International Journal of Innovative Computing, Information and Control
Volume **6**, Number **5**, May **2010**

ICIC International ©2010 ISSN 1349-4198
pp. **2371–2381**

PARALLEL PROCESSING FRAMEWORK BASED ON DISTRIBUTED COMPUTATION OF SPECIALIZATION

Hidemi Ogasawara[1], Kiyoshi Akama[2] and Hiroshi Mabuchi[3]

[1]School of Information Science and Technology
Chukyo University
101 Tokodate, Kaidu-cho, Toyota, Aichi 470-0393, Japan
hidemi@sist.chukyo-u.ac.jp

[2]Information Initiative Center
Hokkaido University
Kita 11, Nishi 5, Kita-ku, Sapporo 060-0811, Japan
akama@iic.hokudai.ac.jp

[3]Faculty of Software and Information Science
Iwate Prefectural University
152-52 Sugo, Takizawa, Iwate 020-0193, Japan
mabu@soft.iwate-pu.ac.jp

Received Feburary 2009; revised July 2009

Abstract. *The purpose of this paper is to propose a framework for generating correct parallel processing programs. We adopt an Equivalent Transformation Framework (ETF) as the base of our study. In an ETF, a set of rules that defines correct computations is generated from a problem specification described by a set of definite clauses. A correct program that executes the actual computation defined by the rule set is then generated. This paper discusses program generation for distributed memory parallel computers from a set of rules within ETF. For distributed parallel processing, we extract a part of a computation, characterized by a general concept called "specialization" in ETF from the computation defined by the set of rules. Specialization is a kind of atom operation that includes unification in logic programming. In this paper, we present a parallel processing program generation framework based on specialization, its theoretical background, and its implementation. The correctness of the program generated is also discussed.*
Keywords: Parallel processing, Program correctness, Equivalent transformation, Specializer generation rule

1. **Introduction.** As the Internet and PCs have become more popular, parallel processing programs that run on distributed-memory parallel computers (DMPC) such as PC clusters have recently been in great demand. Development of parallel processing programs to solve a problem requires to find factors that are proccessed in parallel, i.e. a parallelism, from the problem or computations to solve the problem. Also the parallelism extracted from the problem or the computations is affected by a framework in which the problem is defined or the computations are described. For example, parallel processing languages derived from the logic programming (LP) are based on the "AND/OR" parallelism [5, 7, 11]. These parallelisms are extracted naturally in LP because it represents problems and computations in logic, which consists of multiple independent goals and candidate clauses. This example shows a fact that a framework to describe problems and computations does affect parallelism extracted from them. We set the goal of our study to propose parallel processing programs based on a new parallelism extracted from another framework. To achieve this goal, our study employs an Equivalent Transformation