

NAME

Fmd1() – Virtual waiting time distribution function

IntFmd1() – Virtual waiting time distribution function for integral values of x

SumMd1() – Calculates the state probabilities of a M/D/1 queue

RecMd1() – Calculates the state probabilities of a M/D/1 queue using a recursive algorithm

SYNOPSIS

```
#include <queue.h>
```

```
double Fmd1(double x, double rho);
```

```
double IntFmd1(int N, double rho);
```

```
double *SumMd1(double x, double rho);
```

```
double *RecMd1(int x, double rho);
```

DESCRIPTION

delim \$\$ These functions return the state probabilities or the virtual waiting time distribution of a M/D/1 queue.

Fmd1() is a model for the M/D/1 queuing system with Poisson arrivals and deterministic (constant) service time. Parameter x is the amount of unfinished work in the system. ρ is the load level of the system.

ALGORITHM

M/D/1 waiting time distribution is calculated using the following algorithm (Iversen):

$$P\{W \leq T + \tau\} = e^{\lambda \tau} \sum_{0 \leq n \leq T} \{(-\lambda \tau)^n \sup_{n!}\}$$

$$\text{over } \{n!\} P\{W \leq T - n\},$$

where $P\{W \leq T - n\}$ is the waiting time for integral values of x :

$$P\{W \leq t\} = P\{0\} + P\{1\} + \dots + P\{t\}.$$

The state probabilities can be calculated by using a general or a recursive algorithm (Fmd1() uses the recursive algorithm, because it is more accurate):

$$P(0) = 1 - A$$

$$P(1) = (1 - A)(e^A - 1)$$

$$P(2) = (1 - A)(-e^A(1 + A) + e^{2A})$$

$$A = \lambda h \text{ (If } h=1, A=\lambda \text{)}.$$

General algorithm:

$$P(i) = (1 - A) \sum_{1 \leq n \leq i} (-1)^{n-i} e^{nA} \left(\sum_{nA}^{i-n} \right) \text{ over } \{(i-n)!\}$$

$$+ \left(\sum_{nA}^{i-n-1} \right) \text{ over } \{(i-n-1)!\}, i=2,3,\dots$$

(The last term always equals e^{iA} .)

Recursive algorithm:

$$P(i+1) = \left(1 \text{ over } \{P(0,h)\} \right) \{P(i) - [P(0) + P(1)] P(i,h) - \sum_{2 \leq n \leq i} P(n) P(i-n+1,h)\}, i=3,4,\dots$$

$$P(i,h) = \left(\sum_{i!}^{(\lambda h)} \right) e^{-\lambda h}$$

ERRORS

When ρ is close to 1, these functions might give inaccurate results.

SEE ALSO

COST 224: Performance evaluation and design of multiservice networks

