Estimating Force Requirements for Crisis Response Operations

Analytical Methods for the Numerical Analysis of Stochastic Loss Systems with Rotational Service

Sebastian Schäfer and Reiner K. Huber

Institute for Applied Systems Science and Operations Research,
University of the Federal Armed Forces Munich, Germany

Abstract

Similar to emergency and rescue operations, Crisis Response Operations (CRO) are triggered by events which occur repeatedly at irregular intervals. However, in contrast to the former they typically last for extended time periods thus requiring regular rotation of the forces involved. Since both inter-arrival times and duration of CRO can be described in terms of probability functions, various stochastic simulation models have been developed to generate, through a series of simulation experiments, the data base required for estimating so-called mission stacks and force requirements for CRO. However, based on experiments using the simulation model STORM (Stochastic Requirements Model) it was found that a data base sufficient for force planning purposes requires an excessive number of experiments. Therefore, as the structure of STORM corresponds to what is known in telecommunication systems analysis as a *multiple service multiple resource loss network* (MSMR), albeit without rotational service, the authors extend the analytical methods used in telecommunication analysis and develop a technique for approximating blocking probabilities that account for different rotational service strategies.

1. Introduction

Historical evidence indicates that, similar to emergency and rescue operations, Crisis Response Operations (CRO) are triggered by events which occur repeatedly at irregular intervals. However, in contrast to the former they typically last for extended time periods thus requiring regular rotation of the forces involved. When analyzing the data on operations-other-than-war (OOTW) compiled by John Sherwood (1995) it was discovered that their inter arrival time as well as their duration can be described in terms of exponential probability distribution functions. Therefore, Cherry/Huber/Hodgson (1998) suggested that queuing theory models might be used to analyze CRO requirements. They illustrated the idea by modeling the process of allocating units to CRO in form of a simple M/M/n queuing system having the capacity of handling up to a given number of CRO of one type simultaneously. Subsequently, this initial prototype simulation model was enriched and became known as the Stochastic Requirements Model (STORM) which was used by Huber and Cherry (1998) to investigate the relation between readiness and unit rotation policy and by Huber and Schäfer (2002) for analyzing German Army structural bottlenecks in CRO similar to SFOR and KFOR.[1]

---

[1] It should be pointed out, however, that the first applications-oriented analysis applying queuing theory in support of force planning was done under the acronym SADE at the US Army's Concepts Analysis Agency (CAA) by Patrick DuBois (1998) who developed a simulation model for generating, based on historical data such as those compiled by Sherwood, so-called simultaneity stacks for small scale contingencies (SSC) as a basis for predicting likelihood of concurrent SSC operations in the period of 1998 and 2006. STORM can be considered an extension of the SADE

The study experience with STORM revealed, however, that an excessive number of simulation experiments was required to generate a data base sufficient for force planning purposes in particular with regard to accounting for readiness requirements and mission priorities. Therefore, the development of analytic approximation methods permitting straightforward computation of force requirements as a function of readiness or availability specifications became the focus of a research project reported in this paper. Since the quality of the analytic methods is assessed in terms of how well they approximate the results obtained from simulation experiments, a review of the essential characteristics of STORM is given first. A detailed description of the model is provided by Schäfer (2003).

2. The Stochastic Requirements Model (STORM)

STORM performs a discrete event simulation over a time period that is long enough so that a stationary phase is reached when the variables of interest assume stable values. The input data fall into two categories:

- the *force pool*, comprising different types ($j = 1,\dots m$) and numbers b_j of force modules from which CRO task forces are compiled, denoted by the vector $b = (b_1, \dots b_m)$;
- the set of different *mission types* ($i = 1,\dots n$) each specified by their expected *occurrence* per time unit (arrival rate λ_i), expected *duration* T_i or service rate μ_i, the types and numbers of *force modules required* $[a_{ij}]$ (with $0 \leq a_{ij} \leq b_j$), and the *rotation policy*. Both arrival rate and duration of missions are stochastic variables determined by specific probability distributions (e.g. exponential distribution) and their parameters.

Rotation is necessary because in most cases the duration of a mission will exceed the maximum acceptable deployment time of an individual force unit or module. In STORM the *rotation policy* is defined by the tuple (T_D, T_R). T_D specifies the maximum time of continuous deployment after which the forces have to be replaced, directly followed by the minimum recovery time T_R indicating the time after which the respective unit or module is available for the next deployment at the earliest. The resultant rotation process is illustrated by Figure 1.

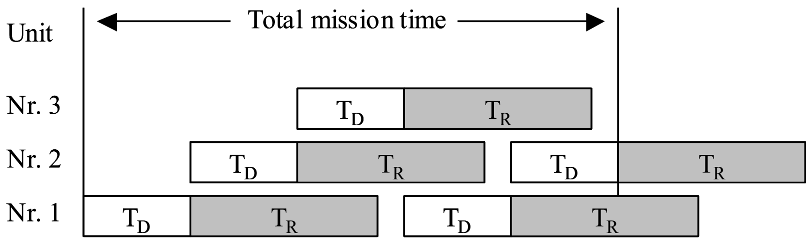

FIGURE 1. Servicing a requested mission by rotation

The rotation policy divides a mission into a series of consecutive *deployments*. Because of the recovery time a multiple of the forces deployed at any point in time must be assigned to each mission. To assure continuous service the rotation requires that each unit keeps a calendar of its future deployments (advance booking).

Since there may be more than one unit in the pool eligible for deployment, a *deployment strategy* is required that defines *which* unit is to be selected and *when* it should definitely be assigned to service the respective request. There are several strategies that may be considered such as the following:

simulation model since it does not only simulate the occurrence of different missions over time as SADE does. It also allocates the forces required for a mission from the force pool.

Random	Random selection from the pool of available units
Fixed	Check units always in the same order and select the first one available
Deploy	Like Fixed, but put the selected unit on top of the list thereafter; resulting in groups of units serving a particular mission
Cyclic	Like Fixed, but put the selected unit to the end of the list thereafter; resulting in a uniform rate of deployment (utilization) within the units of the same type
Pool	All units of one type are centrally managed, particular units are assigned right at the beginning of their deployment
Block	All units of a type are managed as block, i.e., at each point in time only the *sum* of free and deployed units is of interest.

The first four strategies book each particular unit right at the time of arrival of the mission, whereas the Pool strategy delays the definite assignment until the unit is actually deployed. In the Block strategy the question of booking a single unit does not apply at all since it does not consider units as individual entities. Serviceability only depends on whether the *sum* of units required for the particular time frames is available. Therefore the Block strategy neglects a major aspect of rotation and yields an upper bound for the number of requests that may be serviced.

Figure 2 shows the trajectories of the blocking probabilities for four different deployment strategies in a simple loss system obtained from discrete event simulations involving a sufficient number of experiments to reach stationary state in each case.

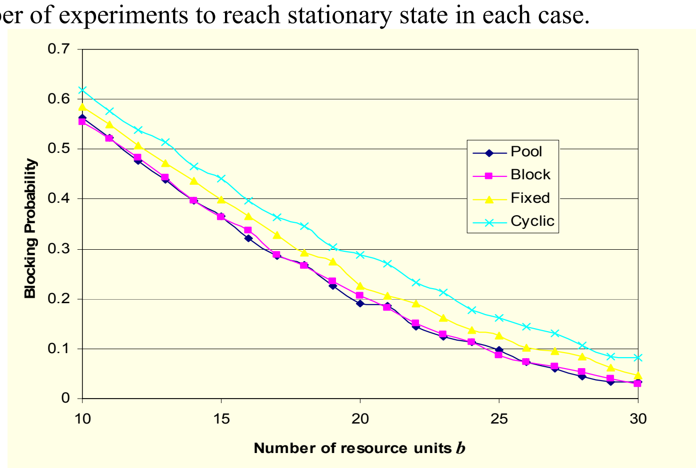

FIGURE 2. Trajectories of the blocking probabilities for four different deployment strategies in a scenario with one force type requested by one mission type with λ=5, μ^{-1}=4 and (T_D,T_R)=(0.5,0.6).

Early booking (Fixed and Cyclic) obviously results in more rejections due to the possibility of booking the “wrong” unit resulting in an sub optimal partition of the time frames. The Pool strategy yields results very similar to those of the Block strategy which yields the lower bound for the blocking probabilities (for a simple proof of this equivalence see Schäfer 2003).

3. Analytical Methods

As pointed out above STORM was designed to provide information on the responsiveness and robustness of a given force structure or pool (measured in terms of the probability that a set of specified missions can be served) and the impact of deployment policies. To generate data for addressing the question of a robust force pool, however, an excessive number of simulation experiments are required. This motivated the developments of analytical approximation methods as an alternative to the time consuming simulation.

Disregarding rotation, the process modeled by STORM essentially resembles a *multiple service multiple resource* (MSMR) loss network. Abstract models of this kind are quite well known in the analysis of telecommunication networks for which literature provides a rich collection of analytical solution methods (Kelly 1991; Ross 1995). Therefore, the question underlying the research reported in this paper was to find an analytical approach that permits to capture the process modeled by STORM by an MSMR including rotation.

3.1. Multidimensional Loss Models without rotation

A MSMR represents a multidimensional extension of the classical Erlang loss system. As in STORM it consists of $m \in N$ different *resources* each comprising a fixed number of units. The resources are requested by $n \in N$ different types of *services* arriving as independent Poisson streams. Depending on type, each service requires that a specified set of resource units is available for an exponentially distributed holding time. A service request is processed only if the associated resource requirements can be met at the very time of arrival. In that case, the service begins immediately with a simultaneous booking of the required resources. Otherwise the request is lost with no further impact to the system.

Assuming exponentially distributed arrival and service times it is well known from the theory of Markov processes that the systems behavior can be modeled as a birth death process which reaches a stationary equilibrium independent of the initial configuration. The current number of processed services of each type $x = (x_1,\dots,x_n)$ is sufficient for describing its state and the corresponding stationary distribution is given by the so-called product form solution

$$\pi(x) = \frac{\prod_{i=1}^{n} \frac{\rho_i^{x_i}}{x_i!}}{\sum_{y \in Z} \left(\prod_{i=1}^{n} \frac{\rho_i^{y_i}}{y_i!} \right)} \quad \forall x \in Z \tag{3.1}$$

where $\rho = (\rho_1 ,\dots,\rho_n)$, denotes the loads of each service calculated by $\rho_i = \lambda_i / \mu_i$ and $Z = \{x \,,\, | \, xA \le b\}$ the state space of the system determined by the demand matrix A and the resource vector b as defined above.

As a remarkable feature of equation (3.1), known as the *insensitivity property*, is that it holds for arbitrary independent service time distributions (Burmann/Lehoczsky/Lim 1984). It is easy to verify that for the *simple loss system* with $n=m=1$ equation (3.1) reduces to the classical Erlang formula.

The key figure of a loss system is the fraction of rejected requests, known as the *blocking probability* B_i of each service type $i =1,\dots,n$. In case of a simple loss system the blocking probability can be calculated by

$$B = Erl(\rho, b) = \pi(x = b) = \frac{\frac{\rho^b}{b!}}{\sum_{k=0}^{b} \frac{\rho^k}{k!}} \tag{3.2}$$

The multidimensional case requires that the bounding states F_i, defined as $F_i = \{x \,,\, | \, x \in Z \wedge (x_1,\dots,x_i+1,\dots,x_n) \notin Z\}$, are applied to differentiate between the different requirements of each service type resulting as

$$B_i = \sum_{x \in F_i} \pi(x) = \frac{\sum_{x \in F_i} \left(\prod_{i=1}^{n} \frac{\rho_i^{x_i}}{x_i!} \right)}{\sum_{x \in Z} \left(\prod_{i=1}^{n} \frac{\rho_i^{x_i}}{x_i!} \right)}. \tag{3.3}$$

However, the simple form of (3.1) or (3.3) notwithstanding, the calculation of the blocking probabilities requires a summation over the entire state space which grows exponentially with the number of service types and resources. A common alternative for large multi-service systems is known as the *Reduced Load Approximation* which permits approximating the blocking probabilities without having to explore the entire state space. If the blocking probabilities are known, however, most of the figures of interest for a loss system can be calculated.

3.2. Approximation Methods for MSMR with rotation

Due to the fixed values set by the rotation policy and the resulting change in the demand of resources when servicing a request, an exact analytical solution of the blocking probabilities is not possible for a loss system with rotation. Therefore, the basic idea presented in this paper is to map the system with rotation to a system without rotation that shows a very similar behavior and can be solved by the known methods.

3.2.1. Simple Loss Systems with Rotation

To derive a suitable approximation only simple loss systems, of one resource and one type of request ($m=n=1$), are considered at first. It is assumed that the one resource operates on the basis of the block strategy yielding a performance similar to the pool strategy. The duration of each request is assumed to be fixed which does not affect the blocking probability in a system without rotation due to insensitivity property of (3.1). Moreover assuming the duration of requests being a multiple of the maximum deployment time assures there are complete deployments only.

With these assumptions the timely resource requirements for servicing a single request can be portrayed in form of a *demand profile* as shown in Figure 3. Since it is impossible to account for the exact profile in an analytical model, the very idea of the approximation is to reduce the profile to a rectangle of equal area. Every rectangle in the diagram represents the properties of a request in a loss system without rotation. The length of the rectangle defines the duration and the height gives the number of units required. Since there is only one type of request in the system, a demand of more than one unit can be normalized to a simple system by dividing the total number of resource units by the height.

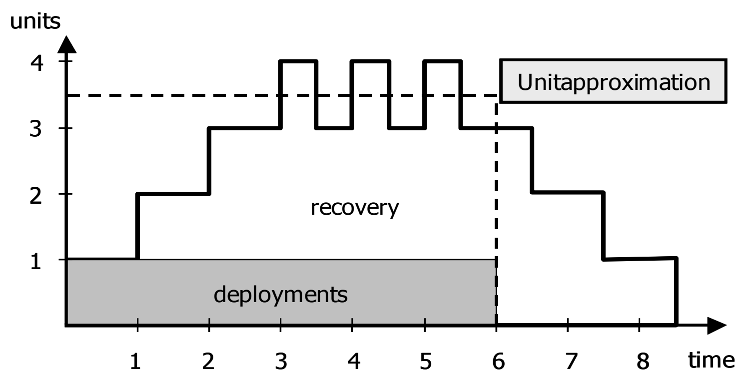

FIGURE 3. Demand profile while servicing a request with μ^{-1}=6 and (T_D,T_R)=(1,2.5).

The area under the curve of a demand profile results as product of the duration of the service and the effective demand per unit of time given by the *rotation factor* (*RF*) defined by

$$RF := 1 + \frac{T_R}{T_D} \cdot \tag{3.4}$$

It is easily verified from the Erlang formula (3.2) that two rectangles with the same area, but of different shape yield different blocking probabilities. Since the height determines the resource demand of a single request, greater height implies less flexibility and, therefore, more rejections. The lower bound for the blocking probabilities may be estimated by what is called the *load approximation* (LA) which implies fixing the height of the rectangle to one unit and adjusting the length appropriately. A (rather rough) upper bound (UB) can be estimated by determining the smallest rectangle that fully covers the profile.

Rather than fixing the height and adjusting the duration as in the LA, one may use the *unit approximation* (UA) which implies leaving the duration or load unchanged and adjusting the units involved as illustrated in Figure 3. In this case the blocking probability for the simple loss system can be calculated by the Erlang formula (3.2) as

$$B_{UA} = Erl(\,\rho,\, b/RF\,) = Erl(\rho,\, b * \frac{T_D}{T_R + T_D})$$

For non-integer values of *b/RF* the blocking probabilities may either be obtained by a linear interpolation of the probabilities resulting with the two neighboring integers or by the exact extension of the Erlang formula using the incomplete gamma function (see for example Jagermann 1974).

The main advantage of the unit approximation is that the solution is quite accurate because of two approximation errors running in opposite directions. On the one hand the rectangle smoothes the demand profile by cutting off the peaks causing an underestimation of the blocking probability. On the other hand it cuts the tail thus making it more compact which results in an overestimation. However, in most cases the first error will prevail, whereas the exact amount is difficult to quantify since it depends mainly on the particular shape of the demand profile as determined by the total number of deployments, the fractional part of the rotation factor and the average parallelism of services in the system.

Figure 4 shows the blocking probabilities resulting for a simple loss system from the approximation methods in comparison to the results obtained from simulation. It can be seen that the results of unit approximation corresponds best to the simulation results.

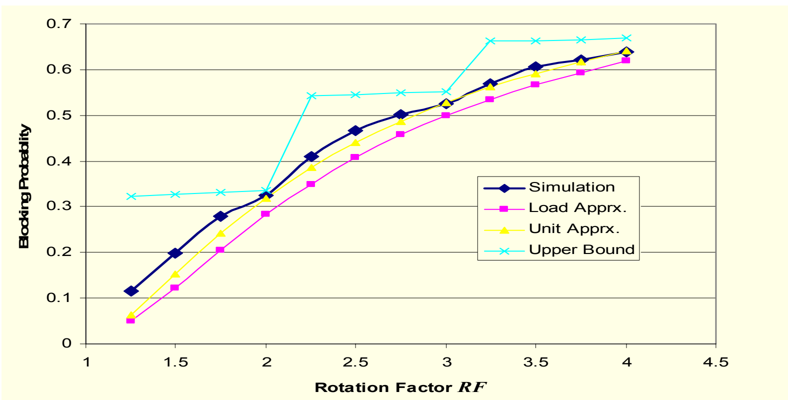

FIGURE 4. Trajectories of the blocking probabilities determined by simulation and the different approximation methods in a simple loss system with (λ,μ^{-1},b)=(4,4,25) and T_D=0.5.

3.2.2. MSMR Loss Systems with Rotation

It is assumed that in the loss systems with multiple services and multiple resources under consideration each service type $i=1,...,n$ may feature a specific rotation policy (T_{Di}, T_{Ri}).[2] In addition, as in the previous section it is assumed that all resources follow the Block deployment strategy and the duration of services is fixed.

The state space of a MSMR system is determined by the demand of each service type. Since the unit approximation is based on an adjustment of the demand, it will change the state space, but it also poses the problem of non-integer demand. Since an exact solution like the extension of the Erlang formula by the Gamma function is not known for multiple services, a pragmatic and easy to handle solution is proposed: scaling of the whole system (resources and demand) to account for a given precision.

In a MSMR system requests with different rotation policies compete for the same resource and the unit approximation presumes a different demand right from the start. But according to the rotation process two requests with similar maximum deployment times and different recovery times have the same demand profile in the beginning. They only differ in the maximum height reached eventually, whereas the probability that an additional service has terminated in the meantime rises nearly exponentially (see Schäfer 2003).

Thus, instead of individual demand adjustments an average recovery time T^*_{Rj} is calculated in an multiple service environment for each resource $j=1,...,m$,

$$T^*_{Rj} = \frac{\sum_{i \in M_j} \rho_i * \frac{T_{Ri}}{T_{Di}}}{\sum_{i \in M_j} \rho_i}, \tag{3.5}$$

where M_j denotes the set of services using resource j. With the adjusted demand vectors, obtained by

$$a^*_{ij} = a_{ij} * \left(1 + T^*_{Rj}\right), \tag{3.6}$$

the new state space is determined for calculating the blocking probabilities using (3.3). The blocking probabilities are the basis for all performance indicators of interest in a multidimensional loss system. For the respective formulas see Schäfer (2003).

Table 1 presents some examples of the blocking probabilities in a MSMR system with resource vector b=(60,60,80) and three types of services with different rotation policies. It will be noticed that the unit approximation (UA) works best if all service types have similar blocking probabilities and if the differences in the maximum deployment times T_D are not too large. The former is due to the fact that the adjustment of the minimum recovery time by (3.5) and (3.6) does not account for the actual throughput of the service types. Thus large differences between the blocking probabilities can cause errors in the approximation. As an easy solution it might be appropriate to apply the method twice and include the results from the first step in the second calculation of (3.5).

Deviations from the simulation results may also occur if the number of missions carried out in parallel is small. This is because in systems with a small ratio of resource units to demand a modification of one resource by a single unit can cause significant changes in the blocking probabilities of all services. This affects especially the results of the unit approximation which entails the simplification to a homogenous demand and a significant change in the state space.

Different maximum deployment times have very different demand profiles which can cause effects that are very difficult to capture in an analytical approximation. Significant deviations

[2] An alternative model with resource side rotation policies is discussed in Schäfer (2003).

from the simulation results might occur. Thus, for large differences in the T_D the use of the unit approximation reaches it limits. However, as of now no better solution has been found to estimate systems behavior under these conditions.

TABLE 1. Blocking probabilities in a MSMR system determined by simulation (S), unit approximation without (UA) and with (UA+) modification (3.6)

Nr.	λ, μ^{-1}	a			T_D, T_R	B^S	B^{UA}	B^{UA+}
	4, 7	1	1	0	0.5,0.8	.622	.532	.599
	1, 7	1	0	1	0.5,1.2	.126	.260	.118
	2, 7	0	1	1	0.5,1.8	.562	.658	.567
					total_rejrate	.533	.529	.521
	1, 7	1	1	0	0.5, 1.6	.376	.416	.377
	2, 8	1	0	1	0.8, 1.2	.222	.194	.236
	3, 6	0	1	1	1.2, 1.5	.211	.185	.220
					total_rejrate	.243	.227	.252
	1, 7	2	0	0	0.5, 1.2	.275	.260	.316
	1, 8	1	0	3	0.8, 2.2	.590	.650	.602
	2, 6	0	2	2	1.2, 1.5	.393	.301	.368
					total_rejrate	.414	.382	.414

3.4. Application to Optimization

The unit approximation provides the basis for a time-efficient resource optimization of MSMR loss systems with rotation. Given a set of service types and an objective function such as, e.g., the weighted sum of blocking probabilities, the aim is to determine the most cost effective resource pool meeting the requirements. Most objective functions (like total service rate, throughput, utilization etc.) can be expressed in a weighted sum of the blocking probabilities,

$$\text{Minimize} \quad \Sigma_i \, w_i * B_i \tag{3.7}$$

where only the constants w_i have to be adjusted appropriately.

However, even though the analytical methods do not provide a direct way of calculating the inverse function, they still reduce the calculation time for a scenario to a fraction of a second. To further reduce the total number of calculations, a simple *hill climbing* approach is used which yields results close to the optimum obtained by a total enumeration. Starting from an initial resource vector $b^0 = (b_1^0, \ldots, b_n^0)$ several approximations are performed identifying, in each step, the current *bottle neck* and increasing the respective resource by one unit. This iterative process ends when the desired values are achieved. Among the options available for the decision variable the *shadow prices* (see for example Kelly 1988) yielded the best results in most scenarios.

This very general approach works quite well for minimizing objective functions of form (3.7), more detailed objectives are covered in Schäfer (2003).

4. Illustrative Example

A simple scenario case is used to illustrate the application of STORM and the analytical methods described above. The underlying assumptions are presented in Table 2. The force pool is assumed to comprise three types of force modules/units of different numbers each. The scenario features two types of missions (MT) which occur randomly over time and demand deployment of different force contingents, or numbers of modules of each type, throughout the duration of the mission.

TABLE 2. Illustrative Scenario: Force Pool, Mission Characteristics and Deployment Requirements

	MT 1	MT 2	
Avg. Occurrence	3 years	5 years	
Avg. Duration	2 years	5 years	
Max. Deployment	0.5 years	0.5 years	
Min. Recovery	1 year	2 years	
Force Type	Total	MT 1	MT 2
Infantry Btl	20	2	1
Helicopter Sqd	18	2	2
Logistics Units	27	1	3

The results obtained with these data from simulation experiments (Sim.) and calculations with the unit approximation (Analyt.) are listed in Table 3. The numbers reveal that, due to the comparatively high timely demand for MT 2 in combined with a rotation factor of five, not more than one mission of type 2 can be processed at the same time as opposed to about two for mission type 1. This small parallelism for MT 2 explains the deviations in the analytical approximation for the service rate. For the unit deployments per year (*udpy*), however, analytical approximation yields very accurate results for each unit.

TABLE 3. Results for the Scenario

Service rate	Sim.	Analyt.
MT 1	80%	81%
MT 2	57%	47%
Total	71%	68%

udpy	Sim.	Analyt.
InfBtl	0.16	0.15
HelSqd	0.24	0.24
LogUnit	0.17	0.18

As an example for an "optimization" of the force pool the following objective is assumed: determination of the minimal force pool under the restriction that the total rejection rate for the scenario should not exceed 10%, or the service rate not fall below 90%. The new configuration of the force pool obtained from calculations using unit approximation and the hill climbing approach is shown in Table 4. As would be expected from the data in Table 2, the helicopter squadrons represent the bottleneck and, therefore, required the largest increase to meet the optimization criteria.

TABLE 4. Optimization of the Force Pool

Service rate	Sim.	Analyt.
MT 1	96%	96%
MT 2	83%	79%
Total	92%	91%

	Total	Difference
InfBtl	28	+8
HelSqd	34	+16
LogUnit	36	+9

The numbers in Table 4 show that the condition, that only the total rejection rate must not exceed 10%, is met by serving most missions of type 1 and fewer of type 2. Therefore the optimization was repeated for the restriction that the service rate of mission type 2 were not to fall below 90%. The respective results listed in Table 5 show a significant increase in the logistics units caused by their comparatively high demand in missions of type 2.

TABLE 5: Reducing the individual rejection rate of MT 2

Service rate	Sim.	Analyt.
MT 1	96%	96%
MT 2	93%	92%
Total	94%	93%

	Total	Difference
InfBtl	23	+3
HelSqd	36	+18
LogUnit	48	+21

5. Conclusions

In conclusion it should be pointed out that the simple examples presented in this paper were conceived only for the purpose of illustrating the basic functions of STORM and the use of analytical models for approximating simulation results in one prototypical scenario. However, in the context of realistic force planning, the entire spectrum of missions and scenarios must be considered that force planners may have to face in the future given the uncertainty of the security environment. To this end, it is proposed to define, on the basis of the analysis of historical mission and military experience, a set of representative generic mission types and mission-specific sets of both military and non-military capabilities required for servicing each mission . These capabilities are offered to varying degrees by the modules or units in the force pool. Via special matching methods appropriate force contingents can be derived as a basis for simulation experiments and the development of analytical solutions for efficient force pools.

REFERENCES

BURMANN, D.Y., LEHOCZKY, J.P. & Lim, Y. 1984. Insensitivity of blocking probabilities in a circuit switching network. *J. of Applied Probability*, **21**:850-859

CHERRY, Peter, HUBER, Reiner K. & HODGSON, Thom. 1998. Ground force requirements for post-cold war operations: A systems analysis approach. In Woodcock and Dave Davis, editors, Analysis for and of the Resolution of Conflict. The Canadian Peacekeeping Press, Clemensport, NS, 157-166

DUBOIS, Patrick L. 1998. Stochastic Analysis for US Army Deployments and Excursions (SADE). Paper presented at 15 ISMOR, Military College of Science, Shrivenham, UK

HUBER, Reiner K. & CHERRY, Peter 1998. On Restructuring Alliance Ground Forces and Adapting Deployment Policies for the New Mission Environment. In Woodcock and Dave Davis, editors, Analysis for Crisis Response and Societal Reconstruction. The Canadian Peacekeeping Press, Clemensport, NS, 226-241

HUBER, Reiner K. and SCHÄFER, Sebastian. 2002. STORM: Ein Modell zur Unterstützung fähigkeitsorientierter Streitkräfteplanung unter Unsicherheit unter besonderer Berücksichtigung des erweiterten Aufgabenspektrums. Bericht Nr. S-0201. Institut für Angewandte Systemforschung und Operations Research (IASFOR). Universität der Bundeswehr München

JAGERMAN., D. L. 1974. Some properties of the Erlang loss function. *The Bell Technical Journal*, **53(3):**525-551

KELLY, F. P. 1988 The optimization of queuing and loss networks. In O.J. Boxma and R. Syski, editors, Queuing Theory and its Applications. North Holland, Amsterdam

KELLY, F. P. 1991. Loss networks (special invited paper). *Annals of Appl. Prob*. **1(3):**319-378

ROSS, Keith W. 1995. Multiservice Loss Models for Broadband Telecommunication Networks. Springer, London

SCHÄFER, Sebastian. 2003. Analytische Methoden zur Kräftebedarfsplanung bei Bedienung stochastisch auftretender Aufträge im Rotationsprinzip. Dissertation. Der Andere Verlag, Osnabrück

SHERWOOD, John D.1995. US army operations other than war since 1989. Technical report, US Army Center of Military History