Comparisons of stochastic task-resource systems

Bruno Gaujal   Jean-Marc Vincent

INRIA and LIG

Fréjus, 4 Juin 2007

Outline

Probabilistic models

The main object of this lecture is the task-resource model.

Probabilistic models

The main object of this lecture is the task-resource model.
Tasks are characterized by the arrival times that form a point process $0 \leqslant T_1 \leqslant T_2 \leqslant T_3 \leqslant \cdots$ over the positive real line and by the sizes of tasks that is a sequence of real numbers $S_1, S_2, \ldots$.

Probabilistic models

The main object of this lecture is the task-resource model.
Tasks are characterized by the arrival times that form a point process $0 \leqslant T_1 \leqslant T_2 \leqslant T_3 \leqslant \cdots$ over the positive real line and by the sizes of tasks that is a sequence of real numbers $S_1, S_2, \ldots$.

Resources are characterized by their number $K \in \mathbb{N} \cup \{+\infty\}$ and the respective speeds $v_1, \ldots, v_K$.

In the following the size of the tasks is often given is seconds (time for a resource of speed 1 to treat a task).

Probabilistic models

The main object of this lecture is the task-resource model.
Tasks are characterized by the arrival times that form a point process $0 \leqslant T_1 \leqslant T_2 \leqslant T_3 \leqslant \cdots$ over the positive real line and by the sizes of tasks that is a sequence of real numbers $S_1, S_2, \ldots$.

Resources are characterized by their number $K \in \mathbb{N} \cup \{+\infty\}$ and the respective speeds $v_1, \ldots, v_K$.

In the following the size of the tasks is often given is seconds (time for a resource of speed 1 to treat a task).

Additionally, tasks and ressources may be constrainted by dependencies, synchronizations, availability conditions, matchings, ...

Probabilistic models

The main object of this lecture is the task-resource model.
Tasks are characterized by the arrival times that form a point process $0 \leqslant T_1 \leqslant T_2 \leqslant T_3 \leqslant \cdots$ over the positive real line and by the sizes of tasks that is a sequence of real numbers $S_1, S_2, \ldots$.

Resources are characterized by their number $K \in \mathbb{N} \cup \{+\infty\}$ and the respective speeds $v_1, \ldots, v_K$.

In the following the size of the tasks is often given is seconds (time for a resource of speed 1 to treat a task).

Additionally, tasks and ressources may be constrainted by dependencies, synchronizations, availability conditions, matchings, . . .

Here we will mostly consider very simple systems with one common features : randomness. Basically, the arrival times and/or the task sizes will be random processes.

Stochastic Orders

There are many ways to compare two random variables (and random processes). The most obvious one is to compare the means : $X \leqslant_\mu Y$ if $E(X) \leqslant E(Y)$.

However, this order is rather crude and may not capture a lot of insight in the comparison of two stochastic systems, in particular in the scheduling context.

Stochastic Orders

There are many ways to compare two random variables (and random processes). The most obvious one is to compare the means : $X \leqslant_\mu Y$ if $E(X) \leqslant E(Y)$.

However, this order is rather crude and may not capture a lot of insight in the comparison of two stochastic systems, in particular in the scheduling context.

Consider a two task-resource systems with one resource. One has arrivals every 4 seconds of tasks of size 4 and the other has arrivals of tasks of size 2 at times $5n$ and $5n + 1$.

Stochastic Orders

There are many ways to compare two random variables (and random processes). The most obvious one is to compare the means : $X \leqslant_\mu Y$ if $E(X) \leqslant E(Y)$.

However, this order is rather crude and may not capture a lot of insight in the comparison of two stochastic systems, in particular in the scheduling context.

Consider a two task-resource systems with one resource. One has arrivals every 4 seconds of tasks of size 4 and the other has arrivals of tasks of size 2 at times $5n$ and $5n + 1$.
Then the expected input loads per second are $E(L_1) = 1 > E(L_2) = 4/5$ respectively.

Stochastic Orders

There are many ways to compare two random variables (and random processes). The most obvious one is to compare the means : $X \leqslant_\mu Y$ if $E(X) \leqslant E(Y)$.

However, this order is rather crude and may not capture a lot of insight in the comparison of two stochastic systems, in particular in the scheduling context.

Consider a two task-resource systems with one resource. One has arrivals every 4 seconds of tasks of size 4 and the other has arrivals of tasks of size 2 at times $5n$ and $5n+1$.
Then the expected input loads per second are $E(L_1) = 1 > E(L_2) = 4/5$ respectively.
As for the expected waiting times, $E(W_1) = 0 < E(W_2) = 1/2$.

Stochastic Orders

There are many ways to compare two random variables (and random processes). The most obvious one is to compare the means : $X \leqslant_\mu Y$ if $E(X) \leqslant E(Y)$.

However, this order is rather crude and may not capture a lot of insight in the comparison of two stochastic systems, in particular in the scheduling context.

Consider a two task-resource systems with one resource. One has arrivals every 4 seconds of tasks of size 4 and the other has arrivals of tasks of size 2 at times $5n$ and $5n + 1$.
Then the expected input loads per second are $E(L_1) = 1 > E(L_2) = 4/5$ respectively.
As for the expected waiting times, $E(W_1) = 0 < E(W_2) = 1/2$.
Even if waiting times are increasing functions of loads (see later), this is not the case for the μ order.

Stochastic Orders

There are many ways to compare two random variables (and random processes). The most obvious one is to compare the means : $X \leqslant_\mu Y$ if $E(X) \leqslant E(Y)$.

However, this order is rather crude and may not capture a lot of insight in the comparison of two stochastic systems, in particular in the scheduling context.

Consider a two task-resource systems with one resource. One has arrivals every 4 seconds of tasks of size 4 and the other has arrivals of tasks of size 2 at times $5n$ and $5n + 1$.
Then the expected input loads per second are $E(L_1) = 1 > E(L_2) = 4/5$ respectively.
As for the expected waiting times, $E(W_1) = 0 < E(W_2) = 1/2$.
Even if waiting times are increasing functions of loads (see later), this is not the case for the μ order.

There exists several stochastic orders (in the book *Comparison methods for stochastic models and risks* (Muller and Stoyan, 2002), 49 different orders are defined with different applications in mind.

The usual stochastic order

The usual stochastic order (also called the strong order) is defined as follows (for real random variables).
$X \leqslant_{st} Y$ if $F_X(a) = P(X \leqslant a) \geqslant P(Y \leqslant a) = F_Y(a)$ for all a.

The usual stochastic order

The usual stochastic order (also called the strong order) is defined as follows (for real random variables).
$X \leqslant_{st} Y$ if $F_X(a) = P(X \leqslant a) \geqslant P(Y \leqslant a) = F_Y(a)$ for all a.

The *st* order has several other characterizations :

- **Sample path definition** There exists two variables X' and Y' in (Ω, A, P) with the same distribution as X and Y such that $X(\omega) \leqslant Y(\omega)$ for each $\omega \in \Omega$.
- **Integral definition** For all increasing function f, $E(f(X)) \leqslant E(f(Y))$.

The usual stochastic order : examples

1. Show that $X \leqslant_{st} Y \Rightarrow X \leqslant_{\mu} Y$.

The usual stochastic order : examples

1. Show that $X \leqslant_{st} Y \Rightarrow X \leqslant_{\mu} Y$.
2. Show that $X \leqslant_{st} Y$ and $EX = EY \Rightarrow F_X = F_Y$.

The usual stochastic order : examples

1. Show that $X \leqslant_{st} Y \Rightarrow X \leqslant_{\mu} Y$.

2. Show that $X \leqslant_{st} Y$ and $EX = EY \Rightarrow F_X = F_Y$.

3. Compare the following integer random variables :

X = 1 w.p. 1/4, 2 w.p. 1/2, 3 w,.p. 1/4
Y = 2 w.p. 1/2 , 3 w.p. 1/2
Z = 1 w.p. 1/3 , 3 w.p. 2/3

Stronger stochastic orders

Some orders are stronger than *st* :

Consider the following case : somebody wants to buy a car and can choose between two models with lifetimes X and Y. If the price is the same and $X \leqslant_{st} Y$ them, she ought to buy model Y. Now what happens if both cars are used (one year old), is Y still a better choice ?

Stronger stochastic orders

Some orders are stronger than *st* :

Consider the following case : somebody wants to buy a car and can choose between two models with lifetimes X and Y. If the price is the same and $X \leqslant_{st} Y$ them, she ought to buy model Y. Now what happens if both cars are used (one year old), is Y still a better choice ?

well, not necessarily :

Stronger stochastic orders

Some orders are stronger than *st* :

Consider the following case : somebody wants to buy a car and can choose between two models with lifetimes X and Y. If the price is the same and $X \leqslant_{st} Y$ them, she ought to buy model Y. Now what happens if both cars are used (one year old), is Y still a better choice ?

well, not necessarily :

Assume that X is uniform over $[0, 3]$ (with db F) and Y has a distribution with density $1/6, 1/2, 1/3$ on $[0, 1],]1, 2],]2, 3]$ (with db G).

Stronger stochastic orders

Some orders are stronger than *st* :

Consider the following case : somebody wants to buy a car and can choose between two models with lifetimes X and Y. If the price is the same and $X \leqslant_{st} Y$ them, she ought to buy model Y. Now what happens if both cars are used (one year old), is Y still a better choice ?

well, not necessarily :

Assume that X is uniform over $[0, 3]$ (with db F) and Y has a distribution with density $1/6, 1/2, 1/3$ on $[0, 1],]1, 2],]2, 3]$ (with db G).

Then, $X \leqslant_{st} Y$ ($F \geqslant G$). However $X_1 = (X|X > 1)$ and $Y_1 = (Y|Y > 1)$ are not st-comparable : X_1 is uniform over $[0, 2]$ (with density 1/2) and Y_1 has a distribution with density $3/5, 2/5$, on $[0, 1],]1, 2]$.

Stronger stochastic orders

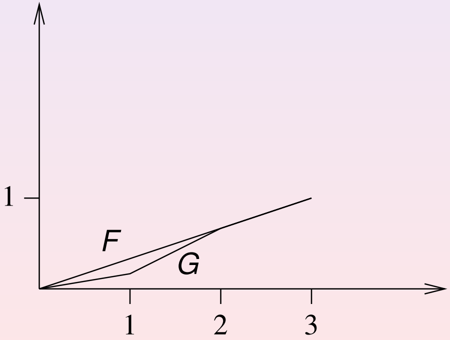
Some orders are stronger than *st* :

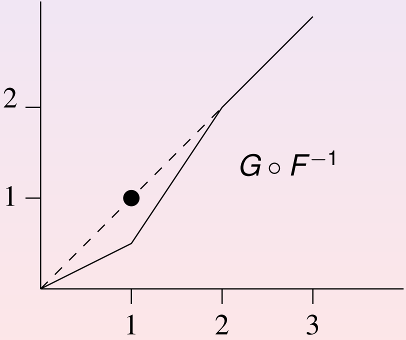
Consider the following case : somebody wants to buy a car and can choose between two models with lifetimes X and Y. If the price is the same and $X \leqslant_{st} Y$ them, she ought to buy model Y. Now what happens if both cars are used (one year old), is Y still a better choice ?

well, not necessarily :

Assume that X is uniform over $[0, 3]$ (with db F) and Y has a distribution with density $1/6, 1/2, 1/3$ on $[0, 1],]1, 2],]2, 3]$ (with db G).

Then, $X \leqslant_{st} Y$ ($F \geqslant G$). However $X_1 = (X|X > 1)$ and $Y_1 = (Y|Y > 1)$ are not st-comparable : X_1 is uniform over $[0, 2]$ (with density 1/2) and Y_1 has a distribution with density $3/5, 2/5$, on $[0, 1],]1, 2]$.





Stronger stochastic orders

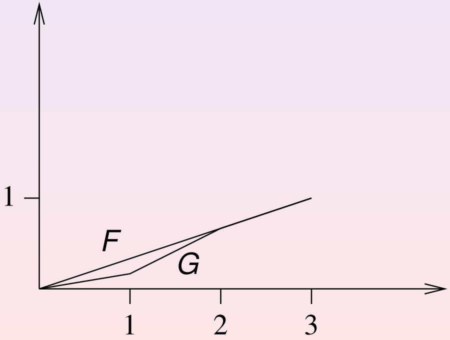
Some orders are stronger than *st* :

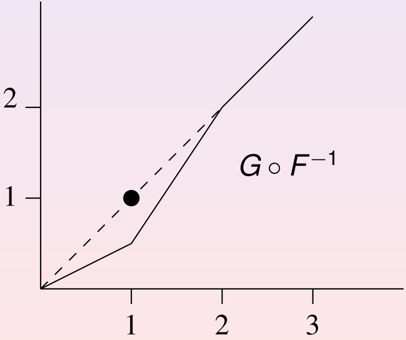
Consider the following case : somebody wants to buy a car and can choose between two models with lifetimes X and Y. If the price is the same and $X \leqslant_{st} Y$ them, she ought to buy model Y. Now what happens if both cars are used (one year old), is Y still a better choice ?

well, not necessarily :

Assume that X is uniform over $[0, 3]$ (with db F) and Y has a distribution with density $1/6, 1/2, 1/3$ on $[0, 1],]1, 2],]2, 3]$ (with db G).

Then, $X \leqslant_{st} Y$ ($F \geqslant G$). However $X_1 = (X|X > 1)$ and $Y_1 = (Y|Y > 1)$ are not st-comparable : X_1 is uniform over $[0, 2]$ (with density 1/2) and Y_1 has a distribution with density $3/5, 2/5$, on $[0, 1],]1, 2]$.





What order is preserved under aging ?

The hazard rate order : *hr*

The hazard rate (or failure rate) is defined by :

$$r_X(t) = \lim_{\varepsilon \to 0} \frac{P(X < t + \varepsilon | X > t)}{\varepsilon} = \frac{f_X(t)}{1 - F_x(t)} = -\frac{d}{dt} ln(1 - F_X(t))$$

Definition

$X \leqslant_{hr} Y$ if $r_X(t) \geqslant r_Y(t)$.

The hazard rate order : *hr*

The hazard rate (or failure rate) is defined by :

$$r_X(t) = \lim_{\varepsilon \to 0} \frac{P(X < t + \varepsilon | X > t)}{\varepsilon} = \frac{f_X(t)}{1 - F_x(t)} = -\frac{d}{dt} ln(1 - F_X(t))$$

Definition

$X \leqslant_{hr} Y$ if $r_X(t) \geqslant r_Y(t)$.

- the Proba-Proba plot $G(F^{-1}(t))$ is star shaped with respect to (1,1).

The hazard rate order : *hr*

The hazard rate (or failure rate) is defined by :

$$r_X(t) = \lim_{\varepsilon \to 0} \frac{P(X < t + \varepsilon | X > t)}{\varepsilon} = \frac{f_X(t)}{1 - F_x(t)} = -\frac{d}{dt} ln(1 - F_X(t))$$

Definition

$X \leqslant_{hr} Y$ if $r_X(t) \geqslant r_Y(t)$.

- the Proba-Proba plot $G(F^{-1}(t))$ is star shaped with respect to (1,1).
- $Eg(X^*, Y^*) \leqslant Eg(Y^*, X^*) \quad \forall g$ s.t. $\quad g(x, y) - g(y, x)$ increasing in $x, \forall x \geqslant y$.

The hazard rate order : *hr*

The hazard rate (or failure rate) is defined by :

$$r_X(t) = \lim_{\varepsilon \to 0} \frac{P(X < t + \varepsilon | X > t)}{\varepsilon} = \frac{f_X(t)}{1 - F_x(t)} = -\frac{d}{dt} ln(1 - F_X(t))$$

Definition

$X \leqslant_{hr} Y$ if $r_X(t) \geqslant r_Y(t)$.

- the Proba-Proba plot $G(F^{-1}(t))$ is star shaped with respect to (1,1).
- $Eg(X^*, Y^*) \leqslant Eg(Y^*, X^*) \quad \forall g$ s.t. $\quad g(x, y) - g(y, x)$ increasing in $x, \forall x \geqslant y$.
- The *hr* order is preserved under aging and is stronger than the *st* order

The likelihood order : *lr*

Another order which is even stronger than *hr* is the likelihood ratio which preserves *st* under any conditioning :

Definition

$U = [a, b], V = [c, d], U < V$ $X \leqslant_{lr} Y$ if $P(X \in V)P(Y \in U) \leqslant P(X \in U)P(Y \in V)$ or equivalently $(X|X \in U) \leqslant_{st} (Y|Y \in U)$

The likelihood order : *lr*

Another order which is even stronger than *hr* is the likelihood ratio which preserves *st* under any conditioning :

Definition

$U = [a, b], V = [c, d], U < V$ $X \leqslant_{lr} Y$ if $P(X \in V)P(Y \in U) \leqslant P(X \in U)P(Y \in V)$ or equivalently $(X|X \in U) \leqslant_{st} (Y|Y \in U)$

- The P-P plot is convex.

The likelihood order : *lr*

Another order which is even stronger than *hr* is the likelihood ratio which preserves *st* under any conditioning :

Definition

$U = [a, b], V = [c, d], U < V$ $X \leqslant_{lr} Y$ if $P(X \in V)P(Y \in U) \leqslant P(X \in U)P(Y \in V)$ or equivalently $(X|X \in U) \leqslant_{st} (Y|Y \in U)$

- The P-P plot is convex.
- $Eg(X^*, Y^*) \leqslant Eg(Y^*, X^*) \quad \forall g$ s.t. $\quad g(x, y) - g(y, x) \geqslant 0, \quad \forall x \geqslant y.$

The likelihood order : *lr*

Another order which is even stronger than *hr* is the likelihood ratio which preserves *st* under any conditioning :

Definition

$U = [a, b], V = [c, d], U < V$ $X \leqslant_{lr} Y$ if $P(X \in V)P(Y \in U) \leqslant P(X \in U)P(Y \in V)$ or equivalently $(X|X \in U) \leqslant_{st} (Y|Y \in U)$

- The P-P plot is convex.
- $Eg(X^*, Y^*) \leqslant Eg(Y^*, X^*) \quad \forall g$ s.t. $\quad g(x, y) - g(y, x) \geqslant 0, \quad \forall x \geqslant y.$
- The *lr* order is preserved under any conditioning and is stronger than the *hr* order.

Other stochastic orders : convex orders

The convex orders are used to compare the variability of stochastic variables.

Other stochastic orders : convex orders

The convex orders are used to compare the variability of stochastic variables.

Definition

$X \leqslant_{cx} Y$ if $Ef(X) \leqslant Ef(Y)$ for all convex functions f.

Other stochastic orders : convex orders

The convex orders are used to compare the variability of stochastic variables.

Definition

$X \leqslant_{cx} Y$ if $Ef(X) \leqslant Ef(Y)$ for all convex functions f.

Definition

$X \leqslant_{icx} Y$ if $Ef(X) \leqslant Ef(Y)$ for all increasing convex functions f.

Strassen Representation Theorem :

Theorem

$X \leqslant_{cx} Y$ iff there exist two r.v. X' and Y' with the same db as X and Y such that $X' = E(Y'|X')$.

Other stochastic orders : convex orders

The convex orders are used to compare the variability of stochastic variables.

Definition

$X \leqslant_{cx} Y$ if $Ef(X) \leqslant Ef(Y)$ for all convex functions f.

Definition

$X \leqslant_{icx} Y$ if $Ef(X) \leqslant Ef(Y)$ for all increasing convex functions f.

Strassen Representation Theorem :

Theorem

$X \leqslant_{cx} Y$ iff there exist two r.v. X' and Y' with the same db as X and Y such that $X' = E(Y'|X')$.

Corollary if X and Z are independent and $E(Z) = 0$ then, $X \leqslant_{cx} X + Z$.

Discrete dynamical systems

There are two types of models in scheduling.

static models :$X = \phi(Z_1, \cdots, Z_N)$ and
dynamic models $X_n = \phi_n(X_{n-1}, Z_n), \quad \forall n \geqslant 0$.

Discrete dynamical systems

There are two types of models in scheduling.

static models :$X = \phi(Z_1, \cdots, Z_N)$ and
dynamic models $X_n = \phi_n(X_{n-1}, Z_n), \quad \forall n \geqslant 0$.

A dynamical system is time -monotone for order F if $X_n \leqslant_F X_{n-1}$.
A system (static or dynamic) is F- isotone if $Z_k \leqslant_F Z'_k \Rightarrow X_n \leqslant_F X'_n$.

Discrete dynamical systems

There are two types of models in scheduling.

static models :$X = \phi(Z_1, \cdots, Z_N)$ and
dynamic models $X_n = \phi_n(X_{n-1}, Z_n), \quad \forall n \geqslant 0$.

A dynamical system is time -monotone for order F if $X_n \leqslant_F X_{n-1}$.
A system (static or dynamic) is F- isotone if $Z_k \leqslant_F Z'_k \Rightarrow X_n \leqslant_F X'_n$.

Comparison are often proved using **mapping, coupling, association and monotony.**

Mapping techniques

Principle : Prove comparability by comparing the inputs of functionals.
for a static system,

Theorem

if $(Z_1, \dots, Z_n) \leqslant_{st} (Z'_1, \dots Z'_n)$ *and are independent, then if* Φ *is increasing , then* $\Phi(Z_1, \dots, Z_n) \leqslant_{st} \Phi(Z'_1, \dots Z'_n)$.
if $(Z_1, \dots, Z_n) \leqslant_{icx} (Z'_1, \dots Z'_n)$ *and are independent, then If* Φ *is increasing and convex then* $\Phi(Z_1, \dots, Z_n) \leqslant_{icx} \Phi(Z'_1, \dots Z'_n)$.

For a dynamic system,

Theorem

if $(Z_1, \dots, Z_n) \leqslant_F (Z'_1, \dots Z'_n)$ *and all* φ_n *are increasing, (resp. increasing and convex) then* $X_n \leqslant_F X'_n$*, with* $F = st$ *(resp.* $F = icx$*).*

Association

Two random variables X and Y are associated if $cov(g(X), f(Y)) \geqslant 0$ for all increasing f and g.

Association

Two random variables X and Y are associated if $cov(g(X), f(Y)) \geqslant 0$ for all increasing f and g.

$X_n = \phi(X_n, Z_n)$ if ϕ is monotone on both variables, then if Z_n are independent or associated then X_n are associated.

Association

Two random variables X and Y are associated if $cov(g(X), f(Y)) \geqslant 0$ for all increasing f and g.

$X_n = \phi(X_n, Z_n)$ if ϕ is monotone on both variables, then if Z_n are independent or associated then X_n are associated.

Finally, $(X_1, \ldots X_n)$ associated implies that $(X_1, \ldots X_n) \leqslant_o (X_1^*, \ldots X_n^*)$ where $(X_1^*, \ldots X_n^*)$ are independent versions of $(X_1, \ldots X_n)$.

Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively).

Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively). One has $\mathbf{C}_n = \mathcal{R}(\mathbf{C}_{n-1} + S_n e_1 - \delta_n 1)_+$

Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

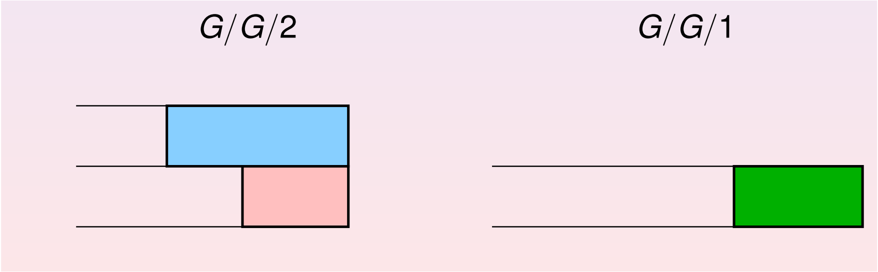
The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively). One has $\mathbf{C}_n = \mathcal{R}(\mathbf{C}_{n-1} + S_n e_1 - \delta_n 1)_+$ Then, it should be clear that the s first components of $\mathbf{C}_n$ and $\mathbf{C}'_n$ are comparable, by induction.

Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively). One has $\mathbf{C}_n = \mathcal{R}(\mathbf{C}_{n-1} + S_n e_1 - \delta_n 1)_+$ Then, it should be clear that the s first components of $\mathbf{C}_n$ and $\mathbf{C}'_n$ are comparable, by induction. Therefore, $C_n =_{st} \mathbf{C}_n(1) \leqslant_{st} \mathbf{C}'_n(1) =_{st} C'_n$.

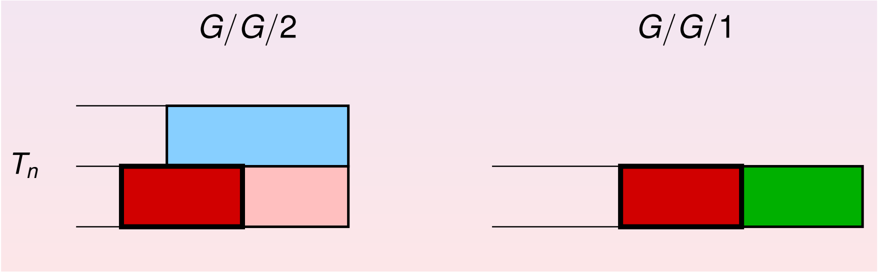


Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively). One has $\mathbf{C}_n = \mathcal{R}(\mathbf{C}_{n-1} + S_n e_1 - \delta_n 1)_+$ Then, it should be clear that the s first components of $\mathbf{C}_n$ and $\mathbf{C}'_n$ are comparable, by induction. Therefore, $C_n =_{st} \mathbf{C}_n(1) \leqslant_{st} \mathbf{C}'_n(1) =_{st} C'_n$.

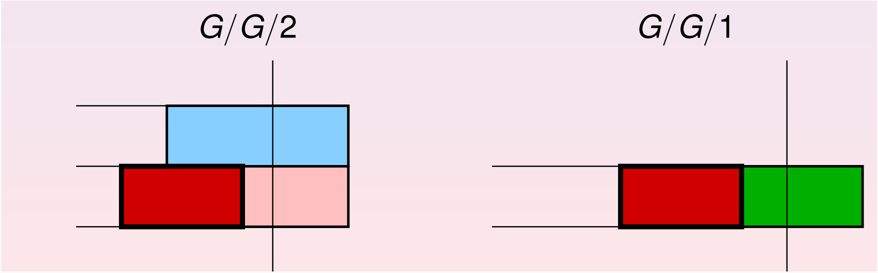


Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively). One has $\mathbf{C}_n = \mathcal{R}(\mathbf{C}_{n-1} + S_n e_1 - \delta_n 1)_+$ Then, it should be clear that the s first components of $\mathbf{C}_n$ and $\mathbf{C}'_n$ are comparable, by induction. Therefore, $C_n =_{st} \mathbf{C}_n(1) \leqslant_{st} \mathbf{C}'_n(1) =_{st} C'_n$.

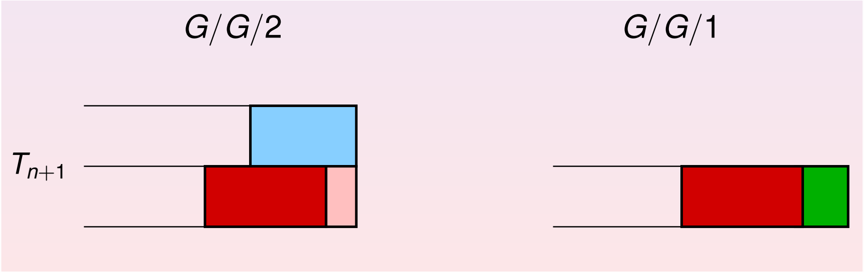


Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively). One has $\mathbf{C}_n = \mathcal{R}(\mathbf{C}_{n-1} + S_n e_1 - \delta_n 1)_+$ Then, it should be clear that the s first components of $\mathbf{C}_n$ and $\mathbf{C}'_n$ are comparable, by induction. Therefore, $C_n =_{st} \mathbf{C}_n(1) \leqslant_{st} \mathbf{C}'_n(1) =_{st} C'_n$.

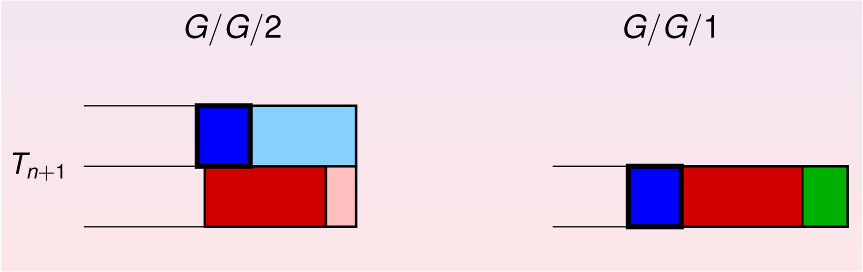


Coupling technique

Principle : Prove comparisons using sample paths. Coupling often provides more powerful results.

Example : compare the load C for two task-resource systems with s and s' resources, respectively :
How to show that $C_n \leqslant_{st} C'_n$? (loads at the nth arrival, T_n)

The best is by using a coupling method, *i.e.* by considering a unique input process sample ω and compare the two systems over that single sequence of sizes and arrival times. The workload vectors $\mathbf{C}_n$ and $\mathbf{C}'_n$ are the increasingly ordered workloads at time T_n in the different resources (they have s and s' components respectively). One has $\mathbf{C}_n = \mathcal{R}(\mathbf{C}_{n-1} + S_n e_1 - \delta_n 1)_+$ Then, it should be clear that the s first components of $\mathbf{C}_n$ and $\mathbf{C}'_n$ are comparable, by induction. Therefore, $C_n =_{st} \mathbf{C}_n(1) \leqslant_{st} \mathbf{C}'_n(1) =_{st} C'_n$.



The problem $1||\sum C_i$

This is one of the simplest scheduling problem : one resource with no scheduling restriction on N tasks, all arriving at time 0, while the objective is to minimize the sum of the completion times (or the average completion time)
We consider all tasks to be independent of sizes $S_1, \ldots, S_N$.

The problem $1||\sum C_i$

This is one of the simplest scheduling problem : one resource with no scheduling restriction on N tasks, all arriving at time 0, while the objective is to minimize the sum of the completion times (or the average completion time)
We consider all tasks to be independent of sizes $S_1, \ldots, S_N$.

For a given schedule (or permutation) σ, the objective function is
$T_\sigma = \sum_{i=1}^{N} C_i = \sum_{i=1}^{N}(N - i + 1)S_{\sigma(i)}$.

The problem $1||\sum C_i$

This is one of the simplest scheduling problem : one resource with no scheduling restriction on N tasks, all arriving at time 0, while the objective is to minimize the sum of the completion times (or the average completion time)
We consider all tasks to be independent of sizes $S_1, \ldots, S_N$.

For a given schedule (or permutation) σ, the objective function is
$T_\sigma = \sum_{i=1}^{N} C_i = \sum_{i=1}^{N}(N-i+1)S_{\sigma(i)}$.
We consider two particular schedules : SEPT (Shortest Expected Processing Time) and LEPT (Largest Expected Processing Time).

It should be clear that for any permutation σ, $ET_{SEPT} \leqslant ET_\sigma \leqslant ET_{LEPT}$.
Indeed, $ET_\sigma = E\sum_{i=1}^{N} C_i = \sum_{i=1}^{N}(N-i+1)ES_{\sigma(i)}$.

The problem $1||\sum C_i$

This is one of the simplest scheduling problem : one resource with no scheduling restriction on N tasks, all arriving at time 0, while the objective is to minimize the sum of the completion times (or the average completion time)
We consider all tasks to be independent of sizes $S_1, \ldots, S_N$.

For a given schedule (or permutation) σ, the objective function is
$T_\sigma = \sum_{i=1}^N C_i = \sum_{i=1}^N (N-i+1)S_{\sigma(i)}$.
We consider two particular schedules : SEPT (Shortest Expected Processing Time) and LEPT (Largest Expected Processing Time).

It should be clear that for any permutation σ, $ET_{SEPT} \leqslant ET_\sigma \leqslant ET_{LEPT}$.
Indeed, $ET_\sigma = E\sum_{i=1}^N C_i = \sum_{i=1}^N (N-i+1)ES_{\sigma(i)}$.

But can we say more ?

The problem $1||\sum C_i$ revisited

Theorem (Shanthikumar, Yao, 1993)

If $S_i \leqslant_{lr} S_{i+1}$ *for all i, then* $T_{SEPT} \leqslant_{st} T_\sigma \leqslant_{st} T_{LEPT}$ *.*
If $S_i \leqslant_{hr} S_{i+1}$ *for all i, then* $T_{SEPT} \leqslant_{icx} T_\sigma \leqslant_{icx} T_{LEPT}$ *.*

Proof uses a classical interchange argument (done for *hr*)

The problem $1||\sum C_i$ revisited

Theorem (Shanthikumar, Yao, 1993)

If $S_i \leqslant_{lr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{st} T_\sigma \leqslant_{st} T_{LEPT}$.
If $S_i \leqslant_{hr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{icx} T_\sigma \leqslant_{icx} T_{LEPT}$.

Proof uses a classical interchange argument (done for *hr*)

For all $\sigma \neq SEPT$ there exists k such that $\sigma(k) = j, \sigma(k+1) = i$ with $i < j$. Let $\mu = \sigma$ except $\mu(k) = i, \mu(k+1) = j$ (μ is closer to $SEPT$ than σ).

The problem $1||\sum C_i$ revisited

Theorem (Shanthikumar, Yao, 1993)

If $S_i \leqslant_{lr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{st} T_\sigma \leqslant_{st} T_{LEPT}$.
If $S_i \leqslant_{hr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{icx} T_\sigma \leqslant_{icx} T_{LEPT}$.

Proof uses a classical interchange argument (done for *hr*)

For all $\sigma \neq SEPT$ there exists k such that $\sigma(k) = j, \sigma(k+1) = i$ with $i < j$. Let $\mu = \sigma$ except $\mu(k) = i, \mu(k+1) = j$ (μ is closer to $SEPT$ than σ).

Now, $T_\sigma = X_j + k(X_i + X_j) + Y$ and $T_\mu = X_i + k(X_i + X_j) + Y$ where Y is the contribution of the other jobs, independent of S_i and S_j.

The problem $1||\sum C_i$ revisited

Theorem (Shanthikumar, Yao, 1993)

If $S_i \leqslant_{lr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{st} T_\sigma \leqslant_{st} T_{LEPT}$.
If $S_i \leqslant_{hr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{icx} T_\sigma \leqslant_{icx} T_{LEPT}$.

Proof uses a classical interchange argument (done for *hr*)

For all $\sigma \neq SEPT$ there exists k such that $\sigma(k) = j, \sigma(k+1) = i$ with $i < j$. Let $\mu = \sigma$ except $\mu(k) = i, \mu(k+1) = j$ (μ is closer to $SEPT$ than σ).

Now, $T_\sigma = X_j + k(X_i + X_j) + Y$ and $T_\mu = X_i + k(X_i + X_j) + Y$ where Y is the contribution of the other jobs, independent of S_i and S_j.

Moreover $g(x, y) = f(x + k(x + y))$ satisfies $g(x, y) - g(y, x)$ is increasing as long as f is convex and increasing. Therefore, $S_i \leqslant_{hr} S_j$ implies $Ef(X_j + k(X_i + X_j)) \geqslant Ef(X_i + k(X_i + X_j))$ for all increasing convex f.

The problem $1||\sum C_i$ revisited

Theorem (Shanthikumar, Yao, 1993)

If $S_i \leqslant_{lr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{st} T_\sigma \leqslant_{st} T_{LEPT}$.
If $S_i \leqslant_{hr} S_{i+1}$ for all i, then $T_{SEPT} \leqslant_{icx} T_\sigma \leqslant_{icx} T_{LEPT}$.

Proof uses a classical interchange argument (done for *hr*)

For all $\sigma \neq SEPT$ there exists k such that $\sigma(k) = j, \sigma(k+1) = i$ with $i < j$. Let $\mu = \sigma$ except $\mu(k) = i, \mu(k+1) = j$ (μ is closer to $SEPT$ than σ).

Now, $T_\sigma = X_j + k(X_i + X_j) + Y$ and $T_\mu = X_i + k(X_i + X_j) + Y$ where Y is the contribution of the other jobs, independent of S_i and S_j.

Moreover $g(x, y) = f(x + k(x + y))$ satisfies $g(x, y) - g(y, x)$ is increasing as long as f is convex and increasing. Therefore, $S_i \leqslant_{hr} S_j$ implies $Ef(X_j + k(X_i + X_j)) \geqslant Ef(X_i + k(X_i + X_j))$ for all increasing convex f.

Finally, $X_i + k(X_i + X_j) \leqslant_{icx} X_j + k(X_i + X_j)$ implies $T_\mu \leqslant_{icx} T_\sigma$.

PERT Graph

A PERT graph is a more general static model : N tasks are to be executed over an infinite number of resources and are constrained by an acyclic graph.

PERT Graph

A PERT graph is a more general static model : N tasks are to be executed over an infinite number of resources and are constrained by an acyclic graph.

PERT graphs are impossible to solve (compute the makespan) analytically in general (Kamburowski, 1992). However, one can use comparisons to prove several results.

PERT Graph, continued

Here are the ingredients used to compare (and compute bounds) for PERT graphs.

PERT Graph, continued

Here are the ingredients used to compare (and compute bounds) for PERT graphs.

1- Using the mapping technique :
$C = \Phi(X_1, \dots X_N) = \max_{c \in \mathcal{P}(G)} \sum_{i \in c} X_i$.

PERT Graph, continued

Here are the ingredients used to compare (and compute bounds) for PERT graphs.

1- Using the mapping technique :
$C = \Phi(X_1, \ldots X_N) = \max_{c \in \mathcal{P}(G)} \sum_{i \in c} X_i$.
Note that Φ is convex and increasing. This implies the following first result. $Z_i \leqslant_F Z'_i$ implies $C \leqslant_F C'$ with $F = st$ or icx.

PERT Graph, continued

Here are the ingredients used to compare (and compute bounds) for PERT graphs.

1- Using the mapping technique :
$C = \Phi(X_1, \ldots X_N) = \max_{c \in \mathcal{P}(G)} \sum_{i \in c} X_i$.
Note that Φ is convex and increasing. This implies the following first result. $Z_i \leqslant_F Z'_i$ implies $C \leqslant_F C'$ with $F = st$ or icx.

2- Next, if tasks are independent (or associated), then the paths are all associated and are therefore bounded by independent versions :

PERT Graph, continued

Here are the ingredients used to compare (and compute bounds) for PERT graphs.

1- Using the mapping technique :
$C = \Phi(X_1, \dots X_N) = \max_{c \in \mathcal{P}(G)} \sum_{i \in c} X_i$.
Note that Φ is convex and increasing. This implies the following first result. $Z_i \leqslant_F Z_i'$ implies $C \leqslant_F C'$ with $F = st$ or icx.

2- Next, if tasks are independent (or associated), then the paths are all associated and are therefore bounded by independent versions :
for all c, $S_c \leqslant_{st} S_c^*$ and $C \leqslant_{st} C^* = \max_{c \in \mathcal{P}(G)} S_c^*$

PERT Graph, continued

Here are the ingredients used to compare (and compute bounds) for PERT graphs.

1- Using the mapping technique :
$C = \Phi(X_1, \dots X_N) = \max_{c \in \mathcal{P}(G)} \sum_{i \in c} X_i$.
Note that Φ is convex and increasing. This implies the following first result. $Z_i \leqslant_F Z'_i$ implies $C \leqslant_F C'$ with $F = st$ or icx.

2- Next, if tasks are independent (or associated), then the paths are all associated and are therefore bounded by independent versions :
for all c, $S_c \leqslant_{st} S^*_c$ and $C \leqslant_{st} C^* = \max_{c \in \mathcal{P}(G)} S^*_c$

3- Next, if X_i is NBUE (New Better than Used in Expectation : $E(X - t | X > t) \leqslant EX$) then $X_i \leqslant_{cx} exp(E(X_i))$.

PERT Graph, continued

This allows us to show that

$$\max_{c \in \mathcal{P}(G)} \sum_{i \in c} EX_i \leqslant_{icx} C \leqslant_{icx} \max_{c \in \mathcal{P}(G)} \sum_{i \in c} exp(E(X_i))$$

PERT Graph, continued

This allows us to show that

$$\max_{c \in \mathcal{P}(G)} \sum_{i \in c} EX_i \leqslant_{icx} C \leqslant_{icx} \max_{c \in \mathcal{P}(G)} \sum_{i \in c} exp(E(X_i))$$

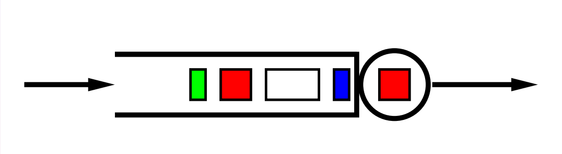
and

$$C \leqslant_{icx} \max_{c \in \mathcal{P}(G)} exp(\sum_{i \in c} E(X_i)) =_{db} \prod_{c \in \mathcal{P}(G)} 1 - exp(-t \sum_{i \in c} E(X_i)),$$

as soon as X_i are all NBUE and associated.

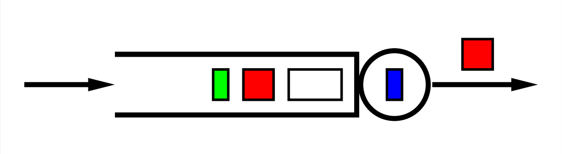
Queues

Queues are among simplest dynamic systems, but are still the source of many open problems.
Tasks do not have any constraints, sizes and arrival times are often independent.



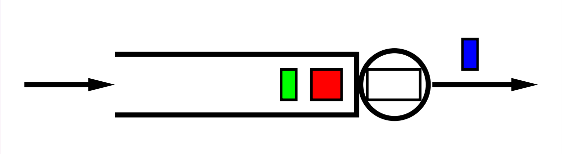
Queues

Queues are among simplest dynamic systems, but are still the source of many open problems.
Tasks do not have any constraints, sizes and arrival times are often independent.



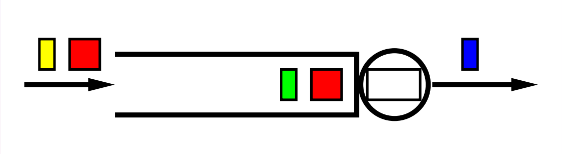
Queues

Queues are among simplest dynamic systems, but are still the source of many open problems.
Tasks do not have any constraints, sizes and arrival times are often independent.



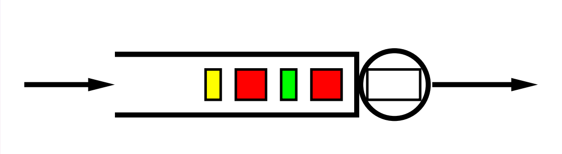
Queues

Queues are among simplest dynamic systems, but are still the source of many open problems.
Tasks do not have any constraints, sizes and arrival times are often independent.



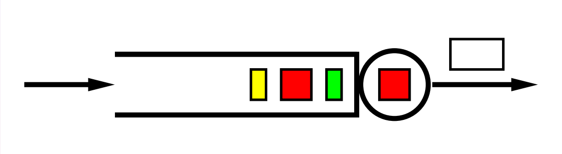
Queues

Queues are among simplest dynamic systems, but are still the source of many open problems.
Tasks do not have any constraints, sizes and arrival times are often independent.

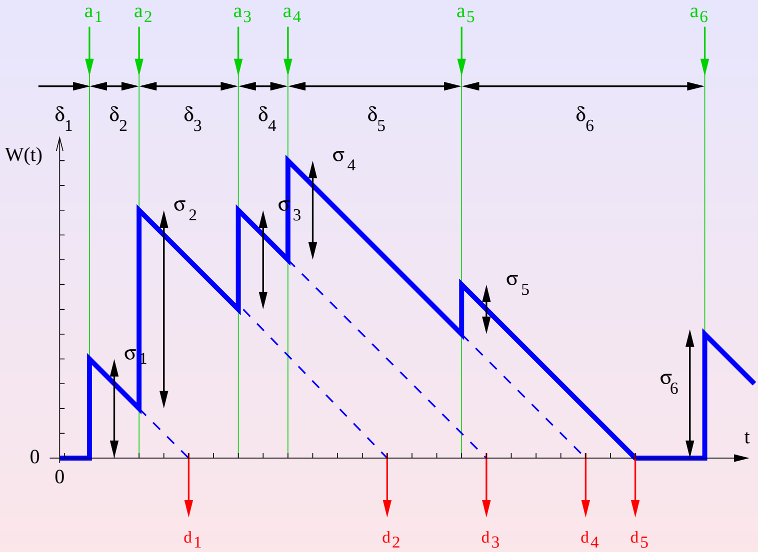


Queues

Queues are among simplest dynamic systems, but are still the source of many open problems.
Tasks do not have any constraints, sizes and arrival times are often independent.



Lindley's formula



W_n is the waiting time of the n-th task. It is a dynamical system of the form $W_n = \varphi(W_{n-1}, X_n)$ with $X_n = S_{n-1} - \delta_n$ and φ defined by the Lindley's equation :

$$W_n = \max(W_{n-1} + X_n, 0) \ .$$

Loynes' scheme

Theorem

$W_n \leqslant_{st} W_{n+1}$ in a G/G/1 queue, initialy empty.

Loynes' scheme

Theorem

$W_n \leqslant_{st} W_{n+1}$ in a G/G/1 queue, initialy empty.

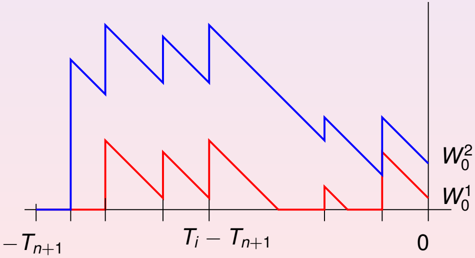
Proof. done by a backward coupling known as the Loynes' scheme. Construct on a common probability space two trajectories by going backward in time : $S^1_{i-n}(\omega) = S^2_{i-n-1}(\omega)$ with distribution S_i and $T^1_{i-n}(\omega) = T^2_{i-n-1}(\omega)$, with distribution $T_i - T_{n+1}$ for all $0 \leqslant i \leqslant n+1$ and $S^1_{-n-1}(\omega) = 0$.

Loynes' scheme

Theorem

$W_n \leqslant_{st} W_{n+1}$ in a G/G/1 queue, initialy empty.

Proof. done by a backward coupling known as the Loynes' scheme. Construct on a common probability space two trajectories by going backward in time : $S^1_{i-n}(\omega) = S^2_{i-n-1}(\omega)$ with distribution S_i and $T^1_{i-n}(\omega) = T^2_{i-n-1}(\omega)$, with distribution $T_i - T_{n+1}$ for all $0 \leqslant i \leqslant n+1$ and $S^1_{-n-1}(\omega) = 0$.
By construction, $W^1_0 =_{st} W_n$ and $W^2_0 =_{st} W_{n+1}$. Also, it should be clear that $0 = W^1_{-n+1}(\omega) \leqslant W^2_{-n+1}(\omega)$ for all ω.
This implies $W^1_{-i}(\omega) \leqslant W^2_{-i}(\omega)$ so that $W_n \leqslant_{st} W_{n+1}$.

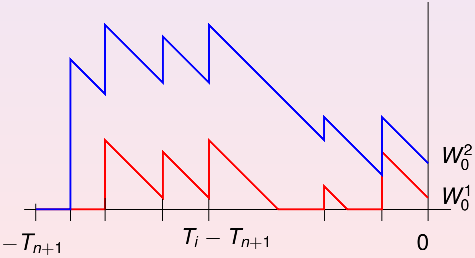


Loynes' scheme

Theorem

$W_n \leqslant_{st} W_{n+1}$ in a G/G/1 queue, initialy empty.

Proof. done by a backward coupling known as the Loynes' scheme. Construct on a common probability space two trajectories by going backward in time : $S^1_{i-n}(\omega) = S^2_{i-n-1}(\omega)$ with distribution S_i and $T^1_{i-n}(\omega) = T^2_{i-n-1}(\omega)$, with distribution $T_i - T_{n+1}$ for all $0 \leqslant i \leqslant n+1$ and $S^1_{-n-1}(\omega) = 0$.
By construction, $W^1_0 =_{st} W_n$ and $W^2_0 =_{st} W_{n+1}$. Also, it should be clear that $0 = W^1_{-n+1}(\omega) \leqslant W^2_{-n+1}(\omega)$ for all ω.
This implies $W^1_{-i}(\omega) \leqslant W^2_{-i}(\omega)$ so that $W_n \leqslant_{st} W_{n+1}$.



This has many consequences in terms of existence and uniqueness of a stationary (or limit) regime for the G/G/1 queue (Baccelli Bremaud, 2002).

Input process

Folk theorem (Ross conjecture, 1978) :*things work better when the input traffic has less variability.*

Theorem

if $(W_0, X_1, \ldots, X_n) \leqslant_F (W'_0, X'_1, \ldots, X'_n)$ *then* $W_n \leqslant_F W'_n$ *(with F = st or icx).*

Input process

Folk theorem (Ross conjecture, 1978) :*things work better when the input traffic has less variability.*

Theorem

if $(W_0, X_1, \ldots, X_n) \leqslant_F (W'_0, X'_1, \ldots, X'_n)$ *then* $W_n \leqslant_F W'_n$ *(with F = st or icx).*

proof $f(x, w) = max(w + x, 0)$ is convex and increasing for both variables.

Examples

1. Show that if the traffic intensity is fixed in a single GI/GI/1 queue, then the average waiting time is smallest when the arrivals are periodic.

Examples

1. Show that if the traffic intensity is fixed in a single GI/GI/1 queue, then the average waiting time is smallest when the arrivals are periodic.
2. Show that if the arrival process in a GI/M/1 queue is NBUE, then the average waiting time can be bounded by $EW \leqslant \frac{1}{\mu - 1/E(T_1)}$.

Extremal input processes

Several extensions are possible :

Theorem (Altman, Gaujal, Hordijk, 2003)

If the arrival sequence $T_1, \ldots, T_n, \ldots$ is fixed in a stochastic FIFO event graph (arbitrary network of queues with no branching enriched with fork and join nodes), then, $S_1, \ldots, S_n \leqslant_{cx} S'_1, \ldots, S'_n$ implies $(W_1, \ldots, W_n) \leqslant_{icx} (W_1, \ldots, W_n)$.

Service discipline

In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, . . .

Service discipline

In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, . . .

PS has insensibility, reversibility and product form properties,

Service discipline

In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, . . .

PS has insensibility, reversibility and product form properties,

FIFO (F) has optimality properties, in terms of waiting times :

Theorem

In a GI/GI/1 queue, $f(\mathbf{W}_n^F) \leqslant_{st} f(\mathbf{W}_n^\pi)$ *for all service discipline* π *and all convex increasing and symmetric* f*.*

Service discipline

In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, . . .

PS has insensibility, reversibility and product form properties,

FIFO (F) has optimality properties, in terms of waiting times :

Theorem

In a GI/GI/1 queue, $f(\mathbf{W}_n^F) \leqslant_{st} f(\mathbf{W}_n^\pi)$ *for all service discipline* π *and all convex increasing and symmetric* f.

Proof. using a coupling technique and majorization.

Since service time and arrivals are independent, we can rearrange the service times in the order of service (and not of arrivals) under policy π and F Let D_i be the departure epochs. They coincide under both policies.

Service discipline

In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, . . .

PS has insensibility, reversibility and product form properties,

FIFO (F) has optimality properties, in terms of waiting times :

Theorem

In a GI/GI/1 queue, $f(\mathbf{W}_n^F) \leqslant_{st} f(\mathbf{W}_n^\pi)$ for all service discipline π and all convex increasing and symmetric f.

Proof. using a coupling technique and majorization.

Since service time and arrivals are independent, we can rearrange the service times in the order of service (and not of arrivals) under policy π and F Let D_i be the departure epochs. They coincide under both policies.

Then $W_i^F = D_i - T_i$ and assume that π interchange the departure of j and $j+1$: $W_j^\pi = D_{j+1} - T_j$ and $W_{j+1}^\pi = D_j - T_{j+1}$

Service discipline

In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, ...

PS has insensibility, reversibility and product form properties,

FIFO (F) has optimality properties, in terms of waiting times :

Theorem

In a GI/GI/1 queue, $f(\mathbf{W}_n^F) \leqslant_{st} f(\mathbf{W}_n^\pi)$ for all service discipline π and all convex increasing and symmetric f.

Proof. using a coupling technique and majorization.

Since service time and arrivals are independent, we can rearrange the service times in the order of service (and not of arrivals) under policy π and F Let D_i be the departure epochs. They coincide under both policies.

Then $W_i^F = D_i - T_i$ and assume that π interchange the departure of j and $j+1$: $W_j^\pi = D_{j+1} - T_j$ and $W_{j+1}^\pi = D_j - T_{j+1}$

Now, it should be obvious that $W_j^\pi + W_{j+1}^\pi = W_j^F + W_{j+1}^F$ and if f is increasing convex and symmetric (or Schur convex) $f(W_j^\pi, W_{j+1}^\pi) \geqslant f(W_j^F, W_{j+1}^F)$.

Service discipline

In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, ...
PS has insensibility, reversibility and product form properties,
FIFO (F) has optimality properties, in terms of waiting times :

Theorem

In a GI/GI/1 queue, $f(\mathbf{W}_n^F) \leqslant_{st} f(\mathbf{W}_n^\pi)$ for all service discipline π and all convex increasing and symmetric f.

Proof. using a coupling technique and majorization.
Since service time and arrivals are independent, we can rearrange the service times in the order of service (and not of arrivals) under policy π and F Let D_i be the departure epochs. They coincide under both policies.
Then $W_i^F = D_i - T_i$ and assume that π interchange the departure of j and $j+1$: $W_j^\pi = D_{j+1} - T_j$ and $W_{j+1}^\pi = D_j - T_{j+1}$
Now, it should be obvious that $W_j^\pi + W_{j+1}^\pi = W_j^F + W_{j+1}^F$ and if f is increasing convex and symmetric (or Schur convex) $f(W_j^\pi, W_{j+1}^\pi) \geqslant f(W_j^F, W_{j+1}^F)$.
In general, consider all tasks (n) within a busy period of the system, then,
$W_1^\pi + \cdots + W_{j+1}^\pi = W_1^F + \cdots + W_n^F$ and interchanging a pair of customers out of order under π, reduces the value of $f(W_1^\pi, \ldots, W_n^\pi)$ down to the value of $f(W_1^F, \ldots, W_n^F)$ for any Schur convex function f.

Service discipline

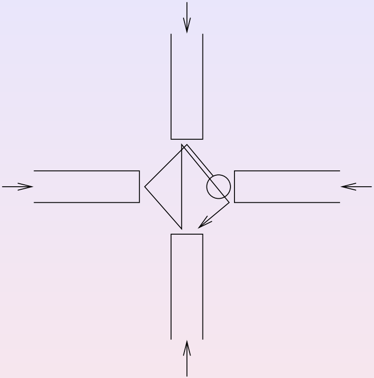
In a queue with one or more servers, tasks may be served according to disciplines : FIFO, LIFO, PS, priority, random, ...
PS has insensibility, reversibility and product form properties,
FIFO (F) has optimality properties, in terms of waiting times :

Theorem

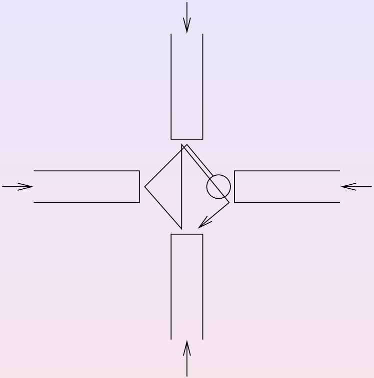
In a GI/GI/1 queue, $f(\mathbf{W}_n^F) \leqslant_{st} f(\mathbf{W}_n^\pi)$ for all service discipline π and all convex increasing and symmetric f.

Proof. using a coupling technique and majorization.
Since service time and arrivals are independent, we can rearrange the service times in the order of service (and not of arrivals) under policy π and F Let D_i be the departure epochs. They coincide under both policies.
Then $W_i^F = D_i - T_i$ and assume that π interchange the departure of j and $j+1$: $W_j^\pi = D_{j+1} - T_j$ and $W_{j+1}^\pi = D_j - T_{j+1}$
Now, it should be obvious that $W_j^\pi + W_{j+1}^\pi = W_j^F + W_{j+1}^F$ and if f is increasing convex and symmetric (or Schur convex) $f(W_j^\pi, W_{j+1}^\pi) \geqslant f(W_j^F, W_{j+1}^F)$.
In general, consider all tasks (n) within a busy period of the system, then,
$W_1^\pi + \cdots + W_{j+1}^\pi = W_1^F + \cdots + W_n^F$ and interchanging a pair of customers out of order under π, reduces the value of $f(W_1^\pi, \ldots, W_n^\pi)$ down to the value of $f(W_1^F, \ldots, W_n^F)$ for any Schur convex function f.
If the first n tasks do not form a busy period, then $W_1^\pi + \cdots + W_{j+1}^\pi \geqslant W_1^F + \cdots + W_n^F$ and again $f(W_1^\pi, \ldots, W_n^\pi) \geqslant f(W_1^F, \ldots, W_n^F)$ for any Schur convex function f

Polling systems

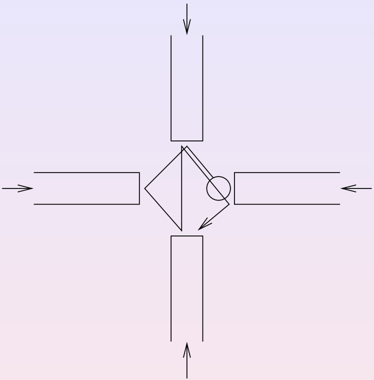


Polling systems



Choosing the best open loop schedule for the server corresponds to choose the most regular service in each queue. (Gaujal, Hordijk, Van der Laan, 2007)

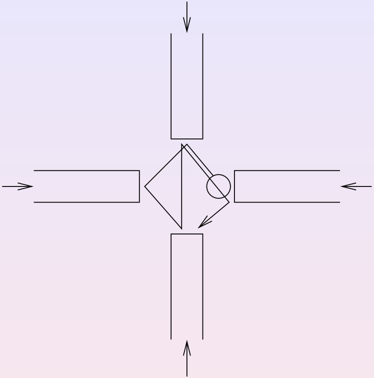
Polling systems



Choosing the best open loop schedule for the server corresponds to choose the most regular service in each queue. (Gaujal, Hordijk, Van der Laan, 2007)

Example : for two queues (1 and 2) 12121212. . . is a better schedule than 1122112211. . .

Polling systems



Choosing the best open loop schedule for the server corresponds to choose the most regular service in each queue. (Gaujal, Hordijk, Van der Laan, 2007)

Example : for two queues (1 and 2) 12121212... is a better schedule than 1122112211...
The main difficulty is to compute the frequency of the visits to each queue.

Polling systems, continued

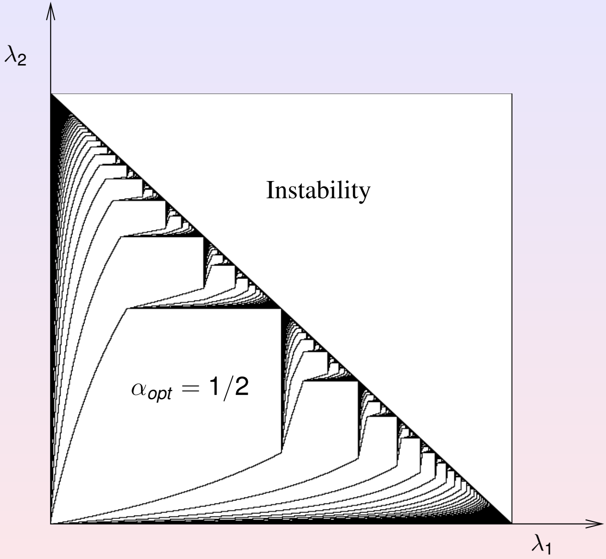


FIG.: The frequency of the server allocations w.r.t input intensities

Polling systems, continued

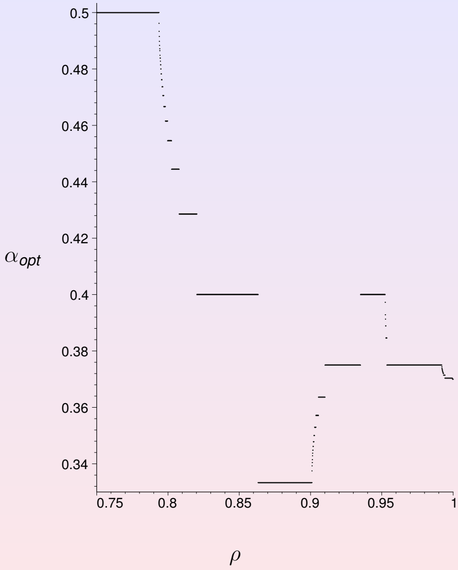


FIG.: The frequency of the server allocations w.r.t the total load, the ratio of input intensities being fixed.

Conclusion

There exists a systematic framework to deal with task-resource systems involving randomness through the theory of stochastic comparisons.

Conclusion

There exists a systematic framework to deal with task-resource systems involving randomness through the theory of stochastic comparisons.

Main actors in that field :
R. Righter, Z. Liu, J . Shanthikumar, C. Cassandras, T. Rolski. . .

Conclusion

There exists a systematic framework to deal with task-resource systems involving randomness through the theory of stochastic comparisons.

Main actors in that field :
R. Righter, Z. Liu, J . Shanthikumar, C. Cassandras, T. Rolski. . .

Main bibliography for that talk :
Comparison Methods for Stochastic Models and Risks (A. Muller and D. Stoyan, 2002).
Stochastic Modeling and the Theory of Queues (R. W. Wolff, 1989).
Discrete-Event Control of Stochastic Networks : Multimodularity and Regularity (E. Altman, B. Gaujal and A. Hordijk, 2003).