Hierarchical Agent Monitored System:

an innovative paradigm for parallel computing

Liang Guang, Jouni Isoaho, Hannu Tenhunen

Dep. of Information Technology, University of Turku, 20520 Turku, Finland
{liagua, jisoaho, hatenhu}@utu.fi

ABSTRACT

Hierarchical agent monitoring design approach is an innovative paradigm to achieve self-aware and autonomic parallel computing in a scalable manner. Run-time monitoring services are assigned to hierarchical agents, which observe system status and reconfigure the components dynamically towards better performance under the influence of variations and errors. Current targeted platform is the NoC-based on-chip parallel system, upon which specific design practices, for instance hierarchical communication power monitoring, are being analyzed. This paper briefly overviews the design approach with reference to our other documents for interested readers.

KEYWORDS: parallel computing; SoC; run-time monitoring; hierarchical agents

1 Run-time Monitoring Services on Parallel SoCs

Constant transistor scaling enables parallel computing in on-chip systems, for instance, TFLOPS multiprocessor has 80 processing cores fabricated in 65nm CMOS technology. Our work is dedicated to ensuring proper working and optimizing the performance of on-chip resources by run-time monitoring operations.

The emerging highly-parallel SoC system suffers from varying and unpredictable run-time status due to three reasons. One is the submicron effects such as crosstalks, and another is the PVT (process, voltage, thermal) variations. Last but not least, errors and faults, both hardware and software ones, have become distributed and unpredictable due the submicron effects and variations. One generic paradigm to ensure system performance under unpredictable platform status is to exploit run-time control, monitoring and optimization techniques. Existing works have already adopted various run-time monitoring techniques, for instance thermal monitoring against hotspot, power monitoring with adaptive voltage scaling or power gating methods, run-time core reconfiguration in case of failures. With the parallelization of components, such techniques need to be incorporated in a scalable manner under the constraints of a feasible SoC implementation, for instance area and power limitations. The design complexity also needs to be carefully examined and design reuse is a major method to lower the design complexity and reduce time-to-market.

2 Hierarchical Agent Monitoring Design Approach

Based on the hierarchical view of parallel system with three levels of component groups (cell, cluster and platform), we define the agents monitoring each level of component as cell agent, cluster agent, and platform agent (Fig. 1). On a regular-layout NoC-based many-core platform (Fig. 2), we consider a cell as the union of a processing element with its network interface and the switch as well the communication channel starting from the switch, monitored by one cell agent. A number of cells are dynamically grouped into a cluster, which is monitored by a cluster agent hosted by one processing unit. The mapping of platform and application agent is flexible, as they may be located outside the processing array.
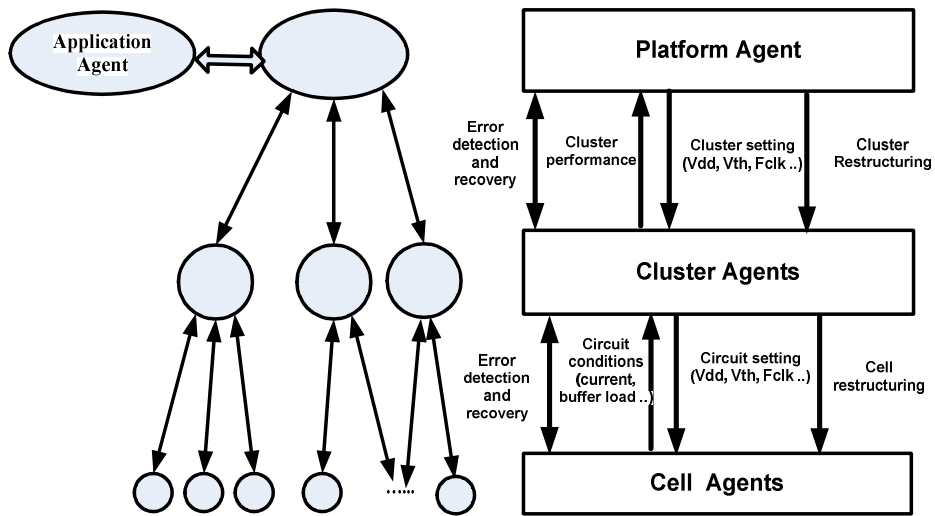

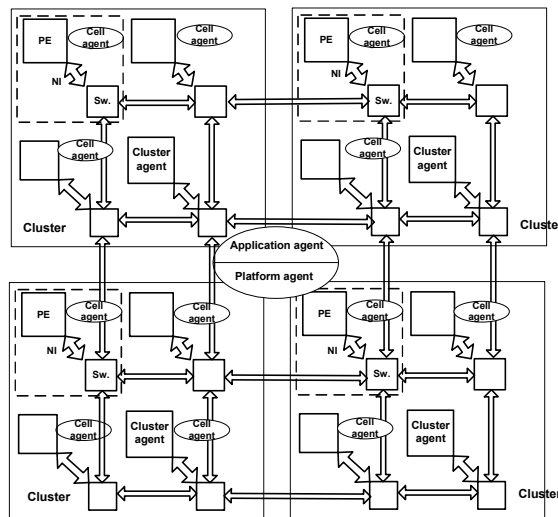
Fig. 1 : Generic Agent Hierarchy for Moring Services

Fig. 2: Illustrative Mapping of Hierarchical Agents on Regular NoC structure

Application agent provides application specification including functional and non-functional requirements, such as computing time and power constraints. Platform agent analyzes the application specification, accordingly configures the network and allocates the tasks and schedules them onto proper clusters. It plays the important role of exploiting task level and physical level parallelism. Cluster agents are the regional monitor in each cluster. They allocate and schedule the instruction segments onto cells, and configure the network within the cluster. Cell agents are the local monitors, which implement various mechanisms of fine-grained tuning and monitoring, for example AVS (adaptive voltage scaling) and ABB (adaptive body biasing), to tolerate variation and minimize power. Very importantly as to provide real-time performance monitoring and fault-tolerance, each level of agents will report the system performance of their corresponding level and any detection of fault to their higher-level agents. Since agents themselves are possible victims of faults and errors, the error state of agents are also traced by their higher-level agents. Common agent interactions in SoCs are illustrated in Fig. 1.

3 Design Practices

Hierarchical agent monitoring architecture can be applied to various types of parallel system for a wide range of monitoring services.

For on-chip system, we currently focus on power monitoring services. By utilizing hierarchically organized agents, we are designing "hierarchical power monitoring" to exploit the energy efficiency at all architectural levels. Conventional techniques are somewhat ad-hoc, and mostly focuse on the optimization at one specific architectural level. Since each level of configuration has direct influence on the general power efficiency of the whole platform, hierarchically monitored system will have better efficiency than conventional techniques. Fig. 3 illustrates the major services at each level of agent. Preliminary results can be found in [1].
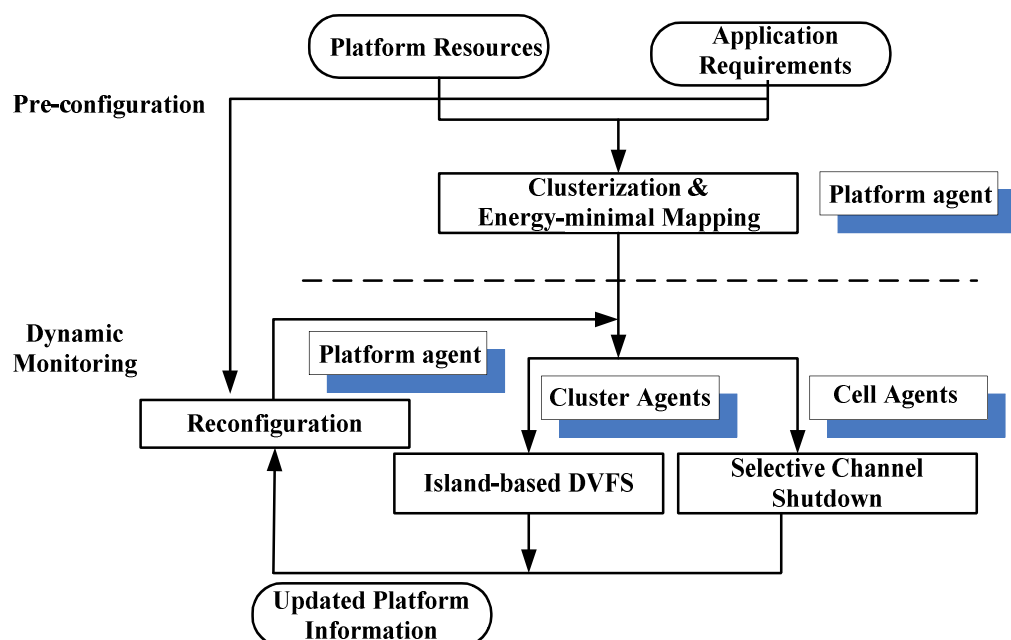
Fig. 3 Hierarchical Power Monitoring Serives

We are also looking at the usage of the hierarchical agent monitoring design approach on large-scale networked systems.

4 Conclusion & On-going Work

We have proposed an innovative monitoring-centric design approach with hierarchical agent monitoring architecture. This approach has the intrinsic scalability to address the constantly parallelizing embedded system.   [3] focuses on presenting the motivation and novelty of our work as a design approach.

Hierarchical agent monitoring architecture can be naturally applied to NoC-based platform, where the regularity and modularity of components provide a straightforward mapping of the agent entities on NoC structures.   Hierarchical monitoring for communication power is one of the major services under study. [1] presents cluster-level autonomous DVFS as initial results.   We are working on the design of power monitoring services at each level. Other services including fault-tolerance and variability are also being analyzed.

Hierarchical agent monitored architecture is a complex design environment. To facilitate accurate specification for designers and future computer-aided design/verification, we are working on its formal model. [4] presents a preliminary theoretic model on the system level.   The formal model is being elaborated at present.

References (selected list of publications)

[1] Liang Guang, Ethiopia Nigussie, Lauri Koskinen, and Hannu Tenhunen, Autonomous DVFS on Supply Islands for Energy-constrained NoC Communication, In Proceeding of ARCS09 (international conference on architecture of computing systems), LNCS 5455, pp.183-194, 2009.

[2] Alexander Wei Yin, Liang Guang, Pasi Liljeberg, Pekka Rantala, Ethiopia Nigussie, Jouni Isoaho, Hannu Tenhunen.   Hierarchical Agent Architecture for Scalable NoC Design with Online Monitoring Services.   First International Workshop on Network on Chip Architectures, held in junction with 41st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO41), 2008.

[3] Liang Guang, Pekka Rantala, Jouni Isoaho, Ethiopia Nigussie, Hannu Tenhunen. Hierarchical Agent Monitoring Design Approach towards Self-Aware Systems. Submitted to ACM Transactions in Embedded Computing Systems (TECS), 2009, under review (minor revision).

[4] Liang Guang, Juha Plosila, Jouni Isoaho, Hannu Tenhunen, Hierarchical Agent Monitoring Services on Reconfigurable NoC Platform : A Formal Approach, presented in DSNOC (Diagnostic Services in Network-on-Chips) workshop in DATE 2009.