Programming Assignment 3: Value-based Scheduling

Instructor: KD Kang

Objective: In this programming assignment, you are required to implement and compare EDF (Earliest Deadline First) with several value-based scheduling algorithms: (i) Highest Value First (HVF), (ii) Highest Value Density First (HVDF), and (iii) your own algorithm for value maximization. Assume that each job is associated with a value (or utility). The system acquires the value when the job finishes within its deadline. The objective is to maximize the total accrued value by intelligently scheduling real-time jobs. You are required to try to outperform HVF and HVDF. Even if your algorithm (or heuristic) cannot achieve a higher total value than the other algorithms do, you will get the full credit as long as you properly design, implement, and evaluate the algorithm against EDF, HVF, and HVDF.

Assignment Due: Through Blackboard, submit your code and short design document including the instruction for compilation and execution of your code by 11:59PM, April 23, 2009. Please note that no late submission will be accepted. Also, please remember to explain your algorithm and reason your algorithm design decisions in the document! You need to discuss why your algorithm could acquire a higher total value.

Simulation Spec. Basically, you can build upon the simulator developed for the first programming assignment. Compare the total value acquired by EDF, HVF, HVDF, and your own algorithm. (The value density of a job is its value divided by its execution time.) Key simulation parameters are specified in the following:

1. Task Arrival Pattern: All tasks are periodic. There is no aperiodic task. In addition, assume that every periodic task arrives at time $t = 0$. You can simulate this by generating all periodic tasks (called sources in the pseudo code of Programming Assignment 1) at $t = 0$.

2. One simulation runs for 10 (simulated) minutes. For each algorithm, run the simulation five times and take the average to get the normalized total value $\frac{\sum_{j=1}^{N_{\text{timely}}} \text{value}_j}{\sum_{j=1}^{N_{\text{total}}} \text{value}_j}$ where $N_{\text{timely}}$ is the number of the jobs finished within their deadlines, $\text{value}_j$ is the value of job $j$, and $N_{\text{total}}$ is the total number of the submitted jobs.

3. Do not apply admission control or imprecise computation.

4. A periodic task $T_i$ generates a job at every period $P_i$ where $P_i$ is uniformly selected in the range [200ms, 500ms], similar to Programming Assignment 1. Assign the execution time $C_i$ to $T_i$ where $C_i$ is uniformly selected in the range [5ms, 20ms].

5. To each job generated by $T_i$, assign a value selected between [0.25, 1] in a uniform random manner.

6. Apply load = 1.0, 1.1, 1.3, 1.4, and 1.5 for this assignment and show the total value acquired by EDF, HVF, HDVF, and your own algorithm.