Fast Algorithm for Large-Scale Optimization Problems Accelerating System Collaboration

Author: *Hideya Shibata**

1. Introduction

Advances in Internet of Things (IoT) technology are making it possible to collect and utilize data from various devices in real time. Along with this, the concept of System of Systems (SoS), in which multiple systems collaborate with each other to make decisions, is becoming increasingly important.

This paper describes a fast algorithm for large-scale optimization problems developed by Mitsubishi Electric to support decision-making in SoS. When considering SoS planning as a large-scale optimization problem, the problem can be modeled as suboptimization for each element system and mutual coordination between element systems (Fig. 1). Based on this concept, the method developed in this study performs fast and highly accurate optimization calculations by decomposition of the problem into element systems and iterative mutual coordination(1). As an application example, the paper also describes a case study of the train timetabling problem for multiple routes(2).

2. Scope

The method developed in this study is intended for SoS planning, and utilizes two characteristics of SoS: “loose coupling between element systems” and “coexistence of two types of decision-making.” In addition, the method focuses on the “time-based order relation” related to planning.

This section describes these three characteristics using the example of the train timetabling problem.

2.1 Loose coupling between element systems

The individual element systems that make up the SoS are themselves closed systems, while also having relationships with other element systems. The former implies that the individual element systems have some degree of independence, and the latter implies that this independence is not perfect. In this paper, such a state is expressed as loose coupling between element systems. Moreover, the interrelationships between element systems are realized by the output from one element system becoming the input to another element system.

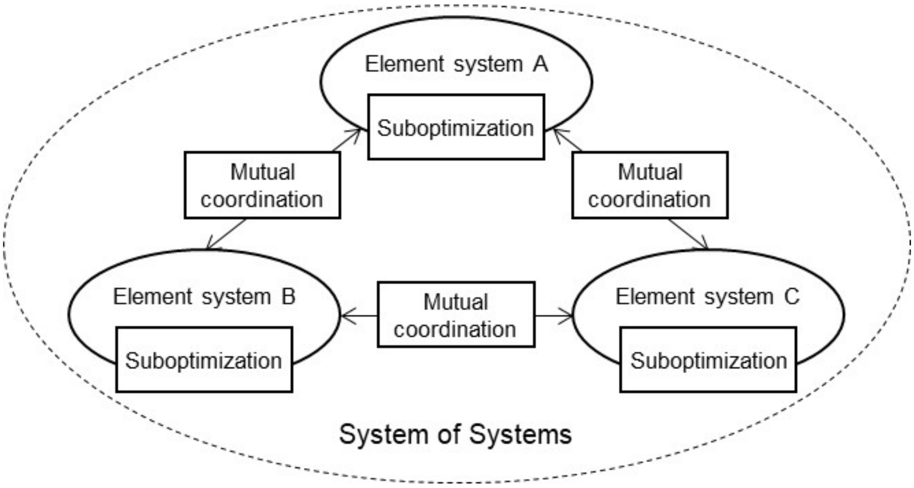

Fig. 1 Model of scheduling for system of systems

**Information Technology R&D Center*

In the example of the train timetabling problem, the railway network can be regarded as an SoS, and each individual route can be regarded as an element system. Individual routes are related to other routes through a limited number of connecting points, or transfer stations. The interrelationships provided by these connecting points are expressed in the form of the arrival time of transfer passengers at transfer stations. In other words, the arrival time (output) of a passenger from one route corresponds to the departure time (input) of the passenger on another route. Viewed in this way, it can be seen that the target railway network has loose coupling between element systems in the train timetabling problem.

2.2 Coexistence of two types of decision-making

The two types of decision-making mentioned here are suboptimization and total optimization. If we assume that the element systems that make up the SoS have some degree of independence, it is natural that the individual element systems are capable of making decisions based on their own closed purposes and means, and have that tendency. This is the aspect of suboptimization. On the other hand, it is not uncommon for both sides to benefit from the relationship between one element system and another element system, and it is also clear that this benefit cannot be obtained simply by aiming for suboptimization. This is the aspect of total optimization. Both aspects are important in SoS decision-making, and ignoring one or the other is not practical.

In the example of the train timetabling problem, determining an individual timetable by using only information about a single route corresponds to suboptimization. If priority is given to ease of train operation and robustness against abnormalities, decision-making by suboptimization is the simplest and most reliable. However, considering the convenience of passengers who have to change trains, the relationship with other routes cannot be ignored. For example, in order to reduce the waiting time of passengers at transfer stations, suboptimization alone is not enough; it is essential to coordinate the train timetable with other routes, that is, from the perspective of total optimization.

2.3 Time-based order relation

Not limited to SoS, problems generally called planning and scheduling have a time-based ordering relationship between the parameters to be determined. The train timetabling problem dealt with in this paper is a type of scheduling and is included in this category.

The method developed this time focuses on the time-based order relation to achieve both high-speed and high-accuracy planning. The method is described in Section 3.

3. Algorithm

This section describes the algorithm of the method developed in this study, using the train timetabling problem as an example.

3.1 Basic policy

The algorithm is designed based on three basic policies. These policies correspond to each of the three characteristics of the applicable scope described in Section 2.

The first is decomposition of the problem into element systems. By utilizing the loose coupling between element systems, the problem is decomposed and the scale of the problem to be handled at one time is reduced. This improves the speed of optimization calculations.

The second is iterative mutual coordination. Utilizing the characteristic that SoS is essentially the coexistence of two types of decision-making—suboptimization and total optimization—suboptimization for each element system and mutual coordination between the element systems are iteratively performed. Here, mutual coordination is a means of total optimization. Mutual coordination suppresses the degradation of approximation accuracy due to problem decomposition.

The third is the sequential determination of parameters. Using the fact that there is an order relation based on the time between the parameters to be determined, the parameters are determined sequentially in order of the earliest time. This suppresses the number of iterations of mutual coordination to a certain number or less, and speeds up the calculation.

3.2 Algorithm

Figure 2 shows the flow of the algorithm; the details of each step are described below according to the figure. In Step 1, the initial state of each element system, that is, the initial solution, is set. Although the method of setting the initial solution depends on the application, the basic policy is to set it so that each element system can be operated to the maximum extent, ignoring costs. The concept of system operation also depends on the application, but, for example, in the train timetabling problem, the departure of a train from the starting station at a certain time can be regarded as the “operation” of the system at that time. In this case, the initial solution corresponds to, for example, a train timetable that runs trains every minute. This is an ideal situation from the user’s point of view, but is not realistic in terms of cost.

In Step 2, mutual coordination between element systems is performed based on the provisional state of each element system. Expressed mathematically, the optimization problem for the entire SoS is solved with the parameters (variables) to be determined, which are closed to each element system, fixed as temporary solutions. This determines the input and output exchanged between individual element systems. The input/output is mutual coordination in practice. Since the problem of total optimization is dealt with here, the apparent scale of the problem is large, but since closed variables are fixed in the element system, the actual scale of the problem is not as large as it seems.

In the example of the train timetabling problem, Step 2 calculates the arrival time of the passenger at the transfer station. That is, with the train timetable of each route fixed, the most efficient route for each passenger to reach the destination, the train to board, etc. are calculated using the total optimization problem. This gives the time of arrival at the transfer station, which is both an output from one route and an input to another route.

In Step 3, the problem is decomposed into element systems based on the results of mutual coordination in Step 2. Since input from other systems is provisionally provided in Step 2, each element system can consider its own problem as an independent suboptimization problem through this provisional input. In the example of the train timetabling problem, the problem is decomposed into an individual timetabling problem for each route based on the arrival time of a passenger who transfers from another route. The decomposed problem does not distinguish between a passenger transferring from another route and a passenger starting travel on that route.

In Step 4, each problem decomposed in Step 3 is calculated and the calculation result is retained as a temporary solution. This step corresponds to suboptimization. In the example of the train timetabling problem, the timetable for each route is determined individually.

In Step 5, among the undetermined parameters, the one with the earliest time is determined based on the provisional solution calculated in Step 4. The parameters determined here will not be changed in later iterations. Once all time-related parameters are determined, the algorithm ends. If an undetermined time exists, the process returns to Step 2. In the example of the train timetabling problem, the timetable for the first time, say 6:00, is determined in the first iteration. In other words, it is determined whether or not a train departs from each station at 6:00. In the subsequent iterations, the timetable for the earlier time is determined successively at 6:01, 6:02, and so on.

Each of these steps can be mapped to the three policies described in Section 3.1 as follows. That is, the problem decomposition into element systems corresponds to Step 3, the iterative mutual coordination corresponds to repetition of a series of processes from Steps 2 to 5, and the sequential determination of parameters corresponds to Step 5.

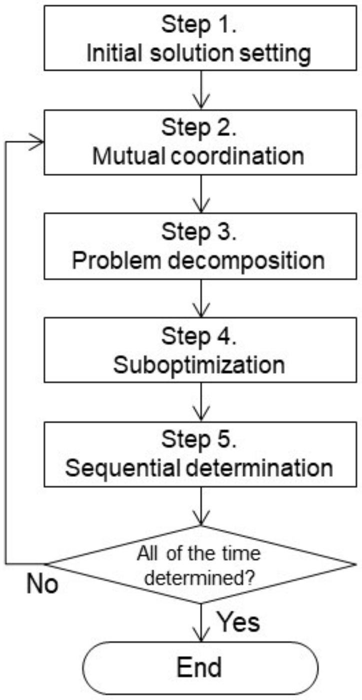

Fig. 2 Flow of algorithm

4. Application Example

This section describes the evaluation results of applying the method developed in this study to the train timetabling problem.

4.1 Evaluation conditions

The evaluation covers a railway network consisting of five routes and 57 stations (Fig. 3). It is assumed that the period to be covered by the train timetable is 1.5 hours (90 minutes) and that approximately 210,000 passengers travel within that period. The quality of the train timetable is determined by the average travel time of passengers.

4.2 Evaluation result

Figure 4 shows the evaluation result. The vertical axis of the figure represents the average travel time of passengers, and the horizontal axis represents the time taken to calculate the train timetable. The solid line corresponds to the method developed in this study, and the dashed line to an existing method for comparison. As the existing method, we set up a local search method that is commonly used in the field of optimization(2). Both the method developed in this study and the existing method are iterative methods; therefore, as the calculation continues, the solution improves and the average travel time decreases. Compared to the existing method, the solution of our method is improved stepwise because the time taken to calculate one iteration is large and the improvement effect is large.

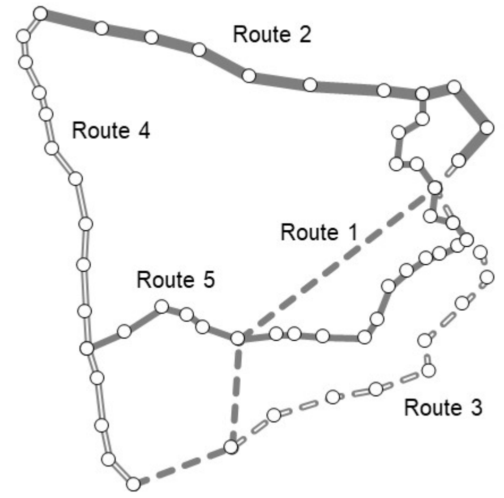

Fig. 3 Target railway network for evaluation

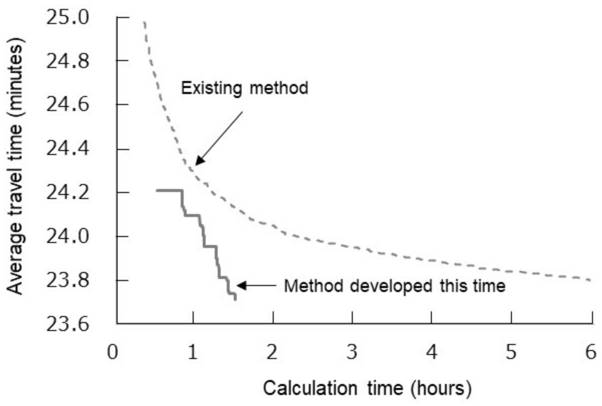

Fig. 4 Evaluation result

Figure 4 shows that the developed method produces a more accurate solution faster than the existing method, deriving a solution with an average travel time of 23.7 minutes and taking 90 minutes for the calculation. On the other hand, the existing method does not reach the same level of accuracy even if the calculation time exceeds 6 hours. The results demonstrate the superiority of our method for the train timetabling problem for a scale of five routes.

5. Conclusion

This paper described the fast algorithm for large-scale optimization problems to support decision-making in the SoS. As an application example, the paper also described the results of evaluating the train timetabling problem consisting of multiple routes, and demonstrated its superiority.

Although not fully described here, we are also trying to apply our method to the unit commitment problem in the electric power sector(3). With the progress of IoT technology, SoS decision-making will become increasingly important. Accordingly, quantitative planning methods based on data will become crucial. In the future, we will expand the scope of application of this method and accelerate system collaboration.

References

(1) H. Shibata, et al.: Optimization Method Based on Subproblem Decomposition for Scheduling Problems Across Multiple Systems, Abstracts of Spring Forum of the Operations Research Society of Japan, 30–31 (2020)

(2) T. Matsunaga, et al.: Decomposition Algorithm for Train Timetabling Problem Based on Travel Demand, Transactions of the Operations Research Society of Japan, 64, 101–125 (2021)

(3) T. Matsunaga, et al.: A Fast Algorithm for Large-scale Unit Commitment Problem based on Decomposition Method, Proceedings of the National Conference of the Institute of Electrical Engineers of Japan, 6-076 (2021)