# FlexRay Static Segment Scheduling on Two Independent Channels with Gateway

Jan Dvořák
Department of Control Engineering
FEE, Czech Technical University in Prague
Prague, Czech Republic
dvoraj57@fel.cvut.cz

Zdeněk Hanzálek
Department of Control Engineering
FEE, Czech Technical University in Prague
Prague, Czech Republic
and Porsche Engineering Services, s.r.o.
hanzalek@fel.cvut.cz

## Abstract

*The FlexRay bus is a modern standard used in the automotive industry. It offers deterministic message transmission in the static segment following a time-triggered schedule. The scheduling problem for case of use two independent communication channels that can intercommunicate through the gateway node is investigated in the paper. Furthermore, a heuristic algorithm is proposed and evaluated.*

## 1. Introduction

Nowadays, the automotive industry is evolving fast. This causes the increase in the number of electronic control units (ECUs) in cars and also the amount of messages that have to be exchanged among these units. Therefore, it is hard for conventional communication buses, such as CAN bus, to satisfy the requirements. The FlexRay bus has a ten times higher transmission rate compared to the CAN bus and it is also well suited for real-time and criticality-related requirements. The static segment with time division multiple access (TDMA) scheme can handle these requirements while a communication in this segment follows a schedule which must be known in advance. Thus, the FlexRay bus is also often used as a backbone network that interconnect the other buses (e.g. CAN or LIN) that are connected with the FlexRay bus via a gateway node.

### 1.1. Motivation

Although the FlexRay bus provides a significantly higher transmission rate than e.g. CAN bus its bandwidth limits could be reached soon because the number of ECUs is growing fast. The problem is currently often solved by splitting the whole network into separate buses which are interconnected by gateways. But this solution gives rise to synchronization problems and it is also economically inconvenient. One way how to efficiently use the provided bandwidth is to create an efficient schedule for communication. This way is common to all the communication protocols that are based on the TDMA scheme. The second way is specific to the FlexRay standard which offers two channels that can be used independently. This property is usually overlooked by scheduling algorithms.

In this paper, we will combine both methods to save as much bandwidth for the static segment as possible.

### 1.2. FlexRay overview

The FlexRay bus offers two independent channels for communication - channel A and channel B. The ECU can be connected to both channels or just to one of them. At least two ECUs, called synchronization nodes, must be connected to both channels. The communication can operate in two modes from the channels point of view: in the independent mode (when the communication on channel A is independeto of communication on channel B) and in the fault tolerant mode (when communication on channel B follows a communication on channel A).

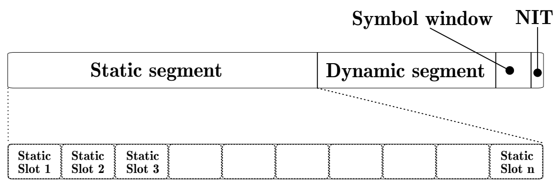


**Figure 1. FlexRay communication cycle**

The communication on the FlexRay bus is based on communication loops called communication cycles. It is possible to differentiate among 64 communication cycles which form a hyperperiod. The cycles in the hyperperiod are periodically repeated. In one communication cycle (presented in Figure 1) we can distinguish four segments:

- Static segment
- Dynamic segment
- Symbol window
- Network idle time

In the static segment, the highly critical signals are exchanged using a time-triggered communication scheme. The whole static segment is divided into static slots with an equivalent duration. Only one ECU is permitted to transmit its data in one slot (i.e. this slot is allocated to the given ECU in all the cycles). The data structure used by the ECU to transmit data into the network is called a frame. Each frame within the static segment can be identified by its slot number and cycle number. One frame can contain more than one signal but the sum of payloads of the signals must not exceed the duration of the slot. The schedule decides which ECU is able to operate in a particular slot and which signals are transmitted. The schedule must be known for all ECUs in advance.

In this paper, we are especially interested in the static segment.

### 1.3. Related works

Several papers were published recently that focus on the FlexRay protocol and particularly the static segment scheduling problem. The FlexRay 3.0.1 is described in detail by the FlexRay™Communication System Protocol Specification [1]. In the automotive industry, this bus is often used together with the AUTOSAR Specification [2].

A milestone in the static segment scheduling area is the article [3] where the transformation of the basic static segment scheduling problem to a specific two-dimensional bin packing problem was introduced. The authors presented an ILP model and also a successful heuristic based on the first fit policy for the bin packing problem. The objective here is to minimize the number of the scheduled slots and to obtain a suitable extensible schedule. Similar problem extended by time-constrains was proposed in [4]. This paper employs the idea of two-stage heuristics, when in the first step, the frame packing is performed and in the second step, the schedule of time-constrained frames are obtained. In [5] we proposed a heuristic for the time-constrained static segment scheduling problem that takes more vehicle variants into account and creates a multi-schedule for all of them at once. The response time analysis for the rate monotonic scheduling of the static segment was proposed in [6]. But this paper requires modifications of the middleware.

The methods described in the presented papers consider the channels to be set in the fault-tolerant mode. Thus, a duplicated communication is used and the potential of the second channel is wasted. To divide the communication to the channels, it is necessary to decide, for each node, if it should be connected to the channel A, B or to both of them. In computer science, a similar problem is clustering. The classical clustering algorithms are expectation-maximization (EM) [7] and the K-Means algorithm [8]. Graph clustering and spectral clustering methods [9] are important for problems that can be described by graphs.

## 2. Problem statement

The problem tackled in this paper is to create the FlexRay static segment schedules for independent channels that can intercommunicate by a special gateway node. Such model is used in practice for signals where fault-tolerance is not critical (a loss of one signal instance cannot cause a jeopardy). Our aim is to find a schedule that minimizes the number of allocated slots and, consequently, reduces the length of the static segment in the communication cycle as much as possible.

The set of nodes $\mathbb{N}$ is given. This set consists of three disjoint subsets $\mathbb{N} = N \cup N^{\mathrm{Sync}} \cup N^{\mathrm{GW}}$. $N$ is the set of all nodes with one port and without special features in the network. These nodes can be connected either to channel A or to channel B but not to both of them. It happens often that the node connected to one channel needs to receive data from the second channel. For these purposes, there is a special gateway node $N^{\mathrm{GW}}$. The gateway node has no data to transmit. It is just mediator for the communication between channels (i.e. it receives data from one channel and sends them to the second one). $N^{\mathrm{Sync}}$ represents the set of synchronization nodes. These nodes are connected to both channels. They are not allowed to transfer other data between channels. The assignment of nodes to the subsets is given.

The data that have to be exchanged in the network are represented by a signal set $S$. Each signal $s_i$ from the set $S$ has the following parameters:

$n_i$ - unique identifier of the ECU, which transmits $s_i$
$p_i$ - the signal period
$l_i$ - signal length/payload in bits
$R_i$ - the set of signal receivers

The $n_i$ identifier of a signal can be any node from $N$ or $N^{\mathrm{Sync}}$ but it cannot be $N^{\mathrm{GW}}$. The period $p_i$ must be a multiple of the cycle duration $m$ and no jitter is allowed. Furthermore, according to the AUTOSAR specification the period must be equal to $m$ multiplied by some power of two (i.e. $p_i = \{m \cdot 2^n \mid n = 1 \ldots 6\}$). The payload of the signal $l_i$ represents the data payload. Signals can be packed to be transmitted in one frame. The signal receivers set $R_i$ contains identifiers of all ECUs that need to receive the signal. Note that if a receiver is in a different channel than the transmitter then the signal must traverse the gateway node $N^{\mathrm{GW}}$. The duplicates of the signals that are transmitted by the gateway are called signal images and we denote them by $s'_i$.

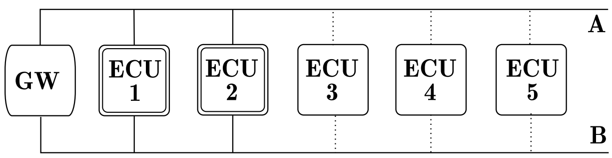


**Figure 2. FlexRay network with two independent channels**

The goal of the scheduling is to find an assignment $s_i \rightarrow \{c_i, l_i, t_i, o_i\}$, where $c_i$ represents the channel to which the signal is transmitted, $l_i$ the identifier of the communication cycle of the first signal occurrence (cycleID) in the schedule, $t_i$ the identifier of the slot (slotID) and $o_i$

is the offset of the signal in the frame and furthermore to find an assignment for signal images $s'_i \rightarrow \{c'_i, l'_i, t'_i, o'_i\}$ for signals that have any receiver scheduled in a different channel than the transmitter. No two signals are tolerated to be overlapped in the schedule for a particular channel. The signal images are, naturally, in the opposite channel. Therefore, we have to know to which channel the node from $N$ is connected to. Consider the unsolved example shown in Figure 2. The synchronization nodes $N^{\text{Sync}}$ are highlighted by double borders. The node labeled as GW is the gateway node $N^{\text{GW}}$. The nodes from $N^{\text{Sync}}$ and $N^{\text{GW}}$ are always connected to both channels. Note that for the nodes from $N$ the assignment to a channel is not known (indicated by a dotted line) and it is also the subject of optimization. Our aim is to find a feasible assignment in such a way that the maximum of the used slots is minimal (i.e. the criterion is $\min \max_i t_i$).

# 3. Algorithm

In this section, we will explain the design of the proposed algorithm. To solve such a problem by exact methods would result in an unacceptable computation time. Thus, a heuristic algorithm is used. The algorithm is divided in two phases. In the first phase, the nodes from $N$ are assigned to the channels and in the second phase the schedules for channels A and B are created.

## 3.1. Node to channel assignment

For each node from $N$ we need to choose one of the channels it will be connected to. Our aim is to find such an assignment which give us the premise to also find good schedules in the second phase. It is assumed, that if there will be a smaller data payload to be exchanged in the channel then even the resulting schedule of the channel will be shorter. According to that we want to find such an assignment that minimizes the number of bits to be exchanged in any channel.

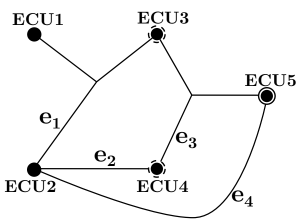


**Figure 3. Hypergraph for Example 1**

The problem can be modeled by the hypergraph. In Fig. 3, the example of the hypergraph with five nodes is depicted. In the hypergraph, each ECU (from $N$ and $N^{\text{Sync}}$) is represented by one vertex. The vertexes are connected by hyperedges. One hyperedge represent one signal. The endpoints are then receivers and the transmitter of the signal. A weight of the hyperedge is the payload of the signal. The task is to mark nodes of the hypergraph according to its assignment to the channels. In Fig. 3, a example of resulting assignment is presented. The ECUs connected to both channels are not marked (node 1 and 2). The nodes in channel A has a black outline (node 5) and the nodes with a dashed outline are connected to channel B.

The criterion value of the given assignment (marking) is evaluated in the following way: If no endpoint of the hyperedge is assigned to channel B, then the signal it represents is only exchanged in channel A and its payload (weight) is added to the payload of A (represented by $P_A$). On the other hand, if none of endpoints is in channel A then the signal is only exchanged in channel B and its payload is added to the payload of B ($P_B$). If the hyperedge has an endpoint from A and also from B then the signal must be exchanged in both channels and also traverse the gateway. Its payload must be added to the payload of both channels ($P_A$ and $P_B$) and also the payload gateway needs to retransmit ($P_G$). Then the criterion value is equal to

$$\max(P_A, P_B) + \alpha \cdot P_G \tag{1}$$

where $\alpha$ is the gateway throughput penalization coefficient. The delays in the gateway can cause problems in the scheduling phase, therefore, assignments leading to the use of the gateway are penalized. It also ensures that from two solutions with the same channel payload, the one with lower gateway throughput will be chosen.

The hypergraph has thousands of hyperedges in real cases. But the hypergraph can be often simplified. If there are two or more hyperedges with the same endpoints it is possible to aggregate them. The new aggregated hyperedge has the weight equal to the sum of the weights (payloads) of the original hyperedges.

The two-partition optimization problem [10] (we are trying to split a set of scalar values to two subsets with as similar sum as possible) can be transformed in polynomial time to the described assignment subproblem. Thus, because the two-partition problem is NP-Hard [10] our subproblem must be also at least NP-Hard.

We are going to solve the subproblem in two ways: by an ILP formulation and by a heuristic approach. The ILP formulation is able to provide optimal solution for easier input instances but it often finish in an unacceptable time for more complex ones. Therefore, the heuristic algorithm was implemented that is able to return near-optimal solution for all the instances in an appropriate time.

## 3.2. Channel scheduling heuristics

In the channel scheduling problem, we have got a set of signals $S$ that should be exchanged among the set of nodes $N \cup N^{\text{Sync}}$. The channel assignment for the nodes from $N$ is known. There is also one special node $N^{\text{GW}}$ that resends signals from one channel to another one if it is necessary.

The FlexRay static segment scheduling methods for single channel were already presented in a number of papers as described in Section 1.3. The basic idea that is used here comes from [3] where the similarity of the

| Set | ILP | CAH | $CAH_{gap}$ | CSH | SCSH |
|---|---|---|---|---|---|
| RealCase | 174843 | 174843 | 0‰ | 121 | 212 |
| synth RealCase | 314748 | 315032 | 0.90‰ | 160.3 | 220.0 |
| $SAE_1$ | 241516 | 241615 | 0.42‰ | 126.0 | 191.4 |
| $SAE_2$ | 259586 | 259822 | 0.91‰ | 134.6 | 191.2 |
| $SAE_3$ | 275871 | 276177 | 1.11‰ | 142.0 | 191.9 |
| $SAE_4$ | - | 300475 | - | 151.9 | 191.3 |
| Average | 272686 | 272916 | 0.83‰ | 142.9 | 197.2 |

**Table 1. Results of individual parts of the algorithm**

static segment scheduling and two-dimensional bin packing problem were shown. In this paper we used a modified algorithm from [5].

## 4. Experimental results

The proposed algorithm was coded in C++ and the GLPK solver for the ILP formulation was used. The timeout for the solver was set to one hour. For the experiments, a few different benchmark sets were used. The first one is the RealCase benchmark instance. This instance reflects the behavior of the algorithm in a real industrial environment. The instance contains more than 5000 signals that are spread to 23 ECUs. This instance was analyzed and its probability model was created. From that probability model, a new synth RealCase benchmark set of 100 benchmark instances was generated. The rest of the benchmark instances are based on the Society of Automotive Engineers (SAE) benchmark instances generated by Netcarbench [11] and extended to include information about the receivers of the signals. Those sets are denoted as $SAE_1$ ... $SAE_4$ and each contains 100 instances.

The SAE benchmark sets differ from each other in distribution of the number of signal receivers. While there are about 75% of messages with only one receiver in $SAE_1$, there are only 20% of messages with only one receiver and more than 50% of signals have 3 and more receivers in $SAE_4$.

In Table 1, the results of the individual parts of the proposed algorithm are presented. The benchmark sets are situated in the rows. The benchmark set caption is in the first column. The second column presents the criterion value of the optimal solution of the channel assignment obtained by the ILP where the $\alpha$ coefficient was set to 0.0001. The hyphen indicates that the solution was not obtained in one hour. The same value for the Channel Assignment Heuristic (CAH) is in the third column. Column $CAH_{gap}$ contains the average optimality gap between the ILP and heuristics solution. It is equal to 0 ‰ for RealCase set because there was just one instance and CAH found the optimal channel assignment for it. An average number of the allocated slots by Channel scheduling heuristic (CSH) is in the fifth column. The last column SCSH contains the number of allocated slots by CSH in the case when all nodes were assigned to the same channel (Single channel scheduling heuristic).

## 5. Conclusion

In this paper, the scheduling problem of independent channels with the gateway was described and the heuristic algorithm was proposed which is the main contribution of the paper. The algorithm was tested on the real case and synthesized instances.

## Acknowledgment

This work was supported by the Grant Agency of the Czech Republic under the Project GACR P103/12/1994.

## References

[1] FlexRay consortium, (Oct 2010), FlexRay Communications System Protocol Specification Version 3.0.1 [Online]. Available: http://www.flexray.com

[2] AUTOSAR Development Partnership, (Oct 2013), AUTOSAR Specification of FlexRay Interface V3.5.0 [Online]. Available: http://www.autosar.org/

[3] M. Lukasiewycz, M. Glaß, J. Teich, and P. Milbredt, "FlexRay Schedule Optimization of the Static Segment", in *Proceedings of the 7th IEEE/ACM International Conference on Hardware/Software Codesign and System Synthesis*, CODES+ISSS '09, 2009, pp. 363–372, New York, NY, USA. ACM.

[4] Z. Hanzalek, D. Benes and D. Waraus, "Time Constrained FlexRay Static Segment Scheduling", in *preprints of the 10th International Workshop on Real-Time Networks (RTN'2011), In conjunction with ECRTS 2011, Porto, Portugal, July 5th*, 2011.

[5] J. Dvorak and Z. Hanzalek, "Multi-variant time constrained FlexRay static segment scheduling", in *Factory Communication Systems (WFCS), 2014 10th IEEE Workshop on*, May 2014, pp. 1–8.

[6] R. Lange, F. Vasques, R. S. de Oliveira, and P. Portugal, "A scheme for slot allocation of the FlexRay Static Segment based on response time analysis", *Computer Communications*, , no. 0, pp. –, 2015.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, vol. 39, no. 1, pp. 1–38, 1977.

[8] S. Lloyd, "Least Squares Quantization in PCM", *IEEE Trans. Inf. Theor.*, vol. 28, no. 2, pp. 129–137, Sept. 2006.

[9] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta, "A survey of kernel and spectral methods for clustering", *Pattern Recognition*, vol. 41, no. 1, pp. 176 – 190, 2008.

[10] J. P. Pedroso and M. Kubo, "Heuristics and exact methods for number partitioning", *European Journal of Operational Research*, vol. 202, no. 1, pp. 73 – 81, 2010.

[11] N. N. Christelle Braun, Lionel Havet, "Netcarbench: A benchmark for techniques and tools used in the design of automotive communication systems.", in *proceedings of the 7th IFAC International Conference on Fieldbuses & Networks in Industrial & Embedded Systems (FeT 2007), Toulouse, France*, Nov 2007.