From Systems to Components: Constructive Methods for Product-Form Solutions: other product-forms

Andrea Marin ¹  Maria Grazia Vigliotti²

¹Dipartimento di Informatica
Università Ca’ Foscari di Venezia

²Department of Computing
Imperial College London
Multiple application of (G)RCAT  A class of non-pairwise cooperations are considered. We show how multiple applications of (G)RCAT can still derive the product-form solution when it exists. Case studies: finite capacity queues with *skipping* [Pittel ’79, Balsamo et al. ’10], G-networks with signals [Harrison ’04b].

Extended Reversed Compound Agent Theorem (ERCAT). The Extended Reversed Compound Agent Theorem [Harrison ’04a] is introduced. Applications for cooperations of pairs of automata which do not yield structural conditions of RCAT are shown.
Part I

Multiple applications of RCAT
Outline

1. A mild introduction

2. Finite capacity queues with skipping: the RCAT solution

3. Product-form solution for G-networks with positive...
• Value $K_a$ may be interpreted as the sum of the reversed rates of the active transitions labelled by $a$ incoming into each state

• In case of Birth and Death processes this may be easily computed, i.e.:

$$K_a = \sum_{j=1}^{N} \lambda_j \frac{\mu_i}{\sum_{j=1}^{M} \mu_j}$$
RCAT or GRCAT?

- The Reversed Compound Agent Theorem (RCAT) [Harrison ’03] requires each state to have one incoming active transition for each synchronising label. Value $K_a$ may be interpreted as the (constant) reversed rate of this unique transition.

- The Generalisation (GRCAT) proposed in [Marin et al. ’10] requires each state to have at least one incoming active transition for each synchronising label. Value $K_a$ may be interpreted as the (constant) sum of the reversed rates of these transitions.
Skipping mechanism for queues with finite capacity

- Consider a tandem of exponential queues, $Q_1$ and $Q_2$
- $Q_1$ has a finite capacity $B_1 > 0$
- Customers arrive according to a homogeneous Poisson process at $Q_1$
- If at the arrival epoch $Q_1$ is saturated, the customer immediately enters in $Q_2$
- After service completion in $Q_1$ customers go to $Q_2$
Standard RCAT analysis

- Processes:

- Clearly, the reversed rates of a-transitions are constant, hence $K_a = \lambda$
- Structural (G)RCAT conditions are satisfied
- Steady-state distribution:

$$\pi(n_1, n_2) \propto \left(\frac{\lambda}{\mu_1}\right)^{n_1} \left(\frac{\lambda}{\mu_2}\right)^{n_2}$$

with $0 \leq n_1 \leq B_1, n_2 \geq 0$
Standard RCAT analysis

• Processes:

\[Q_1\]

\[Q_2\]

• Clearly, the reversed rates of \(a\)-transitions are constant, hence \(K_a = \lambda\)

• Structural (G)RCAT conditions are satisfied

• Steady-state distribution:

\[\pi(n_1, n_2) \propto \left(\frac{\lambda}{\mu_1}\right)^{n_1} \left(\frac{\lambda}{\mu_2}\right)^{n_2}\]

with \(0 \leq n_1 \leq B_1, n_2 \geq 0\)
Standard RCAT analysis

- Processes:

\[ \begin{align*}
    Q_1 &: & 0 \xrightarrow{(a, \mu_1)} 1 \xrightarrow{(a, \mu_1)} 2 \xrightarrow{\lambda} B_1 \\
    Q_2 &: & 0 \xrightarrow{(a, x_a)} 1 \xrightarrow{(a, x_a)} 2 \xrightarrow{\mu_2} 1 \xrightarrow{\mu_2} 2 \xrightarrow{\mu_2} 2
\end{align*} \]

- Clearly, the reversed rates of \( a \)-transitions are constant, hence \( K_a = \lambda \)
- Structural (G)RCAT conditions are satisfied
- Steady-state distribution:

\[ \pi(n_1, n_2) \propto \left( \frac{\lambda}{\mu_1} \right)^{n_1} \left( \frac{\lambda}{\mu_2} \right)^{n_2} \text{ with } 0 \leq n_1 \leq B_1, n_2 \geq 0 \]
Standard RCAT analysis

- Processes:

- Clearly, the reversed rates of $a$-transitions are constant, hence $K_a = \lambda$
- Structural (G)RCAT conditions are satisfied
- Steady-state distribution:

$$\pi(n_1, n_2) \propto \left(\frac{\lambda}{\mu_1}\right)^{n_1} \left(\frac{\lambda}{\mu_2}\right)^{n_2} \text{ with } 0 \leq n_1 \leq B_1, n_2 \geq 0$$
Possible generalisation?

- Consider a sequence of $N$ exponential stations $Q_1, \ldots, Q_N$ with finite capacities $B_1, \ldots, B_N$
- Customers arrive at $Q_i$ according to a homogeneous Poisson process with rate $\lambda_i$, $1 \leq i \leq N$
- At a job completion at queue $Q_i$, the customer tries to enter queue $Q_{i+1}$, $1 \leq i < N$
- A customer is allowed to enter $Q_i$ if this is not saturated, or must try to enter $Q_{i+1}$ otherwise, $1 \leq i < N$
- After a job completion at queue $Q_N$ or if this is saturated, customers leave the system
- Note the system is unconditionally stable
Are these pairwise cooperations?

- Each transition in the system may change the state of only two components but...

- Consider the cooperation between $Q_1$ and $Q_3$: an arrival or a job completion at $Q_1$ may generate an arrival at $Q_3$ depending on the state of $Q_2$!

- The cooperation cannot be described only in terms of pairs of queues in isolation

- These cases may still be studied by RCAT with multiple applications
A mild introduction

Finite capacity queues with skipping: the RCAT solution

Product-form solution for G-networks with positive
Cooperating processes

\[ \begin{align*}
Q_1: & (a_{12}, \mu_1), (a_{12}, \lambda_1) \\
Q_2: & (a_{23}, \mu_2), (a_{23}, \lambda_2) \\
Q_3: & (a_{34}, \mu_3), (a_{34}, \lambda_3) \\
Q_N: & (a_{(N-1)N}, \mu_N), (a_{(N-1)N}, \lambda_N), (a_{(N-1)N}, \mu_N) \\
B_1: & (a_{12}, \lambda_1) \\
B_2: & (a_{23}, \lambda_2) \\
B_3: & (a_{34}, \lambda_3) \\
B_N: & (a_{(N-1)N}, \lambda_N) 
\end{align*} \]
Peculiarity of the model

- For $1 < i < N$ a self-loop of state $B_i$ has two roles:
  - it is passive with respect to cooperation label $a_{(i-1)i}$
  - it is active with respect to cooperation label $a_{i(i+1)}$ and has $K_{(i-1)i}$ as a forward rate
- We apply (G)RCAT multiple times adding at each time a new queue
Application of RCAT to the first two queues

RCAT can be applied because:

- Structural conditions on passive transitions are satisfied
- Structural conditions on active transitions are satisfied
- We have $K_{12} = \lambda_1$
Application of RCAT to the first two queues

RCAT can be applied because:

- Structural conditions on passive transitions are satisfied
- Structural conditions on active transitions are satisfied
- We have $K_{12} = \lambda_1$
Application of RCAT to the first two queues

RCAT can be applied because:

- Structural conditions on passive transitions are satisfied
- Structural conditions on active transitions are satisfied
- We have $K_{12} = \lambda_1$
Application of RCAT to the first two queues

RCAT can be applied because:

- Structural conditions on passive transitions are satisfied
- Structural conditions on active transitions are satisfied
- We have $K_{12} = \lambda_1$
• Structurally, the situation is analogue to the previous case

• Note that state $B_2$ has two transitions incoming with the same label $\Rightarrow$ We apply GRCAT and sum the reversed rates obtaining $\lambda_2 + \lambda_1$

• The reversed rate of the death transitions is $\lambda_2 + \lambda_1$ which is the value of $x_{23}$
Steady-state distribution

- Multiple applications of (G)RCAT lead to the following values of the reversed rates:

\[ K_{i(i+1)} = \sum_{\ell=1}^{i} \lambda_i \quad 1 \leq i < N \]

- The steady-state distribution is in product-form:

\[ \pi(n_1, \ldots, n_N) \propto \prod_{\ell=1}^{N} \rho_{\ell}^{n_{\ell}}, \]

with \( 0 \leq n_{\ell} \leq B_{\ell} \) and

\[ \rho_{\ell} = \frac{\sum_{j=1}^{\ell} \lambda_j}{\mu_{\ell}} \]
The result may be easily extended to more general topologies

Does the product-form yield in case of multiple server stations?
  - Yes! ⇒ the reversed rates do not change!

Does the product-form yield in case of negative customers?
  - No! ⇒ the reversed rates of the “death” transitions are different (smaller) from those of the self-loops
  - But if we properly slow-down the arrival rates to saturated queues we may still obtain a product-form solution!
Some notes

- The result may be easily extended to more general topologies
- Does the product-form yield in case of multiple server stations?
  - Yes! ⇒ the reversed rates do not change!
- Does the product-form yield in case of negative customers?
  - No! ⇒ the reversed rates of the “death” transitions are different (smaller) from those of the self-loops
  - But if we properly slow-down the arrival rates to saturated queues we may still obtain a product-form solution!
Some notes

- The result may be easily extended to more general topologies.
- Does the product-form yield in case of multiple server stations?
  - Yes! ⇒ the reversed rates do not change!
- Does the product-form yield in case of negative customers?
  - No! ⇒ the reversed rates of the “death” transitions are different (smaller) from those of the self-loops.
  - But if we properly slow-down the arrival rates to saturated queues we may still obtain a product-form solution!
A mild introduction

Finite capacity queues with skipping: the RCAT solution

Product-form solution for G-networks with positive
Model description

- Network of $N$ exponential queues $Q_1, \ldots, Q_N$ with external Poisson customer arrivals with rate $\lambda_i$ and service rate $\mu_i$

- At a job completion at $Q_i$ a customer can:
  - go to queue $Q_j$, $j \neq i$, with probability $P_{ij}^+$ as a standard customer
  - go to queue $Q_j$, $j \neq i$, with probability $P_{ij}^-$ as a trigger
  - leave the system with probability $1 - \sum_j (P_{ij}^+ + P_{ij}^-)$

- At a trigger arrival at $Q_j$ it:
  - vanishes if $Q_j$ is empty
  - removes a customer from $Q_j$ and add a customer to $Q_k$, $k \neq j$, with probability $R_{jk}$, if $Q_j$ is non-empty
  - removes a customer from $Q_j$ with probability $1 - \sum_k R_{jk}$, if $Q_j$ is non-empty
• The picture shows just the cooperation among three queues \( Q_i, Q_j, Q_k \) embedded in a general network.
• We focus on the analysis of the trigger behaviours.
• Positive customer analysis is the same of Jackson’s networks.
• A job completion in \( Q_i \) may change the state of three queues simultaneously: \( Q_i, Q_j, Q_k \).
Process underlying a generic queue $Q_i$

- $1 \leq j \leq N$, $j \neq i$
- $a_{ij}^+$: positive customer from $Q_i$ to $Q_j$
- $a_{ij}^-$: trigger from $Q_i$ to $Q_j$
- $b_{ij}$: customer arrival at $Q_j$ caused by a trigger arrival at queue $Q_i$
- We set up the RCAT traffic equations by the analysis of each queue in isolation
- This operation can be done algorithmically
Queue 1

\[
Ka^-_{12} = (\lambda_1 + Ka^+_3 + Ka^+_2)P^-_{12}
\]

\[
Ka^+_{12} = (\lambda_1 + Ka^+_3 + Ka^+_2)P^+_{12}
\]
Andrea Marin

Queue 2

\[
Q_2 \quad 0 \quad 1
\]

\[
(a_{12}^+, xa_{12}^-) \quad (a_{12}^-, xa_{12}^-) \quad (a_{21}^+, \mu_2 P_{21}^+) \quad (a_{21}^-, \mu_2 (1 - P_{21}^+)) \quad \frac{(a_{12}^-, xa_{12}^-)}{(b_{23}, Ka_{12}^-)}
\]

\[
Ka_{21}^+ = \frac{Ka_{12}^+}{\mu_2 + Ka_{12}^-} \mu_2 P_{21}^+
\]

\[
Kb_{23} = \frac{Ka_{12}^+}{\mu_2 + Ka_{12}^-} Ka_{12}^-
\]

Andrea Marin

University of Venice
Queue 3

\[ Q_3 \]

\[ (b_{23}, x_{b_{23}}) \]

\[ \lambda_3 \]

\[ (a_{31}^+, \mu_3) \]

\[ K a_{31}^+ = \lambda_3 + K b_{23} \]
Concluding the example

- The solution of the traffic equations straightforwardly gives the product-form solution
- The traffic equations may be solved either symbolically or numerically
- The algorithm presented in [Marin et al. ’09] applies an iterative schema to efficiently solve such networks of queues
- The approach may be extended to deal with negative triggers (at a trigger arrival the receiving non-empty queue may send a trigger to another queue)
Part II

Extended Reversed Compound Agent Theorem (ERCAT)
Motivations by example

The theorem

Solution of the running example

Open networks of exponential queues with finite capacity and blocking

Conclusion
A system in Boucherie’s product-form

- Two exponential queues $Q_1$ and $Q_2$ with independent Poisson arrival streams with rate $\lambda_1$ and $\lambda_2$
- Service rates are $\mu_1$ and $\mu_2$
- If one of the queues enters in state 0 the other one is blocked (i.e. no arrivals or service completions occur)
- The model is known to be in Boucherie’s product-form
Are (G)RCAT structural conditions satisfied? **NO!**
Motivations

The theorem

Running example

Networks with blocking

Conclusion

Process representation

Are (G)RCAT structural conditions satisfied? NO!
Joint state space

- ERCAT requires to check a rate equation for each state of the irreducible subset of the joint process.
- Often, states can be opportunely clustered and hence the computation becomes feasible.
- The computational complexity is higher than the standard (G)RCAT.
- Let \((s_1, s_2)\) be a state of the irreducible subset of the joint process.
Fundamental definitions

- $\mathcal{P}(s_1, s_2) \rightarrow$: outgoing labels from $s_1$ or $s_2$
- $\mathcal{P}(s_1, s_2) \leftarrow$: incoming passive labels into $s_1$ or $s_2$
- $\mathcal{A}(s_1, s_2) \rightarrow$: outgoing active labels from $s_1$ or $s_2$
- $\mathcal{A}(s_1, s_2) \leftarrow$: incoming active labels into $s_1$ or $s_2$
- $\alpha(s_1, s_2)(a)$: rate of transition labelled by $a$ outgoing from $(s_1, s_2)$
- $\beta(s_1, s_2)(a)$: reversed rate of the passive transition labelled by $a$ incoming into $(s_1, s_2)$
Theorem (ERCAT)

Given two models $Q_1$ and $Q_2$ in which RCAT structural conditions are not satisfied but the reversed rates of the active transitions are constant, their cooperation is in product-form if the following rate equation is satisfied for each state $(s_1, s_2)$ of the irreducible subset of states of the joint process:

$$
\sum_{a \in P^{(s_1, s_2)}_\rightarrow} x_a - \sum_{a \in A^{(s_1, s_2)}_\leftarrow} x_a = \sum_{a \in P^{(s_1, s_2)}_\leftarrow \setminus A^{(s_1, s_2)}_\leftarrow} \beta_a^{(s_1, s_2)} - \sum_{a \in A^{(s_1, s_2)}_\rightarrow \setminus P^{(s_1, s_2)}_\rightarrow} \alpha_a^{(s_1, s_2)}
$$
Motivations by example

The theorem

Solution of the running example

Open networks of exponential queues with finite capacity and blocking

Conclusion
State \((0,0)\)

\[
\begin{align*}
\mathcal{P}^{(0,0)\rightarrow} &= \emptyset & \mathcal{A}^{(0,0)\leftarrow} &= \{a, b\} \\
\mathcal{P}^{(0,0)\leftarrow} - \mathcal{A}^{(0,0)\leftarrow} &= \emptyset & \mathcal{A}^{(0,0)\rightarrow} - \mathcal{P}^{(0,0)\rightarrow} &= \{c, d\} \\
-x_a - x_b &= -\alpha_c^{(0,0)} - \alpha_d^{(0,0)} & \text{Ok}
\end{align*}
\]

Note that:

\[
x_a = \lambda_1, \quad \alpha_c^{(0,0)} = \lambda_1, \quad x_b = \lambda_2, \quad \alpha_d^{(0,0)} = \lambda_2
\]
State (0,0)

\[ P(0,0) \rightarrow = \{\} \quad A(0,0) \leftarrow = \{a, b\} \]
\[ P(0,0) \leftarrow \setminus A(0,0) \leftarrow = \{\} \quad A(0,0) \rightarrow \setminus P(0,0) \rightarrow = \{c, d\} \]
\[ -x_a - x_b = -\alpha_c^{(0,0)} - \alpha_d^{(0,0)} \quad \text{Ok} \]

Note that:

\[ x_a = \lambda_1, \quad \alpha_c^{(0,0)} = \lambda_1, \quad x_b = \lambda_2, \quad \alpha_d^{(0,0)} = \lambda_2 \]
State \((0,n), n>0\)

\[
\begin{align*}
\mathcal{P}^{(0,n)} & \rightarrow = \{a, c\} & \mathcal{A}^{(0,n)} & \leftarrow = \{a, b, d\} \\
\mathcal{P}^{(0,n)} & \leftarrow \setminus \mathcal{A}^{(0,n)} & \mathcal{A}^{(0,n)} & \rightarrow \setminus \mathcal{P}^{(0,n)} = \{b, d\} \\
x_a + x_c - x_a - x_b - x_d & = \beta^{(0,n)}_c - \alpha^{(0,n)}_b - \alpha^{(0,n)}_d \\
\text{Note that:} & \\
x_b & = \lambda_2, \quad x_c = \mu_1, \quad x_d = \mu_2, \quad \beta^{(0,n)}_c = \mu_1, \quad \alpha^{(0,n)}_b = \mu_2, \quad \alpha^{(0,n)}_d = \lambda_2
\end{align*}
\]
State \((0,n), n > 0\)

\[
P(0,n) \rightarrow = \{a, c\} \quad A(0,n) \leftarrow = \{a, b, d\}
\]

\[
P(0,n) \leftarrow \setminus A(0,n) \leftarrow = \{c\} \quad A(0,n) \rightarrow \setminus P(0,n) \rightarrow = \{b, d\}
\]

\[
x_a + x_c - x_a - x_b - x_d = \beta_c^{(0,n)} - \alpha_b^{(0,n)} - \alpha_d^{(0,n)} \quad \text{Ok!}
\]

Note that:

\[
x_b = \lambda_2, \quad x_c = \mu_1, \quad x_d = \mu_2, \quad \beta_c^{(0,n)} = \mu_1, \quad \alpha_b^{(0,n)} = \mu_2, \quad \alpha_d^{(0,n)} = \lambda_2
\]
State \((m,n), \ m, n > 0\)

\[
\begin{align*}
\mathcal{P}(m,n) \rightarrow &= \{a, b, c, d\} & \mathcal{A}(m,n) \leftarrow &= \{a, b, c, d\} \\
\mathcal{P}(m,n) \leftarrow & \setminus \mathcal{A}(m,n) \leftarrow = \{\} & \mathcal{A}(m,n) \rightarrow & \setminus \mathcal{P}(m,n) \rightarrow = \{\}
\end{align*}
\]

0=0

Note that states \((m, 0)\) with \(m > 0\) are similar to \((0, n)\), \(n > 0\).
Conclusion of the running example

- The model, as expected, is in product-form:
  \[ \pi(m, n) \propto \left( \frac{\lambda_1}{\mu_1} \right)^m \left( \frac{\lambda_2}{\mu_2} \right)^n \]

- Note that state \((0, 0)\) is either the only ergodic state or does not belong to the irreducible subset.

- Hence, the normalising constant distinguishes this solution from the case of independent queues.

- Every Boucherie’s product-form with full blocking can be studied by ERCAT [Harrison ’04a].
Motivations by example

The theorem

Solution of the running example

Open networks of exponential queues with finite capacity and blocking

Conclusion
Queues with finite capacity and Repetitive Service (RS) blocking

- We consider a network of queues, $Q_1, \ldots, Q_N$ with finite capacity $B_i$ and service rate $\mu_i$.
- At a job completion at $Q_i$ the customer goes to $Q_j$ with probability $P_{ij}$. If $Q_j$ is saturated the customer service is restarted and a new target station is selected at job completion.
- In open networks $\lambda_i$ is the arrival rate at $Q_i$ and customers leave the system with probability $1 - \sum_j P_{ij}$. Arrivals at saturated queues are not allowed.
Example

Motivations
The theorem
Running example
Networks with blocking
Conclusion

Andrea Marin

University of Venice
• Differently from ordinary queueing networks we use active transitions to model synchronised arrivals and passive to model synchronised departures

• Which states shall we consider?
  1 \((0,0)\)
  2 \((0,K)\) with \(0 < K < B_2\) (and symmetrically we obtain \((K,0)\) with \(0 < K < B_1\))
  3 \((0,B_2)\)
  4 \((K,B_2)\) with \(0 < K < B_1\) (and symmetrically we obtain \((0,K)\) with \(0 < K < B_2\))
  5 \((B_1,B_2)\)

• Note that \(\alpha_a^{(\cdot,\cdot)} = q\mu_2\), \(\alpha_b^{(\cdot,\cdot)} = p\mu_1\) and also \(\beta_a^{(\cdot,\cdot)} = \bar{\beta}_a\) and \(\beta_b^{(\cdot,\cdot)} = \bar{\beta}_b\)
State \((0, B_2)\)

\[
\begin{align*}
\mathcal{P}(0,B_2) \rightarrow &= \{a\} & \mathcal{A}(0,B_2) \leftarrow &= \{b\} \\
\mathcal{A}(0,B_2) \rightarrow \setminus \mathcal{P}(0,B_2) \rightarrow &= \{\} & \mathcal{P}(0,B_2) \leftarrow \setminus \mathcal{A}(0,B_2) \leftarrow &= \{\}
\end{align*}
\]

\[
x_a - x_b = 0 \Rightarrow x_a = x_b \tag{1}
\]
State \((0, K)\)

\[
\begin{align*}
\mathcal{P}_{(0,K)\rightarrow} &= \{a\} & \mathcal{A}_{(0,K)\leftarrow} &= \{b\} \\
\mathcal{A}_{(0,K)\rightarrow} \setminus \mathcal{P}_{(0,K)\rightarrow} &= \{b\} & \mathcal{P}_{(0,K)\leftarrow} \setminus \mathcal{A}_{(0,K)\leftarrow} &= \{a\}
\end{align*}
\]

i.e.:

\[
x_a - x_b = \beta_{(0,K)}^{(0,K)} - \alpha_{(0,K)}^{(0,K)} \quad (1) \quad \beta_{(0,K)}^{(0,K)} \rightarrow \beta_{b} = \alpha_{a}
\]

By symmetry, state \((K, 0)\) gives \(\beta_{a} = \alpha_{b}\)
Motivations
The theorem
Running example
Networks with blocking
Conclusion

State \((0, 0)\)

\[
\begin{array}{c}
\lambda_1 \\
(a, q \mu_2) \\
(1 - p) \mu_1 \\
\lambda_2 \\
(b, p \mu_1) \\
(1 - q) \mu_2 \\
\end{array}
\]

\[
\begin{align*}
P^{(0,0)} & = \{\} \\
A^{(0,0)} & = \{\}
\end{align*}
\]

\[
A^{(0,0)} \setminus P^{(0,0)} = \{a, b\} \\
P^{(0,0)} \setminus A^{(0,0)} = \{a, b\}
\]

\[
\beta_a^{(0,0)} + \beta_b^{(0,0)} = \alpha_a^{(0,0)} + \alpha_b^{(0,0)}
\]

which is a consequence of \((2)\)
States \((K, B_2)\) and \((B_1, K), (B_1, B_2)\)

For these states we have:

- \(\mathcal{P}(\cdot, \cdot)\rightarrow = \{a, b\}\)
- \(\mathcal{A}(\cdot, \cdot)\leftarrow = \{a, b\}\)

Since all the synchronising labels are present in both these sets, the rate equation for these states is an identity.
Conditions derived from the ERCAT rate equations

\[
\begin{cases}
\beta_b = \alpha_a = q\mu_2 \\
\beta_a = \alpha_b = p\mu_1
\end{cases}
\]

The process analysis gives:

\[
\begin{align*}
\beta_b &= \frac{x_b(\lambda_1 + q\mu_2)}{x_b + (1 - p)\mu_1} \\
\beta_a &= \frac{x_a(\lambda_2 + p\mu_1)}{x_a + (1 - q)\mu_2}
\end{align*}
\]

From which we straightforwardly derive:

\[
\begin{align*}
x_a &= \frac{(1 - q)p\mu_1\mu_2}{\lambda_2} \\
x_b &= \frac{(1 - p)q\mu_1\mu_2}{\lambda_1}
\end{align*}
\]
Product-form rate condition

Since $x_a = x_b$ by (1) we have the product-form rate condition:

$$(1 - p)q\lambda_2 = (1 - q)p\lambda_1$$

Under this assumption expressions (3) for $x_a, x_b$ satisfies:

$$x_a = \frac{(x_b + (1 - p)\mu_1)q\mu_2}{\lambda_1 + q\mu_2} \quad x_b = \frac{(x_a + (1 - q)\mu_2)p\mu_1}{\lambda_2 + p\mu_1}$$
• ERCAT may be applied to a set of agent with pairwise cooperations (this is also known a MARCAT)
• In case of QN with RS blocking and general topology in [Balsamo et al. ’10] is proved that:

Theorem

A QN (open or closed) with finite capacity stations and RS blocking policy with reversible routing matrix always satisfies ERCAT rate equations.

• Product-form for reversible routing has been proved in [Akyildiz ’87]
Closed QN with RS blocking

- Consider a closed QN with RS blocking policy.
- Note that the ERCAT rate equation is an identity for state $n$ when none of the stations is empty in $n$.
- We immediately have the following result:

**Theorem (QN with strict non-empty condition)**

A closed QN with finite capacity stations and RS blocking is in product-form if the number of customers is such that none of the station can be empty (strict non-empty condition).

- In [Balsamo et al. ’10] we prove that the same result for QN in which at most one station can be empty (non-empty condition).
4 Motivations by example

5 The theorem

6 Solution of the running example

7 Open networks of exponential queues with finite capacity and blocking

8 Conclusion
In the second part of the tutorial we have shown how to overcome some limitations of original RCAT and GRCAT formulation.

- In case of some non-pairwise cooperations we can apply (G)RCAT iteratively to obtain the product-form.
- In case structural conditions of (G)RCAT are not satisfied, we may apply ERCAT.
- Application of (G)RCAT or ERCAT may be done algorithmically, however the computational cost of ERCAT is higher than that of (G)RCAT.
Applications

- Other models than those presented here may be studied by RCAT and its extensions (e.g. product-form Stochastic Petri Nets)
- New product-form may be derived
- The solution of the traffic equations may be efficiently computed by means of the algorithm presented in [Marin et al. ’09]
  - Numerical and iterative algorithm
- Product-form of models expressed in terms of different formalisms may be derived.
Appendix: Reversible routing matrix

- Consider a queueing network with $N$ stations and fixed routing probability matrix $P = [p_{ij}]$, $1 \leq i, j \leq N$
- $p_{i0}$ is the probability of leaving the network after a job completion at station $i$
- $e_i$ is the (relative) visit ratio to station $i$
- $\lambda_i$ is the arrival rate at station $i$

Definition (Reversible routing matrix)

The routing matrix $P$ is said reversible if:

\[
\begin{align*}
    e_i p_{ij} &= e_j p_{ji} \quad \text{for } 1 \leq i, j \leq N \\
    \lambda_i &= e_i p_{i0} \quad \text{for } 1 \leq i \leq N
\end{align*}
\]


<table>
<thead>
<tr>
<th>Section</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motivations</td>
<td>Andrea Marin</td>
</tr>
<tr>
<td>The theorem</td>
<td>For Further Reading III</td>
</tr>
<tr>
<td>Running example</td>
<td>A. Marin, M.G. Vigliotti: A general result for deriving product-form</td>
</tr>
<tr>
<td></td>
<td>solutions of Markovian models, Proc. of WOSP/SIPEW Int. Conf. on Perf.</td>
</tr>
<tr>
<td></td>
<td>Eval.</td>
</tr>
<tr>
<td>Networks with blocking</td>
<td>S. Balsamo, P. G. Harrison, A. Marin: A unifying approach to product-</td>
</tr>
<tr>
<td></td>
<td>forms in networks with finite capacity constraints, Proc. of the 2010</td>
</tr>
<tr>
<td></td>
<td>ACM SIGMETRICS, pp. 25–36, 2010</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Andrea Marin University of Venice</td>
</tr>
</tbody>
</table>