

INFOKOMMUNIKÁCIÓS RENDSZEREK HATÉKONYSÁG- ELEMZÉSÉRE SZOLGÁLÓ ESZKÖZÖK

TOOL SUPPORTED PERFORMANCE MODELLING OF INFOCOMMUNICATION SYSTEMS

*Sztrik János, jsztrik@inf.unideb.hu
Debreceni Egyetem, Informatikai Kar*

*Kim Che Soong, dowoo@sangji.ac.kr
Sangji University, Korea*

1. Bevezetés

A gyakorlati problémák megoldásánál nagy segítséget jelent, ha olyan eszköz áll rendelkezésünkre, melynek használatával a modellezés bizonyos értelemben automatizálódik. Sokszor a legnagyobb probléma az, hogy hogyan tudjuk megfogalmazni a feladatot, majd anélkül, hogy megadnánk az egyensúlyi állapotegyenleteket kiszámítani a rendszerjellemzőket. Ebben segítenek az ún. "tools"-ok, melyek valamilyen sajátos programnyelven lehetővé teszik, hogy leírjuk vizsgálandó rendszert. Ez általában az SPN (Stochastic Petri Nets , Sztochasztikus Petri hálók), vagy különböző fajtái ún. színezett, általánosított, stb segítségével tehető meg. Manapság nagyon gyakran grafikus szerkesztő áll rendelkezésünkre. Ezután a program generálja a leíró sztochasztikus-folyamatot, ami analitikus esetben általában Markov-lánc. Ezt követően az ún SOLVER feladata, hogy megadja a rendszerjellemzőket. A különböző programoktól függ, hogy milyen módszert alkalmaznak. Ez történhet a lineáris egyenletrendszerek megoldására kifejlesztett eljárással, közelítéssel, szimulációval stb. Természetesen több neves kutatóközpont, egyetem foglalkozik ilyen eszközök fejlesztésével. Az igazán használható szoftverek azonban nagyon drágák, ezért különös gondot kell fordítani az ingyen elérhető tool-okra. Néhány forrás pld.

- A dortmundi egyetemen kifejlesztett programok:
HIT, MACOM, HiQPN,DSPNexpress
<http://ls4-www.informatik.uni-dortmund.de/tools.html>
- Az erlangeni egyetemen kifejlesztett programok:
PEPSY, MOSEL
<http://www4.informatik.uni-erlangen.de/Projects/MOSEL/>
- A Duke egyetem CACC-Duke kutatóközpontjában elkészített programok:
SHARPE, SPNP, iSPN, RAFT
<http://www.ee.duke.edu/~kst/tools.html>

A kutatásokhoz a KOSEF-HAS, 2004, koreai-magyar tudományos együttműködési pályázat részleges anyagi támogatást nyújtott

2. A MOSEL tool

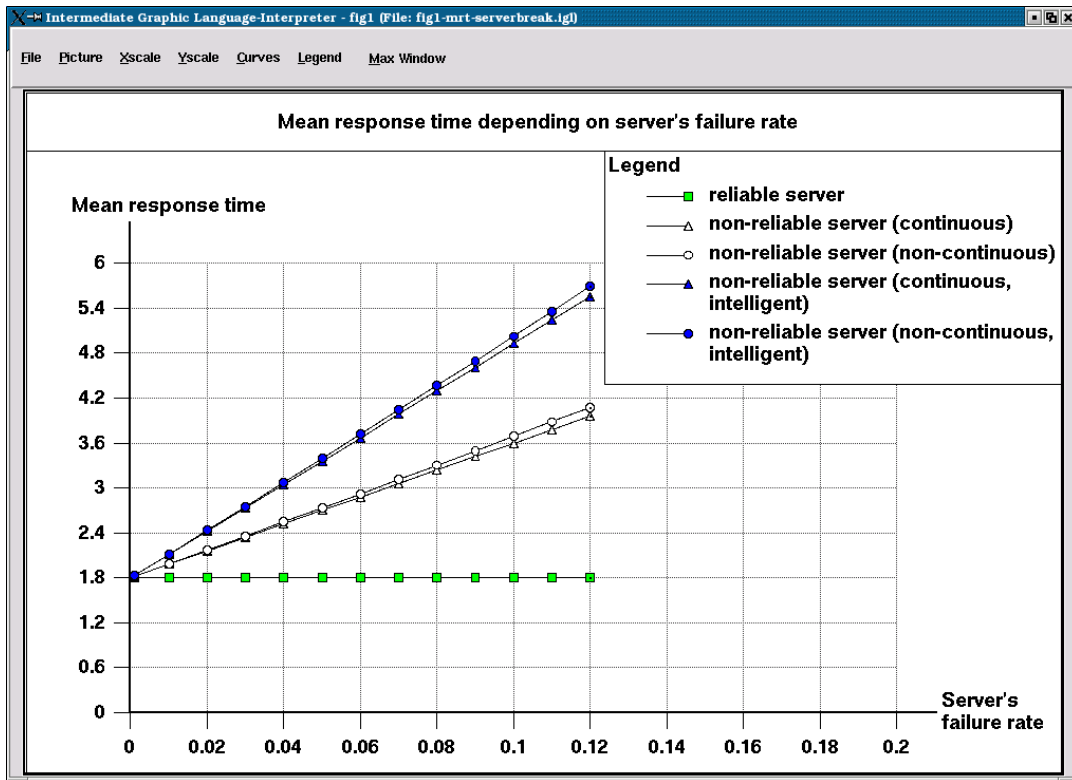
Ez a fejezet egy véges forrású, visszatérési sorbanállási rendszer teljesítményelemzésével foglalkozik, melyben az igényforrások és a kiszolgálók is azonos paraméterekkel rendelkeznek. Feltesszük, hogy a kiszolgálók egy véletlen idő eltelte után meghibásodnak és javításra szorulnak. A modellalkotás során minden valószínűségi változót exponenciális eloszlásúnak és egymástól kölcsönösen függetlennek tételeztünk fel. A vizsgált rendszerjellemzőket a MOSEL hatékonyságvizsgálati szoftver segítségével számítottuk ki. Néhány fontosabb rendszer-paraméternek (új igény érkezési, újragenerálási, kiszolgálási és a kiszolgáló egység meghibásodási illetve javítási intenzitásának) az átlagos válaszidőre, a rendszer össz-kihasználtságára, a rendszerben tartózkodó igények átlagos számára gyakorolt hatását a numerikus eredmények grafikus ábrázolásával lehet elemezni. Az alábbiakban látható hogyan fogalmazható meg a probléma.

```

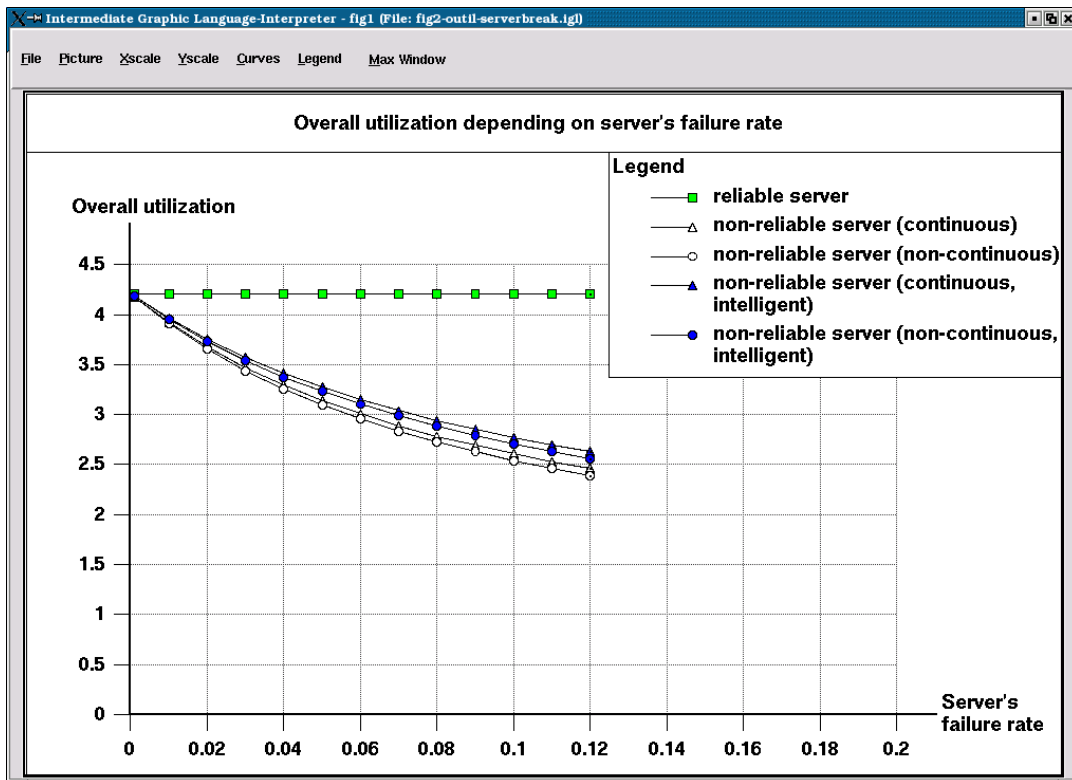
/* retrievalnr-hom-cpu-cont.msl begins */
/*----- Definitions -----*/
#define NT 6
VAR double prgen;
VAR double prretr;
VAR double prrun;
VAR double cpubreak_idle;
VAR double cpubreak_busy;
VAR double cpurepair;
enum cpu_states {cpu_busy, cpu_idle};
enum cpu_updown {cpu_up, cpu_down};
/*----- Node definitions -----*/
NODE busy_terminals[NT] = NT;
NODE retrying_terminals[NT] = 0;
NODE waiting_terminals[1] = 0;
NODE cpu_state[cpu_states] = cpu_idle;
NODE cpu[cpu_updown] = cpu_up;
/*----- Transitions -----*/
IF cpu==cpu_up FROM cpu_idle, busy_terminals
TO cpu_busy, waiting_terminals W prgen*busy_terminals;
IF cpu==cpu_up AND cpu_state==cpu_busy FROM busy_terminals
TO retrying_terminals W prgen*busy_terminals;
IF cpu==cpu_up FROM cpu_idle, retrying_terminals
TO cpu_busy, waiting_terminals W prretr*retrying_terminals;
IF cpu==cpu_up FROM cpu_busy, waiting_terminals
TO cpu_idle, busy_terminals W prrun;
IF cpu_state==cpu_idle FROM cpu_up TO cpu_down W cpubreak_idle;
IF cpu_state==cpu_busy FROM cpu_up TO cpu_down W cpubreak_busy;
FROM cpu_down TO cpu_up W cpurepair;
/*----- Results -----*/
RESULT>> if(cpu==cpu_up AND cpu_state==cpu_busy) cpuutil += PROB;
RESULT>> if(cpu==cpu_up) goodcpu += PROB;
RESULT if(cpu==cpu_up) busyterm += (PROB*busy_terminals);
RESULT>> termutil = busyterm / NT;
RESULT>> if(cpu==cpu_up) retravg += (PROB*retrying_terminals);
RESULT if(waiting_terminals>0) waitall += (PROB*waiting_terminals);
RESULT if(retrying_terminals>0)
retrall += (PROB*retrying_terminals);
RESULT>> resptime = (retrall + waitall) / NT / (prgen * termutil);
RESULT>> overallutil = cpuutil + busyterm;
/* retrievalnr-hom-cpu-cont.msl ended */

```

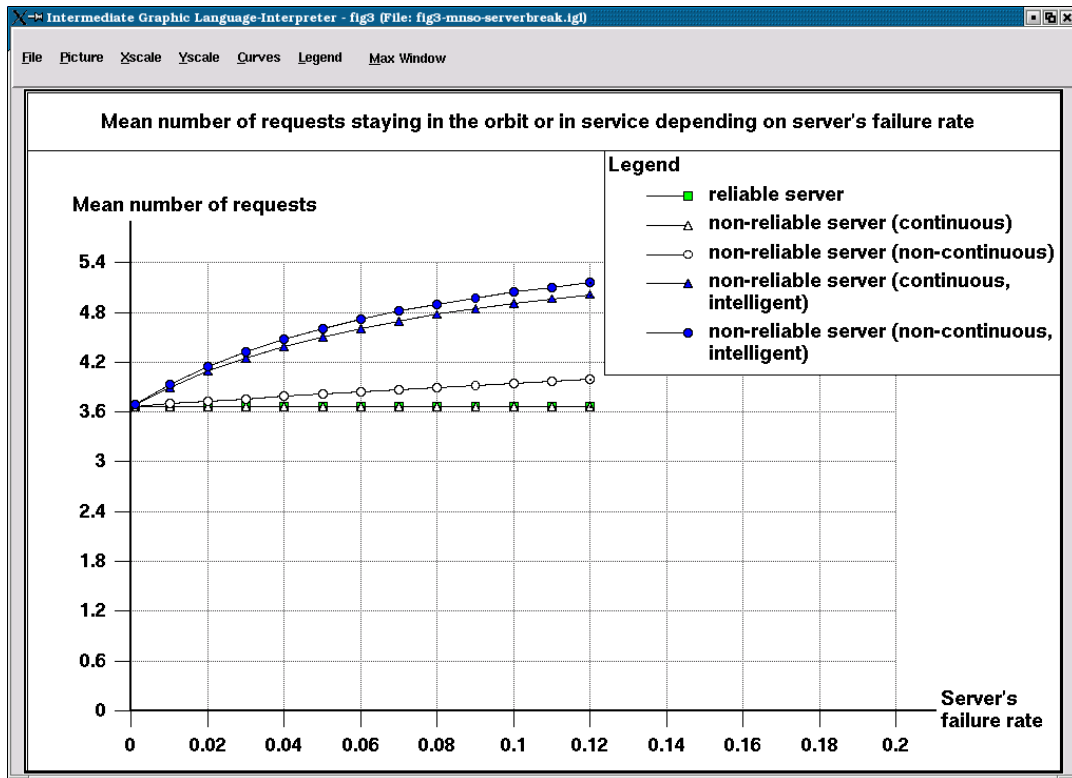
A rendszerjellemzőket a MOSEL által generált grafikonokon jelenítettük meg. Arra voltunk kíváncsiak, hogy milyen hatással van a kiszolgáló egység meghibásodási illetve javítási intenzitása az átlagos válasz időre, az össz-kihasználtságra és a rendszerben tartózkodó igények átlagos számára. Az 1-3 Ábrák a meghibásodás a 4-6 Ábrák pedig a javítás hatását szemléltetik.



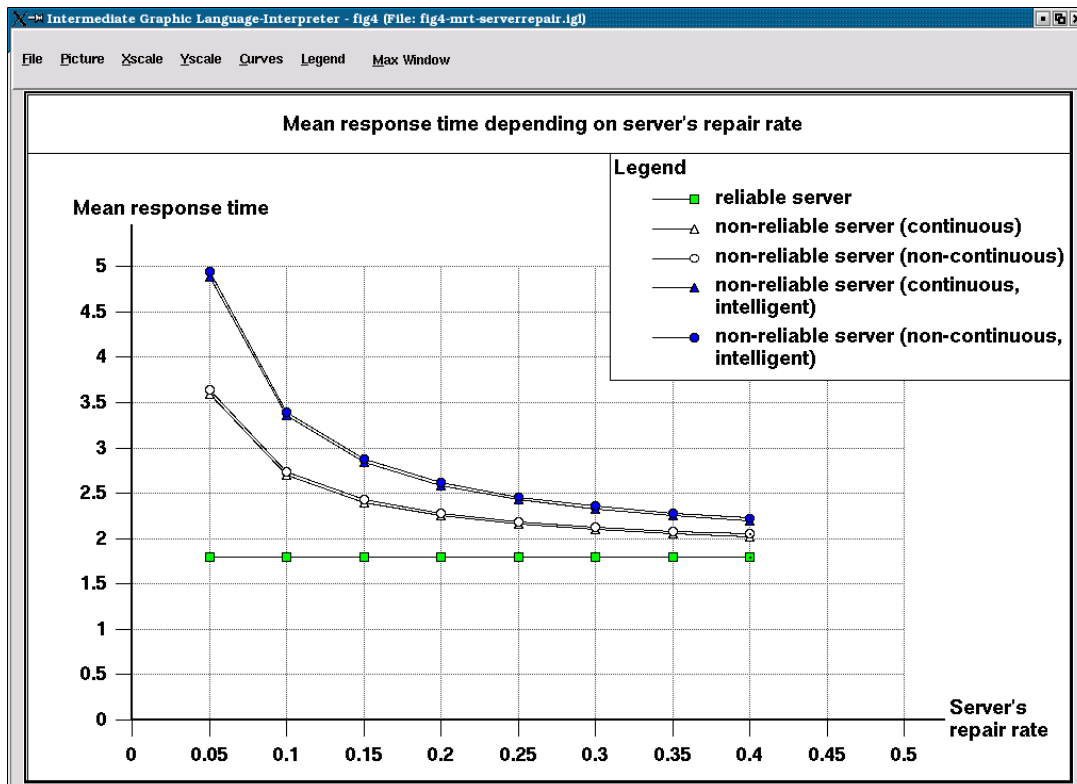
1. Ábra



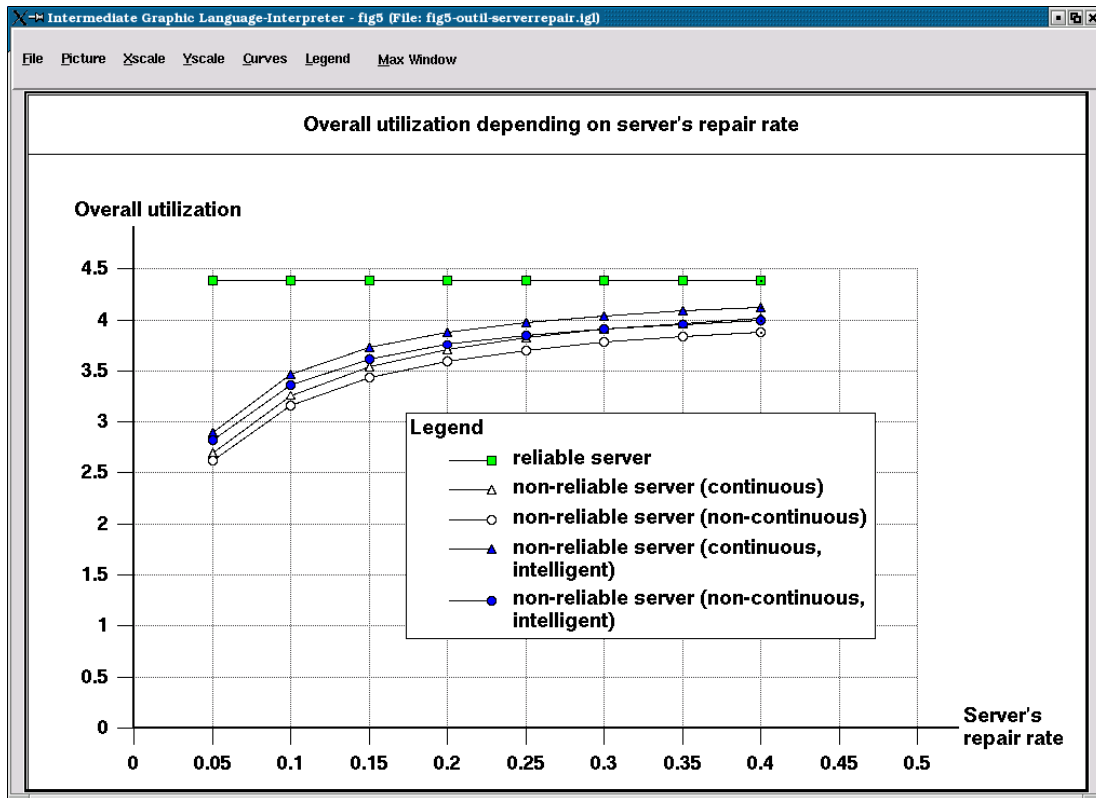
2. Ábra



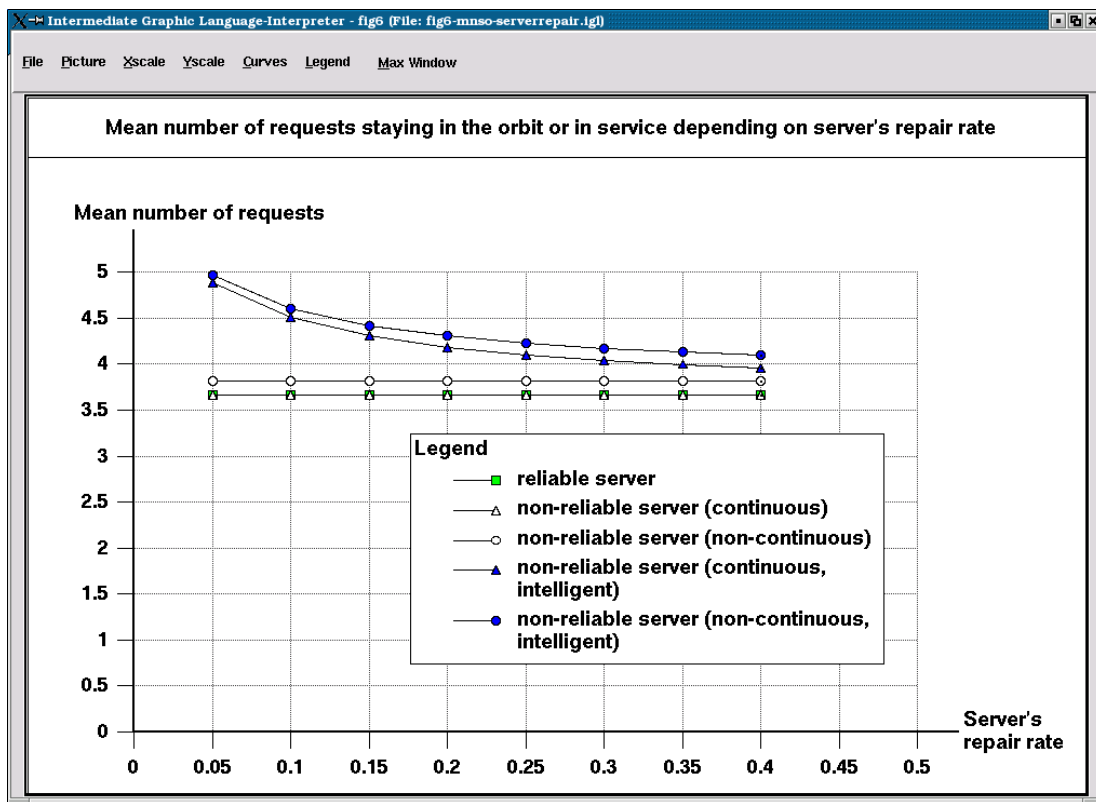
3. Ábra



4. Ábra



5. Ábra



6. Ábra

Természetesen ezek a jellemzők nagyon függenek a többi intenzitástól is. Ezeken a grafikonokon a tendenciát akartuk megmutatni nem pedig a pontos eredményeket, hiszen már az exponenciális eloszlás feltételezése is kérdéseket vethet fel. A vizsgált modellel és a tool-okkal kapcsolatos legfontosabb munkákat az Irodalomjegyzékben soroltuk fel. Nagy segítséget jelent még a <http://www2.uwindsor.ca/~hlynka/pscpeopl.gif> cím is, amelyen számos más, a sorbanállási elméleti problémákkal kapcsolatos szoftverek találhatunk.

Irodalomjegyzék

1. **Almási B., Roszik J., Sztrik J.:** *Homogeneous finite-source retrieval queues with server subject to breakdowns and repairs*, Technical report, University of Debrecen, Hungary 2002/17.
2. **Aissani A. and Artalejo J.R.:** *On the single server retrieval queue subject to breakdowns*, Queueing Systems, 30(1998), 309-321.
3. **Begain K., Bolch G. and Herold H.:** *Practical performance modeling, application of the MOSEL language*, Kluwer Academic Publisher, Boston, 2001.
4. **Falin G.I. and Artalejo J.R.:** *A finite source retrieval queue*, European Journal of Operational Research, 108(1998), 409-424.
5. **Haverkort B.:** *Performance of Computer Communication Systems*, Wiley, New York, 1998.
6. **Jain R.:** *The Art of Computer Performance Analysis*, Wiley, New York, 1991.