Computers and Electronics in Agriculture 109 (2014) 101–108

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Improving ISO 11783 file transfers into mobile farm equipments using on-the-fly data compression

Natalia Iglesias *, Pilar Bulacio, Elizabeth Tapia

CIFASIS-CONICET Institute, Bv. 27 de Febrero 210 Bis, Rosario, Argentina
Facultad de Cs. Exactas e Ingeniera, Av. Pellegrini 250, National University of Rosario, Argentina

ARTICLE INFO

Article history:
Received 28 February 2014
Received in revised form 12 August 2014
Accepted 19 September 2014

Keywords:
ISOBUS
Lossless data compression
Bus utilization
Object pool

ABSTRACT

It is considered the bandwidth bottleneck problem arising in ISO 11783 networks of mobile farm equipments when large file transfers are performed. To overcome this problem, a compression protocol called ISOBUSComp allowing the implementation of dynamic ("on the fly") data compression services for general ISO11783 file transfers is proposed. As a result, transmitting Electronic Control Units are free to choose any data compression technique they wish and receiving Electronic Control Units need not to be aware of such decisions, but just be able to process a suitable Universal Decompression Virtual Machine. Comprehensive simulation studies show that dynamic data compression services built upon the proposed protocol help to reduce bus utilization of ISO 11783 networks between a 28% and a 63%, thus speeding up the time for large file transfers.

1. Introduction

Over the last decades, a great number of technologies have been introduced to agricultural machines (Vanclay et al., 2013); from embedded systems to key networking technologies (Munack and Speckmann, 2001), more and more electronics continue to be added (Pereira et al., 2010; Wang et al., 2007; Cox, 2002). At the mobile farm equipment level, specific local agricultural networks defined by the ISO 11783 set of standards (ISO11783, 2007) are recommended (Backman et al., 2013). Taking into account that standardization and integration of information are dominant attributes in precision agriculture (Suprem et al., 2013; Aubert et al., 2012), the value of a communication standard like the ISO 11783 is increasingly recognized. It is observed, however, that ISO 11783 functionalities may require suitable extensions to deal with the increasing need of transmitting data of exponential growth (Egorova et al., 2007). For example, in mobile farm equipments supplied with different sensors able to collect large amounts of data while working (Peets et al., 2012; Nash et al., 2009; Steinberger et al., 2009), data gathering may cause undesirable bandwidth bottlenecks. Since bandwidth is not a uniformly available resource in agricultural infrastructure (Lee et al., 2012; Herpel et al., 2009), efficient, dependable and high-speed communication systems are crucial to provide almost real-time performance and thus, to avoid system malfunctioning (Mainoo, 2012).

Nowadays, the efficient management of network bandwidth is a top-of-mind in all areas of life (Passos et al., 2013; Kim and Lee, 2014). In the agricultural context, this issue was first envisioned by Stone (1994) who proposed the use of agricultural networks based on buses of higher bandwidth or multiple buses (Stone et al., 1999). It is observed, however, that these hardware solutions are not straightforward applicable to the ISO 11783 standard. In this regard, an alternative option is the development of applications to reduce the amount of data to be transferred, similar to those used for in-vehicle networks (Miucic et al., 2009; Ramteke and Mahmud, 2005), e.g., by the utilization of data compression methods.

Data compression is a well-known technique to improve network performance of low speed networks. In recent work (Iglesias et al., 2014), it was shown how data compression methods can be used to alleviate bandwidth bottleneck problems of ISO 11783 networks at initialization time. It is observed, however, that in such proposal, the provision of static data compression services is required, i.e., receiving Electronic Control Units (ECUs) have to know in advance the specific data compression method used by transmitting ECUs.

In this paper, built upon our previous work, a data compression protocol allowing the development of dynamic data compression services for ISO 11783 networks called ISOBUSComp is introduced.

* Corresponding author at: CIFASIS-CONICET Institute, Bv. 27 de Febrero 210 Bis, Rosario, Argentina.
E-mail address: iglesias@cifasis-conicet.gov.ar (N. Iglesias).

http://dx.doi.org/10.1016/j.compag.2014.09.012
0168-1699/

As a result, data compression methods can be independently selected at transmitting ECUs and receiving ECUs need not to be aware of such decisions. This unique feature provides a great flexibility in the design and operation of data compression services for general ISO 11783 file transfers.

This paper is organized as follows. In Section 2, a proposal for the implementation of dynamic data compression services in ISO 11783 networks is presented. In Section 3 and Section 4, the impact of such proposal in ISO 11783 bus network performance is evaluated using the CANoe.ISO11783 framework (Vector-Informatik, 2011). Finally, in Section 5, conclusions are presented.

2. On-the-fly data compression for ISO 11783 networks

Over the last decade, with the aim of boosting productivity in sustainable way (Blank et al., 2013), data management has become a topic of high interest in the field of agricultural engineering. On the other hand, the ISO 11783 standard has been originally designed just for the periodic transmission of small amounts of control data (Oksanen, 2010; Godoy et al., 2010). Hence, new methods are required to allow the effective and efficient use of ISO 11783 networks for general data sharing purposes. In this context ISOBUSComp is introduced. ISOBUSComp is a compression protocol designed to transparently enhance the current ISO 11783 standard with a common data compression service interface between applications and transport protocols.

Briefly, this proposal is inspired in the design of the Signaling Compression Protocol (SigComp) (Du et al., 2013; Wu et al., 2010; Bin et al., 2006; Jin and Mahendran, 2005; Price et al., 2003). SigComp is a widely used communication protocol originally designed by the Internet Engineering Task Force to alleviate the signaling overhead required for the provision of Voice over IP (VoIP) services in mobile networks (Rosenberg et al., 2002). SigComp essentially embodies a method for compressing text-based communication data used by VoIP signaling protocols. Similarly to SigComp, the core of ISOBUSComp is a Universal Decompression Virtual Machine (UDVM), a virtual machine[1] like the Java Virtual Machine (JVM) (Downing and Meyer, 1997) optimized for running decompression algorithms. Using a UDVM allows ISO 11783 implementers to select any compression algorithms they wish: compressed data at the transmission side will combine with a set of UDVM instructions (bytecodes) allowing original data to be extracted at the destination side.

2.1. ISOBUSComp architecture

ISOBUSComp protocol architecture is built upon two modules, the Compressor and Decompressor ones (Fig. 1). The Compressor Module is composed of a *Compression Dispatcher* and a *Compression Machine*; the Decompressor Module is composed of a *Decompressor Dispatcher* and a *UDVM unit*. If the application decides a compression, the Compressor Machine joint with the Compressor Dispatcher do the application message by compressing and attaching the *bytecodes* of the specific decompression algorithm. At the receiving ECU, the Decompressor Module decompresses the application message by interpreting received *bytecodes* with the resident UDVM.

For the sake of better understanding, it is consider the example where the Application is the Object Transport Protocol dealing with the transmission of the Object Pool (OP) files during ISO 11783 network initialization. The ISOBUSComp protocol starts with the requirement of an OP file transfer by the Object Transport

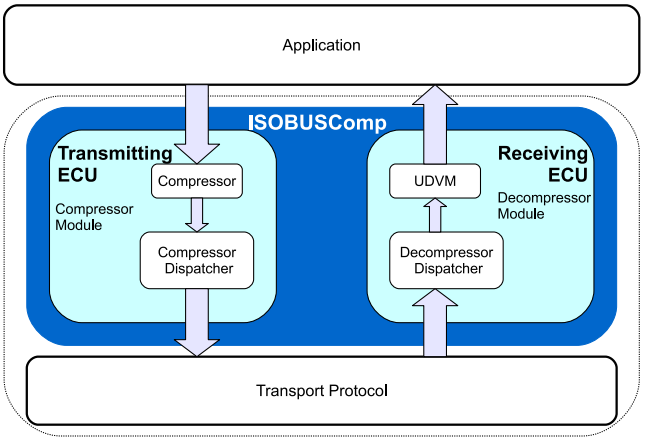

Fig. 1. The ISOBUSComp protocol architecture in ISO 11783 network. A Compressor Module is located at transmitting ECU; a Decompressor Module is located at receiving ECU. Both the Compressor and Decompressor modules provide data compression services to general ISO 11783 applications.

Protocol (see Fig. 2) from a transmitting ECU to a designated receiving ECU, in this case a Virtual Terminal (VT[2]) unit.

The following activities are made by the **Compressor Module**:

Activity 1: The compression process is initiated by the Object Transport Protocol when an OP transfer is required.
Activity 2: The OP file is compressed by the *Compressor Machine* using a compression algorithm.
Activity 3: The *Compressor Dispatcher* receives the compressed OP file from the *Compressor Machine* and builds an *Object Pool Transfer Message* with the following information: length and location of bytecodes, the bytecodes, and the compressed OP file. Note that ISO 11783 does not specify compression techniques at any layer, and thus, a reserved bit for future applications of the ISO 11783 frame (R bit) is used to signal the use of ISOBUSComp data compression services (Iglesias et al., 2014).

The following activities are made by the **Decompressor Module**:

Activity 1: The decompression process is initialized when an *Object Pool Transfer Message* from an ECU is received.
Activity 2: The *Decompressor Dispatcher* checks R bit. If $R = 0$, then OP file is directly sent to the VT. Otherwise, the message is split into *bytecodes* and compressed OP data; both parts are send to the *UDVM unit*.
Activity 3: The *UDVM* unit receives the *bytecodes* and the compressed OP data to make possible the decompression. Finally, the OP file is sent to the VT.

2.2. Implementation details

A proof of concept of the ISOBUSComp protocol was developed using an adaptation of the open source *libsigcomp*[3] library. For convenience of simplicity, the DEFLATE compression algorithm (Deutsch, 1996), already included in *libsigcomp*, was used. Since DEFLATE is closely related to the GZIP compression algorithm (Deutsch, 1996), this choice allowed to build upon (Iglesias et al.,

[1] The LLVM Compiler Infrastructure, http://www.llvm.org/.

[2] ISO 11783-6:2010 describes a terminal unit that can be used by both tractors and implements as universal Virtual Terminal (VT). While AEF (Agricultural Industry Electronics Foundation) call it as ISOBUS Universal Terminal (UT). For practical purposes the VT and UT terms can be used interchangeably

[3] LibSigComp: Open Source SigComp API, https://code.google.com/p/libsigcomp.