# Theory, Implementation, and Applications
# for Structured Parallel Programming

**Project Leader**

MATSUZAKI Kiminori, Ph.D.

Professor, Information Systems Engineering

## 1. Objective

**This project is aimed at:**

Developing the theory for a new paradigm in structured parallel programming and realizing it as library implementations with several application examples. Since the success of Google's MapReduce and Intel's Threading Building Blocks (TBB), structured parallel programming (also called skeletal parallel programming) has grown in importance. To promote parallel programming on multicores, manycores, and large clusters, we have developed a new paradigm in structured parallel programming through intensive work in both theory and practice.

## 2. Project Outline

**To that end, the project will consist of the following phases:**

(a) Theory: Apply concepts from functional programming (such as parallel data structures, program transformation, certified programming) to structured parallel programming.

(b) Implementation: Extend existing frameworks or develop a new library of structured parallel programming. We have been developing a parallel skeleton library, "SkeTo."

(c) Practice: Apply the theory and implementation to several domains, including large simulations, combinatorial optimization problems, and data mining in neuroscience.

## 3. Expected Performance

**In this project, the successful candidate would be expected to:**

(a) Work with the project leader and join the inter-university research team.

(b) Develop a library and applications in C++ and MPI, Java, or CUDA, with performance tuning.

(c) Perform routine work for the maintenance of cluster systems.

## 4. Required Skills and Knowledge

**The successful candidate for this project will have the following knowledge and skills:**

(a) Superior skills in programming (C, C++, Java, etc.) and problem solving with algorithms.

(b) Mathematics, especially algebra and statistics.

(c) Experience in functional programming (Haskell, ML, Scheme, etc.) and parallel programming (OpenMP, MPI, CUDA, MapReduce, TBB, Cilk) are desirable.

**References**

[1] Y. Liu, Z. Hu, K. Matsuzaki: Towards Systematic Parallel Programming over MapReduce. In Proc. EuroPar 2011 Part II, LNCS 6853, pp. 39-50, Springer. (2011)

[2] K. Matsuzaki, K. Emoto: Implementing Fusion-Equipped Parallel Skeletons by Expression Templates. In IFL 2009 Revised Selected Papers, LNCS 6041, pp. 72-89, Springer (2010)

[3] K. Morita, A. Morihata, K. Matsuzaki, Z. Hu, M. Takeichi: Automatic Inversion Generates Divide-and-Conquer Parallel Programs. In *Proc. PLDI 2007*, pp. 146-155, ACM. (2007)

**See my web page:**

https://www.kochi-tech.ac.jp/english/admission/ssp/guideline.html

**See our admission guidelines:**

https://www.kochi-tech.ac.jp/english/admission/ssp_aft19oct/ssp_application_guideline.html

**Contact**

E-mail: matsuzaki.kiminori@kochi-tech.ac.jp