International Journal of Innovative
Computing, Information and Control
Volume **6**, Number **8**, August **2010**

 ISSN 1349-4198
pp. 3519–3527

AN EFFICIENT REAL-TIME CONCURRENCY CONTROL PROTOCOL FOR GUARANTEEING TEMPORAL CONSISTENCY

YingYuan Xiao[1], Yueguo Chen[2], Kevin Lü[3] and Yunsheng Liu[4]

[1]Key Laboratory of Computer Vision and System, Ministry of Education
Tianjin University of Technology
Nankai District, Tianjin 300191, P. R. China
xiaoyingyuan_hust@126.com; yyxiao@tjut.edu.cn

[2]School of Computing
National University of Singapore
21 Lower Kent Ridge Road, Singapore 119077
chenyueg@comp.nus.edu.sg

[3]Brunel University
Kingston Lane, Uxbridge UB8 3PH, United Kingdom
kevin.lu@brunel.ac.uk

[4]Huazhong University of Science and Technology
Wuhan 430074, P. R. China
yshliu@hust.edu.cn

Received February 2009; revised June 2009

Abstract. *A real-time system usually requires maintaining a large amount of temporal data objects. These temporal data objects are accessed by application activities (transactions) to generate the ultimate control actions. In many practical applications, real-time systems require not only ensuring transactions finished in the specified time limits (deadlines), but also guaranteeing temporal consistency of data. In this paper, we first present the formal definitions of temporal consistency for data and transaction. Then, we propose a novel real-time concurrency control protocol (TCHP-2PL), which can guarantee temporal consistency. Further, in order to obtain better real-time performance, the STCHP-2PL, an enhanced version of TCHP-2PL, is proposed by introducing the concept of similarity. Performance test shows that the STCHP-2PL can ensure excellent real-time performance while guaranteeing temporal consistency.*
Keywords: Real-time systems, Real-time concurrency control, Temporal consistency

1. **Introduction.** Real-time systems (RTS), which are often employed to monitor and interact with dynamic environment, are widely applied in time-critical applications, such as robot navigation, medical patient monitoring, programmed stock trading, etc. [1-5]. Typically, a real-time system consists of a controlling system and a controlled system. For example, in an automated factory, the controlled system is the robots, assembling stations and the assembled parts, while the controlling system is the computer and human interfaces that manage and coordinate the activities on the controlled system. Thus, the controlled system can be viewed as the environment with which the computer interacts.

The state of a dynamic environment is often modeled and captured by a set of data objects within real-time systems. The controlling system interacts with its environment based on the data objects about the dynamic environment. The data objects are obtained by various sensors, such as temperature and pressure sensors. Usually, data objects in RTS are classified into two categories: persistent data objects and temporal data objects.