

# Package ‘hpcwld’

June 26, 2022

**Version** 0.6-5

**Date** 2022-06-25

**Title** High Performance Cluster Models Based on Kiefer-Wolfowitz Recursion

**Author** Alexander Rumyantsev [aut, cre]

**Maintainer** Alexander Rumyantsev <ar@sampo.ru>

**Description** Probabilistic models describing the behavior of workload and queue on a High Performance Cluster and computing GRID under FIFO service discipline basing on modified Kiefer-Wolfowitz recursion. Also sample data for inter-arrival times, service times, number of cores per task and waiting times of HPC of Karelian Research Centre are included, measurements took place from 06/03/2009 to 02/30/2011. Functions provided to import/export workload traces in Standard Workload Format (swf). Stability condition of the model may be verified either exactly, or approximately. Stability analysis: see Rumyantsev and Morozov (2017) <[doi:10.1007/s10479-015-1917-2](https://doi.org/10.1007/s10479-015-1917-2)>, workload recursion: see Rumyantsev (2014) <[doi:10.1109/PDCAT.2014.36](https://doi.org/10.1109/PDCAT.2014.36)>.

**License** GPL (>= 2)

**RoxygenNote** 7.1.2

**Repository** CRAN

**Repository/R-Forge/Project** hpcwld

**Repository/R-Forge/Revision** 27

**Repository/R-Forge/DateTimeStamp** 2022-06-25 20:09:01

**Date/Publication** 2022-06-26 21:20:07 UTC

**NeedsCompilation** no

**Depends** R (>= 2.10)

## R topics documented:

hpcwld-package . . . . .	2
ApproxC . . . . .	3
DataToSWF . . . . .	4

DMC	5
FromSWF	5
HPC_KRC	6
HPC_KRC2	7
MaxThroughput2	8
ToSWF	8
Wld	9
X	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

hpcwld-package	<i>Model and data for High Performance Cluster workload</i>
----------------	---

---

## Description

This package contains several models describing the behavior of workload and queue on a High Performance Cluster and computing GRID under FIFO service discipline basing on modified Kiefer-Wolfowitz recursion. Also sample data for inter-arrival times, service times, number of cores per task and waiting times of HPC of Karelian Research Centre are included, measurements took place from 06/03/2009 to 02/30/2011. The stability condition of the model can be verified either exactly, or approximately.

## Details

Package:	hpcwld
Type:	Package
Version:	0.5
Date:	2015-02-14
License:	GNU GPL
LazyLoad:	yes

## Author(s)

Alexander Rumyantsev (Institute of Applied Mathematical Research, Karelian Research Centre, RAS)

## References

E.V. Morozov, A.Rumyantsev. Stability analysis of a multiprocessor model describing a high performance cluster. XXIX International Seminar on Stability Problems for Stochastic Models and V International Workshop "Applied Problems in Theory of Probabilities and Mathematical Statistics related to modeling of information systems". Book of Abstracts. 2011. Pp. 82–83.

A. Rumyantsev. Simulating Supercomputer Workload with hpcwld package for R // Proceedings of 2014 15th International Conference on Parallel and Distributed Computing, Applications and Technologies. IEEE, 2014. P. 138-143. URL: <http://conferences.computer.org/pdcat/2014/papers/8334a138.pdf>

A. Rumyantsev. Evaluating the stability of supercomputer workload model // Journal on Selected Topics in Nano Electronics and Computing, Vol. 2, No. 2, December 2014. P. 36-39.

<http://cluster.krc.karelia.ru>

### Examples

```
Wld(T=rexp(1000,1), S=rexp(1000,1), round(runif(1000,1,10)), 10)
# returns the workload, delay and total cpus used
# for a cluster with 10 CPUs and random exponential times
```

---

ApproxC

*Approximate, dynamic iterative computation of the stability constant for a workload of a High Performance Cluster model*

---

### Description

This function calculates the constant  $C$  that is used in the stability criterion of a supercomputer model, which is basically the following:  $\lambda/\mu < C$ , where  $\lambda$  is the task arrivals rate, and  $\mu$  is the service intensity. The constant depends only on the number of servers in the model and the distribution of classes of customers, where class is the number of servers required. This method of calculation allows to stop on some depth of dynamics, thus allowing to calculate an approximate value in faster time. The constant is valid only for the model with simultaneous service.

### Usage

```
ApproxC(s, p, depth = 3)
```

### Arguments

s	number of servers in the model
p	vector of class distribution
depth	By default calculates up to groups of 3 tasks. When depth=s, calculates the exact value. However, depth=s might take a bit more time.

### Value

The value of a constant  $C$  in the relation  $\lambda/\mu < C$  is returned

### Examples

```
ApproxC(s=2,p=c(.5,.5), depth=3)
```

---

**DataToSWF***Convertor from a dataframe to Standart Workload Format*

---

### Description

Note that this is only a wrapper for the ToSWF command with a dataframe argument. It needs a correctly built dataframe and converts it to a Standart Workload Format used to share the logfiles of High Performance Clusters

### Usage

```
DataToSWF(Frame, filename = "output.swf")
```

### Arguments

Frame	A dataframe containing the variables needed by ToSWF function
filename	The file to store the converted workload (output.swf by default)

### Details

The Standart Workload Format is a single format to store and exchange high performance cluster logs, that is used in Parallel Workload Archive. See references for current standard. The SWF format may contain additional data, but in this package only the 1st to 5th fields are used. One may also need to manually fill in the header of the file in order to completely prepare the resulting SWF file.

### Value

Nothing is returned, but a file is created in the current working directory (with default name output.swf) containing the converted data.

### References

Feitelson, D.G. and Tsafir, D. and Krakov D. 2012 Experience with the Parallel Workloads Archive. Technical Report 2012-6, School of Computer Science and Engineering, the Hebrew University April, 2012, Jerusalem, Israel

<https://www.cs.huji.ac.il/labs/parallel/workload/swf.html>

### Examples

```
## Not run:  
data(HPC_KRC)  
tmp=data.frame(T=HPC_KRC$interarrival, S=HPC_KRC$service, N=HPC_KRC$cores_used, D=HPC_KRC$delay)  
DataToSWF(tmp)  
  
## End(Not run)
```

---

DMC

*Distributional Measure of Correlation*

---

### Description

This is a suggested by Dror Feitelson measure of correlation for dependent variables, that may be successfully used to examine the datasets from a High Performance Cluster logs

### Usage

DMC(X, Y)

### Arguments

X                    First variable (vector)  
Y                    Second variable (vector)

### Value

One value between -1 and 1, characterizing the dependence between the variables

### References

<http://interstat.statjournals.net/YEAR/2004/abstracts/0412001.php?Name=412001>

### Examples

```
data(HPC_KRC)
DMC(HPC_KRC$service[1:1000], HPC_KRC$cores_requested[1:1000])
```

---

FromSWF

*Convertor to a dataset from a Standart Workload Format*

---

### Description

This is a convertor from a Standart Workload Format (used to share the logfiles of High Performance Clusters) to an internally used in a package dataset format

### Usage

FromSWF(filename)

### Arguments

filename            A mandatory field containing the path to SWF file

**Details**

The Standart Workload Format is a single format to store and exchange high performance cluster logs, that is used in Parallel Workload Archive. See references for current standard. The SWF format may contain additional data, but in this package only the 1st to 5th fields are used. One may also need to manually fill in the header of the file in order to completely prepare the resulting SWF file.

**Value**

A dataset is returned, containing 'delay' as a vector of delays exhibited by each task, 'total\_cores' as the total busy CPUs in time of arrival of each task, and 'workload' as total work left at each CPU.

**References**

Feitelson, D.G. and Tsafir, D. and Krakov D. 2012 Experience with the Parallel Workloads Archive. Technical Report 2012-6, School of Computer Science and Engineering, the Hebrew University April, 2012, Jerusalem, Israel

<https://www.cs.huji.ac.il/labs/parallel/workload/swf.html>

---

HPC\_KRC

*Workload data for High Performance Cluster of High Performance Data Center of Karelian Research Center, Russian Academy of Sciences.*

---

**Description**

This is a complete data of the tasks which successfully finished executions at HPC of HPDC KRC RAS for time period 06/03/2009 to 02/04/2011, a total of 8282 tasks. The data contains interarrival times, service times, cores that tasks requested, cores really used (due to administrative limitations) and delays excursed by tasks, all in seconds.

**Usage**

```
data(HPC_KRC)
```

**Format**

A data frame with 8281 observations on the following 5 variables.

interarrival a numeric vector

service a numeric vector

cores\_requested a numeric vector

cores\_used a numeric vector

delays a numeric vector

**Source**

<http://cluster.krc.karelia.ru>

**References**

<http://cluster.krc.karelia.ru>

**Examples**

```
data(HPC_KRC)
```

---

HPC_KRC2	<i>Workload data for High Performance Cluster of High Performance Data Center of Karelian Research Center, Russian Academy of Sciences.</i>
----------	---

---

**Description**

This is a complete data of the tasks which successfully finished executions at HPC of HPDC KRC RAS for time period 02/04/2011 to 16/04/2012, a total of 9389 tasks. The data contains interarrival times, service times, cores that tasks used, and delays excused by tasks, all in seconds.

**Usage**

```
data(HPC_KRC2)
```

**Format**

A data frame with 9389 observations on the following 3 variables.

`interarrival` a numeric vector

`service` a numeric vector

`cores_used` a numeric vector

`delays` a numeric vector

**Source**

<http://cluster.krc.karelia.ru>

**References**

<http://cluster.krc.karelia.ru>

**Examples**

```
data(HPC_KRC2)
```

---

MaxThroughput2	<i>This function gives the maximal throughput of a two-server supercomputer (Markov) model with various service speeds, various rates of classes and random speed scaling at arrival/departure</i>
----------------	--

---

### Description

This function gives the maximal throughput of a two-server supercomputer (Markov) model with various service speeds, various rates of classes and random speed scaling at arrival/departure

### Usage

```
MaxThroughput2(p1, pa, pd, mu1, mu2, f1, f2)
```

### Arguments

p1	probability of class 1 arrival
pa	probability of speed switch from f1 to f2 upon arrival
pd	probability of speed switch from f2 to f1 upon departure
mu1	work amount parameter (for exponential distribution) for class 1
mu2	work amount parameter (for exponential distribution) for class 2
f1	low speed (workunits per unit time)
f2	high speed (workunits per unit time)

### Value

maximal input rate, that is the stability boundary

---

ToSWF	<i>Convertor from a dataset to Standart Workload Format</i>
-------	---

---

### Description

This is a convertor from a correctly built dataset to a Standart Workload Format used to share the logfiles of High Performance Clusters

### Usage

```
ToSWF(T, S, N, D, filename = "output.swf")
```



**Arguments**

T	Interarrival times of tasks (a vector)
S	Service times of tasks (a vector)
N	Number of cores each task needs (a vector)
D	The delays of tasks in a queue (a vector)
filename	The file to store the converted workload (output.swf by default)

**Details**

The Standart Workload Format is a single format to store and exchange high performance cluster logs, that is used in Parallel Workload Archive. See references for current standard. The SWF format may contain additional data, but in this package only the 1st to 5th fields are used. One may also need to manually fill in the header of the file in order to completely prepare the resulting SWF file.

**Value**

Nothing is returned, but a file is created in the current working directory (with default name output.swf) containing the converted data.

**References**

Feitelson, D.G. and Tsafirir, D. and Krakov D. 2012 Experience with the Parallel Workloads Archive. Technical Report 2012-6, School of Computer Science and Engineering, the Hebrew University April, 2012, Jerusalem, Israel

<https://www.cs.huji.ac.il/labs/parallel/workload/swf.html>

**Examples**

```
## Not run:
data(HPC_KRC)
ToSWF(HPC_KRC$interarrival, HPC_KRC$service, HPC_KRC$cores_requested, HPC_KRC$delay)

## End(Not run)
```

**Description**

This function computes the Kiefer-Wolfowitz modified vector for a HPC model. This vector contains the work left on each of 'm' servers of a cluster for the time of the arrival of a task. Two methods are available, one for the case of concurrent server release (all the servers end a single task simultaneously), other for independent release (service times on each server are independent).

**Usage**

```
Wld(T, S, N, m, method = "concurrent")
```

**Arguments**

T	Interarrival times of tasks
S	Service times of customers (a vector of length n, or a matrix nrows=n, ncols='m').
N	Number of servers each customer needs
m	Number of servers for a supercomputer
method	Independent or concurrent

**Value**

A dataset is returned, containing 'delay' as a vector of delays exhibited by each task, 'total\_cores' as the total busy CPUs in time of arrival of each task, and 'workload' as total work left at each CPU.

**Examples**

```
Wld(T=rexp(1000,1), S=rexp(1000,1), round(runif(1000,1,10)), 10)
```

---

X

*Dataset with raw workload data from HPDC KRC RAS*

---

**Description**

Source data for workload of HPC of HPDC KRC RAS. More usable dataset is HPC\_KRC. This are raw times in sec. since 1 January 1970, for tasks arrival times, start of execution times and end times.

**Usage**

```
data(X)
```

**Format**

The format is: num [1:8499, 1:3] 1.24e+09 1.24e+09 1.24e+09 1.24e+09 1.24e+09 ...

**Source**

<http://cluster.krc.karelia.ru>

**References**

<http://cluster.krc.karelia.ru>

**Examples**

```
data(X)
```

# Index

- \* **datasets**

- HPC\_KRC, [6](#)

- HPC\_KRC2, [7](#)

- X, [10](#)

- \* **package**

- hpcwld-package, [2](#)

ApproxC, [3](#)

DataToSWF, [4](#)

DMC, [5](#)

FromSWF, [5](#)

HPC\_KRC, [6](#)

HPC\_KRC2, [7](#)

hpcwld (hpcwld-package), [2](#)

hpcwld-package, [2](#)

MaxThroughput2, [8](#)

ToSWF, [8](#)

Wld, [9](#)

X, [10](#)