Attention, please!

- Homework assignment 1 is available online.

SDSU Ranking

- U.S. News & World Report's annual ranking of America's Best Colleges lists SDSU at No. 68 among public universities, and No. 140 overall among national universities. There are 311 national universities in the US.
- SDSU has been gaining ground over the last five years, moving up 43 spots from No. 183 overall in 2011.

CS696 Research Project (3) – The whole process

Tao Xie

Department of Computer Science
San Diego State University

Content

- Establishing your goal and timing
- Identifying a research problem
- Summarizing existing related work
- Focusing on one particular technique
- Designing your algorithm
- Conducting experiments
- Writing your technical paper

Writing Your Paper
(Enjoy Your Results)

- This topic itself deserves one lecture
- First of all, read good articles from top resources and learn writing from there
- Second, ask your professor to revise your draft
- Normally, a technical paper includes:

Introduction, Related Work, System Model, Algorithm And Its Analysis, Experimental Results, Conclusion And Future Work, References.

Abstract

- No more than 200 words (150~200 is good)
- First sentence: why this topic is critical?
- Second: what is the problem of traditional techniques?
- Third: what this paper will do on this problem?
- Fourth: what methodology or approach will be taken?
- Fifth: what are the merits of our approach?
- Sixth: what are the experimental results?

A Sample Abstract

- High performance, highly reliable, and energy-efficient storage systems are essential for mobile data-intensive applications such as remote surgery and mobile data center. Compared with conventional stationary storage systems, mobile disk-array-based storage systems are more prone to disk failures due to their severe application environments. Further, they have very limited power supply. Therefore, data reconstruction algorithms, which are executed in the presence of disk failure, for mobile storage systems must be performance-driven, reliability-aware, and energy-efficient. Unfortunately, existing reconstruction schemes cannot fulfill the three goals simultaneously because they largely overlooked the fact that mobile disks have much higher failure rates than stationary disks. Besides, they normally ignore energy-saving. In this paper we develop a novel reconstruction strategy, called multi-level caching-based reconstruction optimization (MICRO), which can be applied to RAID-structured mobile storage systems to noticeably shorten reconstruction times and user response times while saving energy. MICRO collaboratively utilizes storage cache and disk array controller cache to diminish the number of physical disk accesses caused by reconstruction. Experimental results demonstrate that compared with two representative algorithms DOR and PRO, MICRO reduces reconstruction times on average 20.22% and 9.34%, while saving energy no less than 30.4% and 13%, respectively.

Introduction
(Precisely And Concisely)

- Problem Statement (background)

  One paragraph

- Motivation

  one or two sentences enough

- Contributions

  Several sentences

- Paper Organization
- The rest of the paper is organized as follows. In the next section we discuss the related work and motivation. In Section 3, we describe the design and implementation of the MICRO strategy. In Section 4 we evaluate performance of MICRO based on synthetic benchmarks. Section 5 concludes the paper with summary and future directions.

Related Work (Summarize In Stead Of Re-state)

- Summarize representative related work
- Point out their upsides and downsides
- Motivation (could be more detailed than the Motivation sentence in Introduction)

Note: Don’t miss some important references.

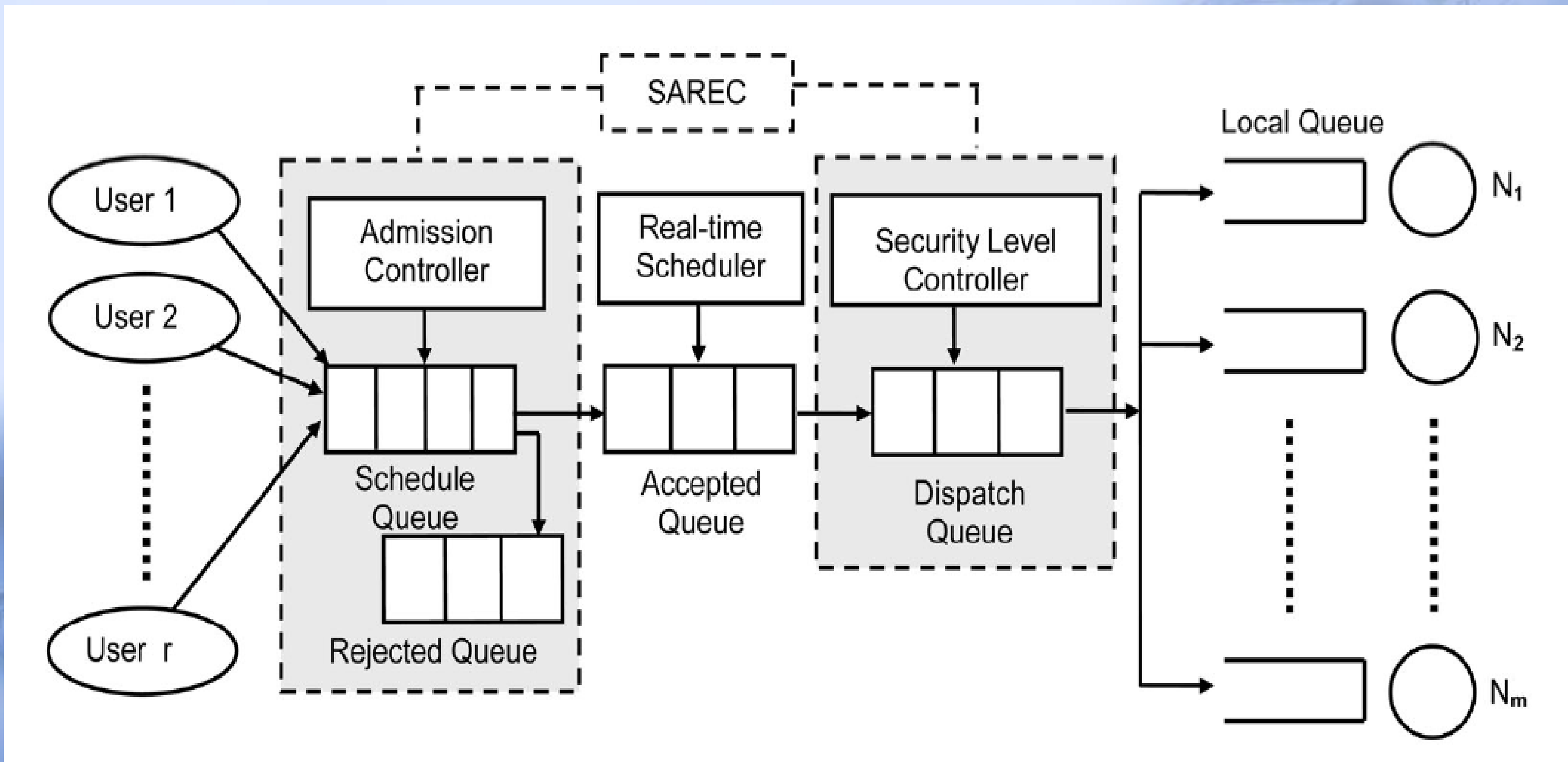
A Sample Motivation

- Although a number of studies reported in the literature have concentrated on how to improve storage system reliability, performance, and energy-efficiency by utilizing caches in either single level or multi-level [6], [33], [37], [38], none of them attempted to address data reconstruction issue by employing multi-level storage caches. On the other hand, existing reconstruction algorithms mainly focus on optimizing data layouts or reconstruction workflow, which are all essentially disk-oriented in the sense that the read/write access requests they generated during recovery are served by physical disks. Based on insightful observations made by research work in multi-level storage caching subsystems discussed above and by our own research, we believe that ….

System Model
(It's better to have a figure)

- Mathematic models that you used
- An architectural level figure of your research
- Use formal way to state the problem and your solution

System Model Example (1)



System Model Example (2)

function. Our proposed security-aware scheduling algorithm strives to schedule tasks in a way to maximize (4):

$$SV(X) = \max_{x \in X} \left\{ \sum_{i=1}^{p} \left(y_i \max_{x_i \in X_i} \left\{ \sum_{j=1}^{q} w_i^j s_i^j(x_i) \right\} \right) \right\}. \quad (4)$$

Algorithm And Its Analysis (Complexity Analysis)

- Time and space cost analysis

- Optimal solution proof

Time Complexity Analysis Example

Theorem 1. *The time complexity of SAEDF is* $O(knm)$, *where* m *is the number of nodes in the cluster,* n *is the number of tasks in the local queue of a node, and* k *is the number of possible security level ranks for a particular security service* v_l $(v_l \in \{a, c, g\}, 1 \leq l \leq 3)$.

Proof. The time complexity of finding the earliest start time for task T_i on a node is $O(n)$ (Step 3). To obtain the minimal security overhead c_i^{min} of task T_i, the time complexity is a constant $O(1)$ (Step 4). Sorting the security service weights in decreasing order (Step 6) will take a constant time $O(1)$ since we only have three

Lemma Proof Example

Lemma 2. *Let I be the set of indices representing files whose access rates and service times satisfy $\sum_{i \in I} \lambda_i \cdot s_i = 2\rho_0$. Let $\{I_1^*, I_2^*\}$ be a PBFA found by Sort partition. Then, for any PBFA $\{I_1, I_2\}$, the following inequalities hold*

$$\sum_{i \in I_1^* \cap I_1} \lambda_i \le \sum_{i \in I_2^* \cap I_2} \lambda_i \tag{5}$$

$$\sum_{i \in I_1^* - I_1^* \cap I_1} \lambda_i \le \sum_{i \in I_2^* - I_2^* \cap I_2} \lambda_i . \tag{6}$$

Proof. We first prove that

$$\sum_{i \in I_1^* \cap I_1} \lambda_i \cdot s_i = \sum_{i \in I_2^* \cap I_2} \lambda_i \cdot s_i .$$

Since both $\{I_1, I_2\}$ and $\{I_1^*, I_2^*\}$ are binary partitions of I, we have $I_1^* - I_1^* \cap I_1 \subseteq I_2$. On the other hand, we have $(I_1^* - I_1^* \cap I_1) \cap I_2^* = \emptyset$, since $I_1^* \cap I_2^* = \emptyset$. Hence,

$$I_2^* \cap I_2 \subseteq I_2 - [I_1^* - (I_1^* \cap I_1)].$$

It follows that

$$\begin{aligned}
\sum_{i \in I_2^* \cap I_2} \lambda_i \cdot s_i &\le \sum_{i \in I_2 - [I_1^* - (I_1^* \cap I_1)]} \lambda_i \cdot s_i \\
&= \sum_{i \in I_2} \lambda_i \cdot s_i - \sum_{i \in I_1^* - (I_1^* \cap I_1)} \lambda_i \cdot s_i \\
&= \sum_{i \in I_2} \lambda_i \cdot s_i - \sum_{i \in I_1^*} \lambda_i \cdot s_i + \sum_{i \in I_1^* \cap I_1} \lambda_i \cdot s_i \\
&= \sum_{i \in I_1^* \cap I_1} \lambda_i \cdot s_i ,
\end{aligned}$$

where the last equality results from the fact that $\sum_{i \in I_2} \lambda_i \cdot s_i = \sum_{i \in I_1^*} \lambda_i \cdot s_i = \rho_0$.

Evaluation

- Describe your experimental configurations
- Give the performance metrics
- Analyze the results based on your observations
- Explain some interesting observations
- This part is used to support your claim and your algorithm analysis section

Evaluation Example

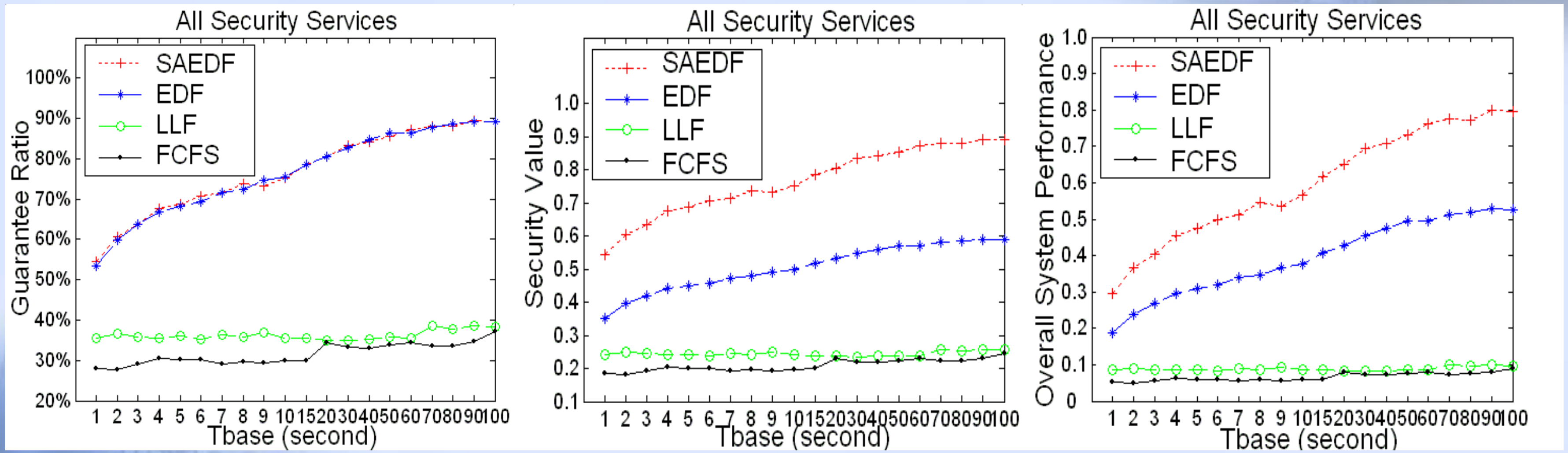


Fig. 5. Simulation performance of four scheduling algorithms.

Conclusions

- Briefly rephrase your work and its contributions
- Mention some possible limitations
- Point out future directions

References

- List all the papers that you cited in alphabetic order
- Obey the IEEE format (see our class web page for IEEE paper format)

References

[1] T.F. Abdelzaher and K.G. Shin., “Combined Task and Message Scheduling in Distributed Real-Time Systems,” *IEEE Trans. Parallel and Distributed Systems,* Vol. 10, No. 11, Nov. 1999.

[2] T.F. Abdelzaher, E. M. Atkins, and K.G. Shin., “QoS Negotiation in Real-Time Systems and Its Application to Automated Flight Control,” *IEEE Trans. Computers,* 49(11), Nov. 2000, pp.1170-1183.

File Assignment In Parallel I/O Systems With Minimal Variance Of Service Time (Research Example)

- We address the problem of assigning non-partitioned files in a parallel I/O system where the file accesses exhibit Poisson arrival rates and fixed service times. We present two new file assignment algorithms based on open queuing networks which aim at minimizing simultaneously the load balance across all disks, as well as the variance of the service time at each disk. We first present an off-line algorithm, Sort Partition, which assigns to each disk file with similar access time. Next, we show that, assuming that a perfectly balanced file assignment can be found for a given set of files, Sort Partition will find the one with minimal mean response time. We then present an on-line algorithm, Hybrid Partition, that assigns groups of files with similar service times in successive intervals while guaranteeing that the load imbalance at any point does not exceed a certain threshold. We report on synthetic experiments which exhibit skew in file accesses and sizes and we compare the performance of our new algorithms with the vanilla greedy file allocation algorithm

How they found this topic?

- Existing algorithms for assigning data to disks in parallel or distributed systems have focused on reducing average response time.
- Algorithms such as Greedy achieved this goal by load balancing.

  i.e., evenly distributing data onto an array of disks
- Two observations they made

1. The mixture of large files and small files will severely delay small files
2. Most visit requests target on small files

Their Idea

- Still, the variance of service times at each disk can affect mean response time.
- Namely, small files will be put together such that no large files can block them.
- It is intuitive that their idea most likely can further improve mean response time.

Two Algorithms

- Sort Partition
- Hybrid Partition

Algorithm Analysis

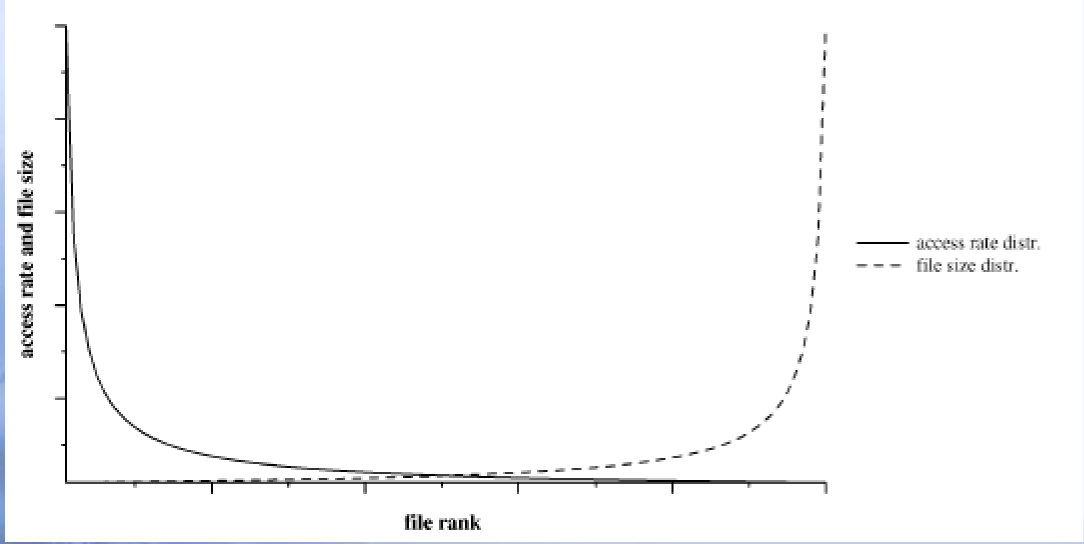
- The heat of a file is defined as

  $h_i = Lambda_i * s_i$

- They proved that among all PBFAs (Perfectly balanced File Assignment), SP finds the assignment guaranteeing minimal system response time

Experiments (1)

- They used FIVE, a parallel I/O system prototype.
- They assumed that the distribution of access rates and file sizes were inversely correlated.



Experiments (2)

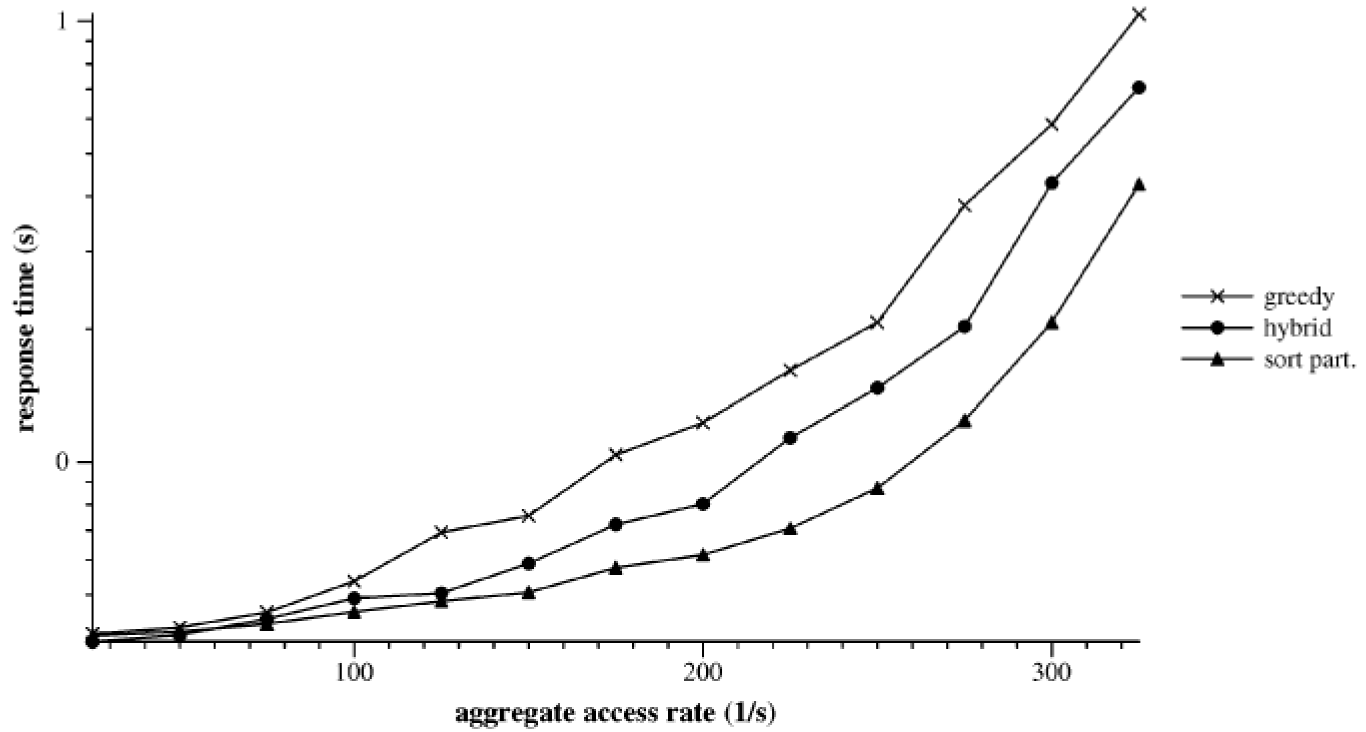


Fig. 8. Average response time for 70/30 skew, one batch.

Please read this paper

- "Overlay Networks" on our class web page.

How to Write a Proposal?

- Introduction
- Motivation
- Project Summary
- Project Details
- Conclusions
- References

Introduction (1)

- Use first 2 or 3 sentences to describe the background and current situation of the problem that you will address in this project

  Parallel applications with energy and low-latency constraints are emerging in various networked embedded systems like digital signal processing, vehicle tracking, and infrastructure monitoring. However, conventional energy-driven task allocation schemes for a cluster of embedded nodes only concentrated on energy-saving when making allocation decisions. Consequently, the length of the schedules could be very long, which is unfavorable or in some situations even not tolerated.

Introduction (2)

- Use one sentence to explicitly tell your readers what is the exact problem you are addressing

  In this project, we address the issue of allocating a group of parallel tasks on a heterogeneous embedded system with an objective of energy-saving and short-latency.

Introduction (3)

- Use one sentence to briefly describe your algorithm

  A novel task allocation strategy, or BEATA (Balanced Energy-Aware Task Allocation), is developed to find an optimal allocation that minimizes overall energy consumption while confining the length of schedule to an ideal range.

Introduction (4)

- Use 3~4 sentences to describe the organization of this proposal

The rest of the proposal is organized as follows. In the next section we describe the motivation of this research. In Section 3, we propose the project summary. Project details are presented in Section 4. Section 5 concludes the proposal with summary.

Attentions to Introduction Section

- Totally, 150 ~ 200 words in one or two paragraphs
- No mathematic formula, reference citation, or fancy nouns without definition !
- No detailed implementation information!
- Make it interesting!

Motivation (1)

- What is the history of the problem and why is this problem important?

  Extensive researches have been conducted to reduce overall energy consumption for a variety of embedded systems using diverse techniques [1][2][3]. In particular, most of recent researches in energy-saving for embedded systems share two common features (1) applications considered are real-time in nature with hard deadlines; and (2) energy-saving is achieved by employing DVS (Dynamic Voltage Scaling).

Motivation (2)

- When and why does the problem occur and what is done now?

  Our work is fundamentally different from the above approaches as we focus on reducing both energy consumption and response time for soft real-time parallel applications running on heterogeneous embedded systems with no DVS available. In a heterogeneous embedded system, different processing nodes have distinct fixed energy consumption rates. Similarly, different communication channels also have various energy assumption rates. The goal of this work is to develop a task allocation strategy that not only conserves energy but also generates a short schedule, which is favorable or even necessary in some scenarios.

Motivation (3)

For example, in a soft real-time embedded system such as a cellular phone [2], it must be able to encode outgoing voice and decode incoming signal during a conversation in a timely manner. Occasional glitches in conversations due to tardy response are not desired. When the response time becomes longer frequent glitches could happen, which are not tolerated at all.

Motivation (4)

Energy-saving and low-latency, however, are two conflicting objectives in the context of allocating a parallel application represented by a task graph onto a set of connected heterogeneous processing nodes in an embedded system. The dilemma arises from a multidimensional heterogeneity bearing by a heterogeneous embedded system. Specifically speaking, a processing node that provides a task with earliest finish time may not be an ideal candidate in terms of energy-saving. This is because the execution time of a task allocated on an embedded node is irrelative to the energy consumption rate offered by the node. Moreover, the computational energy consumption of a task allocated on a node is a product of energy consumption rate of the node and execution time of the task. The motivation of this work is to solve the energy-latency dilemma existed in networked heterogeneous embedded system where both energy-saving and low-latency need to be achieved. In this project, we address the issue by minimizing energy consumption while confining schedule lengths. To this end, we devised an energy-adaptive window to control the trade-off between energy consumption and response time.

Attentions to Motivation Section

- 2 ~ 3 paragraphs with totally 300 ~ 400 words
- Cite papers that relates to your project
- Highlight the differences between your solution and the existing ones
- If you add something new, please explain why this new feature is necessary

Project Summary

In this project, we address the issue of allocating tasks of parallel applications in heterogeneous embedded systems with an objective of energy-saving and latency-reducing. BEATA (Balanced Energy-Aware Task Allocation), a task allocation scheme considering both energy consumption and schedule length, is developed to solve the energy-latency dilemma. To facilitate the presentation of BEATA, we will also propose mathematical models to describe a system framework, parallel applications with precedence constraints, and energy consumption model. Extensive simulations using a real world application as well as synthetic benchmarks will be conducted to compare the performance of existing approaches with that of the BEATA scheme. The experimental results will show that BEATA significantly improves the performance in terms of energy dissipation and makespan time over two baseline allocation schemes.

Attentions to Project Summary

- 1 paragraph with totally 100 ~ 150 words
- Do NOT delve into details or timelines

Project Details (1)

- Architecture

A networked embedded system in the most general form consists of a set, e.g., *P* = {*p1*, *p2*, ..., *pm*}, of heterogeneous embedded computing nodes (hereinafter referred to as nodes or embedded nodes) connected by a single-hop wired or wireless network. The network embedded system can be represented as a graph of nodes along with their point-to-point links. In the system, an embedded node is modelled as a vertex. There exists a weighted edge between two vertices if they can communicate with each other. Each node in the system has an energy consumption rate measured by Joule per unit time. With respect to energy conservation, each network link is characterized by its energy consumption rate that heavily relies on the link's transmission rate, which is modelled by weight *wij* of the edge between node pi and pj. An allocation matrix *X* is an *n×m* binary matrix used to reflect a mapping of n tasks to m embedded nodes. Element *xij* in *X* is "1" if task *ti* is assigned to node *pj* and is "0", otherwise.

Project Details (2)

- New Challenges

1. **Very limited power supply In contrast to high performance computing platforms such as clusters located in data center buildings where electrical power is guaranteed, embedded systems only have very limited power supply provided by batteries. The lifetime of batteries of an embedded system directly determines the lifetime of the embedded system.**

2. **Stringent response requirement Normally, embedded systems are used to collect information in a mobile environment. Quick response time is desirable or mandatory in many situations. Therefore, low latency must be achieved. Otherwise, the value of the results will be degraded or useless.**

3. **Achieving energy-saving and low-latency simultaneously As we discussed in Section 2, energy-saving and low-latency are two conflicting objectives in the context of allocating a parallel application represented by a task graph onto a set of connected heterogeneous processing nodes in an embedded system. How to save energy while still keep response performance becomes more challenging in this context.**

Project Details (3)

- Contributions

  Our main contributions include: (1) developing an energy-delay driven task allocation strategy called BEATA for collaborative applications running in heterogeneous networked embedded systems, (2) constructing an energy consumption model for quantitatively measuring energy caused by both computation and communications, (3) extending a heterogeneity model to reveal an inherent nature of a heterogeneous embedded system and its impact of system performance, and (4) simulating a heterogeneous embedded system where the BEATA strategy is implemented and evaluated.

Project Details (4)

We will use Matlab to compose a simulator, which simulates a set of connected heterogeneous processing nodes in an embedded system. In addition, our trace generator will produce synthetic workload for the simulated embedded system. For real-world application trace data, we will use DSP (Digital Signal Processing) because it is a typical parallel application. The impacts of parameters that will be examined in our experiments are:

Project Details (5)

- Timelines
- Make your timeline reasonable
- Give yourself around 3 weeks to write

Attentions to Project Details

- No source code!
- A graph for architecture is desirable

Conclusions (1)

- **In this project, we address the issue of allocating tasks of parallel applications in heterogeneous embedded systems with an objective of energy-saving and latency-reducing. BEATA (Balanced Energy-Aware Task Allocation), a task allocation scheme considering both energy consumption and schedule length, is developed to solve the energy-latency dilemma.**

Conclusions (2)

- **Future studies in this research can be performed in the following directions. First, we will extend our scheme to multi-dimensional computing resources from which energy-saving can be achieved. For now, we simply consider CPU time and network communication time. Memory access and I/O activities will be considered in the future. Second, we intend to enable the BEATA scheme to deal with real-time parallel applications, where the hard deadlines must be guaranteed.**

References

References

[1] W.S. Conner, L. Krishnamurty, R. Want, "Making Everyday Life Easier Using Dense Sensor Networks," *Ubicomp*, pp. 49-55, 2001.

[2] R. P. Dick, D. L. Rhodes, and W. Wolf, "TGFF:Task graphs for free," *Proc. Int. Workshop. Hard-ware/Software Codesign*, pp. 97-101, Mar. 1998.

[3] D. Estrin, L. Girod, G. Pottie, M. Srivastava, "Instrumenting the world with wireless sensor networks," *Proc. of the Int'l Conf. on Acoustics, Speech, and Signal Processing*, Salt Lake City, Utah, May 2001.