MODELING ON MICROS AND WORKSTATIONS

Edited by
Thomas A. Rathburn

SCS THE SOCIETY FOR COMPUTER SIMULATION

Modeling on Micros and Workstations

Proceedings of the SCS Multiconference on
Modeling on Micros and Workstations
23-25 January 1991
Anaheim, California

Edited by
Thomas A. Rathburn
Kent State University

Sponsored by
The Society for Computer Simulation (SCS)

SCS

Additional copies of the Proceedings are available from:

The Society for Computer Simulation
P. O. Box 17900
San Diego, CA 92177 U.S.A.

ISBN 0-911801-84-7

PRINTED IN THE UNITED STATES

PREFACE

As the performance gap between microcomputers and mainframes continues to close, modeling in the micro and workstation environments continues to become more sophisticated. The papers in this proceeding are the survivors of a process that began in July with the submission of abstracts by authors and culminated in January at the Disneyland Hotel in Anaheim, California.

Throughout this process, many individuals have made important contributions. I am indebted to everyone who made it possible, especially Chip Stockton and the staff of The Society for Computer Simulation, Jay Weinroth at Kent State University, and the authors. I would also like to thank the Department of Administrative Sciences in the College of Business at Kent State University for all their support, monetary and otherwise.

In organizing this meeting and preparing the Proceedings, abstracts were received from industry, government, and academic environments. This Proceedings includes the work of authors from three continents. The papers included in this Proceedings were presented in six sessions over a two-day period. Included are topics ranging from innovative applications to modeling design and evaluation techniques. As always, all credit for an interesting and informative conference goes to the authors whose work is represented here. My biggest hope is that the errors that I have introduced have been minimal and do not seriously detract from the efforts of these individuals.

Thomas A. Rathburn
Kent State University

CONTENTS

CONTENTS

MODELING ON MICROS AND WORKSTATIONS

ABET- AVAILABILITY BENEFIT EVALUATION TOOL

Philip H. Young and Robert L. Abramson
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009-2957

ABSTRACT

The Availability Benefit Evaluation Tool (ABET) provides insight into availability of a deployed and activated constellation of satellites. The ABET methodology analytically computes exact probabilities of having m operational satellites available out of a constellation of n satellites. ABET considers the effects of active sparing as well as various failure and downtime characteristics of the satellites.

I. INTRODUCTION

Estimated cost for various satellite system concepts is often used as an essential figure of merit in comparing one satellite system concept with another. Cost to a large extent depends upon availability (defined as the fraction of time the constellation can maintain its operational status). Availability of a constellation of satellites, in turn, depends upon availability of the individual satellites that constitute the constellation.

The Availability Benefit Evaluation Tool (ABET) is designed to provide insight into availability of a deployed and activated constellation. The ABET methodology analytically computes exact probabilities of having m operational satellites available out of a constellation of n satellites (no Monte Carlo sampling is used). ABET considers the effects of active sparing (spare satellites fail at the same rate as active satellites) as well as various failure and downtime characteristics of the satellites.

Finally, ABET is a system-level methodology, so its purpose is to help evaluate, at a high level of aggregation, one satellite system concept against another for performing a given mission. In this sense, it is a companion to the Launch Interval Forecasting Tool (LIFT) (Young, et al, 1990), whose main objective is to schedule yearly deployment of satellites, given the same type of general information as ABET uses. As with LIFT, ABET methodology is simple enough to easily be implemented on a personal computer.

The methodology will be presented first assuming exponential failure time and downtime distributions for each satellite. Then it will be shown that the same general principles apply when these assumptions are replaced by a general distribution for these quantities.

II. METHODOLOGY DESCRIPTION

This section describes the mathematical methods used in ABET. First some preliminary concepts will be presented, followed by the detailed mathematical structure.

A. Preliminary Concepts

The fundamental parameters and assumptions of ABET, together with a definition of input and output parameters will now be specified. Input parameters are as follows:

1. (n) - Number of satellites in constellation.

2. (m) - Number of satellites required to accomplish mission, $m \leq n$.

3. (s) - Number of active (or "hot") spares in constellation, $0 \leq s < \infty$.

4. (MTBF) - Mean time before failure, $0 < MTBF < \infty$.

5. (MDT) - Mean downtime, $0 < MDT < \infty$.

Fundamental assumptions of ABET are as follows:

1. All satellites operate mutually independent of each other.

2. The entire constellation of satellites (including active spares) is deployed and activated.

3. All satellites have same probability distribution of downtime ($\tilde{t}_d$):

$$Pr(\tilde{t}_d \leq t_d) = F_d(t_d)$$

$$= 1 - \exp\left(-\frac{t_d}{MDT}\right), \text{ where} \quad (1)$$

$$E(\tilde{t}_d) = MDT \quad (2)$$

4. All satellites have same probability distribution of time before failure ($\tilde{t}_f$):

$$Pr(\tilde{t}_f \leq t_f) = F_r(t_f)$$

$$= 1 - \exp\left(-\frac{t_f}{MTBF}\right) \quad (3)$$

$$E(\tilde{t}_f) = MTBF \quad (4)$$

5. A random variable is a real-valued function defined on the space of outcomes (in these cases, the space of outcomes is either time or availability). The ~ sign over a symbol marks that symbol as a random variable; e.g., equations (1) - (4) above.

6. Failure time and downtime are independent.

Output produced by ABET is the probability distribution of the availability of having m operational satellites out of a total of n + s.

B. Mathematical Description

The Appendix describes the methodology used to form the availability distribution for each satellite (i). If r =MDT/MTBF, then the resulting distribution is:

$$Pr(\tilde{Z}_i \leq Z_i) = F(Z_i)$$

$$= \frac{rZ_i}{(r-1)\ Z_i + 1} \quad (5)$$

$$0 \leq Z_i \leq 1$$

The Z_i in equation (5) is the availability of satellite i.

It follows from equation (5) that each of the n+s satellites has the same availability distribution $[F(Z_i)]$. If each of the satellites is activated at the same time, then $\tilde{Z}_1 = \ldots = \tilde{Z}_{n+s} = \tilde{Z}$, and the distribution of $\tilde{Z}$ is F(Z) as defined by equation (5). Since by assumption 1 all n+s satellites are probabilistically mutually independent, the availability of having at least m out of n+s operational satellites is given by:

$$\tilde{W} = \sum_{i=m}^{n+s} \binom{n+s}{i} \tilde{Z}^i \ (1 - \tilde{Z})^{n+s-i}$$

$$= P(\tilde{Z}) \quad (6)$$

The probability distribution of $\tilde{W}$ will be represented by:

$$G(W) = Pr(\tilde{W} \leq W)$$

The kth fractile value of W is obtained by solving the equation:

$$G(W) = k, \ (0 \leq k \leq 1) \quad (7)$$

for $W = W_k$

If, in equation (7) k = 0.5, $W_{0.5}$ is the median.

The following theorem allows W_k to be obtained without explicit knowledge of G(W).

Theorem: Let $\tilde{V}$ be a random variable with probability distribution

$$F(v) = Pr(\tilde{V} \leq v),$$

and $\tilde{U} = h(\tilde{V})$ a random variable, where h is an invertible Borel measurable function. If k is the kth fractile value of F(v), then $u_k = h(v_k)$ is the kth fractile value of the distribution $Pr(\tilde{U} \leq u) = G(u)$.

Proof: Let u = h(v).

Then $v = h^{-1}(u)$, and it follows that:

$$G(u) = Pr(\tilde{U} \leq u) = Pr[\ h(\tilde{V}) \leq u]$$

$$= Pr[\ \tilde{V} \leq h^{-1}(u)\ ] = F[\ h^{-1}(u)\ ].$$

Thus, the kth fractile of $\tilde{V}$, namely $v_k = h^{-1}(u_k)$, is such that

$$G(u_k) = F[\ h^{-1}(u_k)\ ] = F(v_k) = k.$$

Therefore, v_k and u_k both correspond to the same fractile (k).

A tedious computation shows that the function P(Z) of equation (6) is monotonically increasing with respect to Z and, therefore, invertible. Applying the above theorem to equation (6), it follows that:

$$W_k = \sum_{i=m}^{n+s} \binom{n+s}{i} Z_k^i \ (1 - Z_k)^{n+s-i}$$

$$= P(Z_k) \quad (8)$$

where the median (k = 0.5) of W is $W_{0.5} = P(Z_{0.5})$.

Setting F(Z) = 0.5 in equation (A3) of the Appendix implies:

$$Z_{0.5} = \frac{MTBF}{MTBF + MDT} = \frac{1}{1 + r},$$

recalling that r = MDT/MTBF and MTBF and MDT are each from an exponential distribution.

Thus, given the median of Z, the median of W can be obtained. Also, it is clear that given any fractile value of Z, the corresponding fractile value of W can be obtained by using equation (8). Therefore, the probability distribution G(W), can be approximated to any degree of resolution <u>without</u> "Monte Carlo" sampling by just using enough fractile values of Z and obtaining the corresponding value of W. Using equation (5), the value of Z corresponding to the kth fractile of F(Z) is:

$$Z_k = \frac{k}{k\ (1 - r) + r}$$

For large values of n + s, evaluation of equation (8) can be tedious. However, for a small price in accuracy the Normal approximation to W_k can be used to significantly speed up the evaluation.

(For a description of the Normal approximation to the Binomial, see Cramer, 1946, pages 198-203). The form of this is:

$$\mu_{NA}^{(k)} = (n + s)\ Z_k$$

$$\sigma_{NA}^{(k)} = \sqrt{(n + s)\ Z_k\ (1 - Z_k)}$$

$$t_{NA}^{(k)} = [m - \mu_{NA}^{(k)}]\ /\sigma_{NA}^{(k)}$$

Thus, $t_{NA}^{(k)}$ is a standard Normal value and the corresponding W_k can be evaluated as:

$$W_k = \left\{1/\sqrt{2\Pi}\right\} \int_{t_{NA}^{(k)}}^{\infty} \exp\ (-u^2/2)\ du$$

W_k can be obained from standard Normal distribution tables, or by a standard rational approximation to the integral (see Cramer, 1946, pages 557-558).

III. GENERALIZATIONS

If assumptions 3 and 4 in Section IIA are removed, and downtime ($\tilde{t}_d$) and time before failure ($\tilde{t}_f$) distributions are represented in general as:

$$\Pr\ (\tilde{t}_d \le t_d) = F_d\ (t_d), \text{ and}$$

$$\Pr\ (\tilde{t}_f \le t_f) = F_r\ (t_f);$$

then the distribution of availability $[\tilde{Z} = \tilde{t}_f/(\tilde{t}_f + \tilde{t}_d)]$ is represented as:

$$\Pr\ (\tilde{Z} \le Z) = \Pr\ [\tilde{t}_f/(\tilde{t}_f + \tilde{t}_d) \le\ Z] = F(Z)$$

F(Z) is evaluated for any $F_d(t_d)$ and $F_r(t_f)$, by following a similar process to the one described in the Appendix. Thus, the methodology in Section IIB can be applied to F(Z) as follows:

(a) Find Z_k such that $F(Z_k) = k$ (may require a numerical method of solution).

(b) Evaluate $W_k = P(Z_k)$ using equation (8).

(c) W_k, by the theorem in Section IIB is, using equation (8), the kth fractile of the distribution G(W) if W = P(Z) is invertible.

If F(Z) is not a simple analytic form that can be solved for Z_k in closed form (for example when $F_d(t_d)$ and $F_r(t_f)$ are Weibull or Normal distributions), an iterative process, such as Newton's method may be used. In any event, when Z_k is known, the evaluation of $P(Z_k)$ is straightforward.

IV. CONCLUSIONS

The ABET methodology provides a companion analysis tool to the LIFT methodology. Thus, given MDT and MTBF either the LIFT or the ABET methodologies can be utilized to provide insight into satellite constellation characteristics, and consequently cost. If the satellites are described in terms of their components and certain knowledge about the components is available, it is possible to determine the above satellite reliability parameters.

The value of ABET, as stated in the introduction, lies in the system-level trades that evaluate various satellite system concepts, before more detailed methodologies with their concomitant complexity are applied.

A prototypical version of ABET, containing all attributes described here, can easily be implemented using a convenient programming language on almost any personal computer system.

REFERENCES

Cramer, H., "Mathematical Methods of Statistics", Princeton University Press, 1946.

Young, P. H., N. E. King, R. L. Abramson, M.A. Rolenz, "LIFT - The Launch Interval Forecasting Tool", Proceedings of the 1990 SCS Western Multiconference, SCS, January 17-19, 1990.

ACKNOWLEDGEMENT

We wish to thank Stephen A. Book for his perceptive editorial suggestions, and Marie Calamaria for her creativity and resourcefulness in typing the manuscript.

APPENDIX

DERIVATION OF F(Z)

It is assumed that the probability distributions for time before failure ($\tilde{t}_f$) and downtime ($\tilde{t}_d$) are defined as in assumptions 3 and 4 of Section IIA.

Availability for any satellite ($\tilde{Z}$) is the ratio of time before failure to time before failure + downtime, as follows:

$$\tilde{Z} = \frac{\tilde{t}_f}{\tilde{t}_f + \tilde{t}_d} \qquad \text{(A1)}$$

Thus, the probability distribution [F(Z)] of $\tilde{Z}$ (from A1) is:

$$F(Z) = Pr\ (\tilde{Z} \le Z) = Pr\left\{\frac{\tilde{t}_f}{\tilde{t}_f + \tilde{t}_d} \le Z\right\}$$

$$= \int_o^\infty \int_L^\infty dF_d(t_d) dF_r(t_f) \qquad \text{(A2)}$$

where: $L = \left\{\frac{1 - Z}{Z}\ t_f\right\}$

Substituting the exponential form for F_d and F_r from equations (1) and (3) in Section IIA, (A2) becomes:

$$F(Z) = \frac{MDT}{(MDT-MTBF)} \times$$

$$\left\{1 - \frac{MTBF}{(MDT-MTBF)\ Z + MTBF}\right\}, \text{ where} \qquad \text{(A3)}$$

$0 \le Z \le 1$

The density function of F(Z) is:

$$f(Z_i) = \frac{(MTBF)(MDT)}{[(MTBF-MDT)\ Z + MTBF]^2}$$

Making the substitution r = MDT/MTBF in (A3)

$$F(Z) = \frac{r}{(1-r)}\left\{\frac{1}{(r-1)\ Z + 1} - 1\right\}$$

$$= \frac{rZ}{(r - 1)\ Z + 1}$$

REAL-TIME MODELS FOR AN ENGINEERING SIMULATOR OF A 660 MWe FOSSIL-FIRED POWER PLANT.

G.B.Garbossa, S.Spelta, P.Groppelli, U.Zoni
ENEL - Research and Development Department
Automatica Research Center
Via Volta, 1 - 20093 COLOGNO M.se (MI) ITALY

A.Rossi
ANSALDO S.p.A. Via Dei Pescatori, 35
16128 GENOVA ITALY

ABSTRACT

This paper describes the models and some results of a project promoted by the Italian National Electricity Board (ENEL) for the development of a real time simulator of the Brindisi Sud station.
In this site four-twin units with multifuel fossil-fired UP boiler, rated at 660 MWe, with supercritical cycle, are under construction.
Other features of this new type of plant for Italy, are related to:
- combustion installations with over fire air injection and external flue gas recirculation (where it is mixed with combustion air), to obtain substantial NOx reductions;
- start-up turbine bypass system.

  To help engineers and plant operators to better understand the behavior of plant equipments and processes and to predict the impact of the new concepts the realization of an engineering real time simulator, using the LEGO code, was established. Other simulator uses include:
- testing of control equipments already installed on the plant;
- control system tuning;
- evaluation of operating procedures.

To cover all these needs the simulator has been set-up to describe the main power plant process behaviour from warm start-up to full power operations.
The scope of simulation includes the following plant subsystems:
condensate, feedwater chain, boiler steam/water, turbine, boiler fuel system, combustion air and flue gas.
In addition the control system related to the boiler, the turbine and the balance of plant are taken into account.
In the paper a brief description of the LEGO code characteristics and a short summary of the simulator features are also presented.

INTRODUCTION

In recent years Italy's Ente Nazionale per l'Energia Elettrica (ENEL), the state-owned Italian power authority, has slowed down its construction program and in some cases stopped the installation of new powerplants. This is because of pubblic opposition to nuclear development and the increasing difficulties to obtain approvals for new large fossil-fired stations.

The need to burn fossil fuels for power well into the future, and to minimize the impact on the environment has markedly influenced boiler design. To achieve the strict pollutant emission limits imposed by Italian laws, for the four units under construction at BRINDISI SUD station, the following plant design criteria, for SO_2, particulates and NOx emissions were adopted:

- flue gas desulphurization,
- electrostatic precipitators,
- flue gas recirculation through the burners and use of overfire NOx ports.

In addition to improve the flexibility, the reliability and the availability of power plant units some modifications to the start-up turbine by-pass system were included. All these new operating conditions justify investing in the realization of an engineering simulator of the plant for training operators to:

- better understanding of the physical phenomena involved;
- acquisition and enhancement of operating procedures;
- manipulation of control parameters to pre-tune control system and reduce start-up time.

In this paper a set of system models capable of describing steady-state and transient performance of BRINDISI SUD Power Station is presented.

These models represent the whole power plant, except auxiliary systems, and provide all the information required to run the simulator.

Up to now the accuracy of models, built utilizing the in-house developed LEGO code, is demonstrated through:

- comparison with the steady-state data calculated by the plant vendor at different power conditions.
- evaluation on engineering basis of the transient calculations, as plant trials are not available.

However comparing simulator model results to start-up plant data, that is the ultimate test of the model's capabilities and accuracy, has been planned.

To support plant personnel during start-up operations the simulator has been installed at BRINDISI SUD station, where many simulations are under way to:

- repeat operations until familiarity is achieved,
- learn operating limitations of equipment,
- experience normal and abnormal plant conditions,
- test the new start-up procedures.

As early examples, two model calculations are presented in this paper:

- Steady-state comparisons at 100% power,
- Load demand increase from about 10% to 20%.

LEGO CODE DESCRIPTION

The LEGO code is a modular package developed at the Research and Development Department of the Italian National Electricity Board (CRA-ENEL) to facilitate modeling of the dynamics of fossil-fueled and nuclear power plants. The LEGO code consists of a library of preprogrammed,

pre-tested and prevalidated modules, that represent power plant components, and a master program which allows the user to build-up a model by automatically interconnecting the modules in the arrangement determined by the modeler.
Each module describes a physical plant component to the prescribed level of fidelity and is independent of any other module.
A module consists of a lumped parameter model, derived from first principles, describing a physical process by means of a system of non-linear algebraic and/or differential equations.
A single component can be represented by different modules of different level of complexity to meet different modeling needs.
The basic characteristics of the LEGO package, described in a previous paper (Marcocci et al. 1983) can be summarized as follows:

- modularity: component models (modules) are available for general plant components (such as pipes, valves, pumps, heat exchangers, tanks, etc.) and the user can connect them in accordance with a specific plant design.
- flexibility: the user can solve special modeling problems by developing the mathematical model of special components, which can be included in the module library of the package.
- reliability: all the numerical algoritms used by the package are centralized in the master program. Module modifications, due to different mathematical modeling assumptions, do not require any numerical algorithm updating. Moreover the modules can exchange information among themselves only by means of the master program so that they can be considered independent.

With respect to the numerical problem the main LEGO features are:

- simultaneous solution of all non-linear algebraic and differential equations, using an implicit formula for the numerical integration method.
- use of sparse matrix techniques in order to reduce computation time, dealing with large power plant models.
- steady-state computation, allowing interchange of the role of input variables, output variables and uncertain constant parameters.

The LEGO code is a flexible and powerful modeling tool suitable for insertion in the iterative process of power plant design and plant troubleshooting. It has been thoroughly valitated against plant trials with respect to the fossil-fired units (Mafezzoni et al. 1983) and to the pressurized water reactor units (Spelta 1985).
Using an appropriated module library the LEGO code has been utilized also for performing a number of other functions in different field (Magnani et al. 1988).
In the last few years an extensive effort has been made to develop an interactive software environment, named LEGOCAD (Anzano et al. 1990). This package suitable for running on workstation has the following features:

- allows a more friendly work session,

- saves time for model building and set-up,
- avoids formal mistake during model handling,
- permits a substantial standardization of models,
- includes a general real time executive characterized by a modular and flexible structure allowing the following simulation functions:
  - start/stop
  - freeze/restart
  - snapshot
  - simulation speed change
  - CPU time statistics

The costruction of a simulator model is made by means of topological data which define the physical connections of the LEGOCAD subsystems. The general real-time executive can accept also "non LEGOCAD subsystems"; the user can include, in his simulator, dynamic models already developed outside the LEGOCAD system.

In addition LEGOCAD has a powerful and easy to use graphic feature: LEGOGRAF. This tool, based on EASE+, allows the user to:
- link the plant mimics to the models,
- change the input variable values,
- manage all simulation conditions.

In conclusion LEGOCAD provides excellent answer to the needs that nowadays are expressed by the plant manufactures and the utilities for:
- performing dynamic simulations for the mechanical and control design as well as for setting up the operating procedures,
- real time simulators for:
  - full scope training simulators,
  - engineering simulators,
  - plant analyser
  - real-time models for the automation equipment testing.

PLANT DESCRIPTION

The Brindisi Sud Station now under construction in Puglia - on Italy's south east coast - consists of four twin-units.
A schematic of the plant, that is composed of a condensate and feedwater system, a boiler and steam generation system, and a turbine and power generation system, is shown in fig. 3.1.
The condensate and feedwater system contains the condenser, three condensate pumps, two parallel strings of two low pressure heaters in the condenser neck, water filters, a deaerator, two motor driven and one turbine driven boiler feedpumps and two parallel strings of three high pressure heaters.
Steam exiting the low pressure turbine is condensed in the condenser shell and then the condensate is recycled, by the condensate pumps, to the low pressure heaters, and the deaerator.
Thereafter the main feedwater is pumped, through the high pressure heaters, to the boiler by the feedpumps.
Feedwater flow control for the feedpumps is by speed control.
Each unit rated for 660 MWe has supercritical once-through, UP type, boiler (Fig. 3.2). The boiler is multifuel fossil-fired, designed for firing front and opposed rear burners in the furnace.
The fuel to the 56 burners, located on four planes of the