Performance assessment for
single echelon airport spare part
management

Annalisa Cesaro¹, Dario Pacciarelli¹

RT-DIA-161-2009 Novembre 2009

(1) Università di Roma Tre,
Via della Vasca Navale, 79
00146 Roma, Italy.

This work is partially supported by the research project FIRB “Advanced tracking system in intermodal freight transportation” of the Italian Ministry of Research, Grant number RBIP06BZW8.
ABSTRACT

In this study we compare approximation techniques for computing the operational availability of a practical corrective maintenance system in charge of the maintenance of 38 Italian Airports for varying the stock level and the demand. We focus on a single echelon one-for-one ordering policy with complete pooling, with a deterministic rule for lateral transshipments. A drawback of this policy is the state dependent nature of re-forwardings in the systems, which does not allow to express the state probabilities of the associated Markov chain model in product form. Therefore, computing the state probabilities is not practical as the number of states in the Markov chain increases. We adapt three approximation techniques to our model and evaluate their performance in terms of computational effort, memory requirement and error with respect to the exact value. Two techniques approximate state probabilities with others that can be expressed in product form, so that the Markov chain can be decomposed. Specifically, we adapt a method by Alfredsson and Verrijdt and the Equivalent Random Traffic method. The third technique is based on the multi-dimensional scaling down approach, which studies an equivalent reduced Markov chain rather than decomposing the original one.
1 Introduction

The ever increasing air traffic demand of passengers and cargos all over the world is currently limited by the capacity of the airports, which are expected to become a serious bottleneck for air traffic in a near future [1]. Facing such growth requires significant investments for developing existing airports and/or constructing new ones. Specifically, there is an increasing need for safety equipments in order to grant airport safety, as well as for supporting the correct execution of airport operations.

In this scenario airports face every day the challenging task of maintaining high standards of safety at a sustainable cost. When a failure of some equipment takes place, failed components must be promptly replenished with new spare parts, since safety standards are not compatible with long repairing times. As observed by several authors, see e.g. by [13], the logistics of spare parts differs from those of other materials in several ways. Equipments may have remarkable costs, long repairing times and sporadic failures. The latter are difficult to forecast and cause relevant financial effects, due to the economical and legal implications of a lack of safety of airport operations. These characteristics are particularly stringent in the airport context.

The sporadic nature of the failure process for a single equipment translates in most cases into very low demand for spare parts. For an item there are typically less than ten working equipments with MTBF equal to six years or more. Therefore, for economical reasons, airports are usually grouped on a regional base and served by a single warehouse. For example, the Italian territory is divided in 17 service regions serving a total of 38 civil airports. Each spare part warehouse manages the aggregated demand of all the airports encompassed in its region. It follows that the aggregated demand rate for some items may be low or high, depending on the number of working equipments in the area.

The spare parts supply chain may typically involve at least three actors: airport authorities, logistics companies and equipment suppliers. The latter are responsible for supplying new components and/or repaired items, which usually require long replenishment times. Intermediate logistic companies are in charge of replenishing spare parts in the short term, by granting minimum levels of operational availability (i.e., the fraction of time during which all working equipments are operative) regulated by contracts with airport authorities. We focus on the point of view of an intermediate logistics company facing the high cost of purchasing and managing spare parts at sustainable cost. Our study is motivated by the practical needs of an Italian logistics company supporting the activity of 38 civil airports spread over the Italian territory. The company handles 17 warehouses and manages the overall processes of purchasing, holding, ensuring that the overall reliability of safety equipments is always within contractual limits. The aim of the company is therefore to grant the prescribed quality of service at minimum cost. To this aim, a two echelon inventory policy without lateral transshipments is currently adopted and the level of stock and geographical allocation of spare parts is obtained with the VARIMETRIC algorithm of [24]. This policy is depicted in figure 1(left).

In Figure 1(right) a single echelon policy is depicted. As observed, e.g., by [18], single echelon models with complete pooling might be more effective for reducing both reaction time to stockouts and inventory levels. At the time of writing, lateral communication is only used by the company in emergency situations, when couriers or overnight carriers rapidly transfer parts to demand locations. However, lateral transshipment is not explicitly included
in the model when deciding the spares allocation among the 17 warehouses. Therefore, the company managers are interested in evaluating the potential benefits deriving from the adoption of a single echelon policy. To this aim, however, there is a need of effective models for assessing the performance of a single echelon replenishment policy. This is not an easy task for large instances even for steady state analysis. In fact, the state probabilities of the associated Markov chain model cannot be expressed in product form, which makes increasingly difficult computing them as the size of the instance increases, as in case of high rate demand. Moreover, the policy for lateral transshipment is deterministic and received little attention in the literature, so far. Among the few exceptions we cite the paper by Kukreja et al. [18].

In this paper, we adapt three approximation methods to compute the overall operational availability at the 38 airports for a given spare parts allocation and evaluate their computational effort, memory requirement and error with respect to the exact value, when available. The latter is computed by directly solving the multidimensional Markovian model, which can be done only for small and medium size instances.

The first two methods are based on decomposition approach. State probabilities are approximated with others that can be expressed in product form, so that the Markov chain can be decomposed and operational availability can be easily computed. Specifically, our first method, referred to in the following as the AV method, is a slight modification of a method by Alfredsson and Verrijdt [2].

The second method is based on ideas successfully used in the field of telecommunications, and specifically in the Interrupted Poisson Process method (IPP method) [20, 22] and the Equivalent Random Traffic method (ERT method) [15]. With the IPP and ERT methods part of the traffic may be lost when no server is available. In our adaptation of these methods we include the presence of an external supplier to avoid lost requests.

The third method is based on the multi-dimensional scaling down approach, which studies an equivalent reduced Markov chain rather than decomposing the original one. A scaling down approach is used by Axsater [4] to study a two-echelon policy. We adapt this method to study the single echelon policy with complete pooling in a Markovian framework without decomposing the original chain.

This paper is organized as follows. Section 2 reviews the literature most related with
this paper. Section 3 formally defines the inventory model studied in this paper. Section 4 presents the multi-dimensional Markovian model, used as benchmark for successive evaluations of approximate models. In Section 5 the two approximation approaches are described: the decomposition and the multi-dimensional scaling down approach. Section 6 presents the numerical comparisons among the models in terms of operational availability, computation time and memory requirements. Some conclusions follow in Section 7.

2 Literature review

The literature on spare part logistics with lateral transshipments is strictly related to the more general context of inventory management. Most contributions focus on the analysis of different inventory management models. Several authors [2, 11] demonstrated the benefits of inventory sharing flexibility provided by complete pooling policies. An extensive overview of the research concerning transshipment modeling in supply chain systems is given by [10]. Kennedy et al. review the different modeling issues in spare part management [17]. Here we limit ourselves to present the foremost works on techniques for assessing the performance of single-echelon systems, applicable in continuous review policies with complete pooling.

As observed, e.g. in [28, 17], at least two main streams of research can be distinguished for approaching the modeling tasks, namely the multi-dimensional Markovian approach and the decomposition approach. A third stream of research that can be cited is based on simulation.

With the multi-dimensional Markovian approach [27], the behavior of the inventory system is modeled with a Markov chain. Studying the Markov chain allows to compute structural properties as well as the state probabilities of the chain, which allows to evaluate the performance of the inventory system. This model becomes impractical for large instances, due to the extremely large number of states of the Markov chain. In order to overcome this drawback, one possibility is to study an equivalent Markov chain with a smaller number of states, even if this approach did not receive much attention in the literature. Axsater [4] suggests and evaluates a similar technique in a two echelon context with continuous review policy and lumpy demand at each warehouse. With the scaling down approach of Axsater, a high-demand system is approximated by a low-demand system. The real customer demand is scaled down such that the ratio between standard deviation and mean value is preserved. According to the author, the scaling down technique is quite effective to speed up the analysis of single queueing systems. However, to the best of our knowledge, little work has been done to assess the effectiveness of this technique in the context of multi-dimensional Markov chains.

A second stream of research is based on an approximate decomposition approach [2]. This approach consists of estimating the state probabilities of the Markov chain, rather than computing their exact values, by studying each queueing system independently from each other, so that the system performance can be easily computed. The basic idea is to adjust the demand flow, so that the lateral transshipments are taken into account. Within this stream of research, Alfredsson and Verrijdt [2] use an iterative method proposed by Axsater [3] in a two echelon context to compute fraction of demand satisfied by different sources by assuming exponentially distributed replenishment times and Poissonian demand. The AV method allows each warehouse to share inventory with every other warehouse, so that all warehouses act as a single big pool. Specifically, lateral transshipment is used when no spare
is locally available and the request is directed toward a randomly chosen closest neighbor warehouse with spares available. An external supplier manages the requests that cannot be filled by other local warehouses or by the central warehouse. The demand and the stock level may differ from one facility to another. The numerical experiments show that the approximate results are very close to the simulation results for low demand rate, while the error may increase remarkably with the demand rate.

The Interrupted Poisson Process (IPP) [20, 22] and the Equivalent Random Traffic (ERT) [15] methods are decomposition approaches used in particular in the design of telecommunication networks to assess the blocking probability of a network [14]. With these methods there are no external entities, such as the external supplier, and requests arriving, when all service centers are busy, are lost. Multiple re-forwardings are possible, by letting requests jumping among the service centers more than one time. With both methods a lateral transshipment from a warehouse (i.e., a re-forwarded request) is viewed as an overflow from the demand arriving at the warehouse, and therefore with variance larger than the mean value. The peakedness of a distribution is the ratio between variance and mean value, which is equal to one for the Poisson process and it is greater than one when dealing with the overflow process at a queue with Poissonian demand and exponential service time.

There are significant differences among the AV, ERT and IPP methods. Besides the presence/absence of the external supplier, a further difference is that AV method models the effective demand at each warehouse as a Poissonian flow, which is therefore described by the first moment of its distribution. The ERT method characterizes the effective demand just by its mean and variance assuming a peakedness greater than one, while the IPP method models the effective demand at each warehouse as hyperexponential, and takes into account its first three ordinary moments in computations.

As observed, e.g. by [9, 8], reliable models of service systems with overflow should take the peakedness into account. In order to explore the potential benefits of including peakedness in the model, we compare two methods, the first assuming Poissonian demand and the second assuming demand with peakedness greater than one.

A third, quite different, stream of research is based on simulation (see, e.g. [25, 19]). In this case the inventory system is modeled as a discrete event system, whose evolution allows to evaluate the original system behavior. Simulation models allows to easily incorporate all relevant practical details of the system, but particular care is necessary to guarantee the statistical relevance of the results achieved. On the other hand, they may require very long computational times in low demand contexts in order to achieve reliable results. Rare event techniques, such as the importance sampling policy [12], can be used in such cases to reduce the simulation times, but the technique can still remain very time consuming [7]. Therefore, such models are not further explored in this paper.

3 The model

The model addressed in this paper is a single item, single echelon, N-locations, continuous review, one-for-one replenishment policy inventory system, which allows for lateral transshipments with complete pooling, emergency transshipments from an external supplier and no negligible transfer times. A deterministic closest neighbor rule is used for lateral transshipment.
3.1 Main notation

In order to formally define the problem we need the following notation. Let \( A = \{1, 2, \ldots, a\} \) be the set of airports, \( W = \{1, 2, \ldots, w\} \) be the set of warehouses, \( s_h \) be the number of spare parts available at warehouse \( h \in W \), \( \mu \) be the service rate of a server at warehouse \( h \), \( T_{hi} \) be the transfer time for a spare from warehouse \( h \) to warehouse \( i \), taking into account also the time needed to issue an order, and \( T_s(j, h) \) be the substitution time, i.e., the time needed to transfer a spare part to the airport \( j \in A \) from the warehouse \( h \in W \) and to physically replace the failed item, and \( T_{0i} \) be the mean emergency replenishment time from the external supplier to warehouse \( i \), taking into account also the time needed to issue an order and the transfer time. Let \( \lambda_{jh} \) be the arrival rate of failure processes from airport \( j \) to warehouse \( h \), \( \lambda_h = \sum_{j \in A} \lambda_{jh} \) be the aggregated arrival rate of failure processes at warehouse \( h \) and \( \lambda'_h \) the effective arrival rate at warehouse \( h \) including transshipments. Let \( P_B \) the network blocking probability, i.e., the probability that a failure occurs at some airport and no warehouse can satisfy the spare demand. Let \( \pi(h, i) \) be the probability of the event: there are no spares in warehouse \( h \in W \) and \( i \in W \) is the closest warehouse with available spares. We also denote with \( MTBF \) the mean time between failures, with \( MCMT \) the mean corrective maintenance time and with \( OA \) the operational availability of all the \( a \) airports.

3.2 System processes and assumptions

When a failure occurs for some component at some airport \( j \), a demand for a new spare part is issued to the associated regional warehouse \( h \). If spare parts are locally available, the component is immediately replaced in the airport using the stock on hand at the local warehouse. Then, the failed component is sent to an external supplier, which can either repair or replace the component with a new item, so that warehouse \( h \) can restore the local stock level for that specific component, after a replenishment time. If no spare part is locally available, warehouse \( h \) forwards the request to the nearest warehouse \( i \) with available spares to satisfy the demand through a lateral transshipment. Then, warehouse \( i \) will issue a replenishment order to the external supplier to restore its stock level. If no spare is available in any warehouse the demand must be satisfied directly by the external supplier through an emergency transshipment, i.e., by using the first repaired/new component available at the supplier. In such a case we say that the warehouses network is blocked, since the failed equipment will not be working at airport \( j \) until after the substitution. Since the replenishment time from the supplier to a warehouse can range up to several months for expensive components, in order to guarantee the high operational availability required by contract with airport authorities, the blocking probability must be kept at a very low level.

In our model we use the Poisson distribution for the demand process, which is a typical assumption for modeling low demand processes [25]. It is worthwhile to mention that the MTBF of an equipment depends on its age and on other exogenous agents, such as the damp, the temperature and other operational conditions. Therefore, in our model we use specific values for each airport. The replenishment time of the external supplier is a random variable, exponentially distributed, with known mean value \( T_{0i} \) for \( i \in W \). By contract its mean value is the same for all warehouses and it is equal to the sum of the mean time to return (MTTR) and the order and ship time (OS). The capacity of the supplier repair shop is assumed to be infinite. It follows that also the number of replenishments from the external
supplier follows the Poissonian distribution. These common assumptions make possible to use the Markovian analysis for modeling the multi-dimensional inventory system. However, as shown by [18], the results obtained with these assumptions holds under less restrictive hypothesis on the replenishment distribution.

We may describe the system dynamics with a suitable queueing network. Each warehouse acts as a single queueing system without buffer, in which the number of servers equals the number of spares in the warehouse. The arrival process (i.e. demands for spare parts) is stochastic. The service time of each request equals the time needed to repair/replenish a spare part. Therefore, the number of busy servers corresponds to the number of outstanding orders of spare parts. In this way each warehouse is a G/M/s_i/0/∞ system. The operational availability OA is defined as in [24]:

\[
OA = \frac{MTBF}{MTBF + MCMT}
\]

This is the performance measure established by contract between the logistic company and the airport authorities. MCMT is the average amount of time an item is not available at an airport. As far as there are spares in the warehouse, this is the time needed to physically substitute the spare. If no spares are locally available, MCMT must also take into account the deterministic transfer time from the closest warehouse with available spares or, if no spare is available elsewhere, the replenishment time from the external supplier.

\[
MCMT = \sum_{h \in W} \sum_{j \in A} \lambda_{jh} \cdot T_s(j, h) + \sum_{h \in W} (\lambda_h \cdot \sum_{i \in W} \pi(h, i) \cdot T_{ih}) + \left(\sum_{h \in W} \lambda_h \cdot P_B \cdot T_0 h\right)
\]

We observe that the first term \(\sum_{h \in W} \sum_{j=1}^{A} \lambda_{jh} \cdot T_s(j, h)\) of Equation (2) only depends on the failure process and on the distance between the airports and their respective regional warehouses. In other words, it does not depend on the spare parts management policy. Moreover, this quantity is typically small with respect to the other terms of Equation (2), therefore for sake of simplicity we assume it negligible in our model and omit its computation in the rest of this paper.

4 Multi-dimensional Markovian approach

We model the system under study with a queueing network with blocking, and study it by using a Markov chain model, with a very similar approach to that of Wong et al. [27]. The main difference is that we explicitly include the external supplier in the Markov chain while in [27] a failure of a part occurring when all warehouses are in stockout condition is lost.

In the Markov chain, a state \(n = (n_1, \ldots, n_w, n_{w+1})\) is a vector, in which \(n_i\) is the number of outstanding requests at warehouse \(i \in W\), and \(n_{w+1}\) is the number of outstanding emergency transshipments issued from all warehouses to the external supplier. Note that the overall number of outstanding requests is \(\sum_{i=1,\ldots,w+1} n_i\). In case of blocked network, if \(n_{w+1} \geq 1\) the first repaired item returned by the external supplier is used for replacing a failed item at some operative site.

There are direct transitions among states just in case of a single arrival event (i.e., a request for a spare at some warehouse) or a single departure event (i.e., the replenishment
of a repaired item by the external supplier). Let \( e^i \) be a vector with \( w + 1 \) elements, all equal to 0 but the element in position \( i \) that is equal to 1, and let \( \psi(h, i) \) be equal to 1 if \( i \) is the warehouse closest to \( h \) with spares available, and be equal to 0 otherwise. More precisely, \( \psi(h, i) = 1 \) if \( n_i < s_i \) and \( n_i = s_i \) for each \( l \in W \) such that \( T_{ih} < T_{lh} \), included \( l = h \). With this notation, \( n + e_i \) is the state of the Markov chain representing an arrival at the \( i \)-th warehouse (with \( n_i < s_i \)), due either to a failure in the \( i \)-th service region or to a forwarded request from some other warehouse \( h \) in stockout conditions for which \( \psi(h, i) = 1 \).

Similarly, \( n - e_i \) is the state with a departure from the \( i \)-th warehouse (if \( n_i > 0 \)). For the external supplier, \( n + e_{w+1} \) represents a new emergency request (if \( n_i = s_i \) for each \( i \in W \)) and \( n - e_{w+1} \) represents the fulfillment of an emergency request (if \( n_{w+1} > 0 \)). The transition rate \( q(n, m) \) from state \( n \) towards state \( m = n \pm e_i \) and \( n \pm e_{w+1} \) is as follows.

- \( q(n, n + e_i) = \lambda_i + \sum_{h \in W - \{i\}} \psi(h, i) \cdot \lambda_h \), for \( i \in W \) and \( n_i = 0, 1, \ldots, s_i - 1 \);
- \( q(n, n + e_{w+1}) = \sum_{i \in W} \lambda_i \), if \( n_i = s_i \) \( \forall i \in W \);
- \( q(n, n - e_i) = n_i \cdot \mu \), for \( i \in W \) and \( n_i > 0 \) and \( n_{w+1} = 0 \);
- \( q(n, n - e_{w+1}) = \sum_{i=1}^{w+1} n_i \cdot \mu \), for \( n_i = s_i \) \( \forall i \in W \) and \( n_{w+1} \geq 1 \).

Figure 2(left) shows an example of a Markov chain for two warehouses, the first having two spares and the second having three available spares. Theorem 1 shows that the blocking probability \( P_B \) of the Markov chain can be easily computed. Let \( S = \sum_{i \in W} s_i \) be the total stock level in the network, and let \( \rho = \frac{\sum_{i \in W} \lambda_i}{\mu} \).

**Figure 2**: A Markov chain (left) and the aggregated birth death model (right)

**Theorem 1** Given a set \( W \) of warehouses, with total stock level \( S \), in which the service process is exponentially distributed with average rate \( \mu \) for each server and the demand flow to warehouse \( i \in W \) is Poissonian with average rate \( \lambda_i \), the blocking probability of all warehouses is \( P_B = 1 - \sum_{k=0}^{S-1} \frac{\rho^k}{k!} e^{-\rho} \).
Proof. Let us consider a cut in the Markov chain grouping all the states $n$ such that $\sum_{h=1}^{w+1} n_h = k$ (in Figure 2(left) is highlighted the case for $k = 2$). For each value $k = 0, 1, \ldots, \infty$ the state aggregation property described in [23, 16] applies, and the states contained in each cut can be replaced with an aggregated state $k$. The demand rate for each aggregated state $k = 1, \ldots, \infty$ is $\sum_{i \in W} \lambda_i$ and the service rate is $k\mu$, as in figure 2(b). The network is therefore equivalent to a virtual single warehouse with combined stock level $S = \sum_{i \in W} s_i$, demand rate $\sum_{i \in W} \lambda_i$ and service rate for each server $\mu$. The overall stockout probability $P_B$ is the probability that the total number of requests is greater or equal to the total number of spares available, i.e., $P_B = \sum_{k=1}^\infty p_k = 1 - \sum_{k=0}^{S-1} p_k$, where $p_k$ is the probability of state $k$ in a queue $M/M/S$, i.e. $p_k = \frac{\rho^k}{k!} \cdot e^{-\rho}$. The blocking probability of all warehouses is therefore

$$P_B = 1 - \sum_{k=0}^{S-1} \frac{\rho^k}{k!} e^{-\rho}.$$ (3)

Unfortunately, this result does not allow to compute the OA of the system, since to this aim the marginal blocking probability of each warehouse is necessary. However, steady state probabilities can be computed for each state in the Markov chain by solving a linear system. To this aim, Theorem 1 can be used to reduce the infinite state space Markov chain to an equivalent one with a finite number of states. Specifically, all the states in which all warehouses are in stockout condition can be replaced with a single state, with probability $P_B$ and with suitable modified departure transition rates.

Let $n^B$ be the state such that $n^B_i = s_i$ for each $i \in W$ and $n^B_{w+1} = 0$. Let $p_{n^B} = \frac{\rho^{n^B}}{n^B!} \cdot e^{-\rho}$ be the probability of state $n^B$ and let $q(n^B, n^B - e_i) = s_i \mu_i$ the departure transition rates from state $n^B$ to state $n^B - e_i$. Let us now replace all states such that $n^B_i = s_i$ for each $i \in W$ and $n^B_{w+1} \geq 0$ with a single state $\hat{n}^B$. To achieve the equivalence with the original Markov chain it is sufficient to set the departure transition rates from state $\hat{n}^B$ to state $\hat{n}^B - e_i$ equal to $q(\hat{n}^B, \hat{n}^B - e_i) = s_i \mu_i F$ for each $i \in W$, where the factor $F$ is equal to

$$F = \frac{p(n^B)}{P_B} = \frac{\frac{\rho^{n^B}}{n^B!} \cdot e^{-\rho}}{1 - \sum_{k=0}^{S-1} \frac{\rho^k}{k!} e^{-\rho}}.$$ (4)

For instance, Figure 3(left) shows a Markov chain with infinite number of states and Figure 3(right) shows its equivalent Markov chain with a finite number of states. In general, the number of states in the finite state space Markov chain is equal to

$$\prod_{i \in W} (s_i + 1).$$ (4)

This number can be exceedingly large as the number of warehouses and spares increases. Therefore, there is a need for approximate methods to compute the OA of large networks.

## 5 Approximate performance computation

In this section we describe three methods for estimating the OA for the single echelon model with complete pooling. The first two methods are based on decomposition.
single-dimensional queueing systems approximate the multi-dimensional original one. The decomposition approach is an exact solution method when the steady state probabilities can be expressed in product form. Unfortunately, the Markov chain model studied in this paper cannot be expressed in product form. However, as described in [5], product form networks provide the basis for many approximate algorithms to solve more general non-product form ones. In this section we describe two decomposition methods. The first adapts the AV method of Alfredsson and Verrijdt [2] to the case of deterministic re-forwardings [18]. The AV method models the demand at each queueing system with a Poissonian independent distribution with adjusted demand rate. The second method assumes non-Poissonian independent demand distributions with adjusted demand rate at each queueing system. It is based on IPP [20, 22] and ERT [15] methods. According to Iversen [14], IPP and ERT methods are particularly suitable to model the peakedness of overflow processes.

The third method is based on the scaling down concept [4]. We apply this concept to the multi-dimensional single echelon with complete pooling context.

5.1 Decomposition approach

The basic ideas of the decomposition methods studied in this paper consists of computing the fraction of the demand $\lambda_i$ at warehouse $i$ that is satisfied by one of three different sources. The first fraction $\beta_i$ (the local fill rate) is directly satisfied by the stock available at warehouse $i$, the second fraction $\alpha_i$ (the transshipment fraction) is satisfied through lateral transshipments from the other warehouses, the third fraction is satisfied by the external supplier through emergency shipments and it is equal to the joint blocking probability $P_B$ of all warehouses, computed as in Theorem 1. Figure 4 shows a pictorial representation of the three flows of spare parts which satisfy the three demand fractions at the first service region of Figure 1(right).

With a decomposition method, each warehouse is studied separately. To this aim, let $p_{s_i}$ denote the probability of having $s_i$ outstanding orders at warehouse $i$, i.e., the probability
of having no stock available at warehouse \( i \). The local fill rate \( \beta_i \) and the transshipment fraction \( \alpha_i \) are therefore:

\[
\beta_i = (1 - p^i_{s_i})
\]

\[
\alpha_i = 1 - \beta_i - P_B
\]  

(5)

The value \( p^i_{s_i} \) for each warehouse can be computed only if the effective arrival rates \( \lambda'_i \) at each warehouse \( i \in W \) are known. To compute the latter values, we let \( o_i \) be the overflow from all other warehouses that is re-forwarded to warehouse \( i \). The closest neighbor sourcing rule for lateral transshipment is taken into account by the probability \( \pi(h, i) \), defined in Section 3.1 as probability of having stock available at warehouse \( i \) and having no stock available at every warehouse \( l \) such that \( T_{hl} < T_{hi} \) (including the case \( l = h \)):

\[
\pi(h, i) = \beta_i \cdot \prod_{l:T_{hl} < T_{hi}} (1 - \beta_l).
\]  

(6)

If values \( \pi(h, i) \) are known, then the values \( o_i \) and \( \lambda'_i \) can be computed as follows:

\[
\begin{align*}
o_i &= \sum_{h=1, h \neq i}^W \pi(h, i) \cdot \lambda_h \\
\lambda'_i &= \lambda_i + o_i.
\end{align*}
\]  

(7)

This expressions are similar to those used in AV method by Alfredsson and Verrijdt [2], with the difference that we use the closest neighbor sourcing rule for lateral transshipments instead of the random sourcing rule used in [2]. Similar modification is applied by Kukreja [18], with the difference that an approximate expression is used in [18], while equation (6) directly follows from the definition of \( \pi(h, i) \).

In order to compute the above quantities, we still need the values \( p^i_{s_i}, \forall i \in W \). The two decomposition methods analyzed in this paper differs in the approach used to estimate these values.
5.1.1 AV method

With the AV method [2], the demand flow at each warehouse is considered Poissonian. Therefore, the steady state probabilities of having \( j \) outstanding requests at warehouse \( i \) are the same that in a Markovian queueing system with \( s_i \) servers and zero buffer. In such a case \( p^i_{s_i} \) may be computed as follows.

\[
p^i_0 = \frac{1}{\sum_{j=0}^{s_i} \frac{(\lambda'_i)^j}{j!}}
\]

\[
p^i_{s_i} = \frac{(\lambda'_i)^{s_i}}{s_i!} \cdot p^i_0
\]

In order to compute \( \alpha_i \) and \( \beta_i \) in steady state an iterative procedure is followed. The iterative procedure starts with \( \beta_i = 1 - P_B \) and \( \alpha_i = 0 \), which implies that \( o_i \) is initially zero for all \( i \in W \). Then, at each iteration, quantities \( \pi(h,i) \) are computed with equation 6 while quantities \( o_i \) and \( \lambda'_i \) are computed with equations 7 and then used to update steady-state probabilities \( p^i_{s_i} \). In the next iteration the values of \( \beta_i, \alpha_i \) are recalculated and used to update the other quantities. This procedure is repeated until the \( \beta_i, \alpha_i \) and \( o_i \) values do not change anymore. These values converge after a few iterations (usually less than 30 in our computational experiments), as experienced also by Axsater [3], Alfredsson and Verrijdt [2] and Kutanoglu [21].

5.1.2 Modified ERT method

With the ERT method [2], demand at each warehouse is only characterized by its mean value and its variance. The basic idea is that the peaked demand at warehouse \( i \) can be viewed as the overflow of another queue \( M/M/k_i/0/\infty \) with \( k_i \) servers, Poissonian demand and exponential service process. Therefore, this method models warehouse \( i \) with a Markovian queueing system with a number of servers \( k_i + s_i \), Poissonian demand flow with average \( A_i \) and zero buffer. The first \( k_i \) servers act as a primary queue, whose peaked overflow is sent to a secondary queue with \( s_i \) servers. Figure 5 is a pictorial representation of the ERT basic idea. The quantities \( k_i \) and \( A_i \) must be determined in order to model the desired peaked effective demand at the secondary queue, with average \( \lambda'_i \) and variance \( v'_i \). In such a case, the values \( p^i_{s_i} \) of Equation (8) are computed as the ratio between the overflow of the queue with \( k_i + s_i \) servers and Poissonian demand \( A_i \) and the effective demand \( \lambda'_i \):

\[
\frac{A_i \cdot E_{k_i+s_i}(A_i)}{\lambda'_i}
\]
In order to compute the values $p^i_s$, we therefore need to compute $A_i$ and $k_i$ values. To this aim, we compute the mean of the effective demand as in equations 7 and the variance of the effective demand, $v'_i$, as follows. Let $v_i$ be the variance of the regular flows for warehouse $i$ and $Z$ be the peakedness factor.

$$v'_i = v_i + \sum_{j=1, j \neq i}^N \pi(j, i) \cdot v_j$$

(10)

Therefore we may solve the following equations 11 w.r.t. $k_i$ and $A_i$:

$$\lambda'_i = A_i \cdot E_k(A_i)$$

$$\frac{v'}{X_i} = Z = 1 - \lambda'_i + \frac{A_i}{k_i + 1 - A_i + \lambda'_i}$$

(11)

This non-linear system has a unique solution [14], and we compute it by using the non linear equations methods of [6].

Differently from [2], we compute the values $\alpha_i$ and $\beta_i$ in steady state with an iterative procedure, similar to the one used in the IPP method [20, 22]. The iterative procedure starts with $\beta_i = 1 - P_B$ and $\alpha_i = 0$, which implies that $o_i$ are initially zero. Then in each iteration $X'_i$ and $v'_i$ are computed, as in equations 7 and 10, and used to compute steady-state blocking probabilities $p^i_s$. The values of $\beta_i$, $\alpha_i$ are then recalculated, used as input to next iteration, and the whole procedure is repeated until the $\beta_i$, $\alpha_i$ and $o_i$ do not change anymore. These values converge after a few iterations.

5.2 Scaling down approach

With the scaling down approach, a system is approximated by scaling the demand, the replenishment time and the stock level of each warehouse using a scale factor $K$. The main purpose is to reduce the stock level at each warehouse $i \in W$ to a new value $\hat{s}_i$, in order to achieve a Markov chain with an affordable number of states, whose probabilities can be efficiently computed.

The intuition behind this method is that the performance levels of the original inventory system depend more on the ratios between demand, replenishment time and stock level than on their absolute values. Their relative sizes may not linearly influence the OA approximation goodness. There are two critical issues in the method. The first issue is the choice of the $K$, the second one is the rounding of the scaled stock levels, which clearly must be integer values. Axsater [4] chooses the scale factor $K$ by keeping the same standard deviation-to-mean ratio in the scaled system as in the original one, while the rounding problem is not addressed since all parameters are assumed to be multiples of $K$. In our procedure we relax the latter assumption and choose a scale factor such that the scaled Markov chain can be solved efficiently.

Our procedure is as follows. Let $MAX$ be the maximum number of states of the Markov chain that can be efficiently managed. In view of equation 4, the number of states that must be taken into account in the scaled Markov chain is equal to $\prod_{i \in W}(\hat{s}_i + 1)$; therefore we set the scale factor $K$ as the minimum integer value such that:
\[
\prod_{i \in W} (\hat{s}_i + 1) \leq \text{MAX.} \tag{12}
\]

Specifically, \( K \) is obtained iteratively as follows. Starting from \( K = 2 \), we set the overall stock level in the reduced system equal to

\[
\hat{S} = \left\lfloor \frac{\sum_{i \in W} s_i K}{S} \right\rfloor + 1 \tag{13}
\]

and then allocate a provisional number of spares \( \left\lfloor \frac{S}{S} \right\rfloor \) to each warehouse \( i \in W \), where \( \hat{S} \) is the overall number of spares in the system. The remaining number of spares \( \hat{S} - \sum_{i \in W} \left\lfloor \frac{s_i}{S} \hat{S} \right\rfloor \) (smaller than \( w \)) is allocated by ordering the warehouses for decreasing value of \( s_i \cdot \hat{S} \cdot S - \left\lfloor s_i \cdot \hat{S} \cdot S \right\rfloor \) and allocating an additional spare to the first warehouses until all spares are allocated, thus obtaining the values \( \hat{s}_i \) associated to the given value of \( K \). In case of tie, the spare is allocated with priority to the warehouse with higher demand. Then, inequality (12) is checked. If \( \prod_{i \in W} (\hat{s}_i + 1) \leq \text{MAX} \) holds, we set \( K \) and \( \hat{s}_i \). Otherwise, we increase \( K \) and repeat the procedure until inequality (12) holds. Finally, the scaled \( \hat{\lambda}_i \) and \( \hat{\mu}_i \) are fixed as \( \hat{\lambda}_i = \frac{\hat{\lambda}}{K} \) and \( \hat{\mu}_i = \frac{\hat{\mu}}{K} \), respectively.

6 Numerical study

In this section we describe our computational experience. We compare the exact results in terms of OA values, computed by directly solving the Markov chain model, with the results obtained with the approximate techniques described in section 5. Besides the percentage error in terms of OA values between the approximate OA value and the Markov chain one, we report on the computation time and on the memory required by the different models. All the experiments are executed on a PC equipped with a processor Intel Core2 Duo CPU (3 GHz), 3.25 GB Ram and Windows operating system.

The set of instances used for our computational study is composed by 60 practical instances from the Italian airport maintenance context plus other 990 randomly generated instances. The experiments are carried out by varying the mean demand to each warehouse, the number of warehouses and the stock levels. The departure transition rate \( \mu_i \) of each server is fixed equal to \( \mu = \frac{1}{3 \text{months}} \) for all instances. This is the value used by the managers in the practical application.

Random instances are classified according to the average arrival rates from all airports \( \hat{\lambda} = \sum_{i \in W} \lambda_i \) and to the stock levels \( s_i, i \in W \). Instances with \( \hat{\lambda} = 0.001 \) are in the low demand class, instances with \( \hat{\lambda} = 0.01 \) are in the medium demand class, and instances with \( \hat{\lambda} = 0.1 \) are in the high demand class. As for the stocking policy, for each demand class we generate 17 distributed instances, in which \( s_i = 1 \forall i \in W \) and \( w = 1, 2, \ldots, 17 \). Other 169 centralized instances are obtained by storing all spares in a single warehouse \( j \in W \) that acts as a central depot, i.e., \( s_i = 0 \) for \( i \in W - \{j\} \) and \( s_j = S \) for \( S = 1, 2, \ldots, 10 \) and for \( w = 1, 2, \ldots, 17 \) (excluding the case \( S = w = 1 \), which is included in the previous group). Finally, 144 hybrid instances are obtained by setting \( s_i = 1 \) for \( i \in W - \{j\} \) and \( s_j \in \{2, 3, \ldots, 10\} \), for \( w = 2, \ldots, 17 \). For each instance, the subset of warehouses, as well as the central depot \( j \) in the latter two groups of instances, is chosen at random among the 17 regional warehouses of the practical application. Overall, there are 17+169+144 random
instances for each demand class, for a total of 990 random instances, besides the 60 practical instances.

As far as the multi-dimensional scaling down approach is concerned, the computer used in our experiments can easily manage Markov chains with $2^{14}$ states. Using a value $MAX = 2^{14}$ in equation (12) results in a scale factor $K = 2$ for all the 1050 instances. Hence, we denote with $K2$ this case. In subsection 6.3 we evaluate the effect of the scale factor on the overall performance of the multi-dimensional scaling down approach.

6.1 Time and memory effort

In practical data experiments, the exact computation with the Markov chain model fails in finding a solution in 18 out of 60 practical instances, due to memory limits. As for the random instances, the exact computation fails in about 30% of the cases. Table 1 shows some practical instances for which the Markov chain approach fails in finding the exact OA value. For each item we show the cost in euro, the replenishment time (months) and the MTBF (hours) of each item. The last three columns report the number of warehouses with at least one spare, the total number of installed items and the total number of allocated spares.

The smallest instances that cannot be solved with the Markov chain model, in terms of the quantity $\prod_{i \in W} (s_i + 1)$, have 12 warehouses and 19 allocated spares for the practical instances, and 12 warehouses and 13 allocated spares for the random instances. However, we observe that also the arrival rate $\bar{\lambda}$ affects the computation time.

<table>
<thead>
<tr>
<th>Item</th>
<th>Cost (euro)</th>
<th>Repair time (months)</th>
<th>MTBF (hours)</th>
<th>num. warehouses</th>
<th>installed items</th>
<th>$\sum_{i \in W} s_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\approx$ 7000</td>
<td>3</td>
<td>76000</td>
<td>12</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>$\approx$ 6000</td>
<td>3</td>
<td>12000</td>
<td>12</td>
<td>19</td>
<td>29</td>
</tr>
<tr>
<td>3</td>
<td>$\approx$ 1800</td>
<td>3</td>
<td>45000</td>
<td>14</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>$\approx$ 3000</td>
<td>3</td>
<td>109000</td>
<td>14</td>
<td>25</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>$\approx$ 3000</td>
<td>3</td>
<td>26000</td>
<td>12</td>
<td>20</td>
<td>24</td>
</tr>
</tbody>
</table>

Table 1: Five non solvable instances.

The time and memory effort needed to solve the Markov chain increase rapidly with the number of warehouses and installed items. In figure 6 we show the computation time required to solve the Markov chain model for the 17 distributed instances and for the three levels of demand $\bar{\lambda}$. Instances with low demand require significantly lower computation time. Moreover, the Markov chain model fails in computing the exact solution in one case for $\bar{\lambda} = 0.001$, in three cases for $\bar{\lambda} = 0.01$ and in four cases for $\bar{\lambda} = 0.1$. Figure 7 shows the memory effort required to solve the same instances with the Markov chain model.

In figures 8 and 9 we compare the computation time and memory effort required by the three approximate models to solve the 60 practical instances. The instances are ordered for increasing number of warehouses and, if equality holds, for increasing number of spares. It can be observed that, even if all the three methods are quite efficient in computing a solution, the multi-dimensional scaling down approach is very fast with all instances. The maximum time required to solve a practical instance is 0.11 seconds. As expected, the ERT method is slightly more time consuming than the AV method, due to the need to solve a non-linear
Figure 6: Computation time for the Markov chain model and distributed instances.

Figure 7: Memory effort for the Markov chain model and distributed instances.
system instead of using a closed form expression as in the AV method. Similar behavior can be observed for the memory effort, shown in Figure 9.

Figure 8: Computation time for the approximate models and practical instances.

In figure 10 we compare the computation time and memory effort required by the three approximate models to solve the 990 random instances.

In the figure we report the average computation time (respectively, the memory effort) required to solve all the instances with the same demand $\lambda$, the same number of warehouses $w$ and the same stock level $S$. The computation times of the three approximate methods increase with $\lambda$ and $w$. With the two decomposition methods AV and ERT, the computation time decreases with $S$, differently from the Markov chain approach. This is due to the overflow reduction caused by an higher $S$, which results in a reduced number of iterations required by the two methods to achieve convergence. As for the memory effort, we observe that with all the three methods the memory occupation is not an issue. For the AV and ERT methods, the memory required to solve every instance only slightly increases with $\lambda$ and $w$, while is almost constant with $S$. The multi-dimensional scaling down approach is
Figure 10: Computation time (left) and memory effort (right) for the approximate models and random instances.

more sensitive to the increase of $\tilde{\lambda}$, $w$ and $S$, but the memory required is always very limited in our experiments, and can be controlled by increasing the scale factor $K$.

6.2 Accuracy analysis

We now analyze the percentage error in OA evaluation for the three approximate methods with respect to the Markov chain solution. Figure 11 shows the percentage error achieved for the 42 practical instances for which the exact OA value can be computed with the Markov chain model. The 42 instances are ordered for increasing value of their exact Operational Availability. The two decomposition techniques ERT and AV provides the same values in practice, since their percentage difference is always smaller than $10^{-6}$. This is mainly due to the fact the estimated peakedness factor is almost 1 for all instances, and therefore modeling the arrival process as a peaked process, as in the ERT method, does not provide benefits with respect to approximating it with a Poisson process, as in the AV method. Therefore, in Figure 11 only one curve is shown for the two decomposition methods. The scaling down method clearly outperforms the decomposition techniques for small OA values ($OA < 0.997$), while the percentage error is similar for larger OA values. Besides the better performance shown in figure, in our experiments the scaling down method provides OA values smaller than the exact ones in more than 80% of the experiments while the decomposition methods find OA values always larger than the exact ones. The scaling down method is therefore more conservative than the decomposition methods, and this is an important feature when the method has to be used within an optimization procedure for spares allocation.

Figure 12 shows the percentage error achieved for the 695 random instances for which the exact OA value can be computed with the Markov chain model. Instances with similar exact OA values are grouped together and the average error is shown in figure for the three
models. Also for the random instances, the scaling down method clearly outperforms the decomposition techniques for small OA values while it behaves similarly for larger OA values.

Figure 11: Percentage error for practical instances

![Figure 11: Percentage error for practical instances](image)

Figure 12: Percentage error for random instances

![Figure 12: Percentage error for random instances](image)

6.3 Influence of the scale factor in the multi-dimensional scaling down approach

In this section we show the influence of the scale factor $K$ in the multidimensional scaling down method on the evaluation accuracy. In Figure 13 we report the OA value computed with the multidimensional scaling down method for four random instances and for $K$ varying from 1 to 10, the value for $K = 1$ corresponding to the exact Markov chain computation. The four instances mainly differ each other for the number of spares $S$. It is interesting to notice that the approximate values are very similar to the exact one when $K$ is sufficiently smaller than $S$, while the estimation deteriorates for $K \geq S$. In fact, the critical points of this method are the computation of the scaled number of spares $\hat{S}$ in Equation (13) and the allocation of these spares, which make the scaled model quite different from the original model as $K$ approaches $S$. 

![Figure 13: OA percentage error w.r.t. Markov chain model](image)
7 Conclusions

In this paper, we have presented three approximate evaluation techniques for estimating the operational availability of a maintenance supply chain with single echelon inventory and complete pooling. An external supplier manages un-satisfied spare part requests when no spare is available in the warehouses of the network. From our computational experiments, carried out on practical and randomly generated instances, it turns out that the two decomposition approaches AV and ERT generate quite the same OA values. Therefore, taking into account the peaky nature of overflows, as in ERT, does not improve the accuracy of the solutions and is more time consuming with respect to modeling overflows with Poissonian flows, as in the AV method. However, the scaling down approach clearly outperforms the two decomposition approaches in terms of both accuracy and computation time.

Future developments of this research include the incorporation of these fast approximation methods in an optimization framework, to optimize the amount of spares and their allocation in order to grant the minimum levels of operational availability required by airport authorities at the minimum cost. To this aim the scaling down approach is preferable to the decomposition methods since the approximate OA values are usually smaller than the exact values computed by solving the Markov chain, and therefore more conservative.

References


