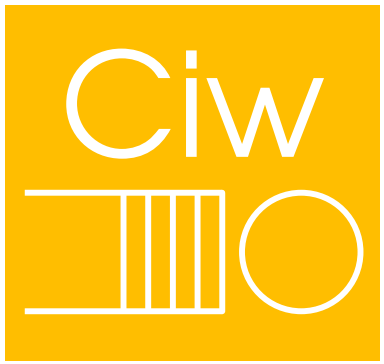
The Ciw Python Library

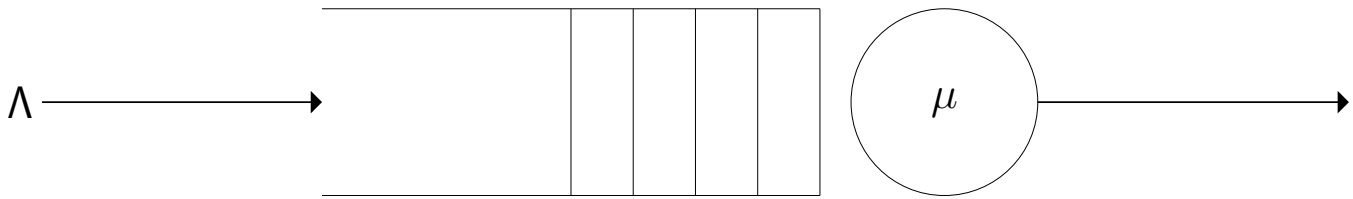
Geraint Ian Palmer
@GeraintPalmer @CiwPython

17th April 2018

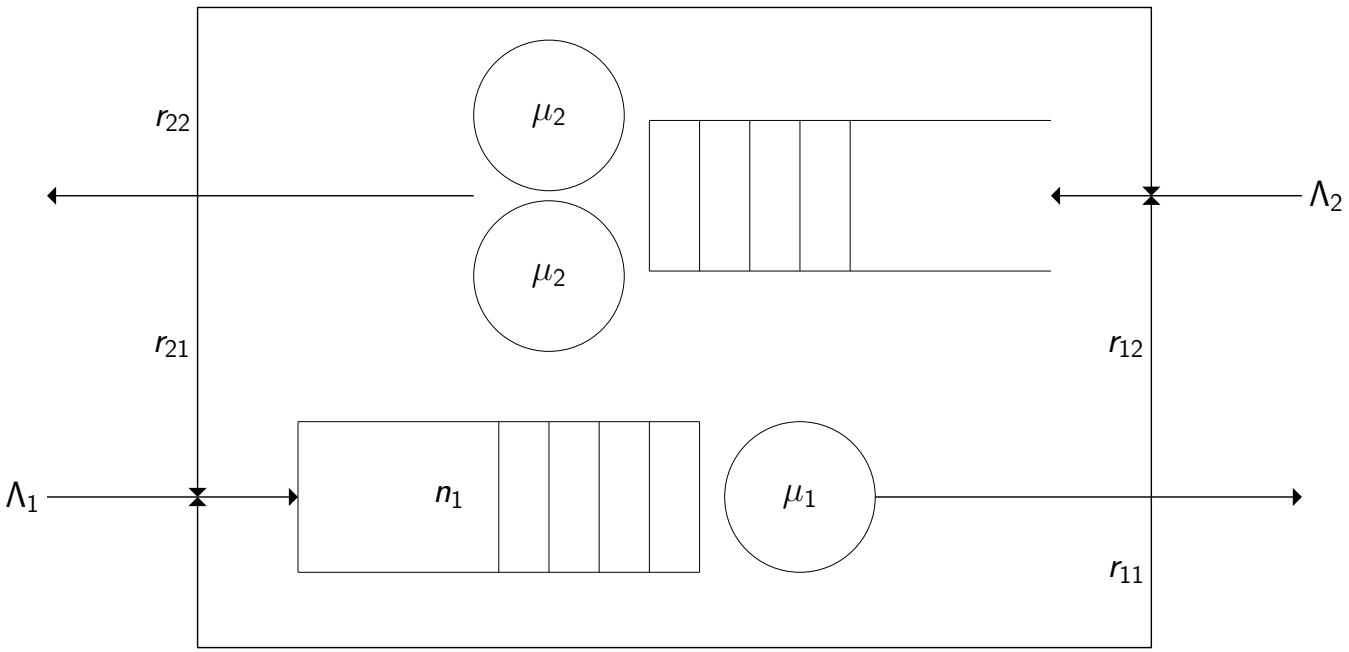



GAME

What is a Queue?



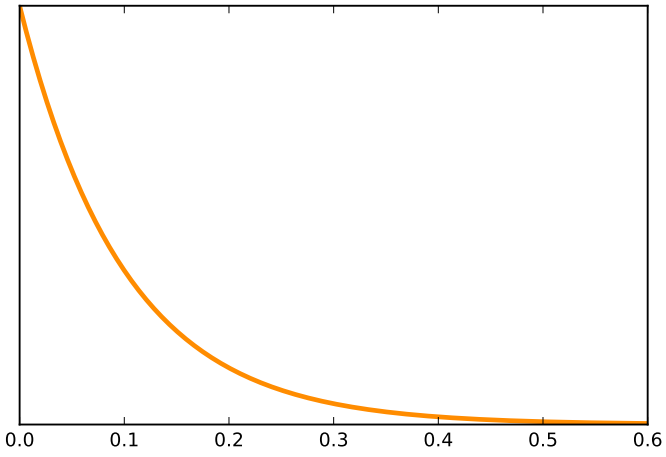
What is a Queue?

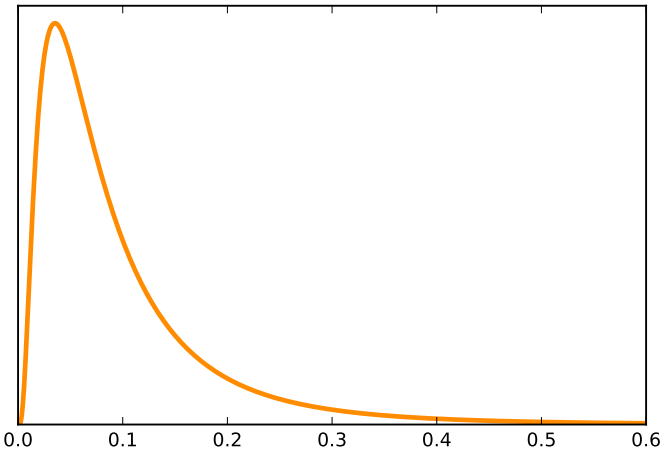


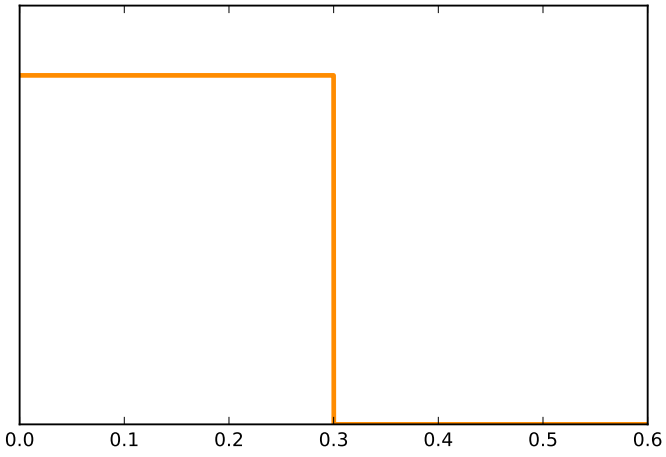
Features

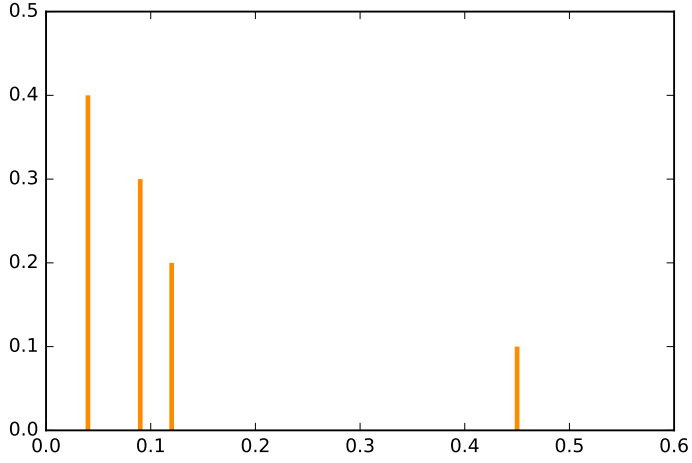
http://ciw.readthedocs.io

Distributions

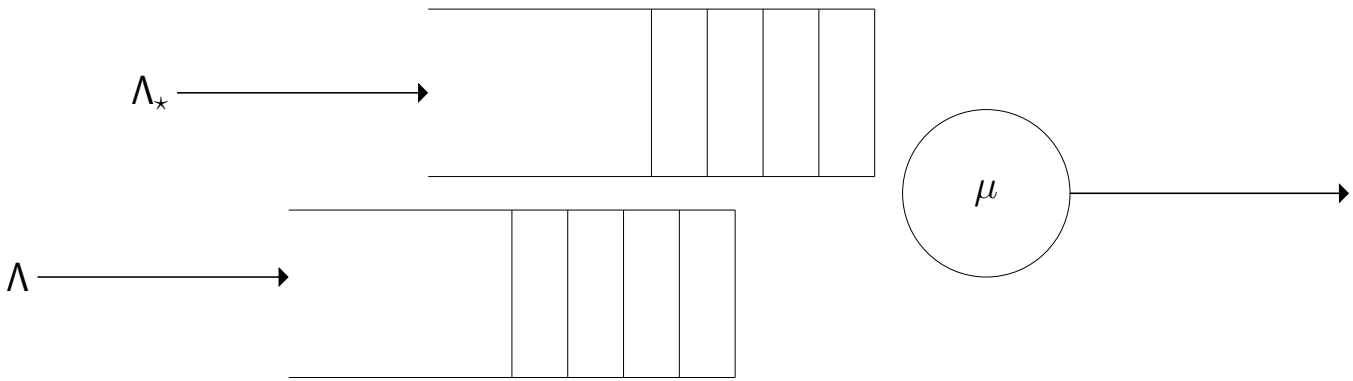




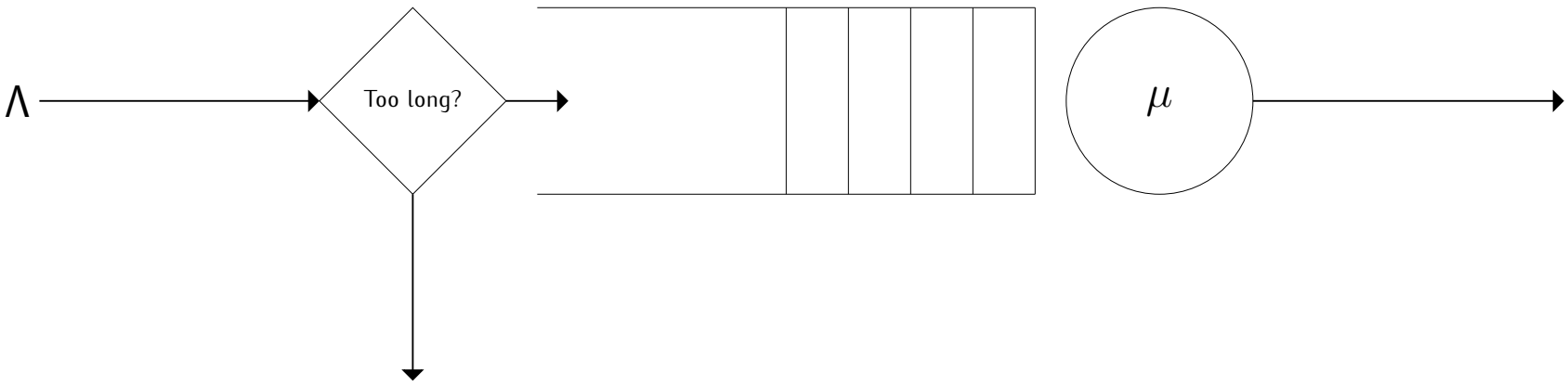




Priority Queues

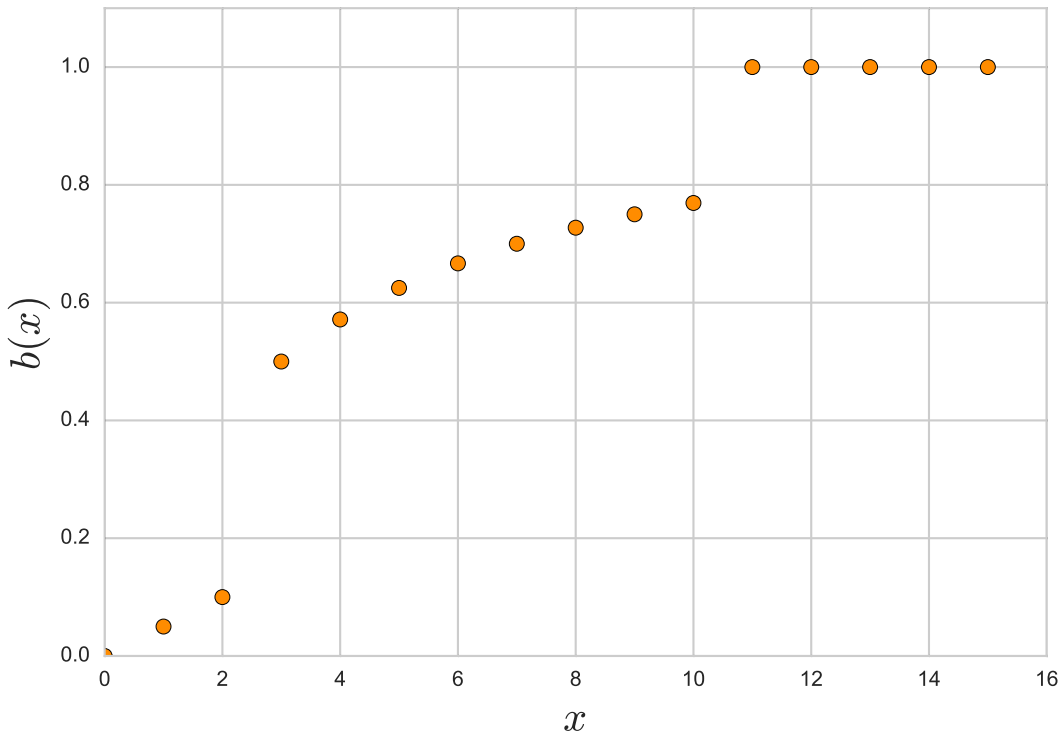


Baulking

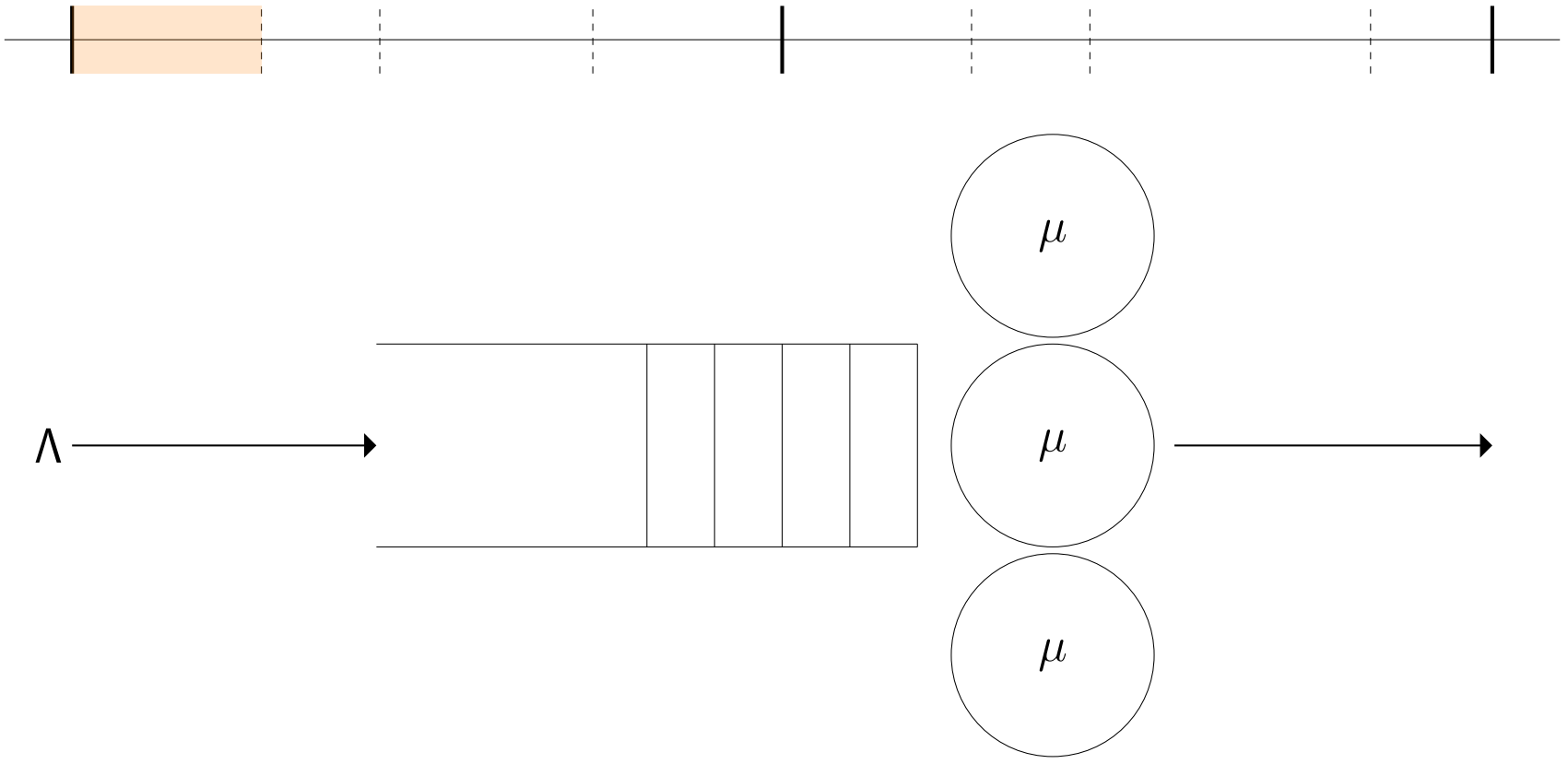


$$P(\text{baulk} \mid x \text{ in queue}) = b(x)$$

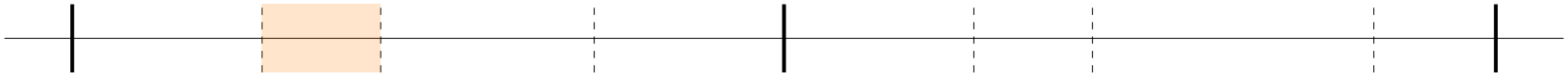
$$b(x) = \begin{cases} \frac{x}{20} & \text{if } x \leq 2 \\ \frac{x}{x+3} & \text{if } 2 < x \leq 10 \\ 1 & \text{otherwise} \end{cases}$$

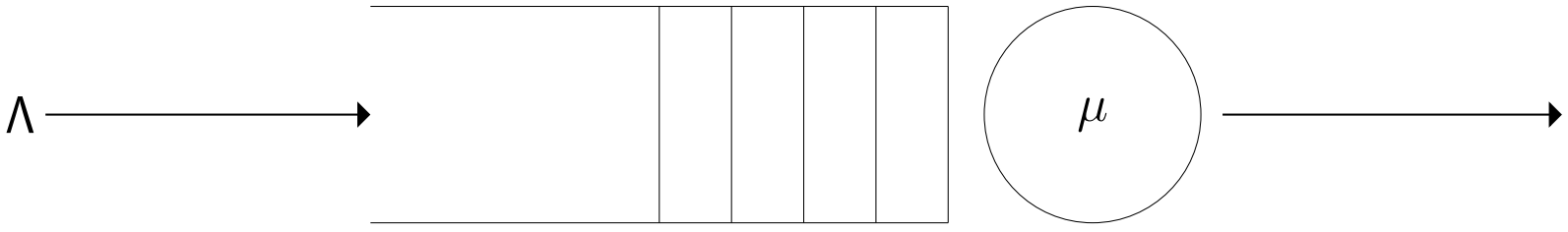


Server Schedules

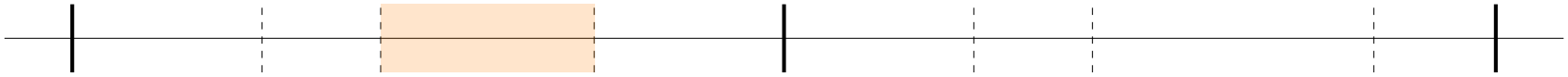


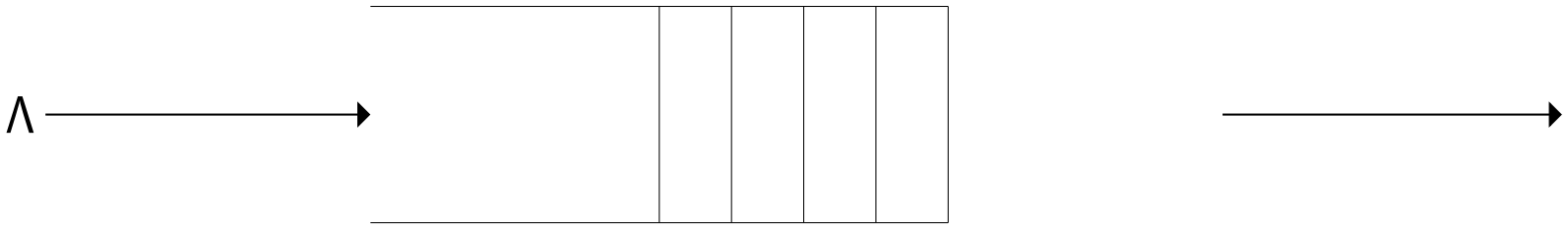
Server Schedules



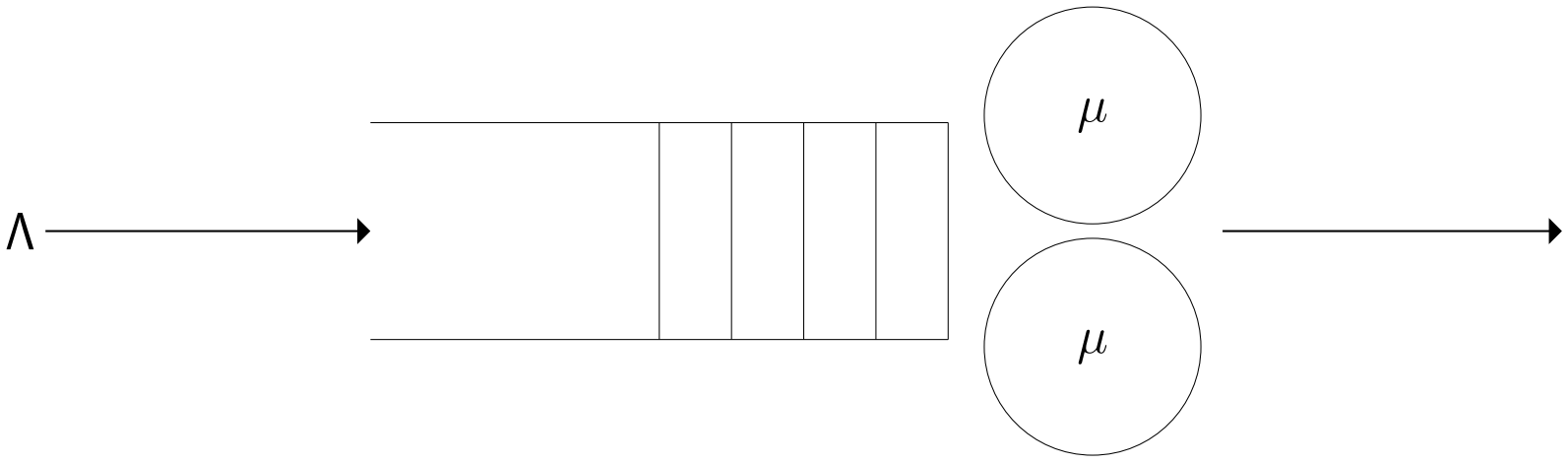


Server Schedules

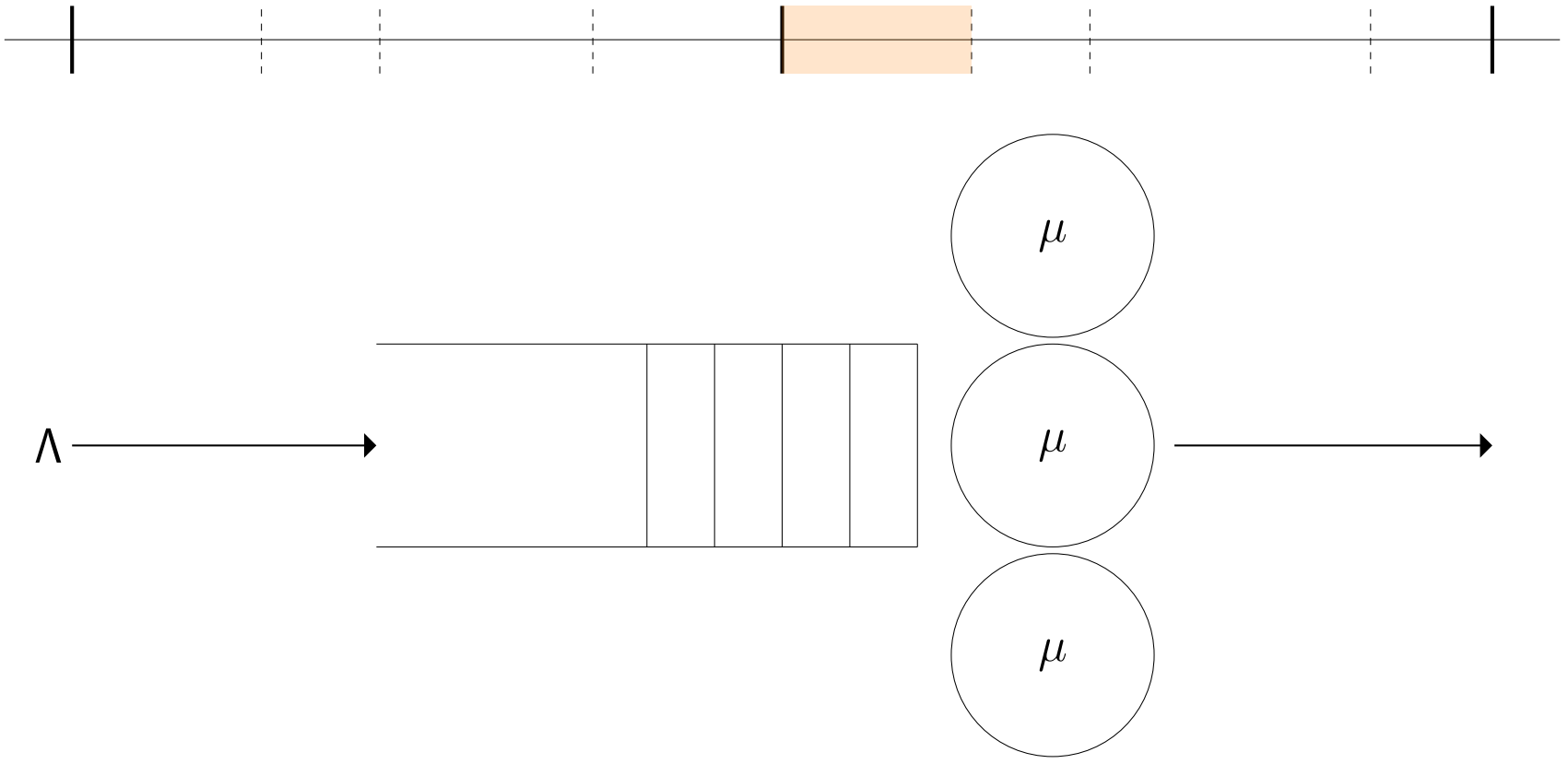




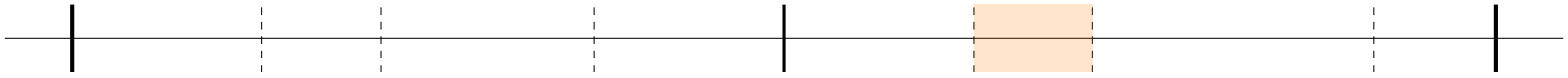
Server Schedules

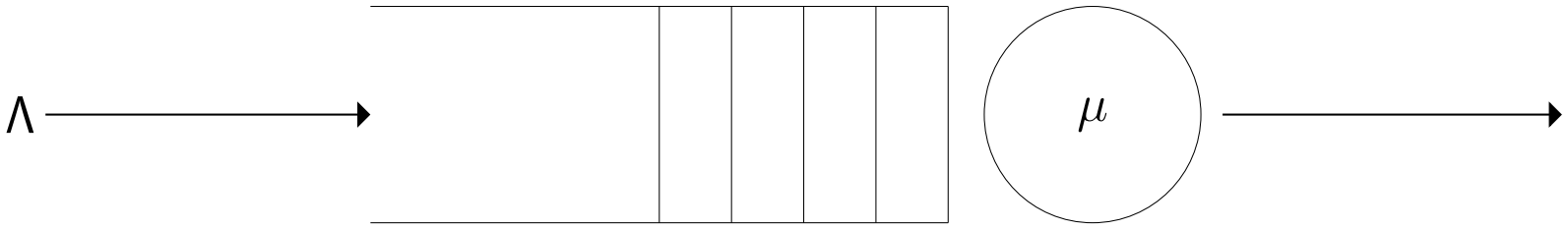


Server Schedules

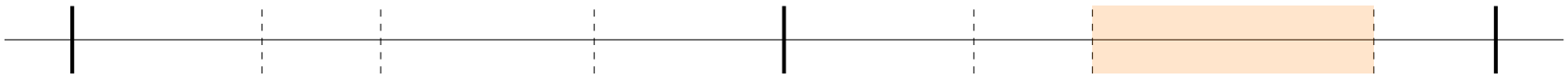


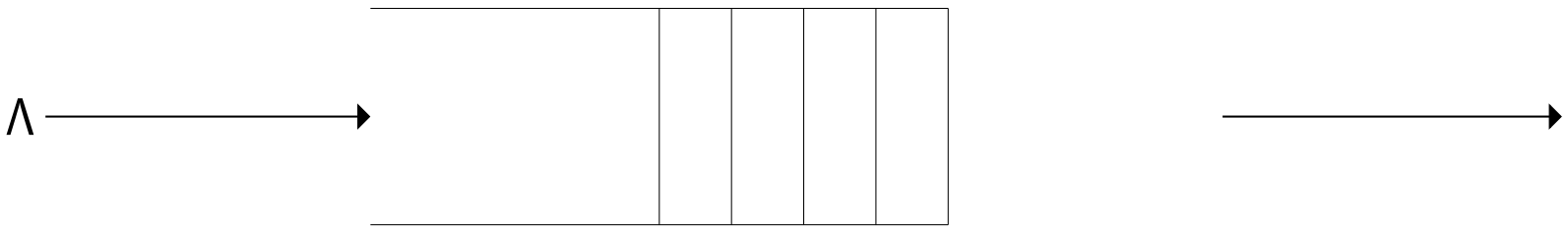
Server Schedules



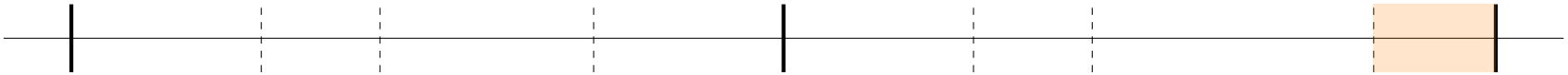


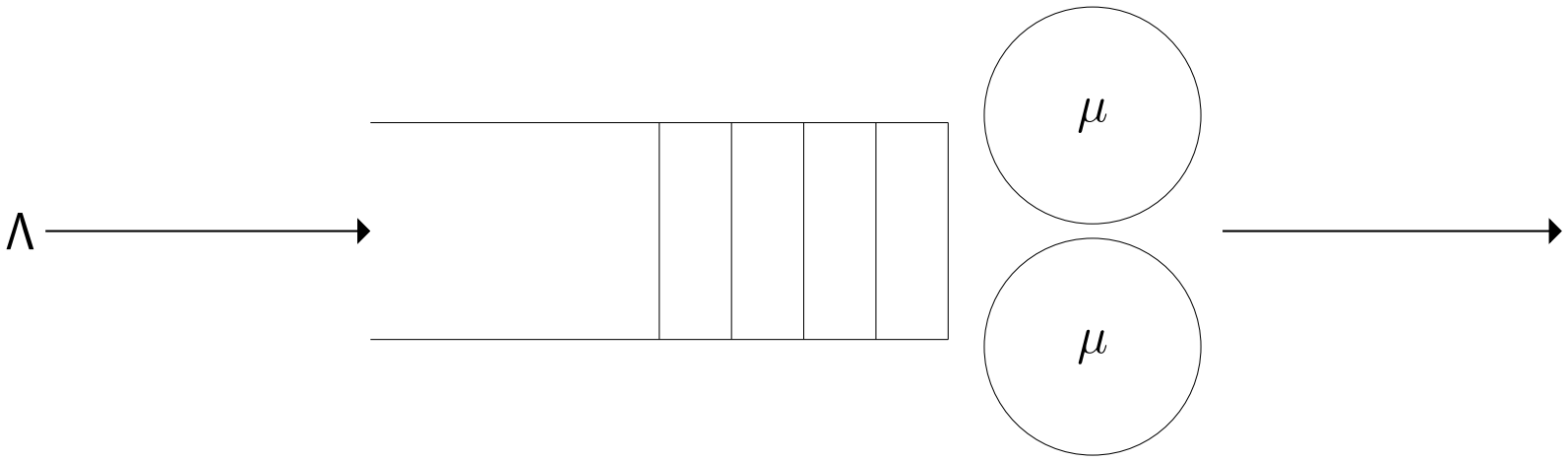
Server Schedules

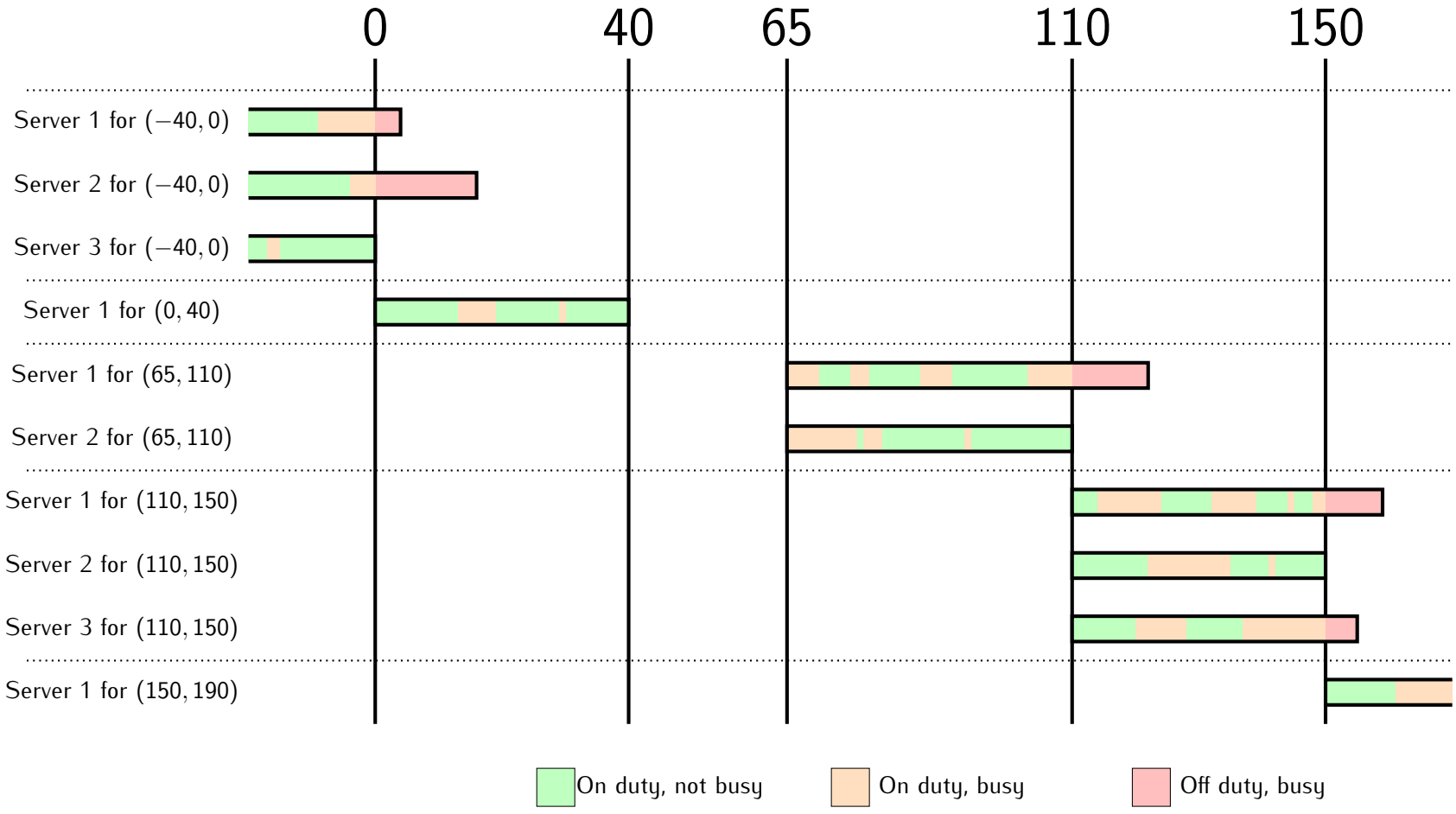




Server Schedules






0
40
65
110
150
Server 1 for $(-40, 0)$
Server 2 for $(-40, 0)$
Server 3 for $(-40, 0)$
Server 1 for $(0, 40)$
Server 1 for $(65, 110)$
Server 2 for $(65, 110)$
Server 1 for $(110, 150)$
Server 2 for $(110, 150)$
Server 3 for $(110, 150)$
Server 1 for $(150, 190)$
On duty, not busy
On duty, busy
Off duty, busy

Academic Uses

Theoretical Work

Investigating deadlock in queueing networks.
(Geraint Palmer, Prof. Paul Harper, Dr. Vincent Knight)

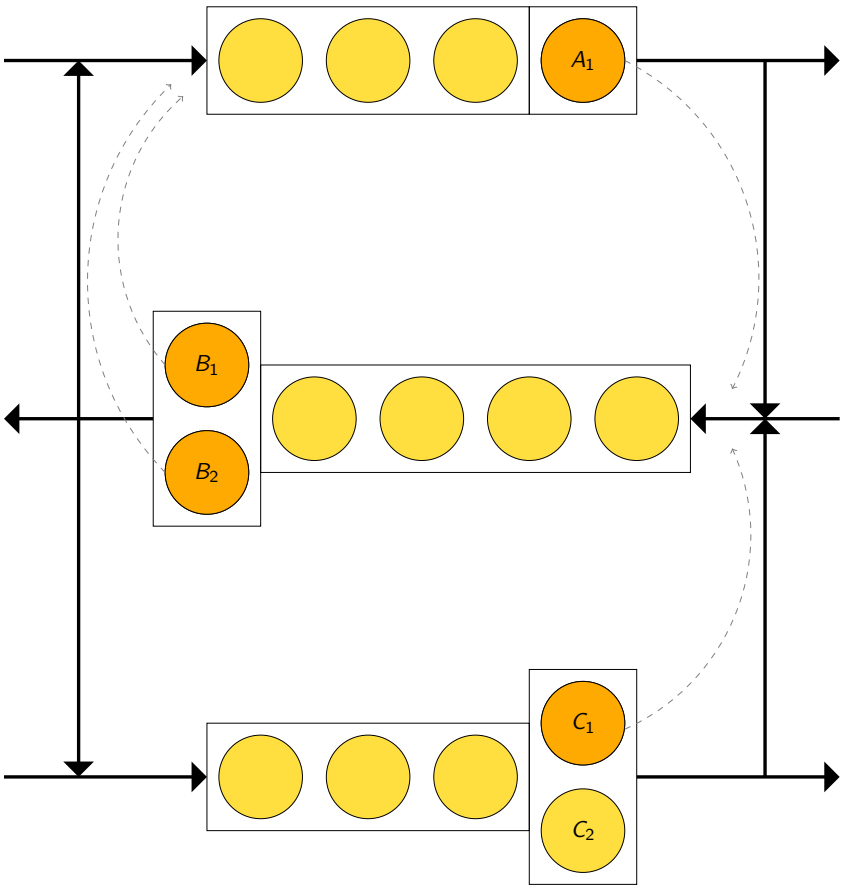
Practical Work

Modelling an ophthalmology clinic to strategise scheduling.
(Lieke Hölscher, Dr. Jennifer Morgan)

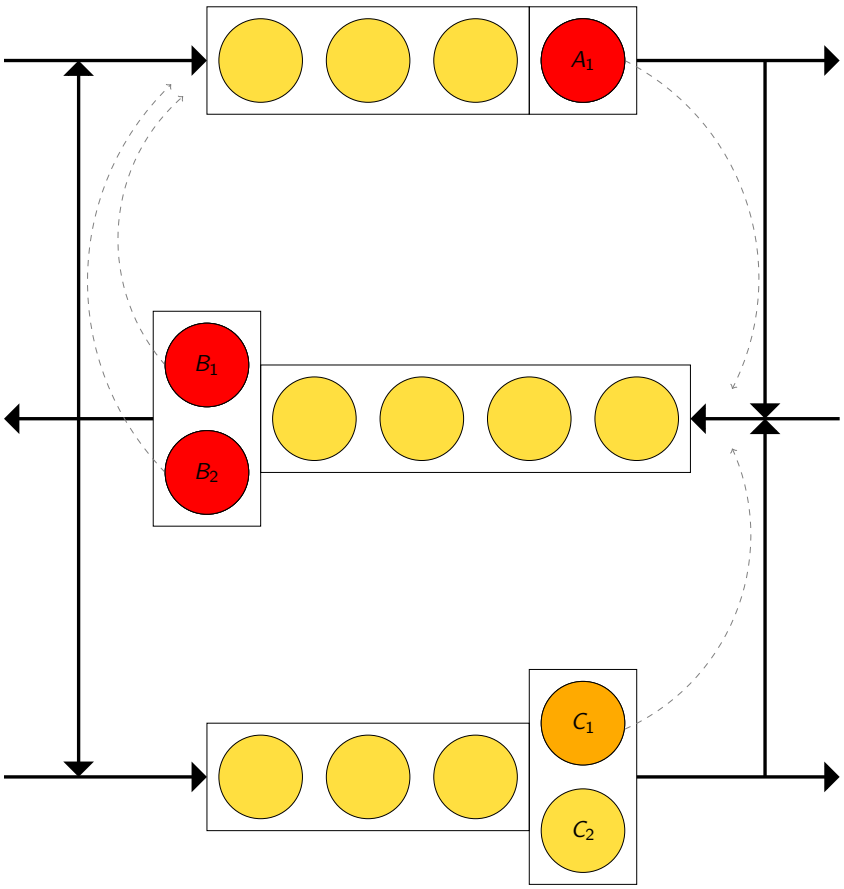
Practical Work

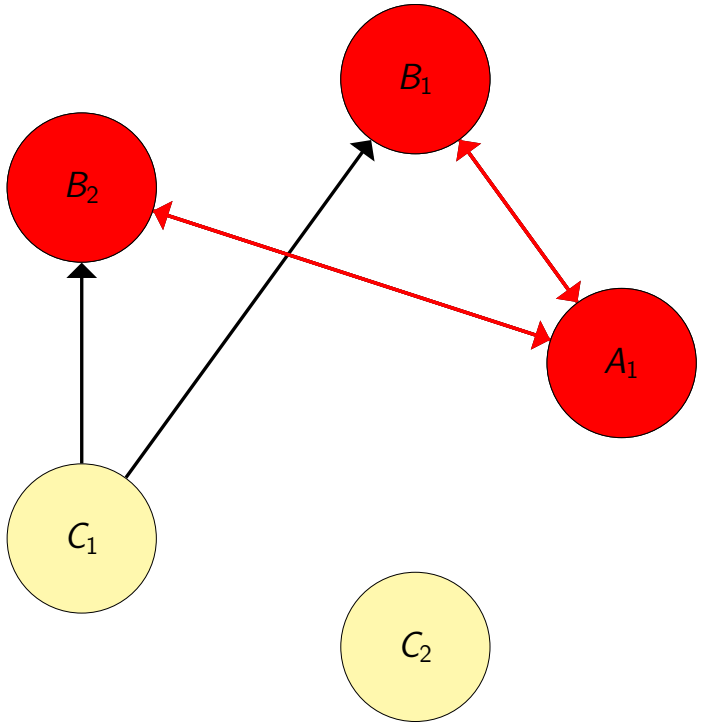
Modelling Stay Well Plans in Gwent.
(Geraint Palmer, Cathy Brooks, Prof. Paul Harper, Dr. Vincent Knight)

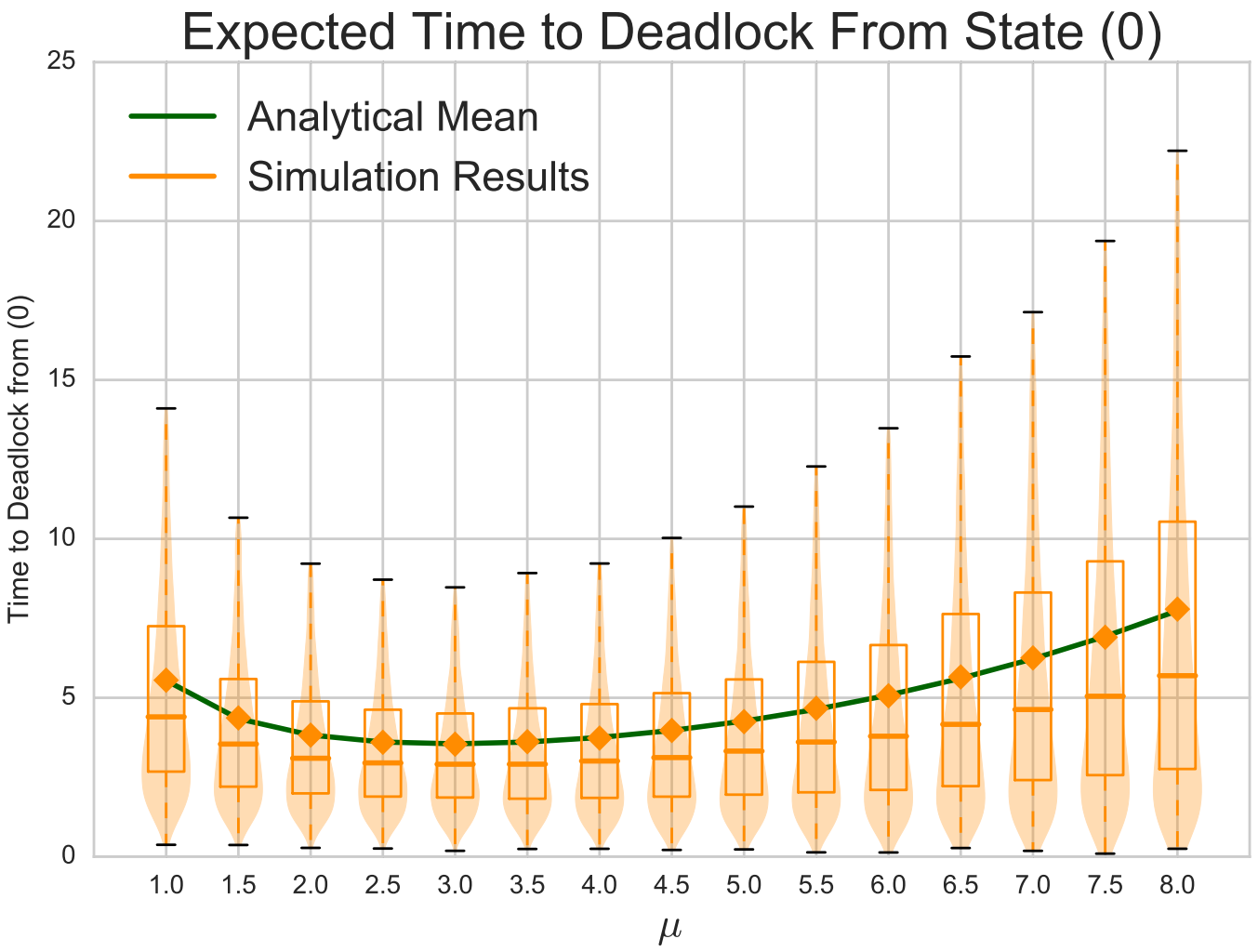
Investigating Deadlock



Investigating Deadlock






Expected Time to Deadlock From State (0)
Analytical Mean
Simulation Results
Time to Deadlock from (0)
0
5
10
15
20
25
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
μ

European Journal of Operational Research 266 (2018) 609–621

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Stochastics and Statistics

Modelling deadlock in open restricted queueing networks

Geraint I. Palmer, Paul R. Harper*, Vincent A. Knight

School of Mathematics, Cardiff University, Senghennydd Road, Cardiff CF24 4AG, United Kingdom

ARTICLE INFO

Article history:
Received 15 June 2016
Accepted 16 October 2017
Available online 7 November 2017

Keywords:
Queueing
Queueing networks
Deadlock
Markov models

ABSTRACT

Open restricted queueing networks give rise to the phenomenon of deadlock, whereby some customers may be unable to ever leave a server due to mutual blocking. This paper explores deadlock in queueing networks with limited queueing capacity, presents a method of detecting deadlock in discrete event simulations, and builds Markov chain models of these deadlocking networks. The three networks for which Markov models are given include single and multi-server networks for one and two node systems. The expected times to deadlock of these models are compared to results obtained using a simulation of the stochastic process, together with the developed deadlock detection method. This paper aims to be of value to simulation modellers of queues.

© 2017 The Author(s). Published by Elsevier B.V.
This is an open access article under the CC BY license. (http://creativecommons.org/licenses/by/4.0/)

1. Introduction

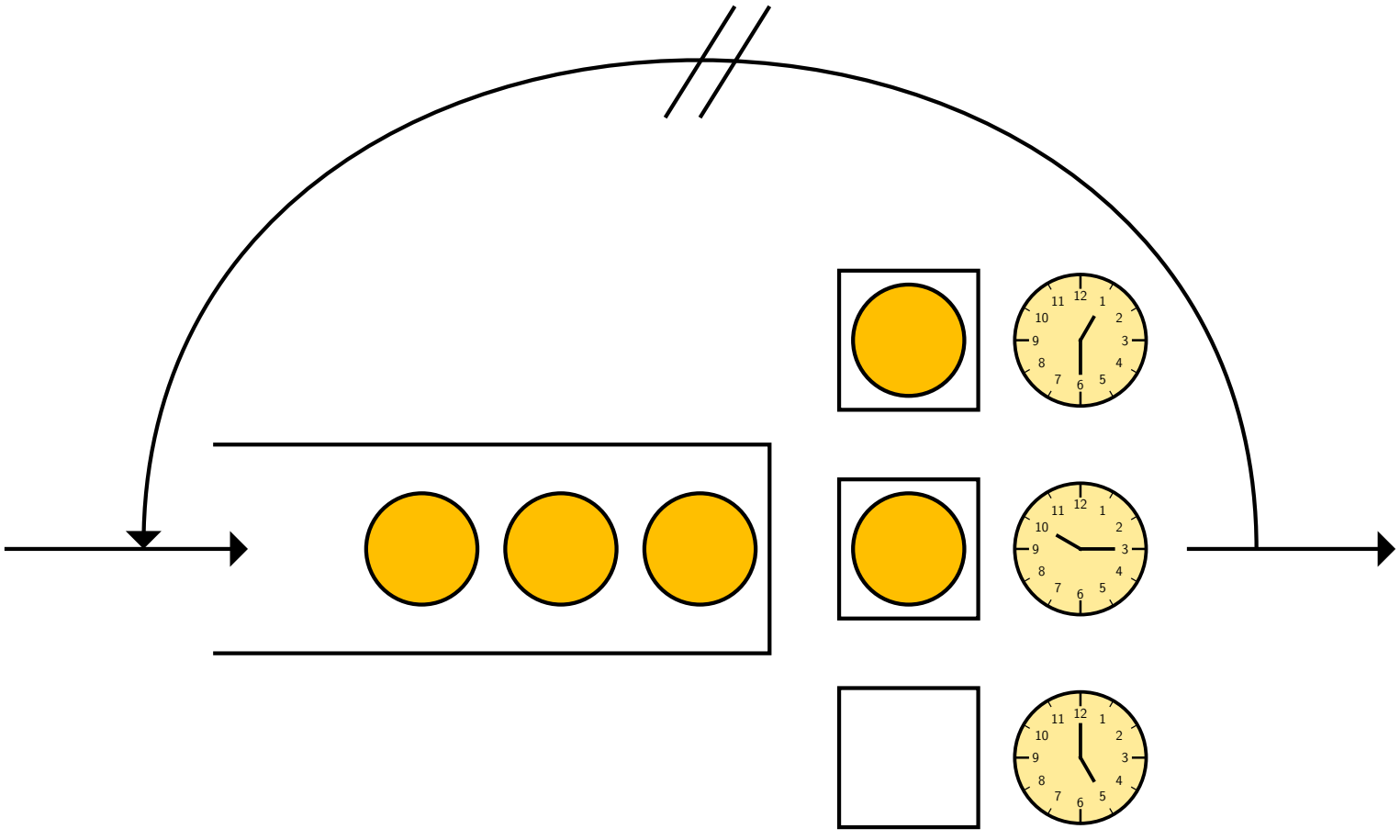
The study and modelling of queueing networks with blocking is an important tool in many aspects of operational research, both analytically and through simulation. These models have applications in many varied settings such as healthcare, supply chains, manufacturing and communications systems. However, these types of models have their limitations, due to their potential to become permanently blocked in deadlock, or a deadly embrace of resources. These deadlocks can be real and observed in reality, in which case accurate modelling of deadlock is needed; or they can be a symptom of a model unable to capture certain behaviours. This may occur in models where deadlock situations are easily adjusted in reality. In this case, such as by swapping two customers,

Definition 1. When there is a subset of blocked customers who are blocked directly or indirectly by customers in that subset only, then the system is said to be in deadlock.

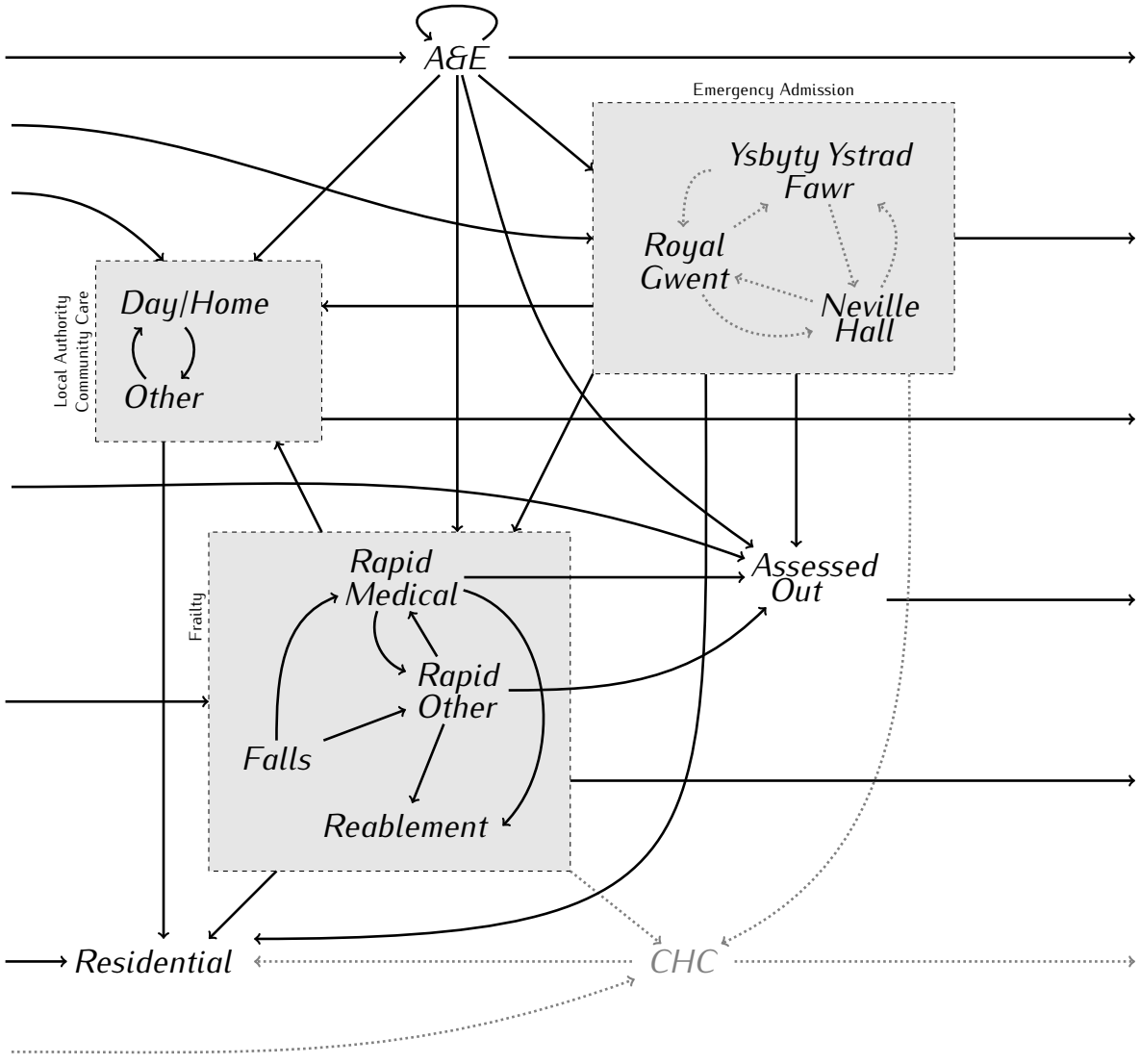
This implies that a system is in deadlock when at least one service station permanently ceases to begin or finish any more services, due to circular blocking. Fig. 1 shows an open two node restricted queueing network in deadlock. The customer at the top server is blocked from entering the bottom node as there is a full queue, and similarly the customer at the bottom server is blocked from entering the top node as there is a full queue. It is clear that by following the rules of blocking defined above, no more natural movement can happen. This system is in deadlock as there is a subset of blocked customers (the customer with server A_1 and

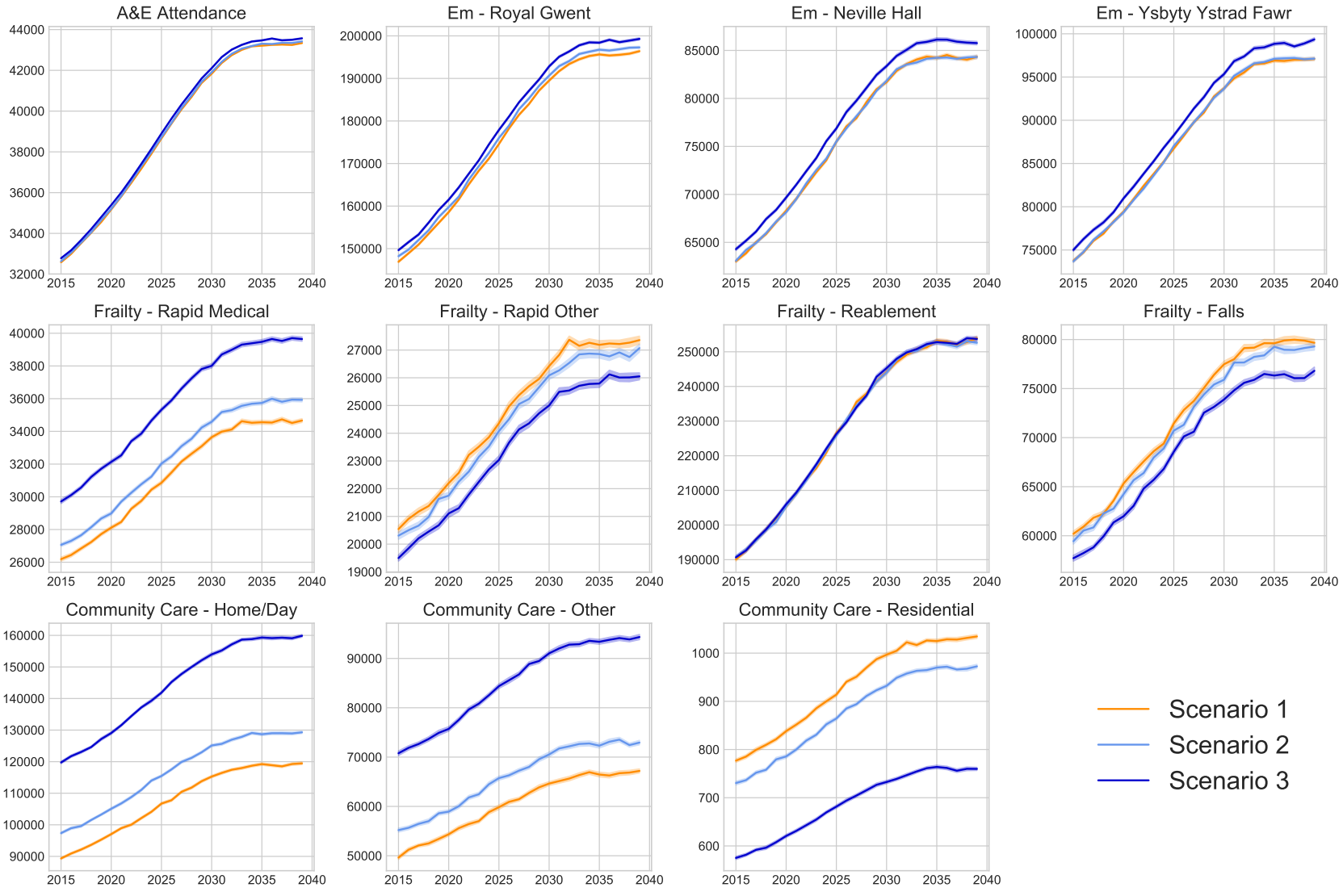
2018: **Modelling Deadlock in Open Restricted Queueing Networks**, *Palmer GI, Harper PR, Knight VA.* European Journal of Operational Research.

Modelling Ophthalmology Clinic



Modelling Stay Well Plans in Gwent




A&E Attendance
Em - Royal Gwent
Em - Neville Hall
Em - Ysbyty Ystrad Fawr
Frailty - Rapid Medical
Frailty - Rapid Other
Frailty - Reablement
Frailty - Falls
Community Care - Home/Day
Community Care - Other
Community Care - Residential
Scenario 1
Scenario 2
Scenario 3

Paper on Ciw Itself

2018: **Ciw: An open source discrete event simulation library**, *Palmer GI, Knight VA, Harper PR, Hawa, AL.* (Under review at the Journal of Simulation.)

`https://arxiv.org/abs/1710.03561`


python
jupyter
GIG CYMRU NHS WALES
Bwrdd Iechyd Prifysgol
Aneurin Bevan
University Health Board
CARDIFF UNIVERSITY
PRIFYSGOL CAERDYDD
Hypothesis
Test faster, fix more
PyYAML
NATCOR
A National Taught Course Centre in Operational Research
Coverage.py
jQuery
write less, do more.
Coleg Cymraeg Cenedlaethol
matplotlib
NumPy
NetworkX
THE OR SOCIETY
PyDiff
Read the Docs
SPHINX
Python Documentation Generator
Software Sustainability Institute
jStat
Bootstrap

Thank you!

@GeraintPalmer

@CiwPython

http://ciw.readthedocs.io

https://github.com/CiwPython/Ciw