Heuristic enhancements of a constraint generation procedure for scheduling of avionic systems

Emil Karlsson · Elina Rönnberg ·
Andreas Stenberg · Hannes Uppman

Abstract Modern aircraft host a huge amount of electronics such as sensors that gather information, units where the information is processed, actuators that control the aircraft, and equipment that presents information to the pilot. Electronics in an aircraft is called avionics and due to the real-time requirements of avionic systems, it is not sufficient that the logical result of a computation is correct, it is also crucial that the result is produced at the correct time. This punctuality can be ensured by a scheduling of all activities in the system.

Scheduling of real-time systems can refer both to on-line scheduling where the scheduling decisions are made at runtime and to pre-runtime (off-line) scheduling where the schedule is created at compile time; this work considers pre-runtime scheduling. For further reading about resource allocation in hard real-time avionic systems, see [1].

This work addresses an industrially-relevant avionics scheduling problem that was introduced in [3]. From a scheduling point of view, this problem can be considered as a multiprocessor scheduling problem with multiple time windows and precedence relations between tasks, combined with the scheduling of a communication network. In this context, a processor is referred to as a module and all tasks are beforehand assigned to a module. The communication network is scheduled by assigning communication messages to time slots. To send a communication message involves the execution of certain tasks on the

The work is part of a project funded by the Center for Industrial Information Technology (CENIIT), Project-ID 16.05. The work of Emil Karlsson is supported by the Research School in Interdisciplinary Mathematics at Linköping University.

Emil Karlsson · Elina Rönnberg
Department of Mathematics, Linköping University, SE-581 83 Linköping, Sweden and
Saab AB, SE-581 88 Linköping, Sweden
E-mail: elina.ronnberg@liu.se

Andreas Stenberg · Hannes Uppman
Saab AB, SE-581 88 Linköping, Sweden

involved modules, and for this reason, the task and communication scheduling are closely integrated. Compared to the problem formulation used in [3], we here also include the possibility of co-allocation of messages in a time slot. Co-allocation of communication message was introduced in [8].

Scheduling has a central role when designing avionic systems. During a development project, that can last for several years, new software functionality is added iteratively to the system. After each update, the scheduling tool has to provide a new schedule for the system or, if it fails, preferably verify that the instance under consideration is infeasible. In the case when no feasible schedule exists, either changes of the software or upgrades of the avionics platform are required. To upgrade the platform is extremely costly due to the rigid certification processes in the aircraft industry, and it is therefore important to have a scheduling tool that can efficiently utilise the existing platform so that upgrades are made only when necessary.

In the previous works [3], [4], and [8], we have derived an exact approach for solving instances of avionics scheduling problems including up to about 20 000 tasks. The approach is based on a constraint generation procedure where the problem is decomposed into a relaxed problem and a subproblem. This decomposition relies on known structures of the avionics scheduling problem under consideration and the relaxed problem is significantly easier to solve than the original one.

The relaxed problem either detects that the problem lacks a feasible solution, or it finds a solution which defines a restriction of the original solution space. Solving the scheduling problem under this restriction forms our subproblem. A solution to the subproblem is either a feasible schedule or, in case a feasible schedule is not found, the solution to the subproblem indicates which constraints that need to be added permanently to the relaxed problem. This detection of constraints is designed such that at least one new constraint is added in each iteration; hence, after sufficiently many iterations, the relaxed problem becomes the original problem. The practical usefulness of this approach relies on that only a small amount of the constraints need to be generated since both the relaxed problem and the subproblem become computationally challenging for large scale instances.

For the approach presented in [3], the largest instance solved had 15 modules and about 20 000 tasks, of which almost 6 000 were placed on the most occupied module. The best computational time for this instance was just below 40 minutes. These results showed that the above described decomposition was efficient for the problem structure, but the industrial need is to solve even larger instances. To facilitate this, our constraint generation procedure is in this work extended into a matheuristic method (see e.g. [5] and [2]) by combining it with an adaptive large neighbourhood search (ALNS).

The ALNS framework was introduced in [9] and [7], where the large neighbourhood search (LNS) from [10] was extended to adaptively choosing the destroy and repair methods to better guide the search. In the original ALNS approach, fast heuristics are used to repair the destroyed solutions, while a later work, see [6], includes the possibility to repair the solution by solving a mixed integer programming (MIP) model.

In our work, we apply ALNS to find a feasible solution to the relaxed problem. This is done by treating most of the constraints in the relaxed problem as penalised soft constraints and performing the search with the objective to minimize these penalties. The destroy methods are designed to destroy a part of the solution that causes a large penalty, and our different destroy methods reflect that the model contains penalised constraints of different kinds. Since also the repair method needs to handle several kinds of constraints, we have chosen to repair the solution by using a MIP-solver. The MIP-solver is applied to the restriction of the model obtained by fixing all variables except those chosen by the destroy method. As initially suggested in [6], this gives the possibility to benefit from the very efficient heuristics implemented in commercial solvers. Preliminary results show that our application of ALNS to the restricted problem makes it possible to solve significantly larger problems than when the exact approach is used.

Keywords avionic system · discrete optimisation · scheduling · matheuristics · constraint generation

References

1. Al-Sheikh, A.: Resource allocation in hard real-time avionic systems. Scheduling and routing problems. Ph.D. thesis, INSA de Toulouse (2011)
2. Archetti, C., Speranza, M.G.: A survey on matheuristics for routing problems. EURO Journal on Computational Optimization **2**(4), 223–246 (2014)
3. Blikstad, M., Karlsson, E., Lööw, T., Rönnberg, E.: An optimisation approach for pre-runtime scheduling of tasks and communication in an integrated modular avionic system. Optimization and Engineering (2018). Doi: 10.1007/s11081-018-9385-6
4. Karlsson, E., Rönnberg, E.: Explicit modelling of multiple intervals in a constraint generation procedure for multiprocessor scheduling. In: N. Kliewer, J.F. Ehmke, R. Borndörfer (eds.) Operations Research Proceedings 2017, pp. 567–572. Springer International Publishing, Cham (2018)
5. Maniezzo, V., Stützle, T., Voß, S. (eds.): Matheuristics – Hybridizing Metaheuristics and Mathematical Programming. Annals of Information Systems. Springer US (2010)
6. Muller, L.F., Spoorendonk, S., Pisinger, D.: A hybrid adaptive large neighborhood search heuristic for lot-sizing with setup times. European Journal of Operational Research **218**(3), 614–623 (2012)
7. Pisinger, D., Ropke, S.: A general heuristic for vehicle routing problems. Computers & Operations Research **34**(8), 2403–2435 (2007)
8. Rönnberg, E.: Co-allocation of communication messages in an integrated modular avionic system. In: N. Kliewer, J.F. Ehmke, R. Borndörfer (eds.) Operations Research Proceedings 2017, pp. 459–465. Springer International Publishing, Cham (2018)
9. Ropke, S., Pisinger, D.: An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. Transportation Science **40**(4), 455–472 (2006)
10. Shaw, P.: Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. In: Principles and Practice of Constraint Programming - CP98. CP 1998. Lecture Notes in Computers Science, vol. 1520, pp. 417–431 (1998)