Designing for High Performance of a Command, Control, Communication & Information Network

Jason Rupe
Texas A&M University, College Station
Srinagesh Gavirneni
Carnegie Mellon University, Pittsburgh

Key Words — **Command & control; Communication & information network; Experimental design; Taguchi; Outlier; Stopping condition**

Summary & Conclusions — The design of effective communication & information networks (CIN) is important to the supported command & control (C^2) structure. The Command, Control, Communication & Information Network Analysis Tool (C3INAT) allows experts in different fields to develop sub-models independently, thus allowing analysts to compare various communication networks under the same C^2 structure. Statistical tools have been applied to C3INAT to reduce the simulation effort and to direct the study toward a near optimal solution through the application of Taguchi's method (experimental design), outlier tests, and stopping conditions. This paper outlines the applications of statistics to the evaluation of CIN.

The application of Taguchi's method has reduced the necessary simulation time immensely while the structure of the method has aided in finding near-optimal conditions faster than a full-factorial analysis. Structuring the experiments before simulation allows future versions of C3INAT to incorporate automatically many of these tasks.

The variance reduction technique and stopping condition place a lot of importance on the quantity ϵ. The algorithm requires minimal computer time & memory, and it can be used in conjunction with other variance reduction techniques. The algorithm, separate from the variance limit, can be applied when the variance limit is substituted with another. The technique can also be applied to post simulation analysis where the data are intact.

1. INTRODUCTION

Acronyms [1]

C^2	command & control
CDB	communication database
CIN	communication & information network
C^3I	command, control, communication & information
C3INAT	C^3I network analysis tool
MOP	measure of performance
MOE	measure of effectiveness.

Members of a given organization, in executing C^2 requirements, continually access the supporting CIN through telephones, fax machines, *etc*. Therefore, the effectiveness of an organization is tightly linked to the efficiency of the supporting communication network, and the whole C^3I system. Because communication resources are expensive, their efficient allocation is important to an organization's health. The problem of constructing & using a CIN is complicated and has not been adequately addressed.

The C3INAT addresses these problems with two unique features:

- Use of established databases, such as the CDB [6]. This feature allows for valid modeling.
- Separation of C^2 and CIN functions, through the construction of a structured interface [11]. This feature simplifies modeling, and allows for efficient analysis.

Through separation of the C^2 and CIN functions, C3INAT has been constructed to compare various CIN designs under controlled C^2 environments. Statistics has been applied to this tool in two forms: 1) Taguchi's method of experimental design, and 2) variance reduction and stopping conditions.

The simulations are specified by a dual-orthogonal design array which separates the controllable parameters from those that are not controlled (noise) into separate orthogonal arrays. The analyst then simulates every specified combination of the experimental conditions. During a simulation run, the collected data are checked for user-specified stopping conditions and for outliers which are then eliminated from the statistical calculations. The result of the limited simulation is an accurate MOP.

Notation

n	number of observations
$x_{(n)}$	order statistic n of batch means, $n = 1,2,\ldots,m$
$\bar{x}$, s	sample [mean, standard deviation] of a given MOP
ChkVar	value that the variance is checked against
NewVar	new variance
ΔVar	tolerance limit for the variance, $\Delta\text{Var} > 0$
ϵ	analyst-specified function of the s-confidence in the MOP
$T_{\min}$	minimum simulation run length.

Other, standard notation is given in "Information for Readers & Authors" at the rear of each issue.

2. BACKGROUND

The task of designing available communication networks has been important, yet difficult for many reasons, one of which is the large, diverse expertise requirements [1]. There have been

[1] The singular & plural of an acronym are always spelled the same.

numerous approaches to the problem of modeling the involved sub-systems. For example, in [14] a mathematical routing model is used to measure the reliability of the communication system. A different model of this sub-system is presented in [7], where a queueing theory approach is briefly explored and a system-dynamics approach is taken in the end. In [5], a different layered structure is used to model a C^2 sub-system; the various possible tasks are categorized to clarify the duties of the sub-system components. The C^2 sub-system is modeled by [10] with a series of algorithms, and rules for which algorithm applies. The human thought & response process is modeled directly as a C^2 sub-model by [9].

Most of these papers have the common weakness that they do not fully address the data collection & use problems. Although simulation is a common tool to model the CIN sub-system, analysts seldom agree on how to model the C^2 sub-system. C3INAT has the advantage of addressing the data collection & use problems, and uses existing databases to emulate the C^2 sub-system behavior.

This study uses experimental design to reduce the number of simulations required while yielding a CIN design robust to uncontrollable factors. The application of Taguchi's method to simulation is relatively new, and it has not been applied to a large network simulation tool like C3INAT. However, Taguchi's method of experimental design does have a good track record; a few examples of this are in [3]. The application of Taguchi's method involves steps that are elaborated upon in [3] and briefly outlined here.

- Identify the possible settings of the design & noise parameters, and the relevant ranges of values to be considered.
- Select orthogonal arrays for the design & noise matrices independently, and plan the experiments, according to desired resolution of effects (main & interactions).
- Conduct the parameter design experiments and evaluate the statistics that measure the system effectiveness from each test run.
- Predict optimal settings from the values of the statistics.
- Verify the predicted improvement. ◄

A design matrix is a subset of all possible combinations of noise factors and design parameters, limited only to those setting combinations actually being experimented. Where a full factorial design would require experimentation under every possible combination of parameter settings, a fractional factorial, such as used in this study, considerably reduces the number of experiments needed. According to Taguchi's method, the analyst must divide the controllable parameters from those that are not usually controllable (noise factors). Then, appropriate factorial designs are picked for each sub-set, and the indicated combinations of design parameters and noise factors are experimented.

The goal of this approach is to use the controllable parameters to design a system robust to the uncontrollable noise. Thus, parameters which have a profound effect on the variation of the MOE but little effect on the MOE itself are set at levels that minimize this variation; the rest of the parameters are set to optimize the MOE. Taguchi recommends that a signal-to-noise ratio be developed from a loss function that is specific to the MOE studied and used to decide the optimal parameter levels. A few of the more commonly used signal-to-noise ratios are developed in [12].

During a simulation run, occasional exceptional observations can cause the sample variance to increase sharply. Because the variance of the observations greatly impacts the simulation time, it is advantageous to filter out such exceptional observations. Through the use of the Central Limit Theorem, batch means can be taken such that they follow an approximate s-normal distribution, thus an outlier test for a s-normal distribution can be applied to the batch means to detect an observation of low probability. Two tests for detecting outliers in a s-normal distribution with unknown parameters, as suggested by Barnett & Lewis [2] are:

$$T_1 = [x_{(n)} - \bar{x}]/s, \tag{1}$$

$$T_2 = [x_{(n)} - x_{(n-1)}]/[x_{(n)} - x_{(1)}]. \tag{2}$$

At observation n, if the test statistic calculated is greater than a critical value, $x_{(n)}$ is dropped from the set of observations used to calculate statistics[2]. This process can be quite cumbersome because these test statistics are defined for every value of n.

The study here reduces simulation time through outlier tests and stopping conditions. The outlined outlier test speeds up the simulation by reducing the estimated variance, and thus the stopping conditions are met faster. Stopping conditions are used to ensure against data loss due to insufficient run lengths, or time wasted on excessive run lengths.

3. INTRODUCTION TO C3INAT

C3INAT is a collection of rules, instructions, and software that was created to allow the many necessarily involved experts to work independently toward comparing alternative CIN designs. Through this tool, MOP lead to a MOE for each network design through which an analyst can compare the network designs. This is one major advantage that the C3INAT has over many other models.

Another important feature of the C3INAT is the separation of C^2 & CIN functions; figure 1 shows the basic model that reflects this separation [8, 13]. This basic model contains 2 major layers and 1 interface, but further layering is neither precluded nor required; the design of C3INAT allows analysts to model all parts as necessary.

---

[2]Dropping an "outlying" observation from the statistical calculations does not mean that the observation should be completely ignored. That observation might provide valuable information about human or machine error.

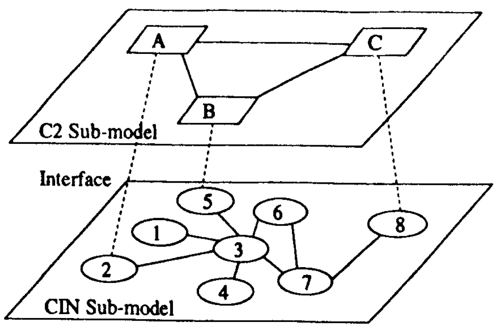

Figure 1. The 2-Layer Model

The C^2 sub-model embodies a description of C^3I requirements for a set of participants (people or information-storage-devices). In C3INAT, a needline defines a specific communication requirement between two or more specific parties under a given set of conditions. The needline information used by C3INAT, derived from the CDB, is a complete list of needlines each specifying information such as: a given commander requiring communication with a specified officer a certain number of times a day in the voice or data mode, in order to convey a specific amount of data. These needlines then are used in simulation to generate call attempts — the action of trying to communicate.

The performance of the CIN can be measured by how well the C^2 requirements are satisfied. The MOE can be determined from this alone, though other MOP can be collected and used, if desired.

The CIN sub-model is constructed by a telecommunication & computer expert so that the results accurately reflect the performance qualities of the CIN to be evaluated. Although a CIN is traditionally composed of nodes connected by links with clusters at the ends, there are no restrictions to the construction of a network. Experts can design CIN completely as they see fit: create dynamic or static networks, include or exclude message relay capabilities, and even include future technology as it is developed.

The interface represents the devices that connect the user to the network (C^2 to CIN). This interface can be as simple as a phone list with one-to-one association between parties and devices, or as complicated as a list of participants associated with devices on given clusters; both extremes have been created. If participants have exclusive access to certain devices, then the interface can be simple. If, however, some participants have overlapping access to devices, the interface must be more complicated.

The C3INAT design allows C^2 experts to work independently of CIN experts by defining the interaction required as a set of rules dictated in the interface. Therefore, the C^2 sub-model may communicate through any CIN sub-model under common interface conditions. The result is a flexible modeling architecture that facilitates experimental design. C3INAT has 4 phases.

- Reference preparation
- Call script generation
- Simulation
- Statistical analysis. ◄

3.1 Reference Preparation

This phase defines the scenario to be tested by identifying & formatting all of the information used in the other phases. To this end, four things must be accomplished:

- Produce a trooplist by identifying & listing the scenario participants.
- Extract the relevant needlines from the CDB.
- Synthesize one or more event scripts on the basis of the scenario information and analysis requirements.
- Develop an interface between C^2 & CIN. ◄

3.2 Call-Script Generation

A call script is a time-ordered list of call attempts to the network. The call-script generation program is a set of computer software files created to generate numerous call scripts through access to the trooplist, needline, and event-script files. Numerous Standard Call Scripts are created to meet the requirements of stochastic simulation and experimental design. The Call Script GENeration program allows an analyst to generate call scripts of any length; therefore, both terminating and steady state experiments can be carried out.

3.3 Simulation

The simulation model has 3 main parts:

- C^2 sub-model,
- CIN sub-model,
- C^2-CIN interface.

This design allows many different communication networks to be simulated in any combination with many different C^2 conditions, and allows experts to work independently under the guidelines of C3INAT.

The C^2 sub-model embodies the communication needs of the parties involved, and their reactions to the limited information usually available to a party (without knowledge of the CIN design being used). Because a particular communication-link begins & ends with participants, the states they are in determine the reactions to the communication requirements, and to the CIN responses.

The CIN sub-model contains all of the communication network information:

- the devices composing the network (links, switches, trunks, *etc*),
- their behavior,
- their arrangement.

The response given to the C^2 sub-model by the CIN sub-model is determined by the availability of a path through the resources of the CIN, searched for by the program.

The interface simply forms the connection between the C^2 sub-model and the CIN sub-model. It passes requests & responses between the sub-models according to the interface language developed, and matches devices with parties when necessary.

A simulation session may consist of several simulation runs with the same call script, but with different CIN conditions, or sub-models. The steps in a simulation session are:

- Session initialization — The simulation reads in the trooplist and the interface information.
- Run initialization — The analyst specifies the CIN sub-model to be used in that particular run.
- Simulation — The calls are read-in individually and loaded onto the pending calls list. All changes to all call-attempts are recorded in an attempt record file so that detailed statistics can be calculated later.
- Statistics summary — Summary statistics for each run are also calculated inside the simulation. ◄

Figure 2 illustrates the structure of the C^2 sub-model with the CIN sub-model, interface, and peripherals [13].

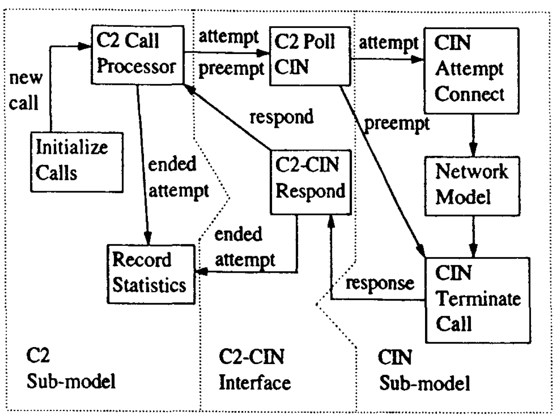

Figure 2. The Simulation Model

A call attempt moves through the simulation sub-models in a recursive "U" pattern in a way that all call attempts go to the CIN through one common interface module, and return through a different common interface module. This design clarifies the construction of the interface. Because the interface has the duty of filtering & changing the CIN responses into a defined status code for the C^2 sub-model and changes C^2 actions into requests to the CIN, construction clarity is important to avoid confusion in modeling.

The main purpose of the C^2 sub-model is to emulate human behavior during a communication attempt. As a call attempt enters this sub-model, it is given an initial status code that changes as the call is processed through the simulation. The next status code that a call attempt receives depends on the C^2 conditions, the CIN response, and the status of the call attempt at that instant. The status codes represent unique call states at various simulation times. Within the simulation, the synthetic human response and the network response causes the call to change states numerous times (transitory) until finally reaching an ending (terminating) state.

A simple CIN (the TN1 sub-model) was constructed for testing purposes yet contains many typical network qualities. The TN1 sub-model:

- Delays responses by a random amount of time
- Creates random failures
- Preempts resources when necessary
- Possesses a typical construction with typical limitations. ◄

The TN1 sub-model, illustrated in figure 3, is constructed of two clusters of devices each connected to one of two switches that are in turn connected by trunks. Arbitrarily, the devices are divided according to even-odd phone number criteria. The number of trunks, and whether or not to enable network failures and preempts are all choices given to the analyst at run time.

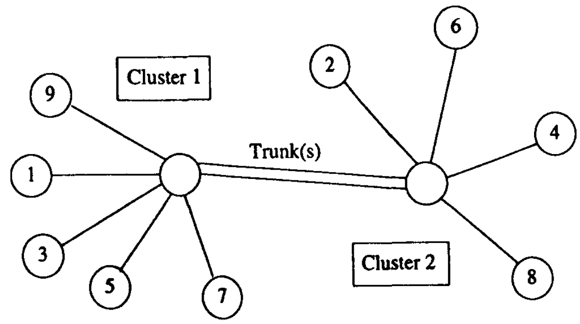

Figure 3. The TN1 Sub-Model

The interface used with the TN1 sub-model is a very simple device-to-*party* relationship such as a phone book. The *party* in this instance can be defined as a person or group of people, without any affect on the simulation model. Because of this simple interface, the availability of a participant indicates the availability of the device.

3.4 Statistical Analysis

Many attempts to analyze a C^3IN system through simulation do not include analysis procedures, perhaps because in some cases the design has no particular unique trait that enhances statistical analysis. The layered structure of C3INAT allows easy separation of design parameters and noise factors in the simulation, thus facilitating the application of experimental design. This is an important advantage of our design.

Two main assumptions were made to facilitate statistical analysis:

- Design parameters and noise factors locally have nearly linear effects on the MOP.
- Batch means (averages) for many MOP are s-normally distributed, according to the Central Limit Theorem. ◄

Statistics have been applied to both ends of the simulation. The remainder of this paper discusses the front end (experimental design) and then the back end (when to stop the simulation). However, before either can be discussed, the MOP must be selected.

4. MEASURE OF PERFORMANCE (MOP)

Simulation of communication networks is a useful technique for estimating network performance. For many systems, especially large scale communication networks, effectiveness is measured by a combination of numerous statistics because no one MOP can unequivocally categorize the quality of such a system. Therefore, several statistics serving as MOP must be recognized, and those that together contribute to measuring the system effectiveness must be isolated. Of the most obvious MOP, some heavily depend upon others. Further statistical analysis of these MOP results in a reduced list:

- ratio of time in the system to call length,
- percentage of parties busy,
- mean length of the queues,
- time interval between deletion of two calls,
- ratio of completed calls to calls attempted.

A function of these MOP can be used as the MOE for the networks simulated. The large, complicated issue of gaining MOE from MOP is beyond the scope of this paper.

5. EXPERIMENTAL DESIGN

C3INAT is constructed to afford the analyst complete control over many aspects of the simulation, and thus easy application of experimental design. Each parameter has many levels from which to choose settings, and some combinations of these yield a nearly optimal, robust design. Experimental design allows the analyst to approach this optimal, robust design with the fewest experiments, thus nearing an optimal setting faster than otherwise possible. The number of experiments is reduced by investigating only two or three settings for each parameter and by reducing the possible full-factorial number of combinations to a reduced set of these, such as a half-factorial design.

Although we have control over every parameter in a simulation, these parameters are not all controllable in the real system being modeled. Using Taguchi's method, the controllable parameters and uncontrollable parameters (noise factors) are divided and applied separately to their own orthogonal arrays. The controllable parameters and noise factors for our application are listed in table 1 along with the type of parameter it is, and the column assigned to each 8-level orthogonal array (OA).

The arrays work well for the very simple TN1 model developed for testing purposes, but larger CIN models require slightly different matrices, especially different in levels tested.

TABLE 1
Parameters and the Array Columns in Which They Reside

System Input	Type	OA Column
Design Array		
Number of trunks	Parameter	1
Number of retries	Parameter	2
Trunks and retries interaction	Interaction	3
Mean time between retries	Parameter	4
Trunks and retry time interaction	Interaction	5
Retries and retry time interaction	Interaction	6
Node configuration	Parameter	7
Noise Array		
Inter-cluster call length	External	1
Intra-cluster call length	External	2
Inter-cluster call frequency	External	3
Intra-cluster call frequency	External	4
Deadline	External	5
Intermittent network failures	Internal	6
error	–	7

The design array contains parameters that can be expanded when necessary. In larger systems requiring more detailed analysis, if the number of trunks between nodes is to vary between different cluster pairs, then each cluster pair linked by trunks might need to be represented by a separate design parameter in the array. If the number & frequency of retries is known to be a function of call priority or another known parameter, then the array might need to be changed also.

Inter-cluster and intra-cluster call length and frequency are represented by four noise factors in the noise array. These might be indirectly controllable in some C^3I systems over a typical time period. However, exceptional events can have unpredictable, devastating effects on the supporting CIN. A good CIN design is robust to these changes. If larger systems are to be studied with localized changes in these four factors, then more factors might need to be introduced into a larger noise array.

With the small design in table 1 for the TN1, the number of necessary experiments is $2^3 \cdot 2^3 = 64$. Although this is still a large number, it is considerably less than the $2^{10} = 1024$ possible combinations of a full factorial design yet obtains more information than a random set of 64 combinations would. Clearly, this savings in experimentation time alone justifies the application of experimental design to this analysis.

Once the selected experiments have been run, the data must be analyzed. The statistical applications yet to be discussed would have been applied during simulation to yield MOP that are used to gain MOE for each configuration simulated. A traditional Taguchi approach can then be used to search for a near-optimal solution, or more statistically sound comparisons can be made, based on the complete simulation output.

Most of the MOP noted earlier have lower variance near their optimal values by the way they are defined. For example, as the portion of completed calls approaches unity, the variation must approach zero. Therefore, instead of using Taguchi's 2-phase approach of first reducing variation and then optimizing the MOP, both phases can be addressed together in these instances.

For MOP that lack this trait, it is important to develop the loss function and signal-to-noise ratios, or refer to well known results when applicable. These results then must be used in the analysis of the data as discussed in the Introduction.

It is extremely important to simulate under the hypothesized-as-optimal conditions indicated by the results, because interactions that cannot be estimated in the fractional factorial design can have an unforeseen effect on the MOE values. If the results of this hypothesized-as-optimal simulation are not as anticipated by the fractional factorial analysis, then additional simulations might be needed to resolve the interaction effects in the results so that near optimal settings can be found.

6. OUTLIERS AND STOPPING CONDITIONS

Within the simulation phase, outliers are eliminated and stopping conditions are introduced. Correctly identified outliers can reduce the variation of the MOP and thus the related MOE. Similarly, correctly formulated stopping conditions can eliminate unnecessary simulation, and prevent the need to rerun an insufficiently long simulation. In a simulation run, as the number of collected batches increases, the analyst anticipates a drop in the observed variance of a MOP, according to assumption 2. This property is used to determine the stopping conditions for a simulation run. Usually, the stopping condition is defined as the maximum allowable ratio between the half-width of a *s*-confidence interval (taken as a function of variance) to the mean of the observations. Thus, the simulation is stopped if:

$$k \cdot s / \bar{x} < \epsilon. \tag{3}$$

The ϵ is specified by the analyst according to the desired *s*-confidence in the performance estimate derived from the data accepted by the outlier test. The *s*-confidence statements about the MOP, if desired, should be calculated from the complete data set.

A tolerance limit for the variance can be calculated from ϵ, as an upper limit to the ratio of the half-width of a *s*-confidence interval to the mean, with:

$$s^2 < [\epsilon \cdot \bar{x}/k]^2 = \Delta\text{Var} \tag{4}$$

The analyst has the assurance that once the variance reaches below a limit then it should not stray far above that limit for long, and that the variance lies between zero and ΔVar. This variance-tolerance can thus be used as an outlier test on batch means.

An algorithm that incorporates these tools has been applied to the C3INAT [4].

Algorithm

1. Input the minimum length of the simulation run, $T_{\min}$.
2. Input ϵ, the maximum allowed value of the applied ratio.
3. Run the simulation for $T_{\min}$.
4. Collect the observations.
   - · Calculate the mean & variance.
   - · Set ChkVar = variance.
   - · Check the stopping conditions. If they are met, then Stop.
5. Store the present values of mean, variance, sum, and sum of squares.
6. Collect the next observation and calculate the new mean and new variance (NewVar).
7. Test NewVar

   Case NewVar < ChkVar:
   - · Accept/store the observation
   - · Set ChkVar = NewVar
   - · Check the stopping conditions. If they are met, then Stop.

   EndCase

   Case NewVar < ChkVar + ΔVar: Accept/store the observation.

   EndCase

   Otherwise: Reject/ignore the new observation.

   EndTest
8. GoTo step 5.

End ◄

The algorithm is applied to every simulation specified by the experimental design. The MOP yield a MOE for each experiment. As specified, each experimented set of parameter levels is repeated under each different noise-factor combination. These 8 MOE are used in the traditional Taguchi 2-phase approach as discussed. However, approximate *s*-confidence bounds on MOP or parameter effects might be obtainable under certain assumptions. The most useful *s*-confidence intervals will bound the MOE of the optimal, robust solution obtained. Depending on the MOE created by the analyst, these might be difficult to obtain.

ACKNOWLEDGMENT

This work has been sponsored in part by the US Army Communications-Electronics Command at Ft. Monmouth, Contract #DAAB 07-89-M-B165, with professor Way Kuo as the principal investigator. We are pleased to acknowledge the contributions of professor Way Kuo to this work and this paper.

REFERENCES

[1] M. Athans, "Command and control (C2) theory: A challenge to control science", *IEEE Trans. Automatic Control*, vol AC-32, 1987 Apr, pp 286-293.

[2] V. Barnett, T. Lewis, *Outliers in Statistical Data*, 1978; John Wiley & Sons.

[3] K. Dehmad, *Quality Control, Robust Design, and the Taguchi Method*, 1989; AT&T Bell Labs, Wadsworth & Brooks/COLF.
[4] S. Gavirneni, "Variance reduction through detection and removal of outliers in C3INAT", *MS Thesis*, 1991; Iowa State University.
[5] J.E. Holmes, P.D. Morgan. "A layered approach to the representation of shipborne naval command systems", *Proc. 2nd Int'l Conf. Command, Control, Communications & Management Information Systems*, 1987 Apr, pp 26-30.
[6] K. Johnson, Y. Wang, J. Mullen, W. Kuo, "Database needs for modeling C3I system performance", *Proc. 1988 Command & Control Research Symp.*, 1988 Jun, pp 526-530.
[7] T.E.G. King, A.F. Martin, "The modeling of communications networks", *Proc. Int'l Conf. Advances in Command, Control & Communications Systems*, 1985 Apr, pp 90-98.
[8] W. Kuo, J. Mullen, J.W. Rupe, *et al*, "C3 information network analysis tool (C3INAT)" (final report), *Technical Report*, 1990; prepared for the US Army Communications-Electronics Command.
[9] D. Leedom, "Representing human thought and response in military conflict simulation models", *Selected Analytical Concepts in Command and Control*, 1982, pp 167-197; Gordon & Breach.
[10] P.D. Morgan, "Modeling the decision maker within a command system", *Proc. Int'l Conf. Advances in Command, Control & Communications Systems*, 1985 Apr, pp 65-69.
[11] J. Mullen, J.W. Rupe, S. Gavirneni, W. Kuo, "The command and control communications and information network analysis tool", *Proc. Winter Simulation Conf.*, 1990, pp 888-894.
[12] P.J. Ross, *Taguchi Techniques for Quality Engineering, Loss Function, Orthogonal Parameters and Tolerance Design*, 1988; McGraw-Hill.
[13] J.W. Rupe, "Application of statistical experiment design to the command control and communication information network analysis tool (C3INAT)", *MS Thesis*, 1991; Iowa State University.
[14] B. Sans, F. Soumis, M. Gendreau, "On the evaluation of telecommunications network reliability using routing models", *IEEE Trans. Communications*, vol 39, 1991 Oct, pp 1494-1501.

AUTHORS

Jason W. Rupe; Industrial Eng'g Dept; Texas A&M Univ; College Station, Texas 77843 USA.

Jason Rupe is a PhD student at Texas A&M University. He received his BS (1989) and MS (1991) in Industrial Engineering from Iowa State University. He has worked on C^3I systems, reliability, and quality.

Srinagesh Gavirneni; Industrial Eng'g Dept; Carnegie Mellon Univ; Pittsburgh, Pennsylvania 15213 USA.

Srinagesh Gavirneni is a graduate student at CMU. He received his BS in Mechanical Engineering from India Institute of Technology, and his MS in Industrial Engineering from Iowa State University.

Manuscript received 1994 August 30.

IEEE Log Number 94-11677

◄TR►