

# HOMOGENEOUS FINITE-SOURCE RETRIAL QUEUES WITH SERVER SUBJECT TO BREAKDOWNS AND REPAIRS

**B. Almási, J. Roszik, J. Sztrik**

Institute of Mathematics and Informatics, University of Debrecen  
Debrecen, Hungary

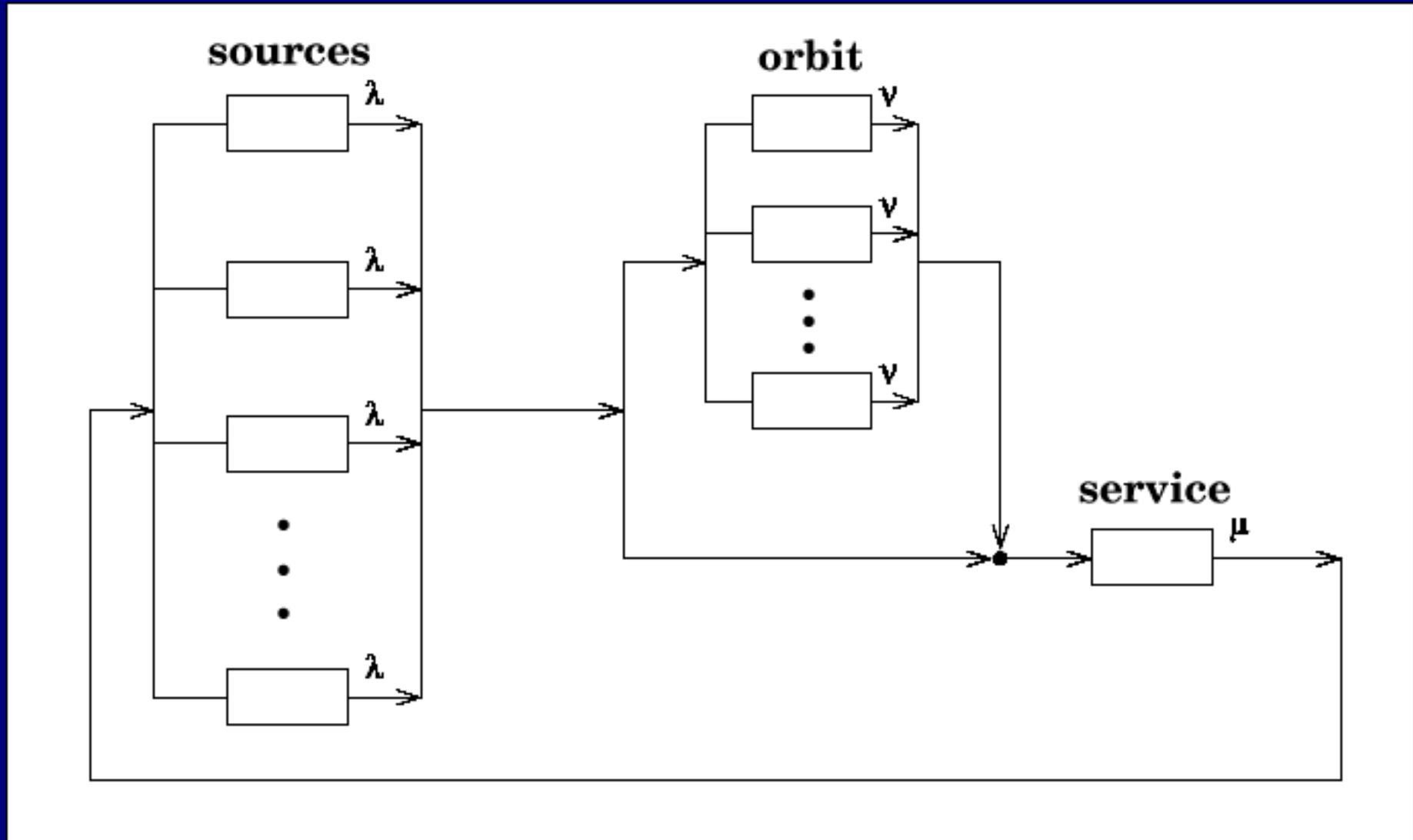
*e-mail: [jsztrik@math.klte.hu](mailto:jsztrik@math.klte.hu)*

*www: <http://it.math.klte.hu/user/jsztrik/>*

# OUTLOOK

- **The queueing model**
- **Applications**
- **Mathematical model**
- **Evaluation Tool MOSEL**
- **Case studies**
- **References**

# The queueing model



# Applications

- **magnetic disk memory systems**
- **local area networks with CSMA/CD protocols**
- **collision avoidance local area network modeling**

# Mathematical model

The system state at time  $t$  can be described with the process

$$X(t) = (Y(t); C(t); N(t))$$

where  $Y(t) = 0$  if the server is up,  $Y(t) = 1$  if the server is failed,

$C(t) = 0$  if the server is idle,  $C(t) = 1$  if the server is busy,

$N(t)$  is the number of sources of repeated calls at time  $t$ .

We define the stationary probabilities:

$$P(q; r; j) = \lim_{t \rightarrow \infty} P(Y(t) = q, C(t) = r, N(t) = j)$$

$$q = 0, 1, \quad r = 0, 1, \quad j = 0, \dots, K - 1.$$

Once we have obtained these limiting probabilities the **main system performance measures** can be derived in the following way.

## 1. Utilization of the server

$$U_S = \sum_{j=0}^{K-1} P(0, 1, j)$$

## 2. Utilization of the repairman

$$U_R = \sum_{q=0}^1 \sum_{j=0}^{K-1} P(1, q, j)$$

## 3. Availability of the server

$$A_S = \sum_{q=0}^1 \sum_{j=0}^{K-1} P(0, q, j) = 1 - U_R$$

#### 4. The mean number of sources of repeated calls

$$N = E[N(t)] = \sum_{q=0}^1 \sum_{r=0}^1 \sum_{j=0}^{K-1} jP(q, r, j)$$

#### 5. The mean number of calls staying in the orbit or in service

$$M = E[C(t) + N(t)] = \sum_{q=0}^1 \sum_{r=0}^1 \sum_{j=0}^{K-1} (r + j)P(q, r, j)$$

#### 6. The mean rate of generation of primary calls

$$\bar{\lambda} = \begin{cases} \lambda E[K - C(t) - N(t); Y(t) = 0] & \text{for blocked case,} \\ \lambda E[K - C(t) - N(t)] & \text{for unblocked case.} \end{cases}$$

## 7. The mean response time

$$E[T] = M/\bar{\lambda}$$

## 8. The mean waiting time

$$E[W] = N/\bar{\lambda}$$

## 9. The blocking probability of a primary call

$$B = \begin{cases} \frac{\lambda E[K - C(t) - N(t); Y(t)=0; C(t)=1]}{\bar{\lambda}} & \text{for blocked case,} \\ \frac{\lambda E[K - C(t) - N(t); C(t)=1]}{\bar{\lambda}} & \text{for unblocked case.} \end{cases}$$

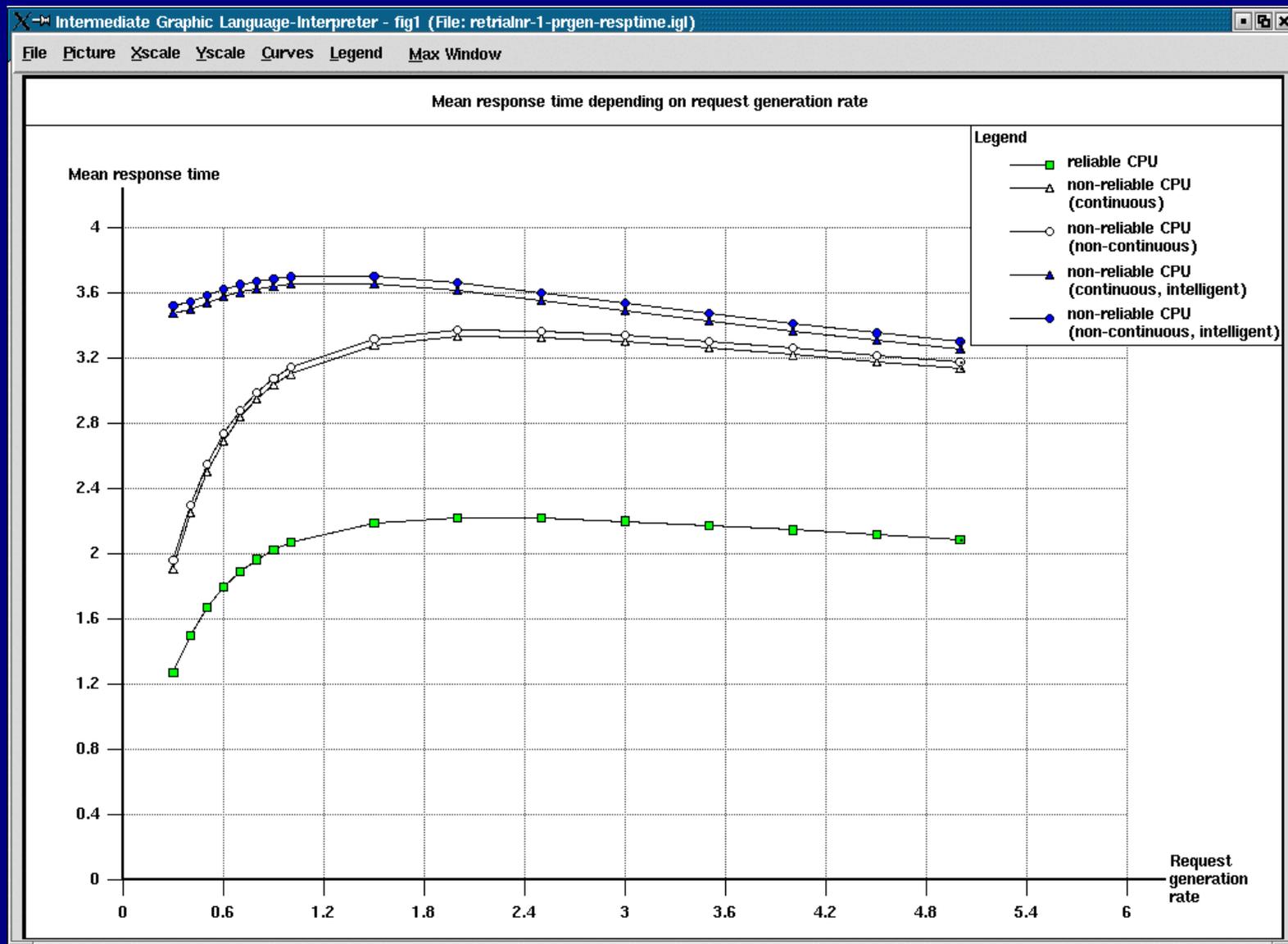
# Evaluation Tool MOSEL

**MOSEL ( Modeling, Specification and Evaluation Language )** developed at the University of Erlangen, Germany, is used to formulate and solved the problem.

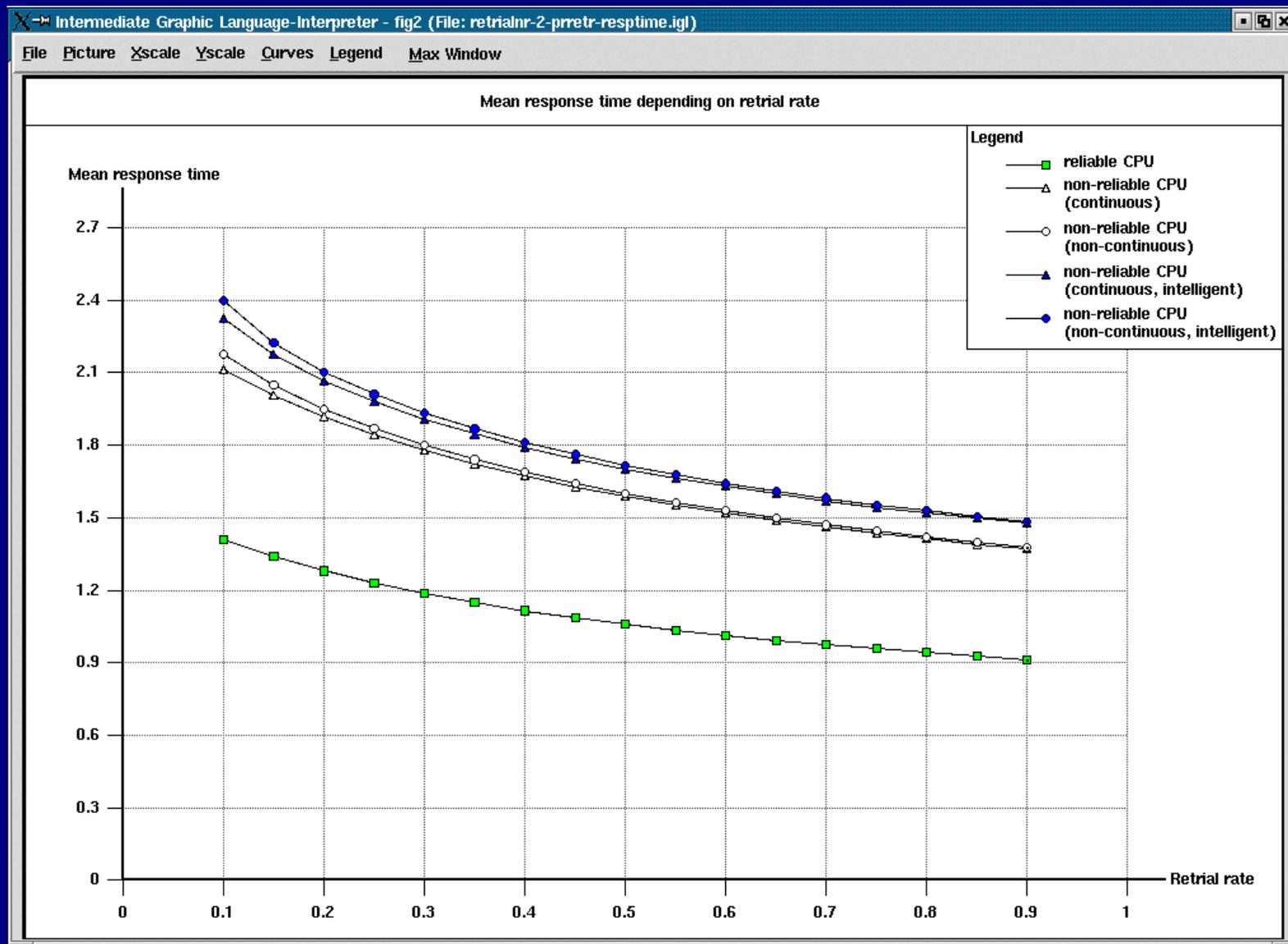
## Case studies

# Validation of results

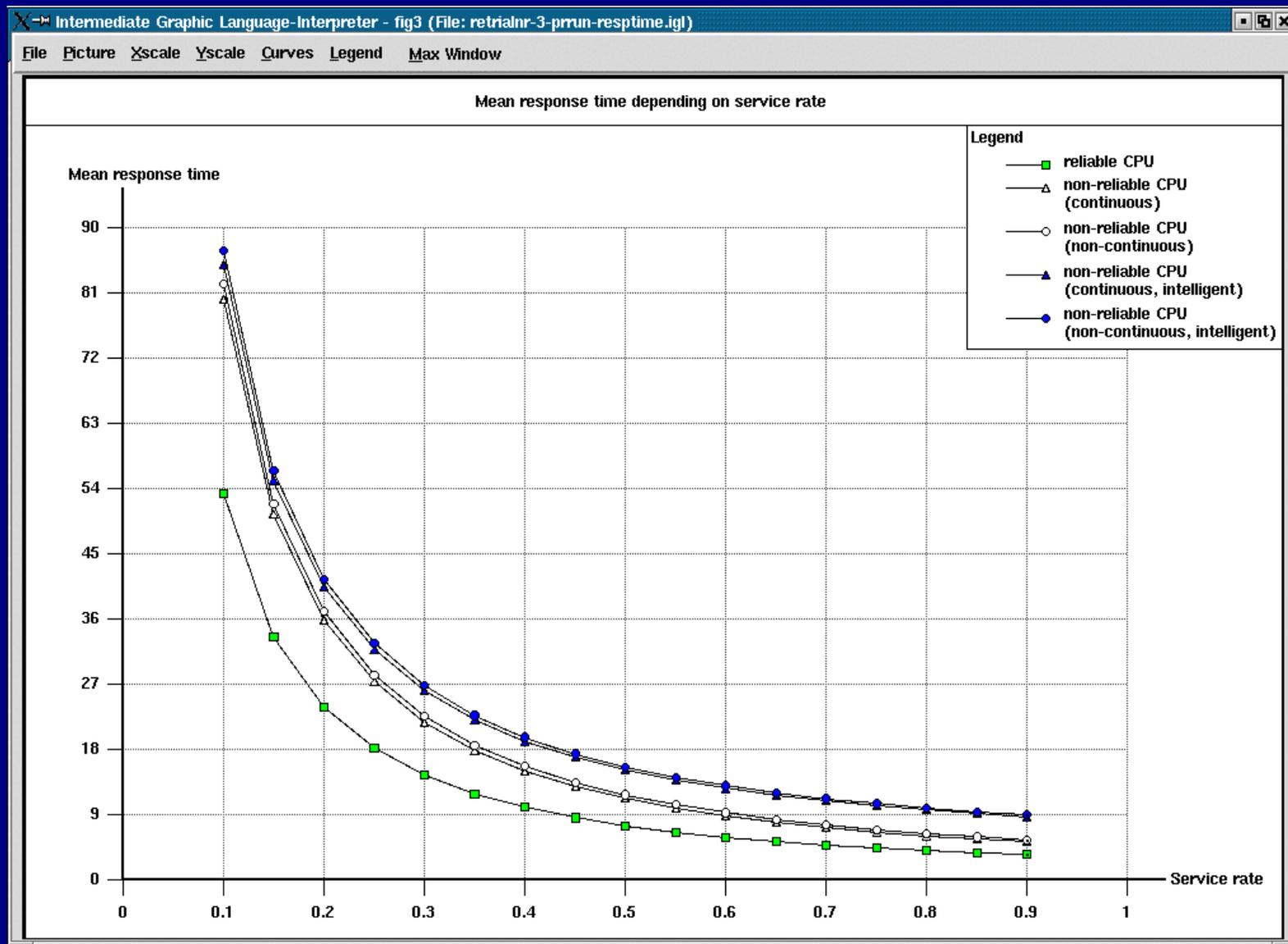
	non-rel. retrial(cont.)	non-rel. retrial(orbit)	non-rel. FIFO
Number of sources:	3	3	3
Request's generation rate:	0.1	0.1	0.1
Service rate:	1	1	1
Retrial rate:	1e+25	1e+25	-
Server's failure rate:	0.01	0.01	0.01
Server's repair rate:	0.05	0.05	0.05
Utilization of the server:	0.22327965614	0.22327965229	0.22327964521
Utilization of the sources:	0.74426549207	0.74426549375	0.74426549684
Mean response time:	1.43606563309	1.43606560277	1.43606554705



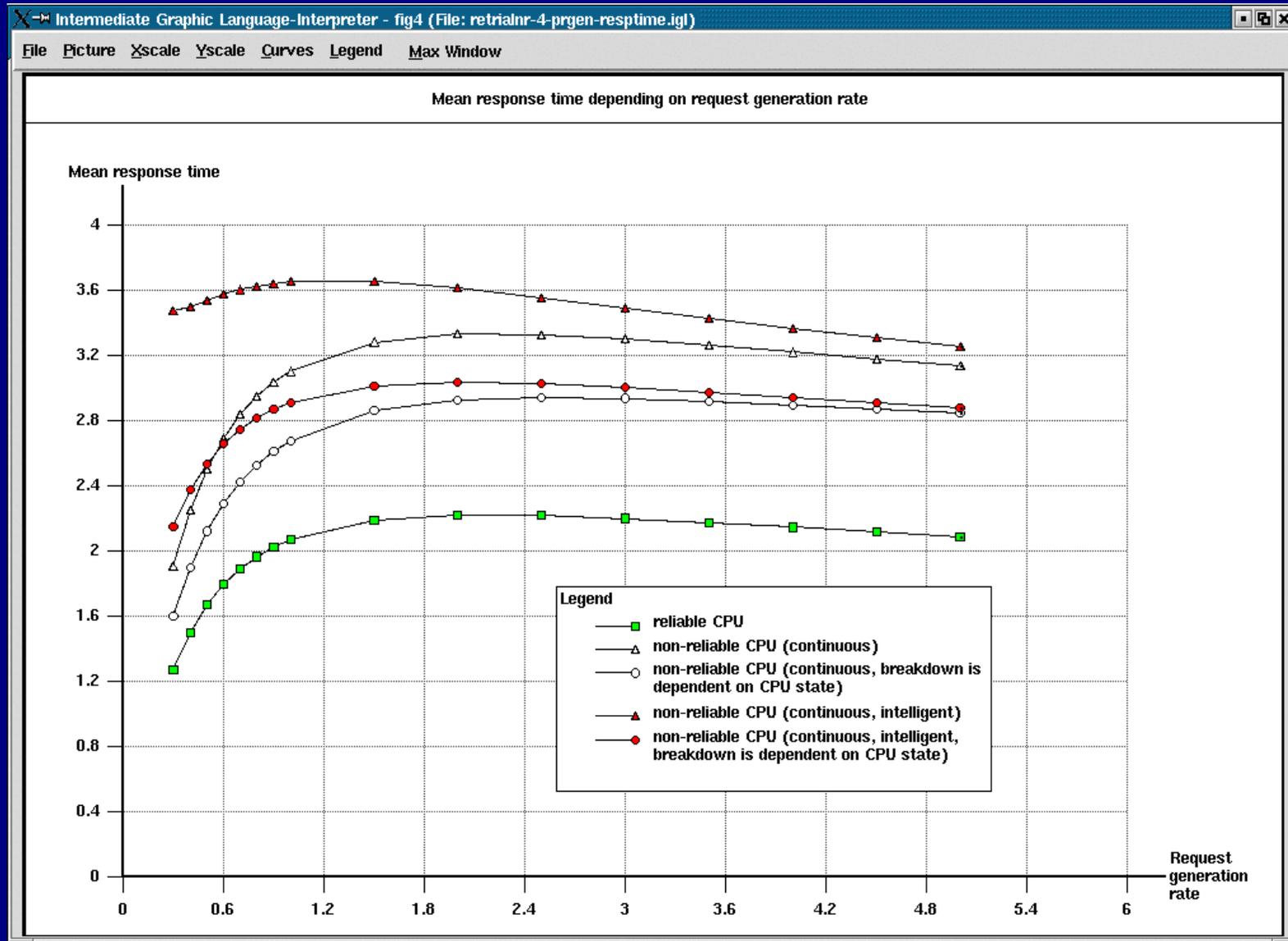
$E[T]$  versus primary request generation rate



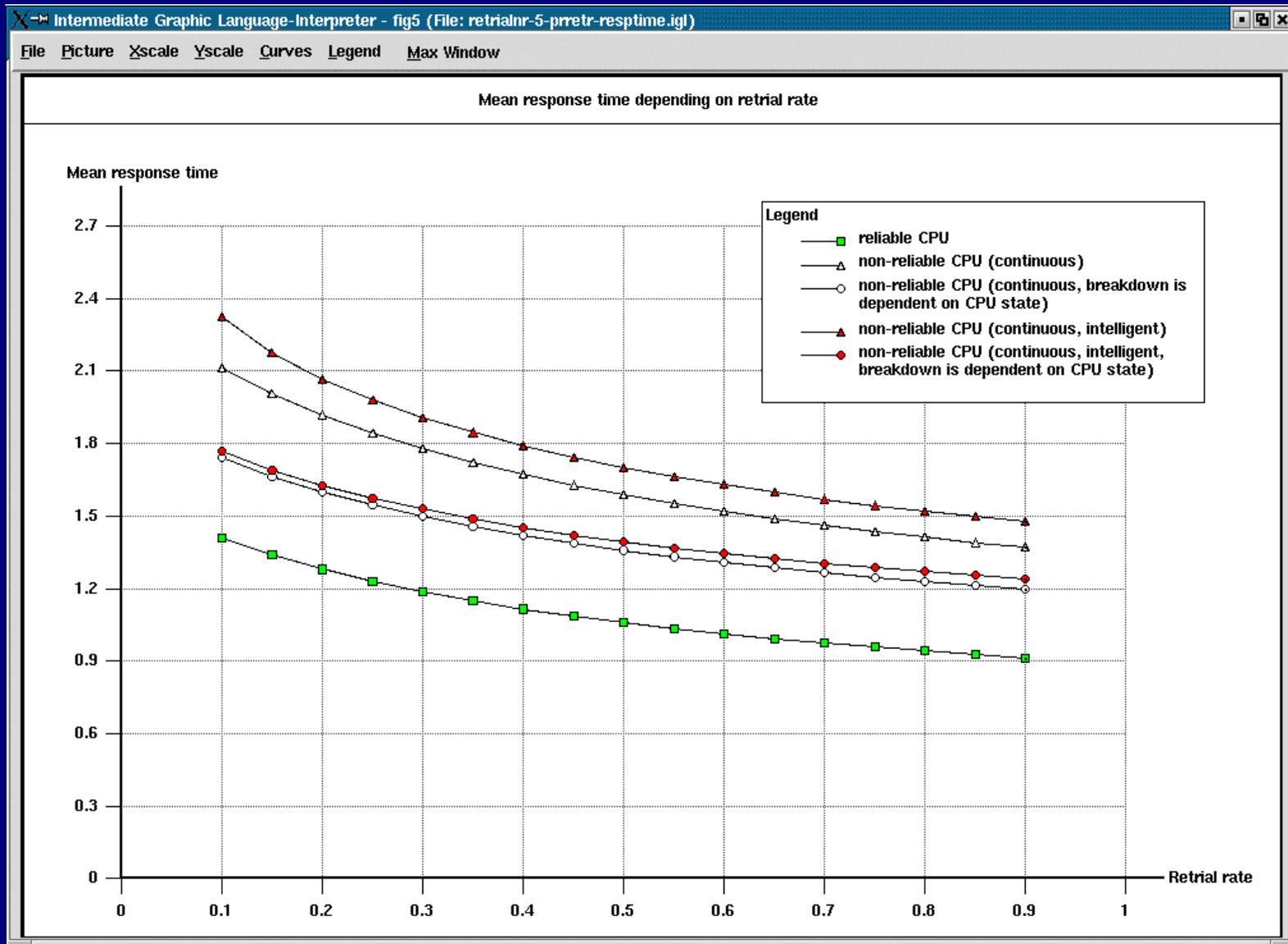
$E[T]$  versus retrieval rate



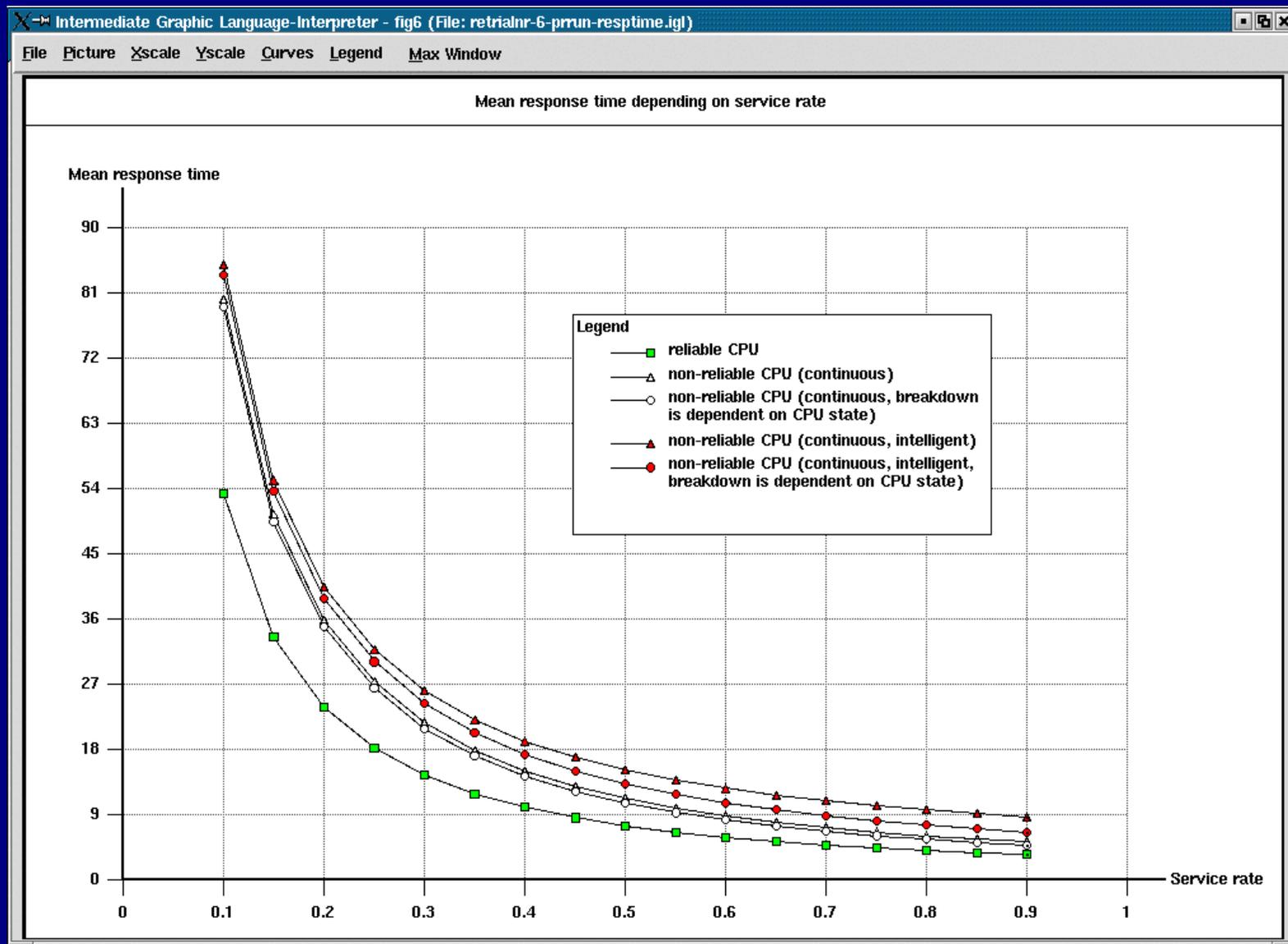
$E[T]$  versus service rate



$E[T]$  versus primary request generation rate



$E[T]$  versus retrial rate



$E[T]$  versus service rate

## References

- [1] **Begain K., Bolch G., Herold H.** *Practical Performance Modeling, Application of the MOSEL Language*, Kluwer Academic Publisher, Boston, 2001.
- [2] **Falin G.I. and Templeton J.G.C.** *Retrial queues*, Chapman and Hall, London, 1997.
- [3] **Falin G.I. and Artalejo J.R.** A finite source retrial queue, *European Journal of Operational Research* 108(1998) 409-424.
- [4] **Sztrik J. and Gál T.** A recursive solution of a queueing model for a multi-terminal system subject to breakdowns, *Performance Evaluation* 11(1990) 1–7.
- [5] **Wang Jinting, Cao Jinhua and Li Quanlin** Reliability analysis of the retrial queue with server breakdowns and repairs, *Queueing Systems Theory and Applications* 38(2001), 363–380.