



[Home Page](#)

[Title Page](#)



Page 1 of 31

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)

Workforce planning and call routing in multi-skill call centers (part I) — Optimization and concavity in call centers (part II)

Ger Koole, Vrije Universiteit Amsterdam
(joint work with Geert-Jan Franx and Auke Pot)

Call center workshop, Montréal, 25 July 2004



Pros and cons multiple skills

Specialized agents:

- need less training
- are more efficient
- are easier to manage

However:

- loss of economies of scale
- loss of flexibility

Thus: **some** cross-trained agents

[Home Page](#)

[Title Page](#)



Page 2 of 31

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



Numerical illustration

- Two skills, specialists and cross-trained agents
- $\lambda_1 = 5.5$, $\lambda_2 = 4.5$, $\beta_1 = \beta_2 = 5$, $\beta_{12} = 5.5$
- 80/20 SL
- No cross-trained agents: $33+28 = 61$ (cheap) agents needed (Erlang C)
- Only c-t agents: 62 (expensive) agents needed (Erlang C)
- Both: $22 + 18 + 18 = 58$ agents is optimal (simulation, # agents minimized)

Robustness: Same solution almost optimal if λ_i changes
(but $\sum \lambda_i$ fixed)

Home Page

Title Page



Page 3 of 31

Go Back

Full Screen

Close

Quit



Two approaches

Approach 1: Skill-based routing

Useful if:

- Reduced efficiency loss
- Many small skills

Approach 2: Dedicated (but cross-trained) agents

Useful if:

- High load for skills
- Focus on efficiency
- Unpredictable load variations

Can be used in combination!

Home Page

Title Page



Page 4 of 31

Go Back

Full Screen

Close

Quit



Objectives talk

- Discuss medium-term scheduling issues (workforce mgmt) under unpredictable load variations (relevant to approach 2 and 1)
- Technical results skill-based routing (relevant to approach 1)

Ongoing work!

Note: current practice WFM-tools is simulation (or worse)

[Home Page](#)

[Title Page](#)



Page 5 of 31

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



Scheduling under load variations: single skill

- Load represented by Poisson process with piece-wise constant rate
- Current practice: point estimate for rate, thus no load unpredictability and constant absenteeism (**shrinkage**)
- However: Actual load overdispersed w.r.t. Poisson process (Jongbloed & K)
- Load at scheduling moment unpredictable

Solution: **Flexibility**

Home Page

Title Page



Page 6 of 31

Go Back

Full Screen

Close

Quit



Scheduling under load variations: example

- $\lambda = 10$, $\beta = 5$, 80/20 SL: 57 agents
- absenteeism average 8%: 62 agents scheduled
- but: $\lambda \in [9, 11]$, shrinkage $\in [5, 10]\%$
- thus: $\lambda = 11$, shrinkage = 10%: 20% SL
- or: $\lambda = 9$, shrinkage = 5%: 74% productivity
- Solution: 54 fixed agents + 14 flexible agents

Simple but (for practice) highly innovative!

Home Page

Title Page



Page 7 of 31

Go Back

Full Screen

Close

Quit



Scheduling under load variations: multiple skills

- Cross-trained agents \Rightarrow Cross-skill flexibility
- Correlation load fluctuations has high impact

Example: claims at insurance company

- Stopped with skill-based routing
- Still many cross-trained agents, gives flexibility in case of:
 - storm (extra home insurance claims)
 - weather conditions (car insurances)
 - holiday season (travel insurances)

[Home Page](#)

[Title Page](#)



Page 8 of 31

[Go Back](#)

[Full Screen](#)

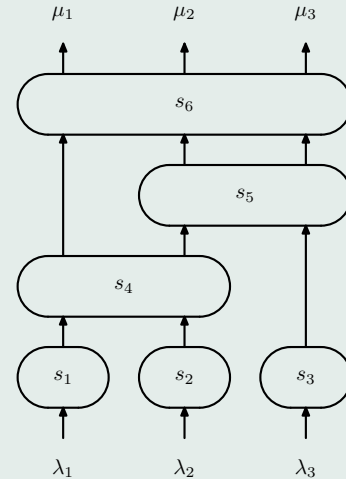
[Close](#)

[Quit](#)



Skill-based routing

- Without queues
- Multi-skill
- Multiple groups
- **Overflow routing**
(industry standard)



What are the blocking probabilities?

- No analytic solution!
- How can we tackle the curse of dimensionality?

Home Page

Title Page



Page 9 of 31

Go Back

Full Screen

Close

Quit



Important aspects

- The blocking model is a simplification: smaller state-space, agent-to-call policy is irrelevant, easier to analyze
- Exact analysis is impossible, because the state-space is multi-dimensional
- Overflow processes become less difficult, but are most often not renewal \Rightarrow still untractable
- Model limitation: it must be possible to arrange all arrows in the figure in such a way that they all point upwards

[Home Page](#)

[Title Page](#)



Page 10 of 31

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



Literature

- Koole & Talim 2000: assumes that the overflow streams are Poisson:
⇒ The burstiness is underestimated
- Cooper 1981: the equivalent random method The call center is decomposed by replacing groups in the same layer by an equivalent group (in the sense that the mean and variance of the overflow processes are equal)
⇒ Only possible to tackle call centers with a special type of overflow routing
- Chevalier & Tabordan: results parallel to ours

[Home Page](#)

[Title Page](#)



Page 11 of 31

[Go Back](#)

[Full Screen](#)

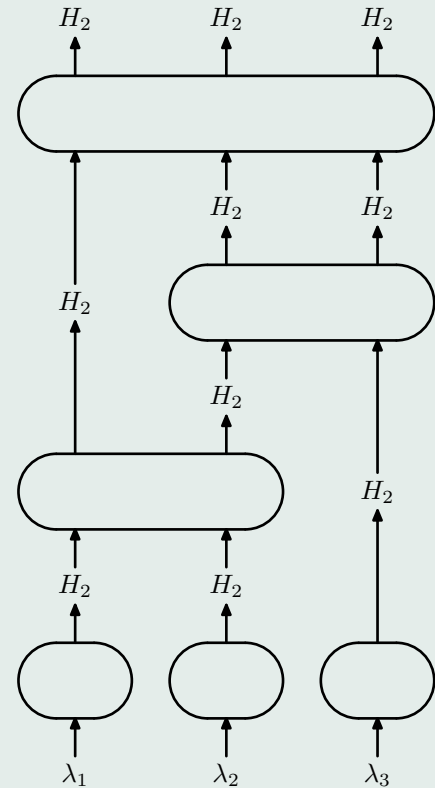
[Close](#)

[Quit](#)



Our approach

- Approximate interoverflow times by H_2 distributions (in the model these are not renewal processes)
- H_2 has 3 parameters \Rightarrow 3 moments needed (calculation: Tijms 86)



Home Page

Title Page



Page 12 of 31

Go Back

Full Screen

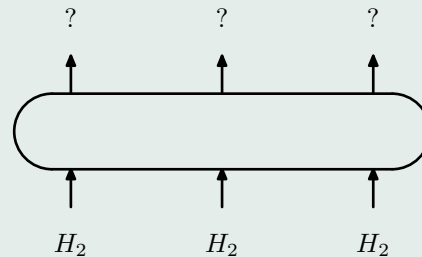
Close

Quit



Hyperexponential decomposition

- Consider a decomposed agent group, having multiple arrival streams with hyperexponential times
- We need approximations for the overflow processes to the next groups



- 1st moment (blocking probability) from Markov process
- 2nd and 3rd moments from 'equivalent' $M/M/s/s$ system

Home Page

Title Page



Page 13 of 31

Go Back

Full Screen

Close

Quit



Home Page

Title Page



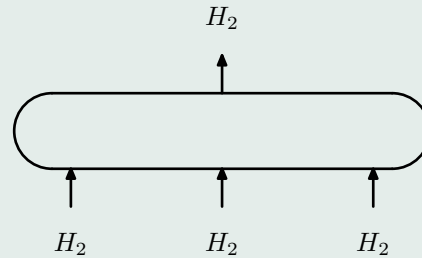
Page 14 of 31

Go Back

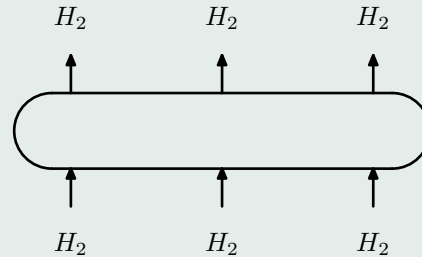
Full Screen

Close

Quit



- Split the stream to make a distinction between the groups in the next layers





Higher moments of the $M/M/s/s$ system

Methods are:

- Riordan '61 or
- simulation

How to circumvent long computation times? Pre-calculation:

- Fill a table with the moments for different combinations of a and s (discretize a in advance)
- Obtain during the algorithm moments for any a and s by interpolation

Home Page

Title Page

◀◀

▶▶

◀

▶

Page 15 of 31

Go Back

Full Screen

Close

Quit



Algorithm

HYPEREXPONENTIAL DECOMPOSITION()

- 1 determine the level of each group, $0, 1, \dots, L$
- 2 **for** level 0 up to L
- 3 **do for** each group in current level
- 4 **do** - calculate the weighted average service rate
- 5 - calculate analytically the blocking
- 6 probability
- 7 - determine the second and third moment
- 8 of the overflow process.
- 9 - approximate the fitted overflow processes
- 10 to the next groups

Home Page

Title Page



Page 16 of 31

Go Back

Full Screen

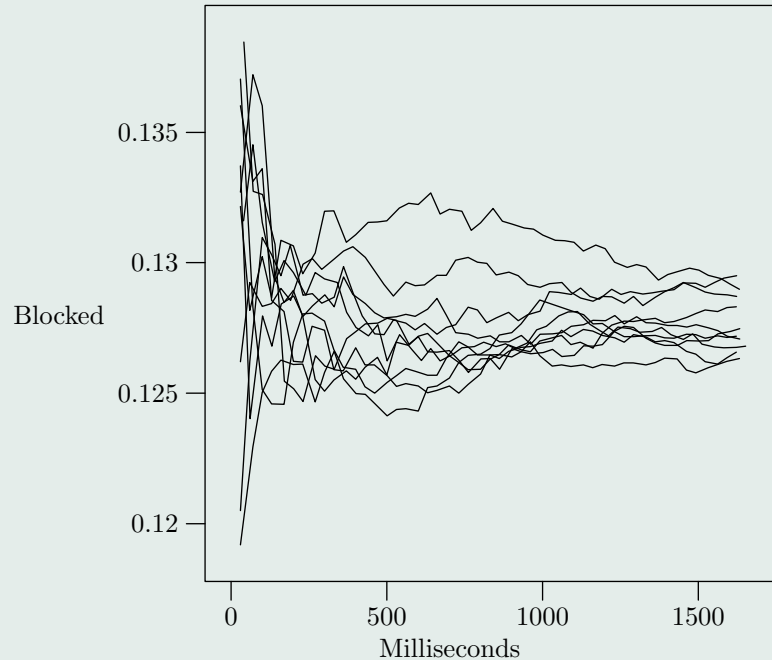
Close

Quit



Computation times

- Sample paths of 10 simulation runs:



- Decomposition algorithm takes only a few milliseconds

Home Page

Title Page



Page 17 of 31

Go Back

Full Screen

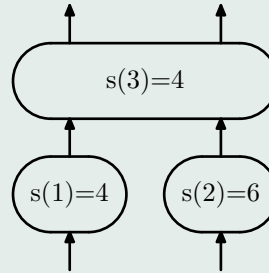
Close

Quit



Numerical results (1)

$$\mu(1) = 1.75 \quad \mu(2) = 0.6$$



$$\lambda(1) = 3 \quad \lambda(2) = 5$$

method	skill 1	skill 2	average
exact	.0201	.1409	.0956
simulation	.0200	.1406	.0954
Poisson	.0175	.1065	.0731
H_2	.0199	.1400	.0946

Home Page

Title Page



Page 18 of 31

Go Back

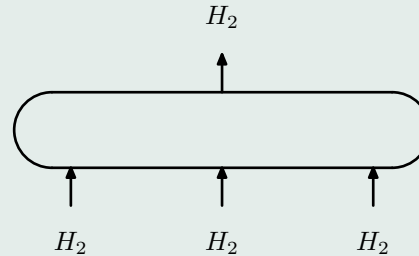
Full Screen

Close

Quit



Numerical results (2)



method	skill 1	2	3	4	average
exact	-	-	-	-	-
simulation	.1364	.1367	.1369	.1377	.1369
Poisson	.1092	.1092	.1092	.1092	.1092
H_2	.1374	.1374	.1374	.1374	.1374
ERM	.1397	.1397	.1397	.1397	.1397

Home Page

Title Page



Page 19 of 31

Go Back

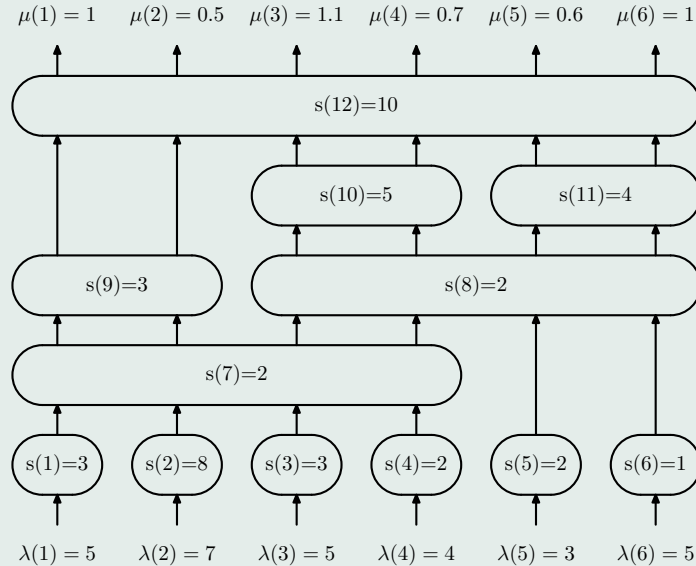
Full Screen

Close

Quit



Numerical results (3)



method	skill 1	2	3	4	5	6	average
exact	-	-	-	-	-	-	-
simulation	0.0961	0.0981	0.0422	0.0538	0.0896	0.1048	0.0823
Poisson	0.0730	0.0676	0.0227	0.0325	0.0658	0.0811	0.0581
H_2	0.1093	0.1018	0.0407	0.0576	0.0922	0.1135	0.0875
ERM	-	-	-	-	-	-	-

Home Page

Title Page



Page 20 of 31

Go Back

Full Screen

Close

Quit



Extensions

- No problem to implement group-dependent service rates (relevant!)
- Randomized overflow routing is easy to implement

Home Page

Title Page



Page 21 of 31

Go Back

Full Screen

Close

Quit



Further research

- Approximation of delay probabilities
- Approximating the waiting time distributions per skill type
- Single-skill blocking model is related to delay model by

$$C(s, a) = \frac{sB(s, a)}{s - a(1 - B(s, a))} \quad (\text{Cooper 1981})$$

Extend this to multiple skills (Koole, Pot & Talim 03)

Home Page

Title Page



Page 22 of 31

Go Back

Full Screen

Close

Quit



Part II: Optimization and concavity in call centers

Ongoing work, references to be added

Final version to be presented at:

Joint GOR/NGB conference, Tilburg, 3 September 2004

[Home Page](#)

[Title Page](#)



Page 23 of 31

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



What are we interested in?

Situation:

- Single-skill call center
- Erlang C, Erlang A, finite # of lines
- PIs: SL, costs, profit, ...

We look at concavity of PIs in number of agents

[Home Page](#)

[Title Page](#)



Page 24 of 31

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



Why is this relevant?

Numerical example:

- Staffing for 2 intervals with $\lambda_1 \neq \lambda_2$
- Objective: weighted average SL \leq some fixed level
- Algorithm: start empty, add agent to interval that increases objective the most
- Optimal if SL concave in # agents
- Note: Concavity \Leftrightarrow Diminishing returns

[Home Page](#)

[Title Page](#)



Page 25 of 31

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



What is known?

Erlang C:

- $W_q(s)$: waiting time in queue, s servers
- $\mathbb{P}(W_q(s) \leq t)$ concave for $s > \lambda\beta$
(Jagers & v Doorn 91)

Erlang A (Erlang C with exponential abandonments):

- $W_q(s)$: waiting time in queue customer with ∞ patience
- Counterexample: $\mathbb{P}(W_q(s) \leq t)$ not concave
(Wattel 04, unpublished)

Home Page

Title Page



Page 26 of 31

Go Back

Full Screen

Close

Quit



What's wrong?

SL definition:

- Mathematics: when objective should be concave, then direct reward "should" be concave
direct reward $\mathbb{I}(W_q(s) \leq t)$ is 0/1 function
- Practice: you don't care about those who wait $> t$ seconds
"rational" call center mgmt ignores these calls
- "Better" SL definition: $\mathbb{E}(W_q(s) - t)^+$

Results:

- Erlang C: again concave in s
- Erlang A: concave in s if $\gamma \leq \mu$
no results (yet) for $\gamma > \mu$

Home Page

Title Page



Page 27 of 31

Go Back

Full Screen

Close

Quit



From cost to profit model

Different business model, different objective:

- Reward r per call
- Cost c per call per time unit
- Cost 1 per agent per time unit
- Maximize profit

Control:

- Admission control (= limiting # lines)
- Determine # agents
- How to find optimal solution?

Home Page

Title Page



Page 28 of 31

Go Back

Full Screen

Close

Quit



How difficult is this problem?

Fixed s : threshold policy is optimal (dp)

Define:

- $g^{s,n}$ average long-run expected profit for s and n waiting lines (thus total $s + n$ lines)
- $g^s = \max_n g^{s,n}$ average long-run expected profit for s and optimal threshold

g^s not concave \Rightarrow no diminishing returns

Home Page

Title Page

◀◀

▶▶

◀

▶

Page 29 of 31

Go Back

Full Screen

Close

Quit



Results

- $g^{s,n}$ increasing up to threshold n
- For $n_s = \arg \max_n g^{s,n}$: $n_{s+1} \geq n_s$ (dp)

Defines search strategy, $O(S + N)$ b-d processes to be analyzed (instead of $O(SN)$)

To be implemented in web tool:

www.math.vu.nl/~koole/ccmath

Home Page

Title Page



Page 30 of 31

Go Back

Full Screen

Close

Quit



Thank you for your attention!

Websites:

- <http://www.math.vu.nl/~koole>
- <http://www.math.vu.nl/obp/callcenters>

Focused issue Mgmt Sc on call centers:

- Deadline: October 1, 2004
- Link to call for papers on <http://www.math.vu.nl/~koole>

Home Page

Title Page



Page 31 of 31

Go Back

Full Screen

Close

Quit