New Methods for Improving the CAN System in Industrial Applications: Online Optimisation of System Parameters in Running Processes

Tobias Graßmeyer and Wolfgang Popp

In industrial processes it is usual practice to configure a communication system when it is put into operation with parameters pinned down to constant values, e. g. the data rate, during the whole runtime. Such a predefined stiff communication behaviour cannot be optimal, it may even collapse in critical situations of the industrial process for which it is responsible.

In this paper a system is presented allowing the permanent and flexible online-adjustment of important configuration parameters of the CAN-system to the real requirements of an industrial process which may even vary heavily. During the operating time it is possible to change e. g. the identifiers, data rate, inhibit time, the bus-allocation procedure or even the bus structure. The data traffic is optimised by a network management. It comprises a central network manager and network assistants placed decentrally in the controllers of the communication partners in the CAN-system. The network assistants are instructed to carry out the optimisation interventions calculated in the network manager. The building up of the system and first experiences are presented.

1. State of the art

The parameters of communication systems used in industrial processes are determined during the installation, fixed and not adapted during the runtime. The fixed parameter setting can not consider modifications in the operating condition of the industrial process. These modifications could be a fluctuating data traffic, steadily or also abruptly changing operating states of the process, defects of the nodes or even the communication system.
The fixed parameter settings of the communication systems are many times oversized or not optimal because the settings don't have to satisfy only stationary operating states but also possible worst case states.
A not considered operating state can lead to a collapse of the communication system.
Parameter changes require at present often a shutdown of several communication partners or even of the whole communication system.

In following a network management for the troller Area Network) [1],[2] will be presented. This management guarantees an optimal adaptation of the parameters of the communication system to the respective operating state during the running processes.

2. Structure of the online optimisation

The network management fulfils the tasks:

- monitoring of the CAN,
- analysis of the CAN and
- adaptation of the communication parameters without interruption of the data traffic of the process.

The network management is subdivided into a centralised function unit, the network manager, and subordinated, decentralised function units, the network assistants.
Fig. 1 shows an example for the integration of the network management into a CAN system. Different CAN nodes are used for the control of an industrial process. One node has the control of the process as an application master, the other nodes fulfil the tasks of application slaves.

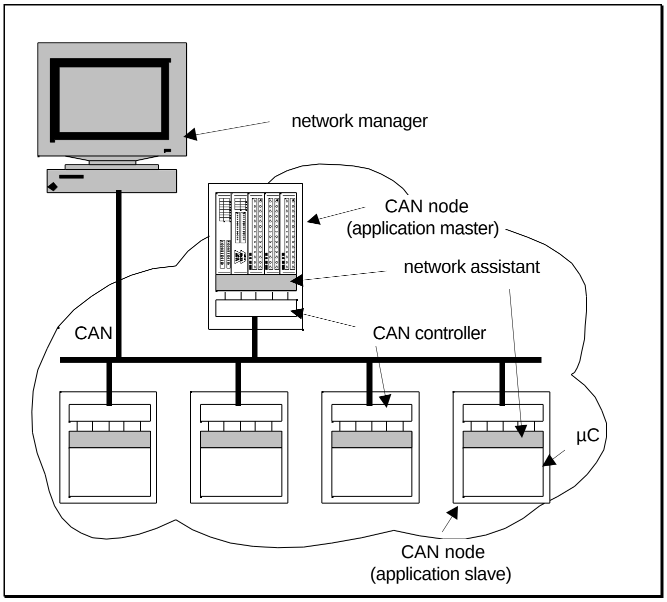

Fig. 1: Example for the network management

The network manager is integrated in the network as an additional communication partner.
The network assistants are implemented in the micro controllers (µC) of each CAN node which contains additionally the application software for the process control. The CAN controllers in the nodes are responsible for communication exclusively.

The network manager offers the following functions:
- initialisation of the CAN,
- monitoring and analysis of the data traffic and
- primarily the parameter adaptation during the runtime.

The changes initialised by the network manager for the parameter adaptation can refer e.g. to the:
- data rate,
- identifier of the message,
- error register and
- node state.

The functions of network assistants are:
- monitoring of the communication behaviour of the node and its internal state,
- transmission of management messages to the network manager and
- reception of management messages from the network manager.

The messages sent by an assistant can refer to the following situations or states:
- connection of a new node,
- life guarding of a node,
- increasing errors in a node or
- announcing a high data volume download.

The network assistants store the configuration of the nodes for all operating states during the installation.

The parameters of CAN adaptable during the runtime are the ***bit timing***, the ***data rate***, the ***identifier*** of the messages, the ***guard time*** and the ***life time*** of the nodes, the ***inhibit time*** of the specified messages, the ***node state***, the ***bus allocation procedure*** or even the ***change of the bus structure***.
Some of these are already named in the CAN specification [2] or in the Draft standard CANopen [3] as essential parameters.
The general, automatic setting of these parameters during the runtime is not described in these documents. In the following such an adaptation will be presented.

3. Online detectable characteristics of CAN-Systems

The network assistants transfer detected changes in the behaviour of the node using corresponding management messages to the network manager.

These messages contain information about:
- long waiting time of a specified message,
- counter values of the error registers,
- imminent transmission of a high data volume,
- connection of new nodes or
- node state.

Long waiting time of a specified message:
The waiting time of a message specified by an identifier cannot be predetermined exactly due to the arbitration procedure. The time depends on the number of messages waiting for sending simultaneously and the identifier has been assigned to a specific message. Only for the message

can be predetermined exactly. The waiting time of any specified message cannot be realised by the receiver of the message.

A method to inform the receiver about the waiting time of a message is the transmission of a time stamp with this message [4]. However, this leads to a more inefficient telegram.

Another method used in the network management discussed here is the supervision of the waiting time in the transmitter of the message: If the time has exceeded a maximum value, the network manager is informed by the network assistant with a correspondent management message.

Counter values of the error registers:

The CAN controllers contain registers which indicate the current values of receiving and transmitting errors. Different rules exist in CAN for fault confinement. The unit creating errors comes earlier into ‘bus off’ state than the others [1],[2].

Based on these rules the network management requires that each network assistant transmits a message whenever its error counter reached 10 or a multiple of 10.

Imminent transmission of a high data volume:

The network assistant will inform the network manager when a high data volume has to be sent.

Definite messages of the nodes and some strategies, like ***putting into operation*** and ***life guarding***, already exist in CANopen.

The network manager evaluates the management messages sent by the nodes and calculates further characteristics like:

- bus load,
- transmission frequency of each specified message and
- distribution of a specified message.

Bus load:

The bus load is an essential indicator for the real time capability of CAN. Here the bus load l is defined as the relation between the duration of the messages to the time interval Δt. The time interval means the duration observed by the network manager. The theoretical bus load produced by the nodes during a time interval may exceed 100 %. A bus monitor, like the network manager, can only receive the messages really transmitted. This means the maximum value of the real bus load is 100 %. An approximate calculation of the bus load is carried out in the network manager according to (1):

$$l = \frac{49 \cdot n_a + 9 \cdot b_a}{\Delta t} \cdot \frac{1}{d} \cdot 100 \qquad (1)$$

n_a is the number of ***all*** messages which were received by the network manager in the time interval Δt and b_a is the sum of ***all*** data bytes contained in these messages. d is the data rate [5].

The exact calculation of the stuff bits used to synchronise the nodes and inserted in the telegrams is approximated by (1) as follows: 2 stuff bits are added to the 47 bits of the standard frame format messages and 1 stuff bit is added to each data byte.

The average number of stuff bits for telegrams with 8 bytes data length is only 3.27 according to a calculation in [6]. The number of stuff bits contained in (1) has been greater chosen than this value in order to have enough safety for the calculation of the bus load.

Transmission frequency of each specified message:

The network manager counts the number of transmissions of each specified message since the start of the communication process.

Distribution of a specified message:

The distribution of a specified message means, whether the specified message is transmitted cyclically or stochastically.

4. Internal structure of the network manager

The network manager contains the following functional units:

- bus analyser for supervising and analysing the CAN,
- database for storing the communication requirements of the CAN nodes, the detected real behaviour of CAN and the chosen parameter adaptation,

- preselector for choosing one optimisation strategy and
- several optimisation strategies which contain different optimisation interventions.

In fig. 2 the structure of the network manager is represented.

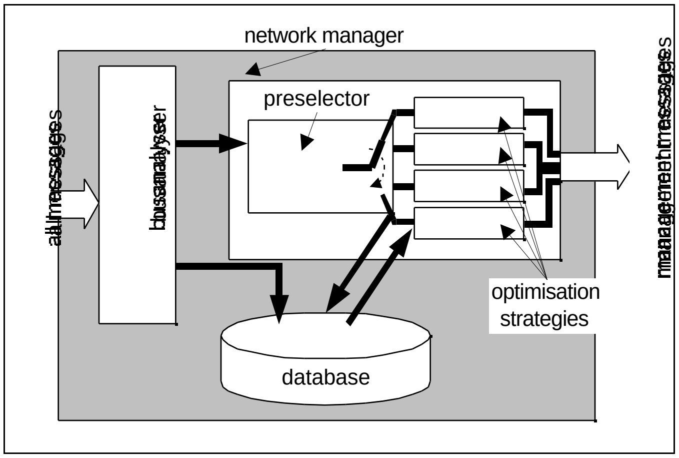

Fig. 2: Internal structure of the network manager

4.1 Methods of the bus analyser and calculations within the bus analyser

The bus analyser detects all management messages transmitted by the nodes and passes these messages to the preselector without waiting for the end of the time interval Δt. Furthermore all messages which the bus analyser receives in the time interval Δt are used in a trend analysis.
Calculated characteristics are:

- bus load (formula 1),
- transmission frequency of identifiers,
- distribution of an identifier,
- weighted data rate (based on formula 2).

The weighted data rate d_w of the system is the result of the addition of the data rates d_{wi} of all messages. The adjusted data rate of CAN may not become smaller than the weighted data rate d_w.
The data rate d_{wi} each single message with the identifier i needs is calculated according to (2):

$$d_{wi} = \frac{n_i(\Delta t) \quad (49 + 9 \quad b_i)}{\Delta t} \quad P_i \qquad (2)$$

$n_i(\Delta t)$ indicates the number of the messages with the identifier i received in the bytes of one message is indicated by b_i and the priority of every message by P_i. P_i is stored in the database. This priority is included in the calculation of the needed d_{wi}. P_i can vary from 1 for low priority messages up to 10 for high priority messages. This means, that the weighted data rate for high priority messages is ten times greater than simple needed data rate.
The data traffic of the last time interval $\Delta t = \Delta t_{i-1}$ is used for a possible adjustment of the data rate for the present time interval Δt_i.

The bus analyser stores all management messages and information from the trend analysis in the database.

4.2 Internal structure of the database

The database stores static characteristics of the CAN, like bus length or communication relations, and dynamic characteristics, like transmission frequency of specified messages, operating states of the nodes and current bus load.

The database containing the data for the online optimisation of CAN uses mainly the following 4 different table types:

- operating states of the industrial process, e.g. putting into operation, flexible production cycle, accident operating state etc.,
- used nodes per operating state,
- specified messages per node and
- communication relations between nodes.

For each operating state of the process the optimal parameters can be dynamically calculated and individually adjusted.

4.3 Choice of an optimisation strategy in the preselector

The preselector has the task of selecting the optimisation strategy by evaluating the messages sent by the network assistants of the nodes and the additional information from the bus analyser and from the database.

The preselector activates one of the fol-

- Reducing the bus load,
- Reaction on a high volume of messages to be sent,
- Increasing the reliability of the network,
- Reducing the waiting time of one specified message,
- Switching over to an other
  - operating state,
  - bus structure and
  - bus allocation procedure.

4.4 Optimisation strategies and interventions

The optimisation ***strategies*** combine different optimisation ***interventions*** which change only one parameter of the CAN.
Four of the above mentioned optimisation ***strategies*** and the corresponding optimisation ***interventions*** shall be discussed more specifically.

4.4.1 Reducing the bus load

All messages are transmitted up to a bus load of 100 %. However, the waiting time increases for messages with lower priority. The low priority messages are ***no longer transmitted*** if the theoretical bus load exceeds 100 %.
Reducing the busload counteracts against the described behaviour. The maximum limit of the bus load is fixed in the described network management to 50 % [6].

This strategy contains the following optimisation interventions:

a) ***Increasing the data rate:***
The maximum value of the data rate depends on the bus length and synchronisation tolerance influenced by the bit timing [7].
The change of the data rate is only possible with a change of the bit timing register (BTR) in the CAN controllers. The register values in the BTR depends on the quartz cycle time and on the choice of the manufacturer of the CAN controller. A certain set of most usual data rates is implemented in the network management of our building up.
The register values in the BTR define the possible resynchronisation tolerance at a given quartz cycle time tolerance and the greatest length of the CAN network. With a jump width of 4 bit-time quanta a high resynchronisation is realised. The sampling at the bit end allows the highest possible data rate for a given length of the CAN network.

The complete change of the data rate needs actually a bus off time of 40 ms for a fault-free switching over. This bus off time guarantees that each node can adjust the new data rate.
The messages arising in the application during this bus off time are stored by the network assistant and transmitted after the switching over procedure.

b) ***Reducing the number of messages to be sent:***
By increasing the cycle time or the inhibit time the bus load produced by these messages is reduced. The time limits are contained in the database.
Similar interventions referring to the bus load restriction are described in [8] for the ***Ethernet*** use in industrial area.

c) ***Disabling data transmission:***
This optimisation intervention goes beyond the previous intervention and does not allow certain message types to be sent or certain nodes to transmit data.
The messages are intermediately stored in the CAN node and transmitted during a low bus load interval.
The database has the information which message types or nodes allow this intervention. The application in the nodes can realise by observing the state of the node
- whether the node itself or
- whether message types and
- which message types

have been disabled.

4.4.2 Reaction on a high volume of messages to be sent

The transmission of greater data packets, like parameter or even program download, always produces a rising bus load. Without limitation of the download speed the measurable bus load can rise to 100%. With a limitation to a low bus load the duration of the download time rises.

The complete bus load for the transmission of great data packets together with

the process data load should not exceed the limit of 50% mentioned before.
The network assistant of a node intending to transmit a high volume of data reports this to the network manager and waits for the permission of the download. During this time the following two optimisation interventions are started in the network manager.

a) ***Increasing the data rate***
A higher data rate increased as far as permitted lowers the bus load produced by the process data to a minimum value.

b) ***Limiting the download***
The bus load produced by the download messages is restricted. The maximum allowed bus load shall not exceed 50 % of the bus load. The bus load produced by download messages equals the allowed bus load reduced by the bus load produced by the process data.
Limiting the download is realised by putting intervals of no transmission into the download stream.
The time interval between two download messages is calculated. The network manager informs the waiting node about this interval value and allows him to execute the download under this condition.

4.4.3 Increasing the reliability of the network

The failure state of the network is evaluated in the network manager depending on the transmitted fault messages from the nodes and also on missing life messages. The network management can eliminate errors which were caused by faulty parameter settings. Errors in the hardware of the nodes cannot be repaired directly. However, their influence can be reduced, e.g. by switching over to a redundant node.

Error scenarios could be:

Slowly growing node failure
At least one error counter has been increased in only one node.

Transmitting error of a node:

The life guarding reports of ***one*** node are missing and the error counters are increasing in all other nodes except of this ***one*** node. This means that this ***one*** node doesn't transmit correctly.

Bad parameter settings
The counter values in the error register have increased in many nodes. The error can be a faulty parameter setting of the CAN.

Interventions to treat above mentioned scenarios could be:

a) Setting new parameters for the faulty node
The failure could be a wrong stored set of parameters in one node. Actually the node is working with these wrong parameters. The new parameter setting can repair this failure.

b) Resetting of the error counters
This prevents the CAN controller of this node from going into the bus off state to early and from being no longer connected to CAN.

c) Early disconnection from the network
With this intervention the node is intentionally turned off into listen only mode, not into the bus off state. This node keeps on being within reach over CAN.

d) Switching over to the accident operating state
The operation of one node is so important that its failure means an heavy accident for the industrial process. The remaining working nodes will bring the process into a stable state and the important data can be exchanged over CAN to avoid critical process states.

e) Shutting down the communication system
This is the strongest intervention in the behaviour of the communication system. No further communication via CAN is possible. For this case appropriate mechanisms must be contained in every node.

Higher bit error rates in the network could

electromagnetic interference. They could be reduced by decreasing the data rate and if needed by the intervention '***Reducing the number of messages to be sent***'.
It is recommended to use in industrial applications only the needed data rate and not the highest possible in order to minimise the bit error rates.

All these optimisation interventions depend on the process.

4.4.4 Reducing the waiting time of one specified message

The network assistant supervises the waiting times of the messages to be sent and reports its exceeding to the network manager. The preselector realises this management message and activates this optimisation strategy, using one of the following optimisation interventions.

a) ***Changing the identifier***

The communication relations between transmitters and receivers are realised via the identifiers.
In the case of the simultaneous access of several messages with various identifiers an automatic sorting of the messages is executed by the arbitration procedure according to the priority of the identifier.
To guarantee that the message with the highest priority gets the identifier with the highest priority the identifiers are assigned to each message according to a specific sorting algorithm in the network manager.
In this algorithm the specified messages are sorted at first by their priority (I.). In order to sort specified messages with the same priority the second sorting criterion 'priority of the transmitter' (II.) is used. In the case of further equal priorities for the specified messages the next sorting criterions from III. to X. are used in the same way.

I. Priority of the specified message
II. Priority of the transmitter
III. Number of the data bytes per message
IV. Minimal inhibit time or cycle time of the specified message
V. Minimal waiting time of the specified message
VI. Priority of the receiver(s) of the specified message
VII. Number of the receivers of the specified message
VIII. Transmission frequency of the specified message during the whole process time
IX. Transmission frequency of the specified message during the last time interval Δt
X. Distribution of the specified messages (cyclic or stochastic behaviour)

Criterions which can be exactly determined only during the runtime are also contained in this algorithm (Point VIII to X).
The data base contains all sorting criterions.

Both transmitter and receiver are equipped with a mechanism for an adaptation of the identifiers during the runtime, which allows effectively simultaneous switching over to the new identifier.
Using the broadcast capability of the CAN all nodes can receive the message to change the identifier simultaneously. This message contains the old and the new identifier.
The function of this mechanism in the network assistants is to compare the identifier in their transmitting and receiving lists with the old identifier. By finding the same identifier the old identifier will be replaced by the new one.

b) ***Increasing the data rate:***

Increasing the data rate reduces the transmission time of the messages and therefore the waiting time.

The order of the mentioned interventions corresponds to the order of their activation in this optimisation strategy.

Only one optimisation strategy which contains several optimisation interventions is activated at a given time. The optimisation interventions in a strategy will not issue conflicting parameter settings.

For all optimisation strategies discussed here opposite interventions exist.

5 Results

In general all communication partners used in the field area are able to fulfil the tasks of the network assistant. They only must contain a µC and an EEPROM to store the code of the network assistant. The code size of the network assistant in our test system is actually below 4 kByte.

The network manager should be implemented in a PC which has enough computing power and memory for the calculations and the database of the network manager.
In our test system a PC with a Pentium II 400 MHz, 128 MB RAM, a hard disc with 6.4 GB is being used.
It should be noted however that the database in the PC will require only up to 10 MB for a full expanded CAN network with 100 nodes and 100 specified messages per node.

The used resources are estimated at 1 % to 20 % of the computing performance depending on the optimisation strategies and the size of the CAN system. In our building up the network manager needs 1 ms for the optimisation strategy 'Reaction on a high volume of messages to be sent' and actually 100 ms for optimisation strategy 'Reducing the waiting time of one specified message' (by 'Changing the identifier'), a time which still will be shortened.

For the management messages 32 identifiers are needed for the network manager and 128 for all network assistants in the nodes.

Principally, it is possible to implement both application master and network manager in the same machine. But for reasons of greater reliability and independence the implementation in different machines is recommended. The communication between the network manager and the application master is realised only by using the CAN. I.e. other communication paths (e.g. internal operating system connections like DDE, OPC or others) do not exist.

The paper presented here has shown the concept of optimally adapting the CAN ess to changing operating conditions. Four important online optimisation strategies have been discussed specifically.

[1] Lawrenz, Wolfhard: CAN Controller Area Network: Grundlagen und Praxis. Hüthig, Heidelberg 1999
[2] Bosch: CAN-Specification 2.0 Part A+B. September 1991
[3] CiA DS-301: CANopen Draft Specification. October 1996
[4] Milke, Burkhard: Integration serieller Feldbussysteme in hochdynamische Regelkreise. Aachen, Shaker 1997
[5] Hanisch, Dirk: Erzeugung und Messung von Verkehrslasten des Kommunikationssystems CAN. diploma thesis, Martin-Luther-Universität Halle-Wittenberg 2000
[6] Rauchhaupt, Lutz: Performance Analysis of CAN Based Systems. 1.st international CAN Conference, Mainz, Sept. 13th-14th 1994, Proceedings p. 1-12 - 1-19
[7] Dietmayer, Klaus: Berechnung des Bit-Timing bei CAN-Bus-Systemen. Elektronik, Franzis, Feldkirchen 1997, part 1: number 21, p. 80 - 84, part 2: number 22, p. 78 - 88
[8] Furrer, Frank: Ethernet-TCP/IP für die Industrieautomation: Grundlagen und Praxis. Hüthig, Heidelberg 1998

Martin-Luther-Universität Halle-Wittenberg
Fachbereich Ingenieurwissenschaften
Lehrstuhl Elektrotechnik
Fon: +049 3461/46-2833
Fax: +049 3461/46-2869
tobias.grassmeyer@iw.uni-halle.de
wolfgang.popp@iw.uni-halle.de
www.iw.uni-halle.de/index.html