Performance-Aware Scheduler Synthesis for Control Systems

Rupak Majumdar[1,2], Indranil Saha[1] and Majid Zamani[1]

[1]University of California, Los Angeles

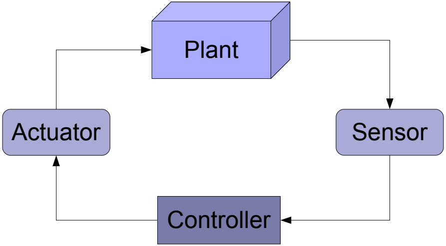
[2]Max Planck Institute for Software Systems

EMSOFT 2011

October 12, 2011

Mathematical Model of a Control System

$$x(k+1) = f\,(x(k), u(k), w(k))$$
$$y(k) = h\,(x(k))$$



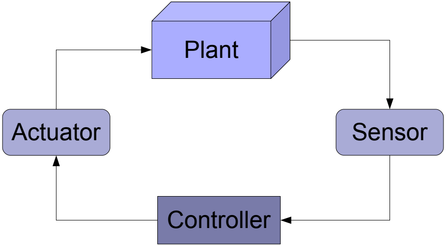
$$u(k) = \kappa(x(k))$$

For linear time-invariant control systems:

$$Plant : x(k+1) = Ax(k) + B_1 w(k) + B_2 u(k)$$
$$y(k) = Cx(k)$$
$$Controller : u(k) = -Kx(k)$$

Mathematical Model of a Control System

$$x(k+1) = f(x(k), u(k), w(k))$$
$$y(k) = h(x(k))$$

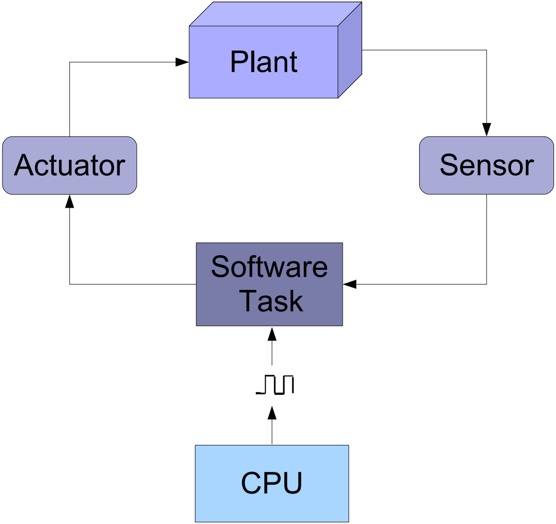


$$u(k) = \kappa(x(k))$$

For linear time-invariant control systems:

$$Plant : x(k+1) = Ax(k) + B_1 w(k) + B_2 u(k)$$
$$y(k) = Cx(k)$$
$$Controller : u(k) = -Kx(k)$$

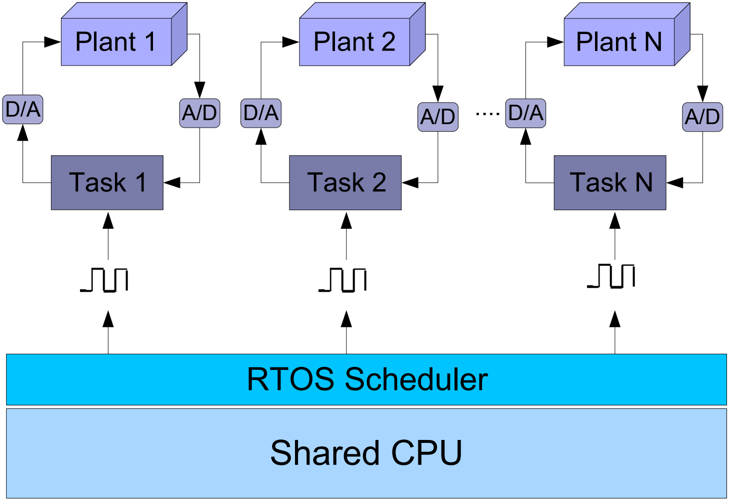
Controller to Software Task



Complex Control Systems

- Today's large control systems have many control units.
  - Boeing 747 has 50 ECUs.
  - BMW has 70-100 ECUs.
- Multiple control loops need to be implemented on a single processor.
  - Helps moving from federated architecture to integrated architecture .
  - Reduces cost.
  - Reduces communication complexity.

Multiple Control Systems with Shared Resources



Multiple Control Systems with Shared Resources

Control: $u_1 = f_1(x)$ Timing: τ_1	Control: $u_2 = f_2(x)$ Timing: τ_2	. . .	Control: $u_k = f_k(x)$ Timing: τ_k

Virtual World: Control Theory

Real World: Real-time OS

Tasks: T_1: *Period* $= \tau_1$, *WCET* $= c_1$ T_2: *Period* $= \tau_2$, *WCET* $= c_2$ . . . . . . T_k: *Period* $= \tau_k$, *WCET* $= c_k$

Schedulable? Schedule tasks

Hard Real-Time Scheduling

- Given tasks with worst case execution times and periods, is there a way to execute them so that all tasks finish executing before their deadlines?
- Key problem in real-time systems.
- System schedulable $\rightarrow$ Implement!

  System not schedulable $\rightarrow$ Send back to designer.

  Or: Throw more resources at it!

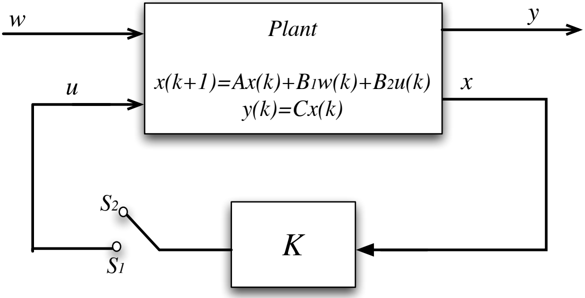
Not-So-Hard Real-Time Scheduling

- Suppose we relax the scheduler:
  - In some rounds, the scheduler can decide not to execute a task.
  - The control input generated in the previous cycle is applied to the plant.
  - Scheduling problem is easier.
- But what happens to the controlled system?
  - If we ignore a control task too many times, the system may become unstable.
  - Even if the system is stable, what happens to the performance?

Not-So-Hard Real-Time Scheduling

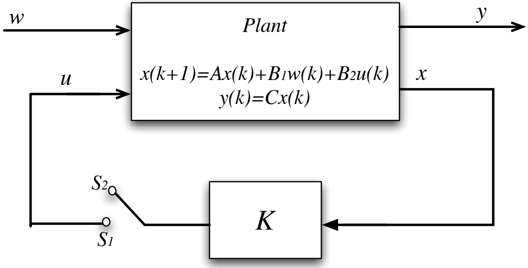
- Suppose we relax the scheduler:
  - In some rounds, the scheduler can decide not to execute a task.
  - The control input generated in the previous cycle is applied to the plant.
  - Scheduling problem is easier.
- But what happens to the controlled system?
  - If we ignore a control task too many times, the system may become unstable.
  - Even if the system is stable, what happens to the performance?

Model of a Control System with Packet Dropouts



When switch is in position S_1 : $u(k) = -Kx(k)$.
When switch is in position S_2 : $u(k) = u(k-1)$.

Successful Transmission Rate



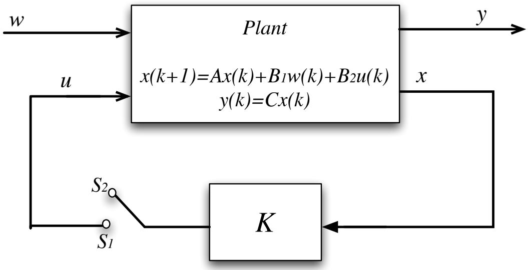
The successful transmission rate is the rate at which the switch is in position S_1. The *successful transmission rate r* is given by

$$r = \lim_{L \to \infty} \frac{1}{L} \sum_{k=0}^{L} (2 - s(k)).$$

The dropout rate means the rate at which the switch is in S_2.

If the successful transmission rate is r, its dropout rate is $1 - r$.

Successful Transmission Rate



The successful transmission rate is the rate at which the switch is in position S_1. The *successful transmission rate* r is given by

$$r = \lim_{L\to\infty} \frac{1}{L} \sum_{k=0}^{L} (2 - s(k)).$$

The dropout rate means the rate at which the switch is in S_2.

If the successful transmission rate is r, its dropout rate is $1 - r$.

Relate Successful Transmission Rate to Stability

Theorem [Branicky et al.-CDC'02]

Consider the control system with packet loss:

$$\begin{aligned} \text{Plant} \quad &: \quad x[k+1] = Ax[k] + B_1 w[k] \\ &\qquad\qquad + B_2(\text{drop?}u[k-1] : u[k]) \\ \text{Controller} \quad &: \quad u[k] = Kx[k] \end{aligned}$$

Assume that r is the successful transmission rate and the closed loop system with no dropout and no disturbance is stable.

The LTI control system with dropout, with no disturbance, is exponentially stable for all $r > r_{min}$

$$\begin{aligned} \text{where} \quad r_{\min} &= \frac{1}{1-\gamma_1/\gamma_2}, \\ \gamma_1 &= \log\left[\max_i |\lambda_i(A - B_2 K)|\right], \\ \text{and} \quad \gamma_2 &= \log\left[\max_i |\lambda_i(A)|\right]. \end{aligned}$$

$\lambda_i(A)$ is the i-th eigen-value of matrix A.

Performance Criteria: $\mathcal{L}_\infty$ to RMS Gain

For a discrete-time LTI control system, the $\mathcal{L}_\infty$ to RMS induced gain from disturbance w to output y [Hassibi et al.-CDC'99] is defined as follows:

$$\sup_{\|w\|_\infty \neq 0, X(0)=0} \frac{\left(\limsup_{l\to\infty} \frac{1}{l}\sum_{j=0}^{l} y^T(j)y(j)\right)^{\frac{1}{2}}}{\|w\|_\infty}$$

$$\text{where} \quad \|w\|_\infty = \sup\{\|w(k)\|_2,\ k \geq 0\},$$

$$\text{and} \quad \|w(k)\|_2 = \sqrt{w^T(k)w(k)}.$$

The $\mathcal{L}_\infty$ to RMS induced gain is a performance criterion showing the effect of the disturbance on the output of the plants.

The Lower is the gain, the better is the performance.

Performance Criteria: $\mathcal{L}_\infty$ to RMS Gain

For a discrete-time LTI control system, the $\mathcal{L}_\infty$ to RMS induced gain from disturbance w to output y [Hassibi et al.-CDC'99] is defined as follows:

$$\sup_{\|w\|_\infty \neq 0, X(0)=0} \frac{\left(\limsup_{l\to\infty} \frac{1}{l}\sum_{j=0}^{l} y^T(j)y(j)\right)^{\frac{1}{2}}}{\|w\|_\infty}$$

where $\|w\|_\infty = \sup\{\|w(k)\|_2,\ k \geq 0\}$,

and $\|w(k)\|_2 = \sqrt{w^T(k)w(k)}$.

The $\mathcal{L}_\infty$ to RMS induced gain is a performance criterion showing the effect of the disturbance on the output of the plants.

The Lower is the gain, the better is the performance.

Performance Criteria: $\mathcal{L}_\infty$ to RMS Gain

For a discrete-time LTI control system, the $\mathcal{L}_\infty$ to RMS induced gain from disturbance w to output y [Hassibi et al.-CDC'99] is defined as follows:

$$\sup_{\|w\|_\infty \neq 0, X(0)=0} \frac{\left(\limsup_{l\to\infty} \frac{1}{l}\sum_{j=0}^{l} y^T(j)y(j)\right)^{\frac{1}{2}}}{\|w\|_\infty}$$

$$\text{where} \quad \|w\|_\infty = \sup\{\|w(k)\|_2,\ k \geq 0\},$$

$$\text{and} \quad \|w(k)\|_2 = \sqrt{w^T(k)w(k)}.$$

The $\mathcal{L}_\infty$ to RMS induced gain is a performance criterion showing the effect of the disturbance on the output of the plants.

The Lower is the gain, the better is the performance.

Relate Successful Transmission Rate to Performance

Theorem

Consider the discrete time LTI control system with the successful transmission rate r. The $\mathcal{L}_\infty$ to RMS gain is less than positive constant γ if there exists a piecewise continuous function $V : \mathbb{R}^{n+m} \rightarrow \mathbb{R}_{\geq 0}$(n and m are dimensions of state space and control input set respectively), such that $V(0) = 0$, and $\gamma_1, \gamma_2 \in \mathbb{R}$ such that

$$r\gamma_1^2 + (1 - r)\gamma_2^2 < \gamma^2$$

and

$$V\left(\widetilde{A}_i X + \widetilde{B}_{1i} w\right) - V(X) \leq \gamma_i^2 w^T w - y^T y, \text{ for } i = 1, 2.$$

We can find upper bound on the gain for different successful transmission rates through convex optimization.

Relate Successful Transmission Rate to Performance

Theorem

Consider the discrete time LTI control system with the successful transmission rate r. The $\mathcal{L}_\infty$ to RMS gain is less than positive constant γ if there exists a piecewise continuous function $V : \mathbb{R}^{n+m} \rightarrow \mathbb{R}_{\geq 0}$(n and m are dimensions of state space and control input set respectively), such that $V(0) = 0$, and $\gamma_1, \gamma_2 \in \mathbb{R}$ such that

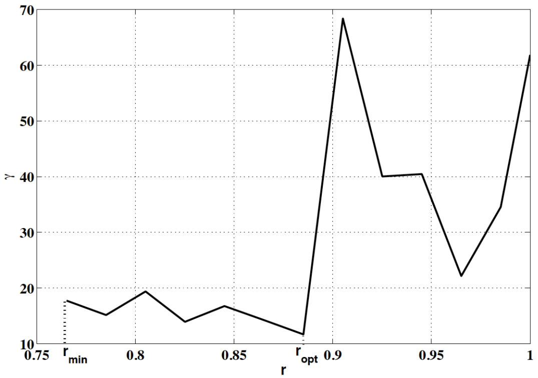
$$r\gamma_1^2 + (1 - r)\gamma_2^2 < \gamma^2$$

and

$$V\left(\widetilde{A}_i X + \widetilde{B}_{1i} w\right) - V(X) \leq \gamma_i^2 w^T w - y^T y, \text{ for } i = 1, 2.$$

We can find upper bound on the gain for different successful transmission rates through convex optimization.

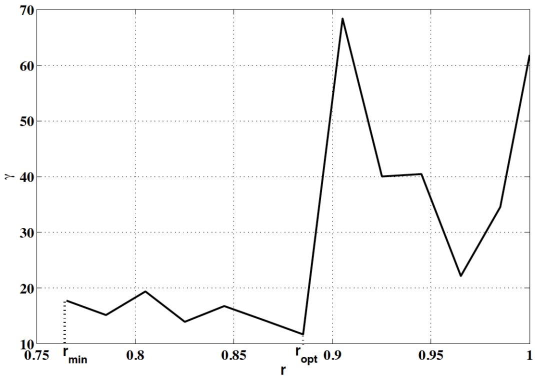
- Performance is Not monotonic with respect to successful transmission rate.
  → Increasing resources may not make the performance better.



Moral: An end-to-end argument can give a better overall system performance, even with lower resources.

Performance vs. Successful Transmission Rates

- Performance is Not monotonic with respect to successful transmission rate.
  → Increasing resources may not make the performance better.



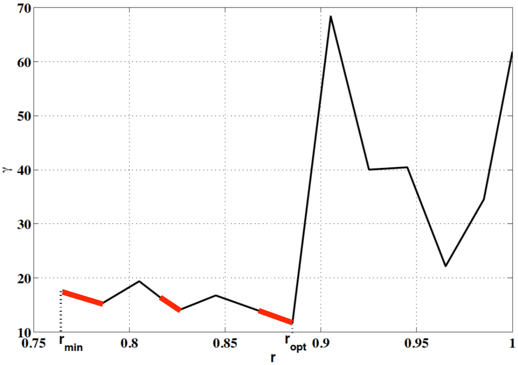
Moral: An end-to-end argument can give a better overall system performance, even with lower resources.

Eligible Successful Transmission Rates

A successful transmission rate r is called *eligible* if it satisfies the following two conditions:

- $r \geq r_{\text{min}}$, where r_{min} is the minimum rate to achieve stability.
- for each $r' \in [r_{\text{min}}, r)$, we have $\gamma(r') \geq \gamma(r)$.

$\gamma(r)$ denote the upper bound on the $\mathcal{L}_\infty$ to RMS gain.



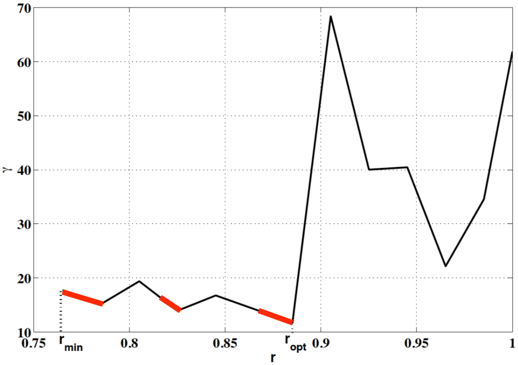
For a chosen discretization for r, the set of eligible rates are denoted by E_i for control system i.

Eligible Successful Transmission Rates

A successful transmission rate r is called *eligible* if it satisfies the following two conditions:

- $r \geq r_{\min}$, where $r_{\min}$ is the minimum rate to achieve stability.
- for each $r' \in [r_{\min}, r)$, we have $\gamma(r') \geq \gamma(r)$.

$\gamma(r)$ denote the upper bound on the $\mathcal{L}_\infty$ to RMS gain.



For a chosen discretization for r, the set of eligible rates are denoted by E_i for control system i.

Optimal Performance Scheduler Synthesis Problem

Choose rates $r_i \in E_i$ such that the system is schedulable and the weighted sum $w_i\gamma(r_i)$ is minimized.

w_i's are weights chosen based on the priority of the control systems.

Formally,

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{N} w_i\gamma(r_i) & \\ \text{such that} & r_i \in E_i & \text{for each } i \in \{1, \ldots, N\} \\ & \sum_{i=1}^{N} c_i r_i/\tau_i \leq 1 & \end{array}$$

The problem is NP-Hard.

- Reduction is from Multiple-Choice Knapsack Problem.

Optimal Performance Scheduler Synthesis Problem

Choose rates $r_i \in E_i$ such that the system is schedulable and the weighted sum $w_i \gamma(r_i)$ is minimized.

w_i's are weights chosen based on the priority of the control systems.

Formally,

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{N} w_i \gamma(r_i) & \\ \text{such that} & r_i \in E_i & \text{for each } i \in \{1, \ldots, N\} \\ & \sum_{i=1}^{N} c_i r_i / \tau_i \leq 1 & \end{array}$$

The problem is NP-Hard.

- Reduction is from Multiple-Choice Knapsack Problem.

Optimal Performance Scheduler Synthesis Problem

Choose rates $r_i \in E_i$ such that the system is schedulable and the weighted sum $w_i\gamma(r_i)$ is minimized.

w_i's are weights chosen based on the priority of the control systems.

Formally,

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{N} w_i\gamma(r_i) & \\ \text{such that} & r_i \in E_i & \text{for each } i \in \{1, \ldots, N\} \\ & \sum_{i=1}^{N} c_i r_i/\tau_i \leq 1 & \end{array}$$

The problem is NP-Hard.

- Reduction is from Multiple-Choice Knapsack Problem.

Our Approach

- Find r_{min} for each control system.
- Find r_{max} for all control systems.
  - Maximize weighted sum of successful transmission rates.
  - Weights are based on the priorities of the control systems.
- Select $r \in [r_{min}, r_{max}]$ such that the performance is the best.
- Synthesize a scheduler based on the selected rates.

Scheduler Synthesis with Task Drops

- Given: Task T_i:
  - WCET c_i.
  - Period τ_i.
  - Successful transmission rate $r_i = \frac{k_i}{K_i}$
- Find: Schedule such that
  - Executions of Task i finish before the deadline.
  - The scheduler drops $1 - r(i)$ fraction of packets in the long run.

Static Scheduling Problem in SMT

We encode constraints as an SMT problem:

- Hyperperiod = {lcm of periods of all tasks (τ_i's)} $\times$ {lcm of the denominators of the rates (K_i's)}.
- Boolean variable $s[i, j]$: if task i is scheduled in round j.

  - If $s[i, j] = 1$, then wcet c_i slots in the j'th period allocated to task i.
- Variable $t[i, j]$: time when task of controller i in the j'th period starts.
- Fraction of the number of periods in the hyperperiod in which task i is chosen = $r(i)$.
- A task should be scheduled after it is generated and should finish before the end of the period.
- A slot should not be assigned to multiple tasks.

Finding the Maximal Transmission Rates

Problem: What is the maximum successful transmission rates for the control systems such that all packets can be scheduled?

Solution: Solving a maximization problem.

- Constraints are same as the previous problem, only the rates are treated as variables.
- The objective function is the weighted sum of the rate variables.
  - Weights are derived from priorities.
- Solve the optimization problem using bisection method and solving a series of feasibility problems.

Example: Inverted Pendulum

$$\dot{x} = Ax + B_1 w + B_2 u$$
$$y = Cx$$

$$A = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} & \frac{\rho}{ml^2} \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 \\ \frac{1}{ml} \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} \quad C = [0.001,\ 0].$$

- x_1 - the angular position
- x_2 - the angular velocity of the point mass
- u - the applied force (control input)
- w - the disturbance input
- m - the mass
- l - the length of the rod
- g - acceleration due to gravity
- ρ - the rotational friction coefficient

Systems	Mass (kg)	Length(m)	Priority	Controller Gain	Sampling Time (s)	Computation Time (s)
System 1	0.50	0.20	1	[5.1, -2.5]	0.010	0.005
System 2	0.50	0.35	2	[5.25, -1.1893]	0.015	0.005
System 3	0.50	0.50	3	[5.4, -0.45]	0.020	0.005

$r_{\min,1} = 0.7651$, $r_{\min,2} = 0.6375$, and $r_{\min,3} = 0.6589$

$r_{\max,1} = 1.00$, $r_{\max,2} = 0.90$, and $r_{\max,3} = 0.70$

$r_{opt,1} = 0.85$, $r_{opt,2} = 0.85$, and $r_{opt,3} = 0.70$

Example: Inverted Pendulum

$$\dot{x} = Ax + B_1 w + B_2 u$$
$$y = Cx$$

$$A = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} & \frac{\rho}{ml^2} \end{bmatrix} \qquad B_2 = \begin{bmatrix} 0 \\ \frac{1}{ml} \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} \qquad C = [0.001,\ 0].$$

- x_1 - the angular position
- x_2 - the angular velocity of the point mass
- u - the applied force (control input)
- w - the disturbance input
- m - the mass
- l - the length of the rod
- g - acceleration due to gravity
- ρ - the rotational friction coefficient

Systems	Mass (kg)	Length(m)	Priority	Controller Gain	Sampling Time (s)	Computation Time (s)
System 1	0.50	0.20	1	[5.1, -2.5]	0.010	0.005
System 2	0.50	0.35	2	[5.25, -1.1893]	0.015	0.005
System 3	0.50	0.50	3	[5.4, -0.45]	0.020	0.005

$r_{\min,1} = 0.7651$, $r_{\min,2} = 0.6375$, and $r_{\min,3} = 0.6589$

$r_{\max,1} = 1.00$, $r_{\max,2} = 0.90$, and $r_{\max,3} = 0.70$

$r_{opt,1} = 0.85$, $r_{opt,2} = 0.85$, and $r_{opt,3} = 0.70$

Example: Inverted Pendulum

$$\dot{x} = Ax + B_1 w + B_2 u$$
$$y = Cx$$

$$A = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} & \frac{\rho}{ml^2} \end{bmatrix} \qquad B_2 = \begin{bmatrix} 0 \\ \frac{1}{ml} \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} \qquad C = [0.001,\ 0].$$

- x_1 - the angular position
- x_2 - the angular velocity of the point mass
- u - the applied force (control input)
- w - the disturbance input
- m - the mass
- l - the length of the rod
- g - acceleration due to gravity
- ρ - the rotational friction coefficient

Systems	Mass (kg)	Length(m)	Priority	Controller Gain	Sampling Time (s)	Computation Time (s)
System 1	0.50	0.20	1	[5.1, -2.5]	0.010	0.005
System 2	0.50	0.35	2	[5.25, -1.1893]	0.015	0.005
System 3	0.50	0.50	3	[5.4, -0.45]	0.020	0.005

$r_{\min,1} = 0.7651$, $r_{\min,2} = 0.6375$, and $r_{\min,3} = 0.6589$

$r_{\max,1} = 1.00$, $r_{\max,2} = 0.90$, and $r_{\max,3} = 0.70$

$r_{opt,1} = 0.85$, $r_{opt,2} = 0.85$, and $r_{opt,3} = 0.70$

Scalability

Number of Pendulums	Sampling Time	Computation Time	t_m	t_s
3	10ms	5ms	3.548s	0.313s
4	15ms	5ms	5.948s	0.591s
5	20ms	5ms	1m34.576s	1.003s
6	25ms	5ms	5m20.364s	1.702s
7	30ms	5ms	11m5.501s	5.945s
8	35ms	5ms	12m39.703s	3.026s
9	40ms	5ms	25m10.479s	5.123s
10	45ms	5ms	11m0.143s	6.485s

Table: Time required to find maximal schedule and optimal schedule

t_m - time required to find the maximal successful transmission rates.
t_s - time required to find the final schedule.

Related Works

- Co-design of feedback controllers and schedulers
  - Choose the sampling time to obtain optimal performance [Seto et al.-RTSS'96, Årzén et al.-CDC'00, Zhang et al.-RTSS'08, and others]
  - Drop some control packets to make the scheduling problem easier without compromising control properties (focus is on Stability) [Branicky et al.-CDC'02, Goswami et al.-ASP-DAC'11]
- Marriage of control theoretic calculation and software verification/synthesis
  - Schedulability and stability [Weiss et al. - HSCC'09]
  - Fixed-point implementation of controller and stability [Anta et al. - EMSOFT'10]

Conclusion

Contributions:

- We present theoretical results as well as a tool for a Controller-Scheduler Co-design problem.
- Co-design lets us relax constraints on the hard real-time scheduling problem, while potentially getting better performance from the system.

Future Work:

- Techniques can be generalized with other sources of error, such as quantization errors or additional network effects.
- Explore how dynamic scheduling policies interact with our control-theoretic analysis.
- Extend our results to more complex hybrid systems with several discrete modes.

Conclusion

Contributions:

- We present theoretical results as well as a tool for a Controller-Scheduler Co-design problem.
- Co-design lets us relax constraints on the hard real-time scheduling problem, while potentially getting better performance from the system.

Future Work:

- Techniques can be generalized with other sources of error, such as quantization errors or additional network effects.
- Explore how dynamic scheduling policies interact with our control-theoretic analysis.
- Extend our results to more complex hybrid systems with several discrete modes.