Energy Efficient
Real-Time Scheduling

Amit Sinha and Anantha Chandrakasan

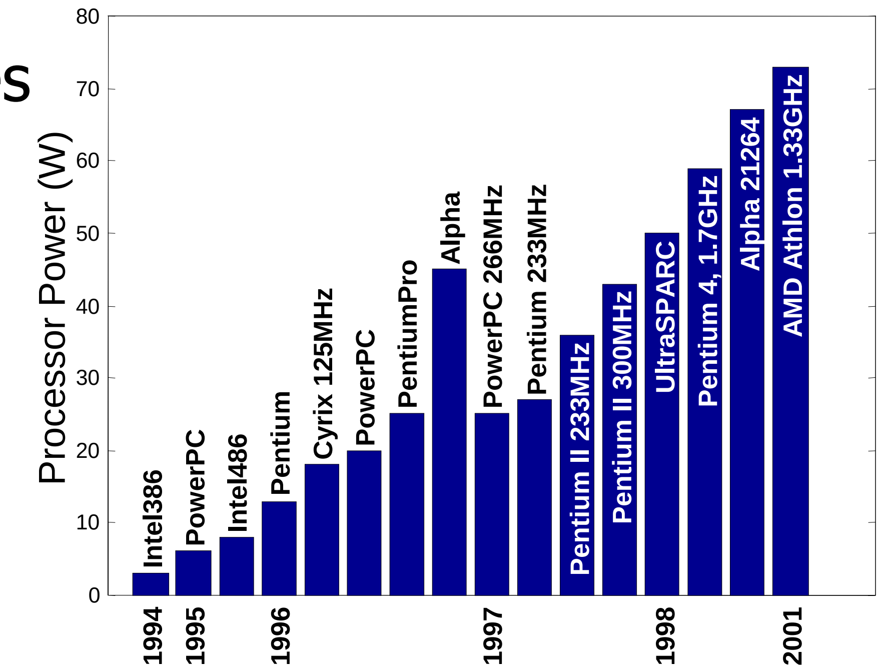
Massachusetts Institute of Technology

Outline

- Dynamic voltage and frequency scaling
  - Overview
  - Energy workload model
- Real-time algorithms
  - Performance metrics
  - Earliest Deadline First (EDF) algorithm
- Slacked Earliest Deadline First (SEDF) algorithm
- Bounds on energy savings
- Rate Monotonic extensions

Motivation for Energy Efficiency

- Proliferation of portable devices
  - Battery technology lags behind
  - 50X μP power vs. 4X in battery



- Electricity cost of servers, desktops and network equipment
  - Currently accounts for about 100 TWHr/year in the US

Dynamic Voltage Scaling

Fixed Power Supply

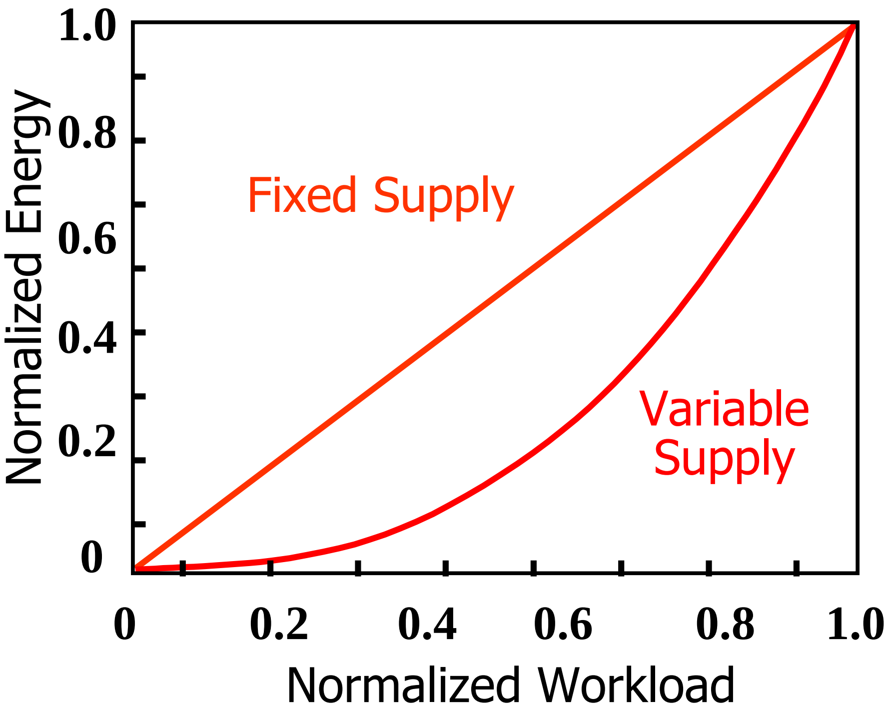
ACTIVE	IDLE

$$E_{FIXED} = \tfrac{1}{2} C V_{DD}^2$$

Variable Power Supply

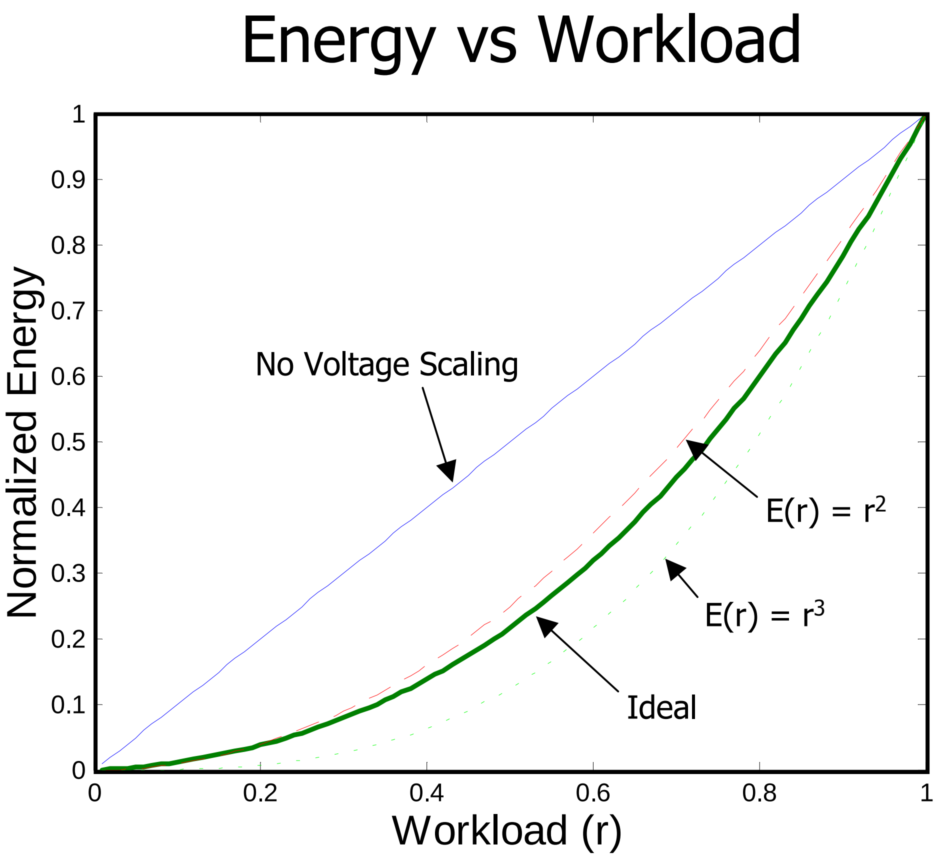
ACTIVE

$$E_{VARIABLE} = \tfrac{1}{2} C (V_{DD}/2)^2 = E_{FIXED} / 4$$



- Variable frequency processors
  - Transmeta's Crusoe
    - LongRun Technology
  - AMD K6-2+
    - PowerNOW!
  - Mobile Pentium III
    - SpeedStep
- StrongARM SA-1100
  - 59 MHz – 206 MHz

Energy Workload Model



$$E(r) = CV_0^2 T_s f_{ref} r \left[\frac{V_t}{V_0} + \frac{r}{2} + \sqrt{r\frac{V_t}{V_0} + \left(\frac{r}{2}\right)^2} \right]^2$$

$$I(r) = I_{ref} r \frac{V_0}{V_{ref}} \left[\frac{V_t}{V_0} + \frac{r}{2} + \sqrt{r\frac{V_t}{V_0} + \left(\frac{r}{2}\right)^2} \right]$$

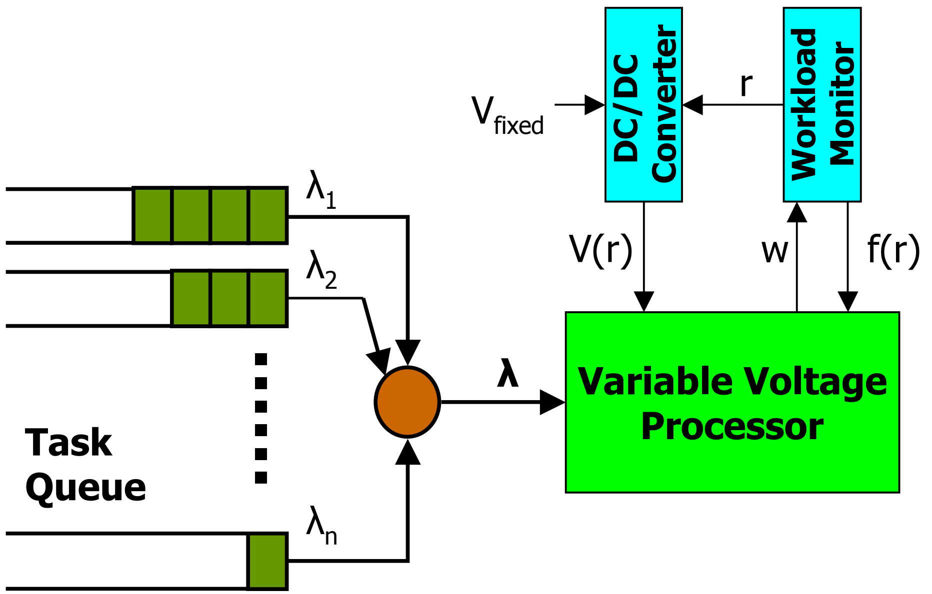
[Gutnik97]

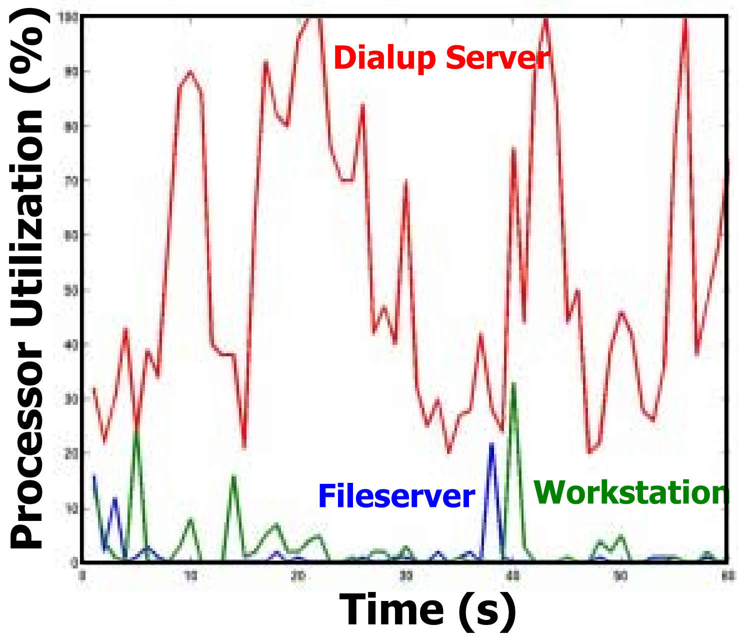
$$E(r) \approx r^2$$

$$E_{save}(r) \approx 1 - r^2$$

- Quadratic model fairly accurate
- E-r graph convex

Real-Time DVS Scheduling





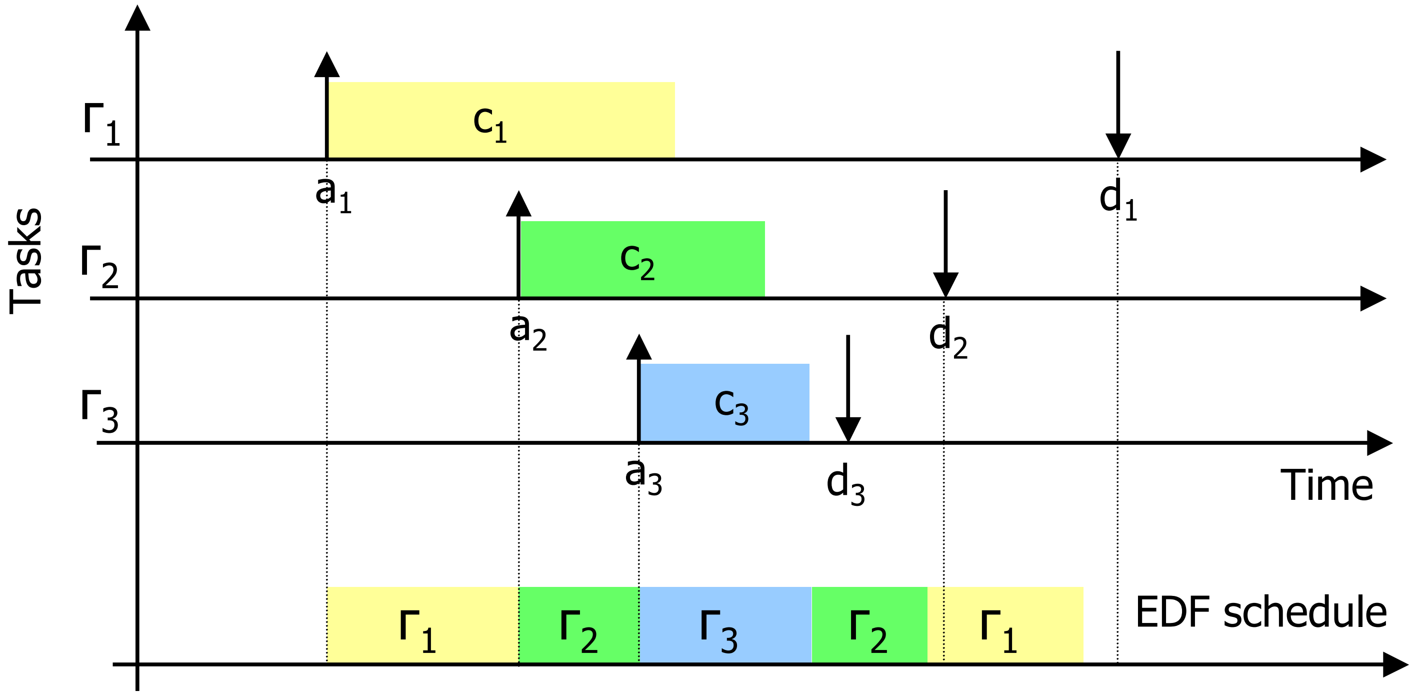
- At time t_i scheduler decides
  - Next task that will run on the processor, τ_i
  - Optimum operating voltage and frequency
- Maximize energy savings and real-time efficiency metric

Real-Time Metrics

Metric	Cost Function
Average response time	$\bar{t}_r = \frac{1}{N}\sum_{i=1}^{N}(f_i - a_i)$
Total completion time	$t_c = \max(f_i) - \min(a_i)$
Weighted completion time	$\bar{t}_w = \sum_{i=1}^{N} w_i f_i$
Maximum lateness	$L_{\max} = \max(f_i - d_i)$
Number of late tasks	$N_{late} = \sum_{i=1}^{N} miss(f_i) \quad miss(f_i) = \begin{cases} 0 & f_i \le d_i \\ 1 & other \end{cases}$

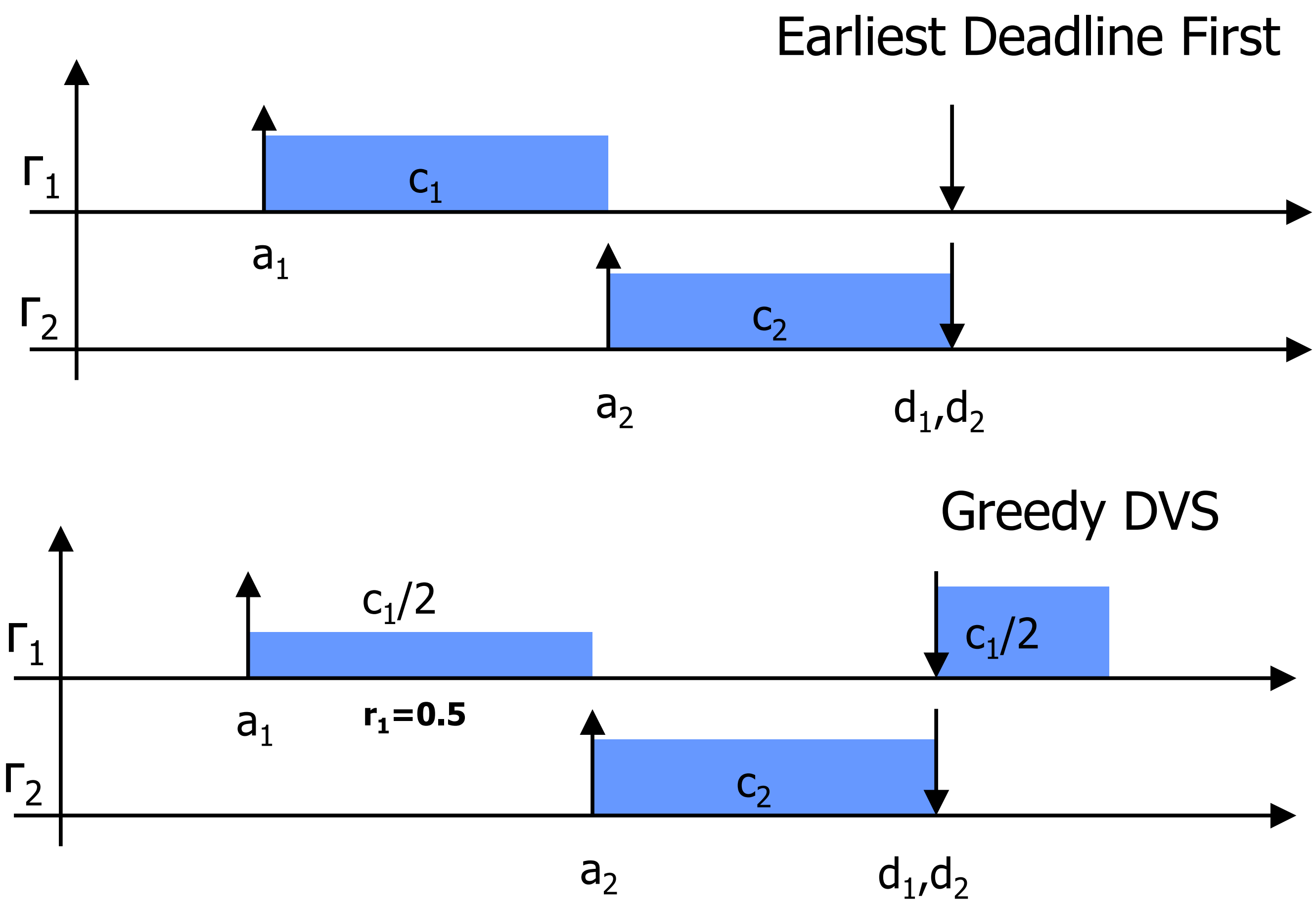
- L_{max} appropriate metric for hard real-time algorithms

Earliest Deadline First



- EDF optimal in minimizing L_{max} amongst all possible dynamic priority algorithms [Dertouzos]

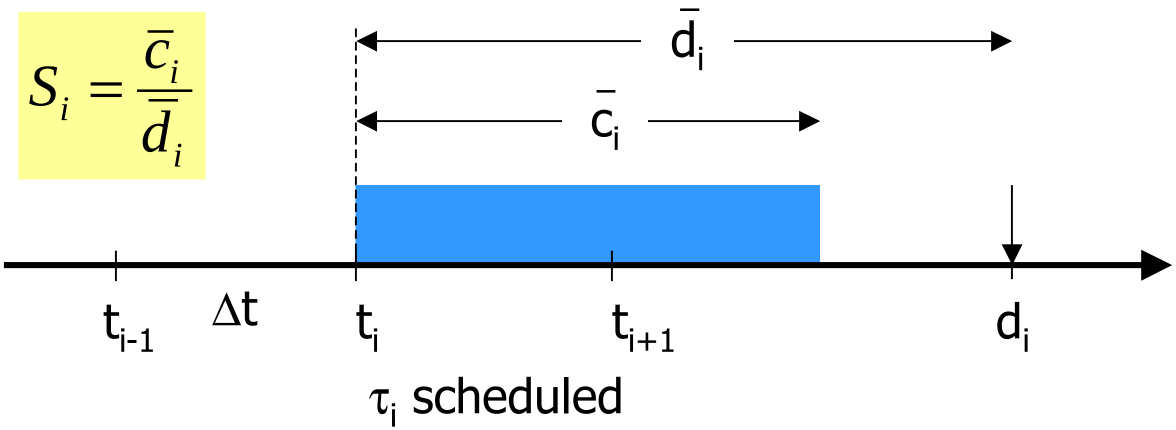
DVS with Real-Time Deadlines



- Need an intelligent scheduling algorithm

Proposed Optimal Scheduling

$$S_i = \frac{\bar{c}_i}{\bar{d}_i}$$



Optimum rate, r

$$\max\{P(r)\cdot E_{save}(r)\}$$

- The Slacked Earliest Deadline First (SEDF) algorithm
- Optimum processing rate is approximated by

$$r_i(S_i, U_i) = \begin{cases} S_i + (1 - S_i)U_i, & 0 < S_i \le 1 \\ 1, & otherwise \end{cases}$$

SEDF Analysis

- Probability that task τ_i completes before its deadline for a given slack, r, and utilization U_i

$$P(r) = \sum_{k=\left\lceil \frac{\bar{c}_i}{r} \right\rceil}^{\bar{d}_i} \binom{\bar{d}_i}{k} (1-U_i)^k U_i^{\bar{d}_i - k}$$

- Expected energy savings

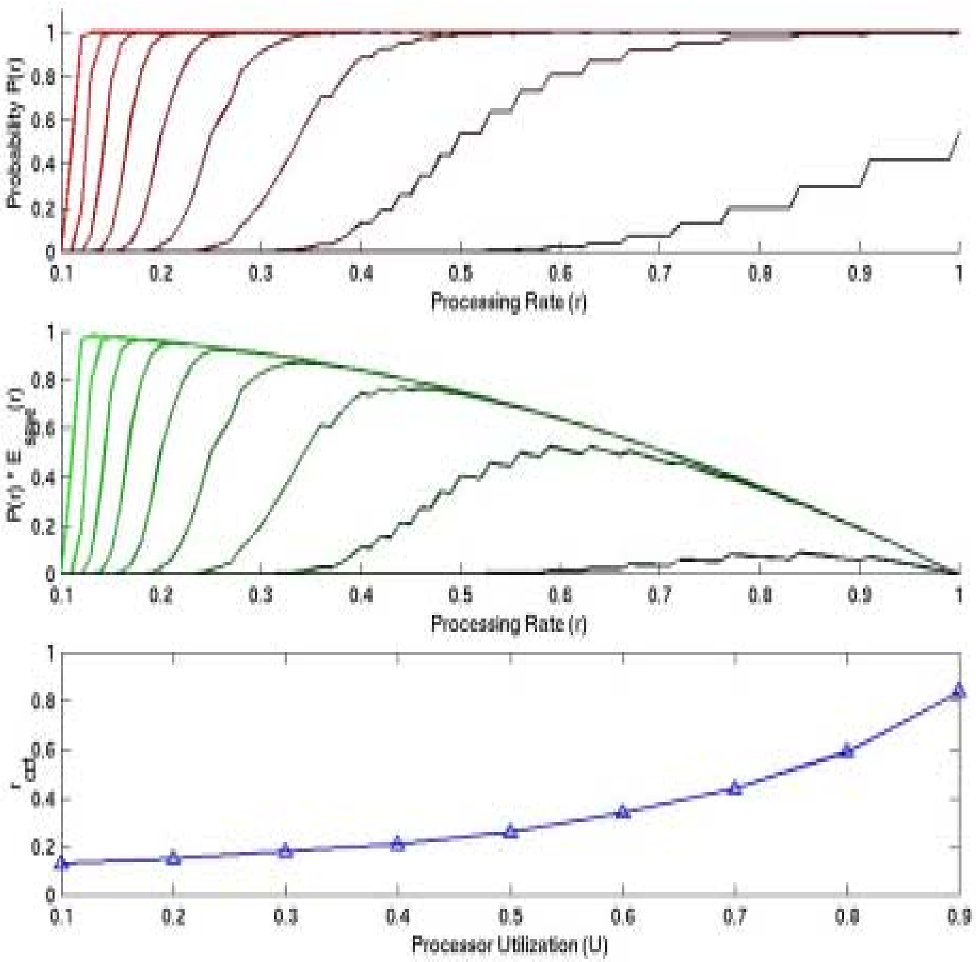
$$\xi(r) = P(r)\left(1 - r^2\right)$$

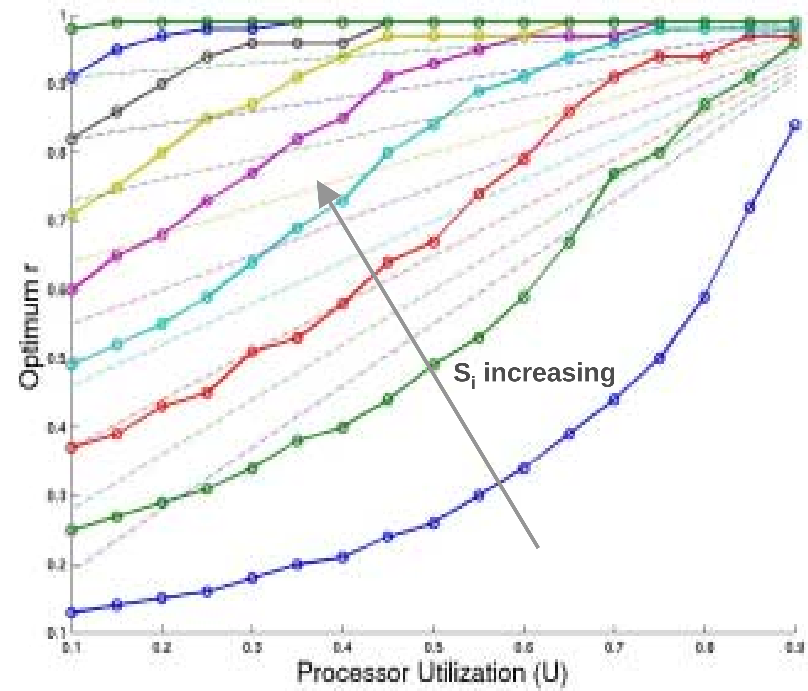
- Optimal slack, r_{opt}

$$\frac{\partial \xi(r)}{\partial r} = 0 \quad \Rightarrow \quad \frac{2r}{1-r^2} = \frac{P'(r)}{P(r)}$$

- Optimal slack well approximated by linear solution

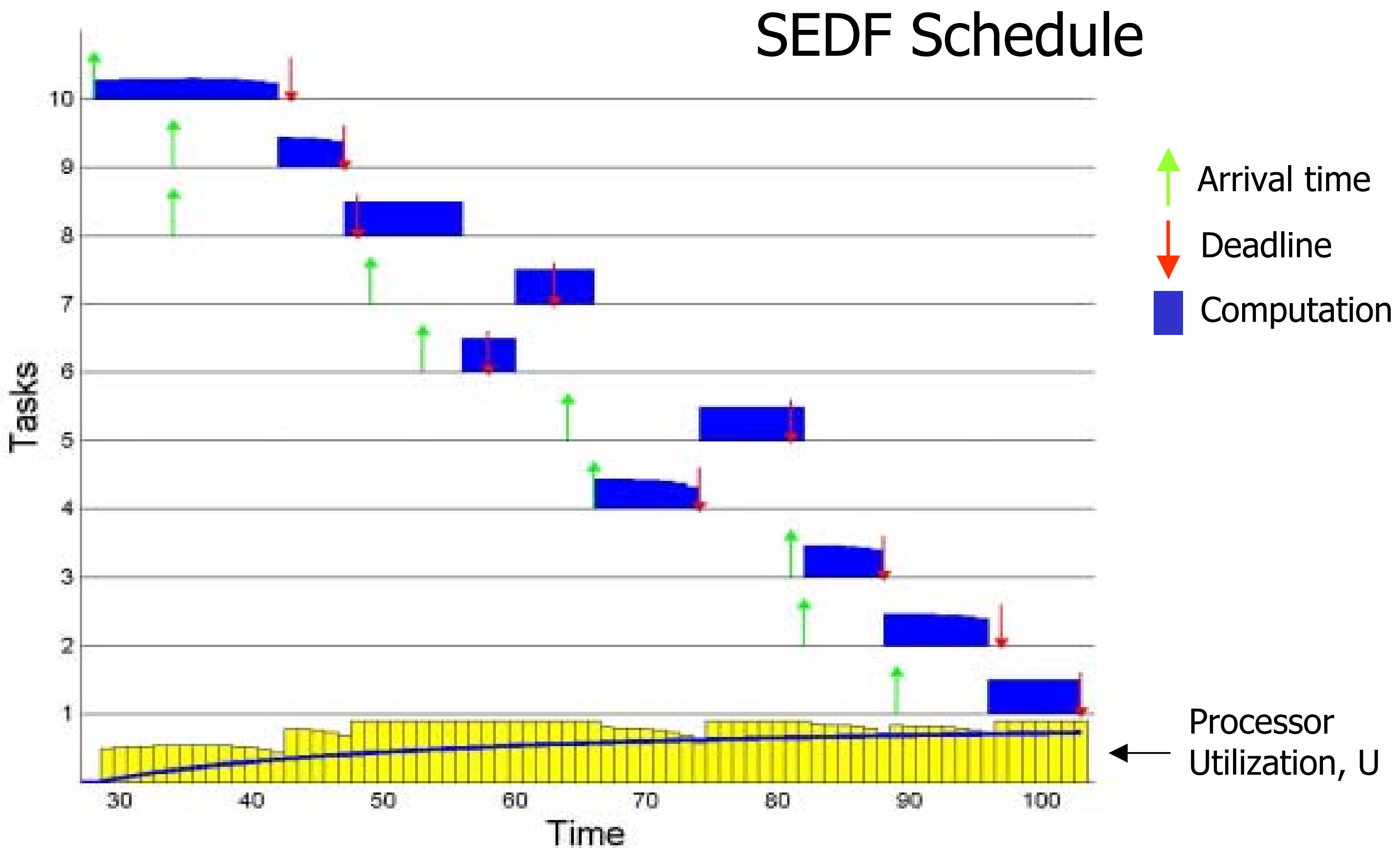
Why SEDF is Optimal





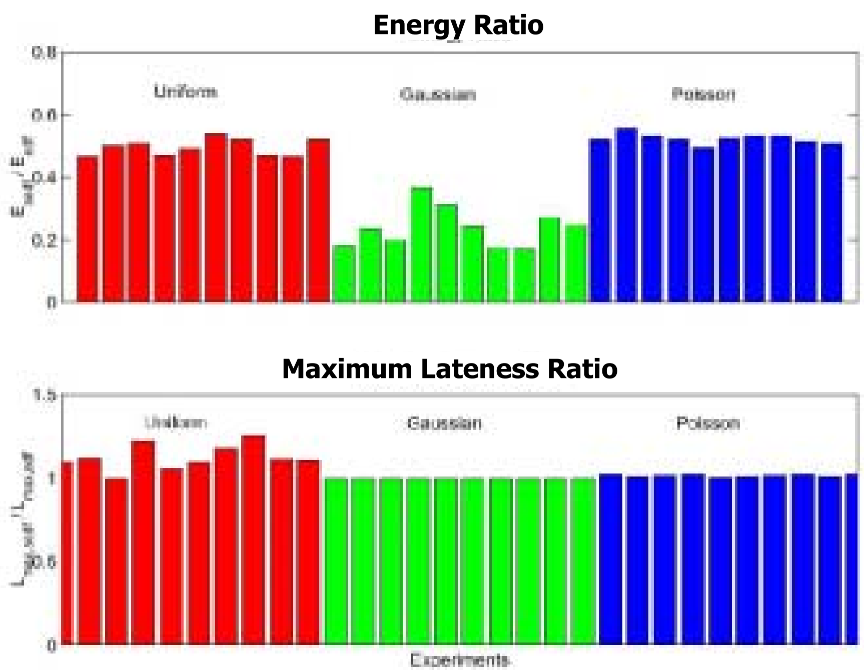
- Maximizes expected energy savings

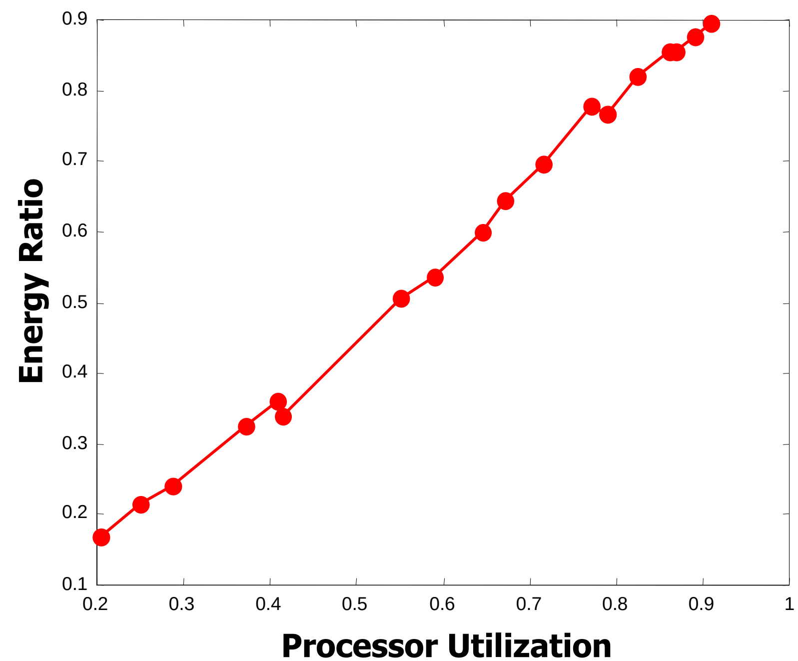
SEDF Scheduling Example



- SEDF is optimal in minimizing maximum lateness (L_{max}) and processor energy

SEDF Results



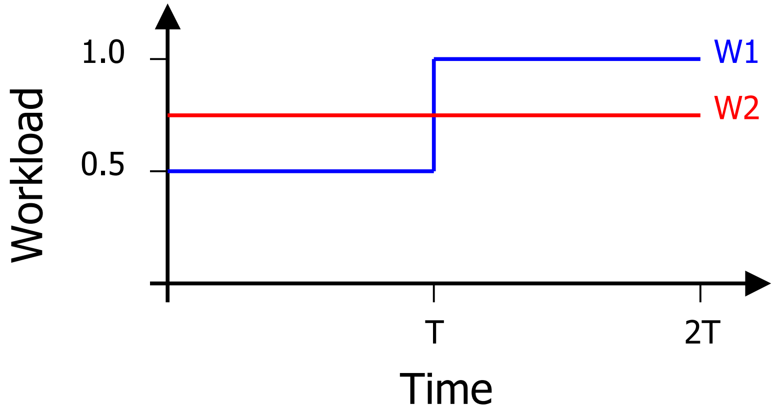


- Energy savings of 60% with 10% L_{max} degradation (averaged over $3x10^6$ experiments)
- SEDF approaches EDF as utilization increases

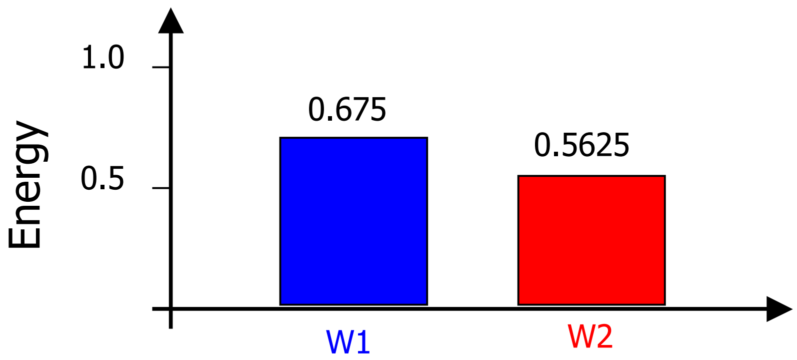
Bounds on Energy Savings

- Averaging is energy efficient because of convex E(r)

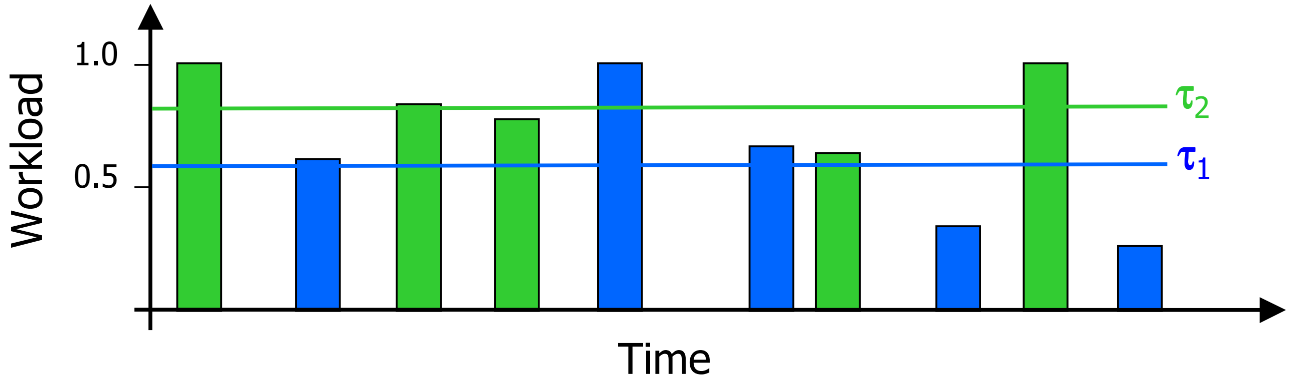
$$\frac{r_1^2 + r_2^2}{2} \geq \left(\frac{r_1 + r_2}{2}\right)^2 \rightarrow \overline{E(r)} \geq E(\bar{r})$$



- Best schedule tries to
  - Minimize workload variance
  - Maximize utilization
  - Use all possible slack



Bounds on Energy Savings [cont ..]



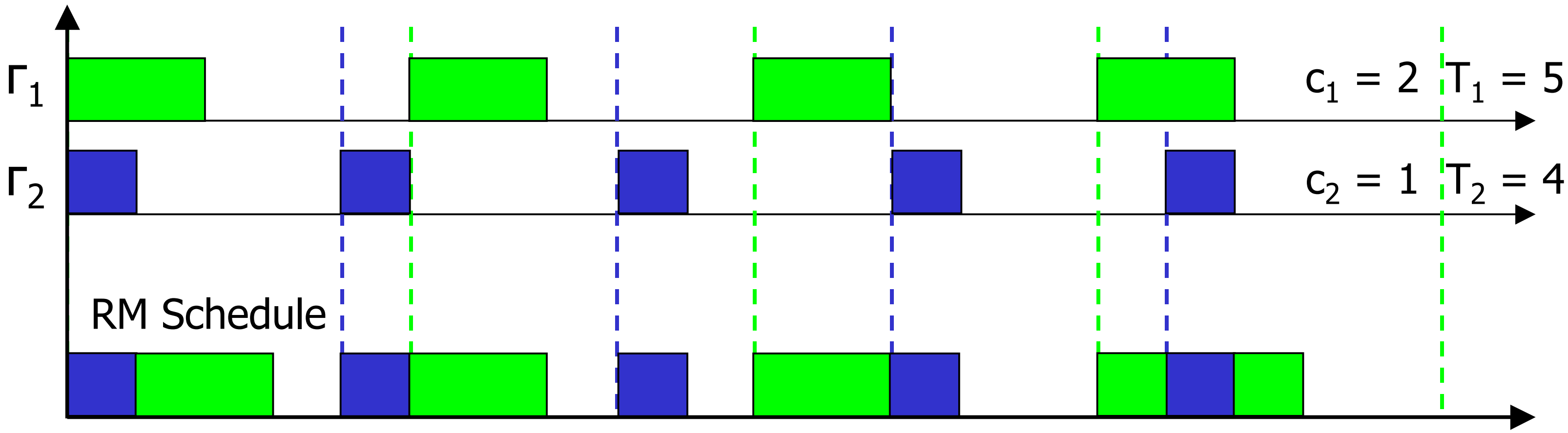
$$\left(\sum_k \frac{c_k}{\bar{r}_k}\right)\left(\sum_k c_k \bar{r}_k\right) \geq \left(\sum_k c_k\right)^2$$

$$\sum_k \frac{c_k}{\bar{r}_k} E(\bar{r}_k) = \sum_k c_k \bar{r}_k \geq \sum_k c_k r_{\min}$$

$$\Rightarrow \quad r_{\min} = \frac{\sum_k c_k}{\max(d_k)}$$

- Maximum energy savings $E_{save,\max} = 1 - r_{\min}^2$

Periodic Scheduling



- Rate Monotonic Analysis (RMA) [Liu73]

$$\sum_i \frac{c_i}{T_i} \le N\left(2^{\frac{1}{N}} - 1\right)$$

Guaranteed schedulability criteria
Smaller period tasks have higher priority

$$\frac{c_1}{T_1} + \frac{c_2}{T_2} = 0.65 \le 2\left(2^{\frac{1}{2}} - 1\right) = 0.83$$

Example

Slacked Rate Monotonic Analysis

- Slack processing rate, r, till utilization approaches RM bound

$$r_{\min} = \frac{\sum_i \frac{C_i}{T_i}}{N\left(2^{\frac{1}{N}} - 1\right)}$$

$T_i \rightarrow$ Period
$N \rightarrow$ Total Tasks
$C_i \rightarrow$ Computation Time

- A set of periodic real-time tasks is guaranteed to be schedulable with maximum energy savings iff processing rate is set to $r_{\min}$

Conclusions

- Dynamic voltage and frequency scheduling can yield quadratic energy savings
- Slacked Earliest Deadline First (SEDF) algorithm is optimal in minimizing expected energy consumption under L_{max} criteria
- Optimal slacking possible under Rate Monotonic scheduling for static periodic task sets