

**Paper Review Title: Thread Scheduling for Multi-Core Platforms,**  
Mohan Rajagopalan Brian T. Lewis Todd A. Anderson

**Reviewer: swayanshu shanti pragnya**

*1. Summarize the (at most) 3 key main ideas.*

In this paper, they have introduced an unique scheduling framework for multi core processor by using the concept RTID(Related thread ID).

The key idea of the paper includes the following points I.e.

1. The frame work allows users to give instructions for guiding thread scheduling. At the time of necessity this frame work allows programmers to control over thread placement which gives flexibility to the programmers.
2. Characteristics of many core architecture along with the interface design.
3. In the other hand they have shown the difficulties this architecture faced and still the performance enhancement

*2. State the main contribution of the paper*

The primary contributions are-

1. Providing flexibility to the programmers over thread placement.
2. Creating a framework for multi core platforms to exhibit thread scheduling
3. A scheduling framework which helps to achieve good performance in many core platforms without compromising on the difficulty of programming.

*3. Critique the main contribution*

*3.a. Rate the significance of the paper on a scale of 5 (breakthrough), 4 (significant contribution), 3 (modest contribution), 2 (incremental contribution), 1 (no contribution or negative contribution). Explain your rating in a sentence or two.*

It is a significant contribution so I would rate as 4 points instead of 5 because they are in the initial implementation for McRT where till now they did not actually evaluated the Run time with experimenting on different scheduling processes. Yet the idea of multi core scheduling allowance to users and programmers really efficient the contribution with the architectural significance.

*4. b. Rate how convincing is the methodology: do the claims and conclusions follow from the experiments? Are the assumptions realistic? Are the experiments well designed? Are there different experiments that would be more convincing? Are there other alternatives the authors should have considered? (And, of course, is the paper free of methodological errors.)*

They created a complete architectural overview for

1. Target architecture (Here, each underlying hardware thread(HWTs) will have a SMP system)

2. RTID placement interface (Here mcrdt threads can share caches explicitly with RTID)

**Here is Problem:**

1. It does not guarantee for the performances if any program will mix the automatic and explicit replacements of threads respectively.

So for these two methods the claim is to avoid the thread replacements which they have fixed in RTID placements by,

1. supporting the explicit data allocation and 2. by allowing clients to specify the data location to place near RTID

These 2 solutions makes the methodology well designed .

I think this is the convincing way to give access for both target and thread location allotment by the user itself

*3. c. What is the most important limitation of the approach?*

The proposed approach of RTID allocation for HWT for performance enhancement is the import limitation. Because the high performance on a particular system might not be able to achieve for other system.

It may perform differently across a range of simultaneous programs on different architecture.

Other limitation is “How to improve application performance?”

*4. Rate the writing in the paper on a scale of 5 (great) to 1 (muddled), and justify your ranking. Did you have to re-read sections? Were algorithms clearly explained? Did the paper have a logical flow?*

Writing-5

The paper had a logical flow and proper explanation of methodology like starting from the paper objective, architectural design, limitations, future goal for improvising the performance are well explained.

Scientific terminology like **RTID**, **HWT**,

Functions like **mcrdtFreeRtid** and **mcrdtAssociateRtidWithHwt** were used appropriately with proper explanation of their functionality usage.

Rereading was not required as the related and introductory part explains all the basic idea behind the paper proposal and advancements.

*5. Answer one of the following three questions (whichever is most relevant for this paper):*

*1. What lessons should system researchers and builders take away from this work?*

*2. What is the lasting impact of this work?*

*3. What (if any) questions does this work leave open?*

This work is having a greater impact for providing flexibility in terms of user and programmer.

The lasting impact will be the Implementation of the paper which will lead a scheduling of threads that will maximize the performances in all the allocated cores of a system.

But at the same time the architectural limitation asks a question about “***How to increase the performance ?***”