this case, the schedulability condition of task τ_i is (Katcher, 1995):

$$\min_{0<t\leqslant D_i} W_i(t)/t \leqslant 1 \tag{3}$$

where:

$$W_i(t) = \sum_{\tau_k \in hp(\tau_i)} \lceil t/T_k \rceil C_k + \sum_{\tau_l \in ip(\tau_i)} C_l \tag{4}$$

where $hp(\tau_i)$ denotes the set of tasks that have priorities higher than that of τ_i and $ip(\tau_i)$ denotes the set of tasks that have the priority the same as that of τ_i.

3. MAXIMUM ALLOWED EXECUTION TIME DICHOTOMY SEARCH.

As mentioned in Section 1, the period and deadline of receiver tasks are determined by the requirements, so T_i and D_i can be treated as fixed/constant parameters. p_i can be obtained with an optimal priority assignment algorithm (Tindell, 1994). Therefore, the focus of this section is on the optimal design of execution time C_i.

When task execution time C_i is longer than a given value, the task set will become unschedulable, according to the constraint for the schedulability condition in Section 2.3. The maximum valid value of C_i is defined as the maximum allowed execution time C_{imax}. If and only if $C_i \leqslant C_{imax}$, the task set is schedulable. According to formula (4), the response time $W_i(t)$ is a monotonic increasing function of parameter C_i. The dichotomy search method is a commonly used quick search method for single-parameter monotonic function optimization. Hence in this research the dichotomy search method is used to find the optimal C_{imax}.

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press

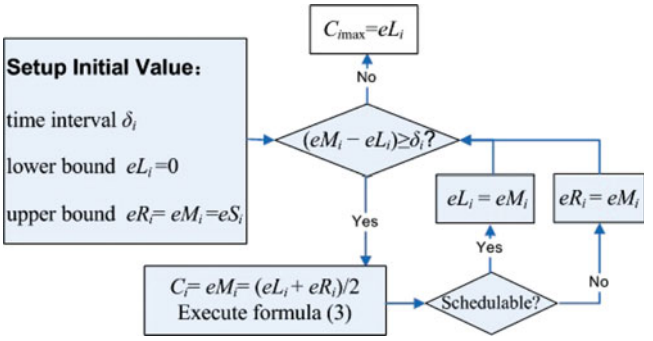


Figure 2. Dichotomy search flowchart for single-task C_{imax}.

3.1. *Dichotomy Search for Single-task* C_{imax}. Searching for the maximum allowed execution time C_{imax} of task τ_i is the key to a scheduling scheme. If $C_i > C_{imax}$, τ_i is unschedulable, otherwise it is schedulable. The minimum value of C_{imax} is 0, and the maximum value of C_{imax} corresponds to the circumstance where the total utilization of the processor is 100%. Taking $eS_i = \left(1 - \sum_{j=1,2,\ldots,N,\, j\neq i} U_j\right) T_i$, then the searching scope of C_{imax} is in $[0, eS_i]$.

For the single task τ_i, both parameters of all other tasks and parameters of p_i, T_i, D_i of task τ_i are fixed, then the flowchart of the dichotomy search for the maximum allowed execution time C_{imax} is shown in Figure 2. The performance of this algorithm mainly depends on the total number of executions of the schedulability condition. $\forall i = 1,2,\ldots,N$, this algorithm executes at most $\log_2(eS_i/\delta_i)$ times of the schedulability condition.

3.2. *Dichotomy Search for Multi-task* C_{imax} *Using Equilibrium Rule*. When the single-task search method cannot meet the application requirements, or, the cost of the single-task optimization is great, a simultaneous optimization of multi-task should be considered. In general, considering the simultaneous optimization of n tasks, the total number of searching is C_N^n. $\forall i = 1,2,\ldots,C_N^n$, where $\boldsymbol{C_{imax}}$ is the vector of the maximum allowed execution time of the n tasks. Its minimum value is the zero vector, i.e., $\mathbf{0}$, and its maximum value corresponds to the circumstance where the total utilisation of the processor is 100%. Let $eS_i = Weight_i T_i \left(1 - \sum_{j=1,2,\ldots,N,\, \tau_j \notin g_i} U_j\right)$, where g_i denotes the task group of the n tasks, $\boldsymbol{Weight_i}$ denotes the normalized equilibrium weight vector.

Based on the relative equilibrium rule, the task's execution time decreases proportionally with the utilization of the processor:

$$\boldsymbol{Weight_i} = \boldsymbol{U_i} / \sum_{\tau_j \in g_i} U_j \tag{5}$$

Based on the absolute equilibrium rule, the task execution time decreases proportionally with execution time:

$$\boldsymbol{Weight_i} = \boldsymbol{C_i} / \sum_{\tau_j \in g_i} C_j \tag{6}$$

The searching scope of $\boldsymbol{C_{imax}}$ is $[\mathbf{0}, \boldsymbol{eS_i}]$, and its dichotomy searching flowchart is shown in Figure 3. $\forall i = 1,2,\ldots,C_N^n$, This algorithm executes at most $\log_2(\max(\boldsymbol{eS_i})/\delta_i)$

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press

Table 1. Task set parameters of a type of navigation receiver.

i	Task Name	p_i	C_i (μs)	$T_i = D_i$ (ms)
1	SST	3	206	1/3
2	SD	2	228	4
3	50 pps DP	2	1408	20
4	AS	2	14	20
5	1 pps DP	2	19	1000
6	NCP	1	15	36
7	PMP	1	14	100

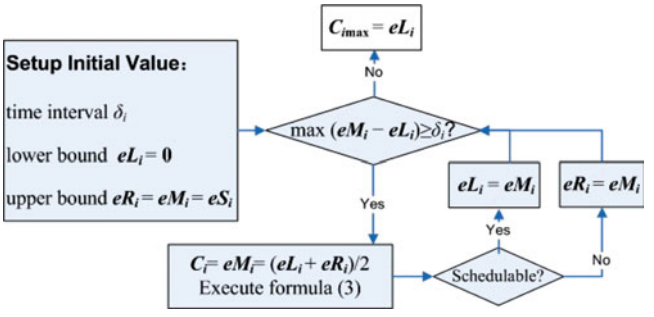


Figure 3. Dichotomy search flowchart for multi-task C_{imax}.

times of the schedulability condition. The calculation amount in the search is predictable.

4. SIMULATIONS AND APPLICATIONS.

The parameters of the task set of a type of navigation receiver are listed in Table 1. According to the schedulability condition in Section 2.3, this task set is schedulable.

4.1. *Dichotomy Search for Single Task.* In addition to those parameters in Table 1, the MDS task $\tau_8 = \{2, 3000\,\mu s, 100\text{ ms}, 100\text{ ms}\}$ was also added to the task set. Using the schedulability condition in Section 2.3, the new task set becomes unschedulable. In this case, the execution time optimization is needed. The dichotomy search method described in Section 3.1 for single-task C_{imax} was used here. The total number of searching is eight, and only two of which are schedulable as shown in Figure 4. According to Figure 4, in one solution, it needs to optimize C_1 from the original values of 206 μs to 128 μs, and in the other solution, it needs to optimize C_8 from the original values 3000 μs to 1151 μs. However, the SST task τ_1 is the most important task, which consumes a great amount of the processor timing. It is difficult to compress a huge amount of the execution time of the SST task. On the other hand, optimizing the MDS task τ_8 decreases the execution time by about 65%, which is a significant amount.

4.2. *Dichotomy Search for Dual-task Using Equilibrium Rule.* Here, simultaneous optimizing of dual-task seems more favourable. Similarly, in addition to those parameters in Table 1, the MDS task $\tau_8 = \{2, 3000\,\mu s, 100\text{ ms}, 100\text{ ms}\}$ is added; and the dichotomy search method described in Section 3.2 for multi-task $\boldsymbol{C_{imax}}$ was used

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press

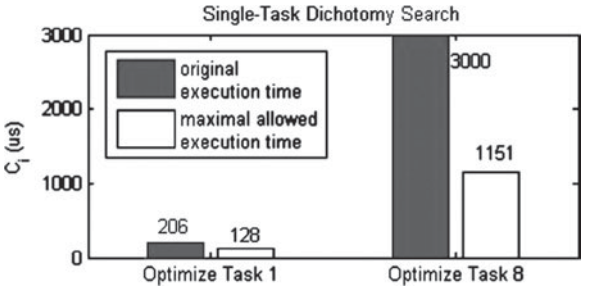


Figure 4. Maximum allowed execution time dichotomy search for single task.

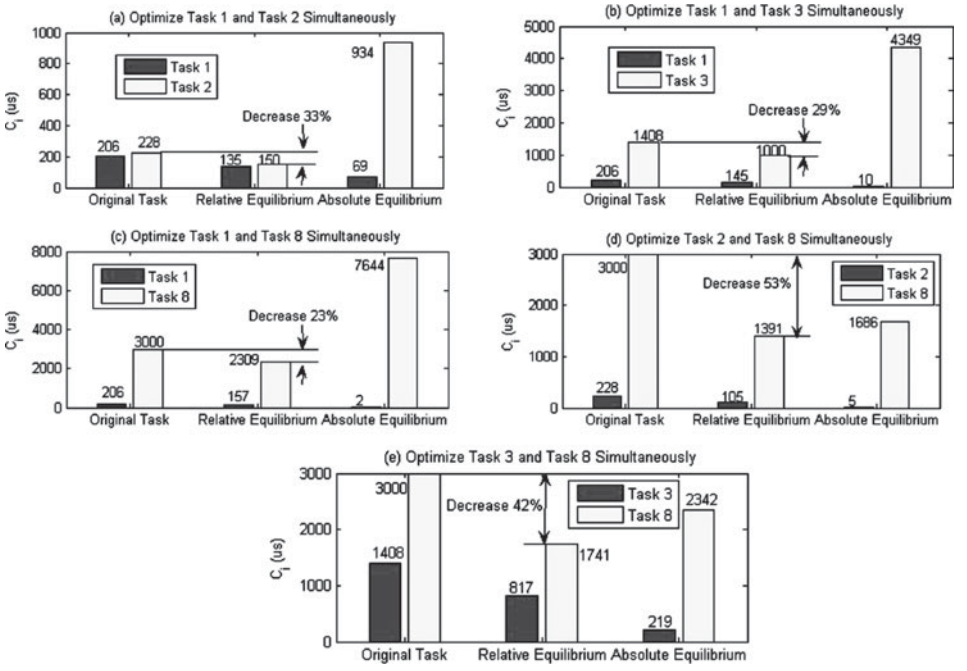


Figure 5. Maximum allowed execution time dichotomy search for dual-task with relative and absolute equilibrium rules.

and based on both relative and absolute equilibrium rules respectively here. For the dual-task search, the total number of searching is $C_8^2 = 28$, and only five of which are schedulable as shown in Figure 5.

According to Figure 5, in the five solutions, the simultaneous optimizing of tasks τ_1 and τ_8 decreases execution times by about 23%, whereas the simultaneous optimizing of tasks τ_2 and τ_8 decreases execution times by about 53%. These results show that simultaneous optimizing of both tasks τ_1 and τ_8 with the relative equilibrium rule can obtain the minimum decrease of the execute time, which provides a valuable reference scheme for receiver task scheduling. On the other hand, the simultaneous optimizing of dual-task with the absolute equilibrium rule needs to compress a great amount of

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press

the execution time of the SST task or the SD task, which is difficult to implement for navigation receivers.

5. CONCLUSION. Signal processing tasks such as the carrier navigation, signal measurement and data communication need to be performed by navigation and communication receivers in real-time. Changing the requirements of a receiver may lead to the increase of task modules and also task complexities. The maximum allowed execution time of the receiver task is very important in the receiver task scheduling. In this paper, a novel dichotomy optimization method based on the execution time is proposed, and the equilibrium rule is also proposed to assist simultaneous optimizing multiple-task dichotomy searching. Simulation results show that the proposed method is convenient and can quickly calculate the upper bound of the task execution time that meets the timing requirement of the receiver. Designers can adjust the task's execution time according to these results. Besides, the relative equilibrium rule is more favourable than the absolute equilibrium rule in the applications of receiver task scheduling. The proposed method also provides quick scheduling approach for the top level task design of navigation receivers, and remarkably improves the development efficiency of receiver embedded software.

REFERENCES

Elliott, D. K. and Christopher, J. H. (2007). *Understanding GPS Principles and Applications(2nd)*, Publishing House of Electronics Industry.

Gold, K. and Brown, A. (2003). A Software GPS Receiver Application for Embedding in Software Definable Radios, *Proceedings of the ION GPS/GNSS 2003*, 2524–2531.

Kai, B., Dennis, M. A., Nicolaj, N., Peter, R. and Jensen, S. H. (2006). *A Software-Defined GPS and Galileo Receiver: a Single-Frequency Approach*, Birkhauser Boston.

Katcher, D. I., Sathaye, S. S. and Strosnider, J. K. (1995). Fixed Priority Scheduling with Limited Priority Levels, *IEEE Transactions on Computers*, 1140–1144.

Jovancevic, A., Brown, A., Ganguly, S., Kirchner, M., Zigic, S., Scott, L. and Ward, P. (2001). Reconfigurable Dual Frequency Software GPS Receiver and Applications, *Proceedings of the ION GPS 2001*, 2888–2899.

Jovancevic, A., Brown, A., Ganguly, S., Goda, J., Kirchner, M. and Zigic, S. (2003). Real-Time Dual Frequency Software GPS Receiver, *Proceedings of the ION GPS/GNSS 2003*, 2572–2583.

SPIRIT Company (2008). *GPS+GLONASS Baseband Software: DuoStar*, from: www.spiritdsp.com/datasheets.

Tindell, K. W., Burns, A. and Wellings, A. (1994). An Extendible Approach for Analyzing Fixed Priority Hard Real-time Tasks, *The Journal of Real-Time Systems*, **6**(2), 133–152.

Wu, W., Liu, X. H., Li, Z. R. and Wang, F. X. (2005). Implementation of Arc Tangent Function Using Assembly in Fixed-point DSP Based on Differential Evolution Algorithm, *Systems Engineering and Electronics*, **27**(5), 926–928.

Wu, W., Ni, S. J. and Liu, X. H. (2009). Schedulability Analysis of Fixed Priority Systems with Limited Priority Levels, *Computer Engineering and Applications*, **45**(5), 32–35.

Xing, J. X., Wang, Y. J., Liu, J. X., Zeng, H. T. and Nasro, M. A. (2007). A Static Priority Assignment Algorithm with the Least Number of Priority Levels, *Journal of Software*, **18**(7), 1844–1854.

https://doi.org/10.1017/S0373463311000464 Published online by Cambridge University Press