The stochastic single node service provision problem

Dye, S.; Stougie, L.; Tomasgard, A.

Published: 01/01/1999

Document Version
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
- A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Memorandum COSOR 99-13

The stochastic single node service provision problem

Shane Dye
Leen Stougie
Asgeir Tomasgard

Eindhoven, August 1999
The Netherlands
The stochastic single node service provision problem*

Shane Dye
University of Canterbury, New Zealand

Leen Stougie
Eindhoven University of Technology, The Netherlands

Asgeir Tomasgard
SINTEF Industrial Management, Norway

Abstract

The service provision problem described in this paper comes from an application of distributed processing in telecommunications networks. The objective is to maximize a service provider's profit from offering computational based services to customers. The service provider has limited capacity of some resources and therefore must choose from a set of software applications those he would like to offer. This can be done in a dynamic manner taking into consideration that demand for the different services is uncertain. This problem is examined in the framework of stochastic integer programming. Approximations and complexity are examined for the case when demand is described by a discrete probability distribution and one resource limits the number of software applications that may be installed. For the deterministic counterpart a fully polynomial approximation scheme is known [2]. We show that introduction of stochasticity makes the problem strongly NP-hard, implying that the existence of such a scheme for the stochastic problem is highly unlikely. For the general case a heuristic with a worst-case performance ratio that increases in the number of scenarios is presented. Restricting the class of problem instances in a way that many reasonable practical problem instances will satisfy, allows for the derivation of a heuristic with a constant worst-case performance ratio. These worst-case results are the first results for stochastic programming problems that the authors are aware of in a direction that is classical in the field of combinatorial optimization. The results do not follow straightforwardly from the deterministic counterparts of the problem.

Key words: distributed processing; telecommunications; service provision; stochastic (integer) programming; strong NP-hardness; approximation; worst-case analysis

Preprint submitted to Elsevier Preprint

17 August 1999
1 Introduction

The service provision problem discussed in this paper comes from an application in telecommunications. It considers how to install different processing based services on a set of computer nodes in a network with distributed processing capabilities. The computers typically have limited resources such as memory, processing capacity and storage capacity. All the services are built from a set of subservices. The subservices are software applications, which run in a distributed manner in the network. The subservices communicate using an underlying transportation network. The focus of this paper is on how to allocate computational resources to a set of subservices in order to meet customer demand for services. Because the resources are limited, it may be necessary to reject some customers, and it is assumed that the service provider tries to maximize his profit when he allocates resources to subservices.

Often it makes sense to look at the allocation of node resources to subservices independently of allocation of transportation resources. This is typically the case when transportation is not a bottleneck for the subservices considered or if transportation capacity is high enough to handle all variations of communications between the subservices.

This change of focus from transportation resources to node resources may follow as a consequence of technological developments increasing the transportation capabilities of networks, such as: digital technology, modern packet switched high speed networks and architectures like ATM [1,8]. Also, many new services and some transportation technologies require more and more processing and node resources, thereby creating a new bottleneck at the computing nodes distributed about the telecommunications network. Add to this the computing industry's influence upon the telecommunication markets and one gets both additional computational power and competition. It is a strong belief in parts of the industry that many new telecommunications services will be more "processing hungry" than the ones seen today. It is difficult to predict which of the bottlenecks, transportation or node resources, will prove to be most influential. From the prognosis that the problem of allocating node resources will be important in near future (as one can already see for the Internet) the authors were asked by the industrial financial contributor to examine

* For this work, financial support has been received from Telenor and Leonardo Da Vinci. Dye and Tomassgard were employees at The Norwegian University of Science and Technology in Trondheim, Norway when the main part of the work was done.
the situation where transportation does not play a role.

Because of the distributed processing capabilities of the network, it may also be possible to consider subservice demand independently of which service generated it. In [10] it is further described when the service provision problem can be solved by allocation of node resources to subservices without considering transportation and customer location. Here it is assumed that the transportation network, the markets, and the distributed processing capabilities of the network nodes make this possible.

This paper considers a variant of the problem with only one node on which to install subservices. Even with only one node, the problem is meaningful in a distributed processing setting. This is typically the situation a service provider faces when he rents capacity from a network provider who manages multiple distributed nodes. The service provider does not take into consideration whether the capacity he has rented is located on one or several computing nodes. He uses it as if it were one continuous block of capacity. The network provider on the other hand is free to replicate and move the various service providers' subservices on all the nodes he manages. He sees a similar problem to the service providers, but with several nodes available. For a further discussion of the roles in the network and a discussion around distribution see [9,10].

The problem arose through a telecommunication company. The managers of this company were interested in how to adapt the configuration of the intelligent nodes of their network to more or less suddenly occurring abnormal demand situations. They had noticed from comparative data studies that at various times the demand pattern shows peaks in demand of services. Typically the demand of only one service peaks at a time. This influences the level of demand for all the subservices used by the service. Moreover, before the moment that a peak actually occurs deviations from the normal demand patterns for subservices can be observed. These deviations can be used as a signal indicating that a peak is about to occur. The signals do not allow determination of the precise character of the possibly upcoming peak as many services may use the same subservices, but any such signal points to a limited number of possible services that might give the demand peak. Of course the exact height of the peak is not indicated by the signal. However, all this implies that for any possible signal a few scenarios often give sufficient description of the situation that is about to occur in terms of subservice demand.

After each such signal there is about enough time to adapt the configuration of the node(s) to anticipate the peak. This is due to the set-up delay for subservices and a two-stage decision situation naturally emerges. In the first stage the decision is which subservices to install given only probabilistic information on demand for subservices. During the set-up time uncertainty resolves itself
and at the time when the second-stage decisions are to be made demand is
known. There is also, typically, a shutdown time; the period between the mo­
ment when a service provider decides to remove a subservice and the moment
when the resources it uses become free. Due to this the only possible recourse
action concerns what demand should be met using the subservices installed in
the first stage. The available capacity is restricted by the first stage decision.

The remainder, will focus on the problem where a single constrained resource
at the single computing node is allocated to subservices, where transportation
is not an issue, and where subservice demand can be treated independently
of the services. The decision process is modelled as a two-stage stochastic
program with recourse, as in [10].

Here the underlying decision process is briefly described. The probability dis­
tribution of uncertain demand is assumed to be discrete and is described in
stochastic programming terminology in terms of scenarios [5]. Denote by m
the number of demand scenarios and by \( p_k \) the probability of scenario \( k \) oc­
curring. In a scenario, demand for each subservice is realized, and a scenario
can therefore be viewed as a vector of demands with an assigned probability.
Both demand and profit are treated in terms of the limited resource related to
the subservices. Then \( \delta_{jk} \) is demand for the resource generated by subservice
\( j \) in scenario \( k \), and \( q_j \) is the corresponding profit obtained from allocating
one resource unit to the provision of this subservice. Let \( n \) be the number of
subservices and \( s \) the resource capacity of the single node. In addition, each
subservice uses a fixed amount of capacity just to be available, this is inde­
pendent of the demand met. This is called the installation requirement of the
subservice and denoted by \( r_j \) for subservice \( j \).

To arrive at a mathematical programming formulation the following decision
variables are introduced. The first-stage variables \( z_j \) indicate whether sub­
service \( j \) is installed, in which case \( z_j = 1 \), or not, indicated by \( z_j = 0 \),
\( j = 1, \ldots , n \). The second-stage variables \( x_{jk} \) denote the amount of the re­
source that is allocated to subservice \( j \) in scenario \( k \). When the node capacity,
the installation requirements, and demands are integral, the \( x \) variables will
automatically be integral.

When uncertain demand for subservices is described by a discrete distribu­
tion, as discussed in [10] a deterministic equivalent [5] can be formulated. This
deterministic equivalent will be a linear mixed integer programming model
(MIP) [7] where the objective is to maximize expected profit subject to two
sets of constraints. The first constraints, called the capacity constraints, ensure
that the capacity of the node is not exceeded in any scenario. The second set,
the demand constraints, make sure that demand for a subservice can only be
met if the subservice is installed at the node. The resulting mathematical pro­
gramming formulation of the stochastic single node service provision problem

4
(SSNP) can be written as:

\[
\begin{align*}
    \max & \quad \sum_{k=1}^{m} p_k \sum_{j=1}^{n} q_j x_{jk} \\
    \text{s.t.} & \quad \sum_{j=1}^{n} (r_j z_j + x_{jk}) \leq s \quad k = 1, \ldots, m, \\
    & \quad \delta_{jk} z_j - x_{jk} \geq 0 \quad j = 1, \ldots, n, \quad k = 1, \ldots, m, \\
    & \quad z_j \in \{0, 1\}, \quad x_{jk} \geq 0 \quad j = 1, \ldots, n, \quad k = 1, \ldots, m.
\end{align*}
\]

(1)

In the remainder of this paper the expected demand for subservice \( j \) will be written in the following manner

\[ E_k[\delta_{jk}] = \sum_{k=1}^{m} p_k \delta_{jk}, \]

diverging slightly from customary notation for expectations in probability theory literature.

To facilitate the exposition the assumption is made that no demand is higher than the node capacity minus the corresponding installation requirement. This can, if necessary, be ensured by preprocessing.

**Assumption 1** For any subservice \( j \) in any scenario \( k \), the support of \( \delta_{jk} \) is in the interval \([0, s - r_j]\).

A consequence of this is that for any subservice the profit of meeting its expected demand is no greater than the optimal profit of the overall problem. Let \( \pi^{\text{OPT}} \) be the optimal value of (1). Then Assumption 1 ensures that

\[ \pi^{\text{OPT}} \geq q_j E_k[\delta_{jk}], \quad j = 1, \ldots, n. \]

(2)

Feasibility of the deterministic service provision problem with multiple nodes and the requirement that all demand must be met is described in [3] and shown to be strongly NP-complete. When demand is deterministic it was shown in [2] that the single node problem where profit is maximized is NP-hard and has a fully-polynomial time approximation scheme. In the same paper it is shown that the multiple node problem is strongly NP-hard and that there exists no fully polynomial approximation scheme even when the number of nodes is fixed. The analysis turned out to have many similarities with the well known knapsack problem [6].

We show in Section 4 that (SSNP) is strongly NP-hard, whereas as noted above the deterministic counterpart admits a fully polynomial approximation
scheme. This is remarkable since the integer variables appear only in the first stage of the two-stage stochastic programming problem. When the number of scenarios is fixed the problem can be solved in pseudo-polynomial time by dynamic programming.

When the number of scenarios is not a constant, there is little hope to find efficient algorithms that solve the problem to optimality or fully polynomial time approximation schemes. It is still possible to find good approximations. This is the motivation behind investigating the LP relaxation. The LP relaxation is discussed in Section 2 together with an approximation method directly based on the LP results. A worst case bound increasing in the number of scenarios is given. In Section 3, for a slightly restricted problem class (to which many reasonable practical problem instances belong) the bound on the ratio between the LP solution value and the optimal integer one is tightened and a constant bound approximation method based on the proof is presented. These are the first worst-case performance results known by the authors for approximation of stochastic integer programming problems.

2 The LP bound and a heuristic

The LP relaxation of (SSNP) replaces the requirement \( z_j \in \{0, 1\} \) in (1) by \( 0 \leq z_j \leq 1 \) for \( j = 1, \ldots, n \). This section describes an optimal basis for the LP relaxation of (SSNP) and uses it to give an upper bound on the ratio of the LP versus the optimal solution. A heuristic based on the bound is given subsequently in Subsection 2.2.

2.1 The LP bound

Relaxing the integrality constraints, consider the resulting LP. The following theorem bounds the number of fractional variables in an optimal LP solution. A variable \( z_j \) is fractional if \( 0 < z_j < 1 \), and a variable \( x_{jk} \) is fractional if \( 0 < x_{jk} < \delta_{jk} z_j \). Note that if \( z_j < 1 \), then it is possible to have \( 0 < x_j < \delta_{jk} \) without \( x_{jk} \) being fractional, as long as it is equal to \( \delta_{jk} z_j \).

Theorem 1 Any basic optimal solution to the LP relaxation of (SSNP) with \( m \) scenarios has at most \( m \) fractional \( z \) and \( x \) variables.

PROOF. Let \( (x^{LP}, z^{LP}) \) be an optimal basic solution to the LP relaxation of (SSNP). Define the reduced problem to be the instance with problem data corresponding to the original, with the exception that subservices for which
$z_j^{LP} = 0$ are removed. The corresponding optimal solution of the reduced problem has the same number of fractional $x$ and $z$ variables. This means the only instances to consider have an LP relaxation with a basic optimal solution, $(z^{LP}, x^{LP})$, for which $z^{LP} > 0$.

Introducing slacks $t_k$ for the capacity constraints and $u_{jk}$ for the demand constraints, results in the following reformulation of the LP relaxation.

$$\max \sum_{j=1}^n \sum_{k=1}^m p_k q_j x_{jk}$$

s.t.

$$\sum_{j=1}^n r_j z_j + \sum_{j=1}^n x_{jk} + t_k = s \quad k = 1, \ldots, m,$$

$$\delta_{jk} z_j - x_{jk} - u_{jk} = 0 \quad j = 1, \ldots, n, \quad k = 1, \ldots, m,$$

$$0 \leq z_j \leq 1, \quad x_{jk}, u_{jk}, t_k \geq 0 \quad j = 1, \ldots, n, \quad k = 1, \ldots, m.$$ (3) (4) (5) (6)

This LP has $m + nm$ constraints, so that at any basic solution at most $m + nm$ variables will lie strictly between their bounds. Let $(t^{LP}, u^{LP}, x^{LP}, z^{LP})$ be a basic optimal solution to the above for which $z^{LP} > 0$. Now, count the number of variables lying strictly between their bounds.

Since $z_j^{LP} > 0$, $j = 1, \ldots, n$, Constraints (5) imply that at least one of $x_{jk}^{LP}$ or $u_{jk}^{LP}$ will be positive for each pair $(j, k)$, $j = 1, \ldots, n, \quad k = 1, \ldots, m$. This accounts for at least $nm$ variables strictly between their bounds. Define the following sets

$$\mathcal{F} = \{ j \mid z_j^{LP} < 1 \},$$

$$\mathcal{U} = \{ (j, k) \mid x_{jk}^{LP} > 0 \text{ and } u_{jk}^{LP} > 0 \}$$

and

$$\mathcal{T} = \{ k \mid t_k^{LP} > 0 \}.$$

Notice that $\mathcal{U}$ is exactly the set of indices for which $x_{jk}^{LP}$ are fractional because they are positive but not equal to $\delta_{jk} z_j^{LP}$.

The number of fractional $z^{LP}$ and $x^{LP}$ is $|\mathcal{F}| + |\mathcal{U}|$ and the number of variables lying strictly between their bounds is $|\mathcal{F}| + |\mathcal{T}| + |\mathcal{U}| + nm$. From the above this is no greater than $m + nm$, implying $|\mathcal{F}| + |\mathcal{T}| + |\mathcal{U}| \leq m$. □
Thus, if \((z^{LP}, x^{LP})\) is a basic optimal solution to the LP relaxation write its optimal value, \(\pi^{LP}\), as

\[
\pi^{LP} = \sum_{j \in W} \sum_{k=1}^{m} p_k q_j x^{LP}_{jk} + \sum_{j \in F} \sum_{k=1}^{m} p_k q_j x^{LP}_{jk}
\]

where \(W = \{j | z_j^{LP} = 1\}\) and \(F = \{j | 0 < z_j^{LP} < 1\}\).

In particular, \(|F| \leq \min\{m, n\}\). Under Assumption 1, the above theorem provides an immediate bound for the optimal value of the LP relaxation in terms of the optimal solution value \(\pi^{OPT}\) of (SSNP).

**Corollary 2** If \(\pi^{OPT}\) is the optimal solution value of an instance of (SSNP) and \(\pi^{LP}\) is the optimal value of the LP relaxation, \(\pi^{LP} \leq \min\{m + 1, n\} \pi^{OPT}\).

**Proof.**

\[
\pi^{LP} \leq \sum_{j \in W} \sum_{k=1}^{m} p_k q_j x^{LP}_{jk} + \sum_{j \in F} \sum_{k=1}^{m} p_k \delta_{jk}
\]

\[
\leq \pi^{OPT} + \sum_{j \in F} \pi^{OPT}
\]

\[
\leq (\min\{m, n\} + 1) \pi^{OPT}
\]

Notice that \(\sum_{j \in W} \sum_{k=1}^{m} p_k q_j x^{LP}_{jk}\) is the value of an integer feasible solution and is therefore no greater than \(\pi^{OPT}\). To obtain the second inequality is then a matter of applying (2). The last inequality is implied by \(|F| \leq \min\{m, n\}\), which is a direct consequence of Theorem 1. Finally, note that if \(|F| = n\), \(W = \emptyset\) so that (8) implies \(\pi^{LP} \leq n \pi^{OPT}\). \(\Box\)

We have no example that shows tightness of this bound. The worst example we found so far has a ratio \(\pi^{LP}/\pi^{OPT} = 4\).

2.2 *The LP round-down heuristic*

This section investigates a heuristic which amounts to rounding down the optimal solution of the LP relaxation of (SSNP). The worst-case performance ratio analysis is related to the analysis for the greedy heuristic of the knapsack problem [6, Subsection 2.4]. In the deterministic case the knapsack LP solution can be found in \(O(n)\) time by a median finding algorithm using the price per unit criterion [2]. Here a similar approach is not known.
The previous section showed that any optimal LP solution of an $m$-scenario problem will have at most $m$ subservices for which the $z^*_{j}\text{-values}$ are fractional. All remaining $z$ are 0 or 1. This motivates the following LP round-down heuristic, which we call LPR: Install each subservice $j$ for which $z^*_{j} = 1$ and no others, that is install all $j \in \mathcal{W}$. Afterwards the remaining capacity is allocated to serve demand of the installed subservices in a greedy manner, starting with the subservices with the highest $q_j$. Assume for simplicity that the subservices are sorted by non-increasing $q_j$. Then there will be a critical subservice $j_k$ in each scenario $k$ for which $x^*_{jk} = \delta_{jk}z^*_{j_k}, j < j_k$ and $x^*_{jk} = 0, j > j_k$. Let $(z^{\text{LPR}}, x^{\text{LPR}})$ be the heuristic solution and $\pi^{\text{LPR}}$ the solution value.

**Proposition 3** A lower bound for the LP round-down heuristic (LPR) value is to allocate only the amount indicated by the LP solution to each installed subservice

$$\pi^{\text{LPR}} \geq \sum_{j \in \mathcal{W}} \sum_{k=1}^{m} p_k q_j x^{\text{LP}}_{jk}.$$ 

**PROOF.** In the LPR heuristic all the space allocated to subservices $j \in \mathcal{F}$ in the LP is free, as these subservices are not installed. This free capacity can potentially be used to meet demand for subservices $j \in \mathcal{W}$. So $x^{\text{LPR}}_{jk} \geq x^{\text{LP}}_{jk}, \forall j \in \mathcal{W}, \forall k$. □

LPR can for some instances of the problem be arbitrarily bad, because a better solution with an arbitrarily higher value may be to install one of the fractional subservices. The heuristic is now modified into a heuristic that we call [LP] to avoid this problem. If the value of installing the best of the services $j \in \mathcal{F}$ is higher than the value of installing all subservices $j \in \mathcal{W}$ then do that instead. Let $\pi^{[\text{LP}]}$ be the optimal value of this heuristic. Then

$$\pi^{[\text{LP}]} = \max\{\pi^{\text{LPR}}, \max_{j \in \mathcal{F}} q_j E_k[\delta_{jk}]\}.$$ 

**Theorem 4** The modified LP round-down heuristic [LP] has a worst case performance ratio of

$$\pi^{\text{OPT}}/\pi^{[\text{LP}]} \leq \min\{m + 1, n\},$$

and this ratio is tight.

**PROOF.**
\[ \pi^{\text{OPT}} \leq \pi^{\text{LP}} = \sum_{j \in W} \sum_{k=1}^{m} p_k q_j x_{jk}^{\text{LP}} + \sum_{j \in \mathcal{F}} \sum_{k=1}^{m} p_k q_j x_{jk}^{\text{LP}}, \]

\[ \leq \pi^{\text{LPR}} + \sum_{j \in \mathcal{F}} q_j E_k [\delta_{jk}], \]

\[ \leq \pi^{[\text{LP}]} + |\mathcal{F}| \pi^{[\text{LP}]} \leq (\min\{m, n\} + 1) \pi^{[\text{LP}]} . \]

Again, the case where \(|\mathcal{F}| = n\) tightens the bound to \(\min\{m + 1, n\} \pi^{[\text{LP}]}\).

A tight example is given here. The problem has \(v + 1\) subservices and \(v\) scenarios, \(v \geq 2\). Let \(q_j = v - \epsilon\), \(r_j = \epsilon\), \(j = 1, \ldots, v\), \(q_{v+1} = v/\epsilon\) and \(r_{v+1} = 0\) where \(0 < \epsilon < 1\). The node size is \(s = 1 + \nu \epsilon\) and all scenarios are equally likely. Demand is defined to be constant over all scenarios for subservice \(v + 1\), \(\delta_{v+1k} = \epsilon/v\), \(k = 1, \ldots, v\). For all other subservices \(j = 1, \ldots, v\) demand is present only in scenario \(j\), with \(\delta_{jj} = 1\) and \(\delta_{jk} = 0\) when \(j \neq k\).

The optimal solution is to install all subservices. Demand for subservice \(v + 1\) is always met completely, while in scenarios \(j = 1, \ldots, v\) the optimal solution has \(x_{jj}^{\text{OPT}} = 1 - \frac{\epsilon}{v}\). The profit from this is \(\pi^{\text{OPT}} = 1 + v - 2\epsilon + \frac{\epsilon^2}{v}\).

The optimal LP solution, \((z_{jk}^{\text{LP}}, x_{jk}^{\text{LP}})\) is \(z_{v+1k}^{\text{LP}} = 1\), \(x_{v+1k}^{\text{LP}} = \epsilon/v\), \(\forall k\), \(x_{jj}^{\text{LP}} = z_{jk}^{\text{LP}} = \frac{1 + \nu \epsilon - \frac{\epsilon}{v}}{1 + \nu \epsilon}, \forall j, x_{jk}^{\text{LP}} = 0, k \neq j\). This solution has \(m = v\) fractional \(z\)-values and no fractional \(x\)-values. The modified LP round-down solution value \(\pi^{[\text{LP}]}\) is the maximum of installing one of the fractional subservices or subservice \(v + 1\): \(\pi^{[\text{LP}]} = \max\{1, \frac{1}{v}(v - \epsilon)\} = 1\). As \(\epsilon\) gets arbitrarily small,

\[ \frac{\pi^{\text{OPT}}}{\pi^{[\text{LP}]}} = 1 + v - 2\epsilon + \frac{\epsilon^2}{v} \]

gets arbitrarily close to \(v + 1\) where \(v\) is the number of scenarios and \(v + 1\) is the number of subservices. \(\square\)

Notice that the given bound on the performance ratio holds for any possible discrete distribution defined in terms of scenarios. It is increasing in the number of scenarios and if the number of scenarios is greater than the number of subservices the bound is even linear in the number of subservices, which, in general, is not a very favourable situation. For the considered application with a limited number of scenarios, it may still be useful. Yet, it would be better to have a constant performance ratio. Next, the bound on the LP ratio is tightened for a class of problem instances and a heuristic with a constant bound for this problem class is defined.
3 A constant bound

The results from Section 2 depend on the demand probability distribution in a fundamental way. It is directly dependent on the number of realizations the random variables may take. This section shows that for a class of service provision problems it is possible to find a worst-case ratio that is independent of the discrete demand distribution.

The class of problems examined are those for which it is feasible (but not necessarily optimal) to install all subservices concurrently. That is, the sum of the installation requirements is less than the node capacity. This assumption is reasonable in many cases for the problem setting. In order to facilitate the exposition, the node capacity is scaled to 1; \( s = 1 \). In this setting the class of problems has

\[
\sum_{j=1}^{n} r_j \leq 1. \tag{9}
\]

In Section 2 the bound was obtained by considering each fractional subservice individually. In this section the bound is improved by considering sets of these subservices together. The important aspect here is the trade-off between the number of sets and the capacity used by the installation requirements of the subservices in each set.

3.1 The \( LP \) ratio

Let \((z^{LP}, x^{LP})\) be a basic optimal LP relaxation solution. Let \( \ell \) be the number of fractional \( z_j^{LP} \) and assume that \( \ell_w \) of these subservices have \( r_j \leq w \) for some \( 0 < w < 1 \) to be chosen later. These subservices will be installed in groups while those with \( r_j > w \) will be installed separately as before. Again, let \( \mathcal{W} \) be the set of subservices with \( z_j^{LP} = 1 \). Without loss of generality let \( 0 < z_j^{LP} < 1 \) and \( r_j \leq w \) for \( j = 1, \ldots, \ell_w \) and \( 0 < z_j^{LP} < 1 \) and \( r_j > w \) for \( j = \ell_w + 1, \ldots, \ell \). Write the optimal LP value as

\[
\pi^{LP} = \pi_0^{LP} + \pi_1^{LP} + \pi_2^{LP} \tag{10}
\]

where

\[
\pi_0^{LP} = \sum_{j \in \mathcal{W}} q_j E_k[z_{jk}^{LP}].
\]
\[ \pi_1^{LP} = \sum_{j=1}^{\ell_w} q_j E_k[x_{j,k}^{LP}], \]

and

\[ \pi_2^{LP} = \sum_{j=\ell_w+1}^\ell q_j E_k[x_{j,k}^{LP}]. \]

Feasible solutions generated from the LP solution will be used to bound parts of (10). From Section 2

\[ \pi^{OPT} \geq \pi^{LPR} \geq \pi_0^{LP}. \] (11)

Next \( \pi_1^{LP} \) is bound. First, define \( \sum_{j=1}^\ell r_j z_j^{LP} = A \) and note that \( \sum_{j=1}^{\ell_w} x_{j,k}^{LP} \leq 1 - A \) for each \( k = 1, \ldots, m \). Integer feasible solutions are generated for which the capacity used by the \( r_j \)'s of the installed subservices is close to some constant \( \beta \). First partition the set \( \{1, \ldots, \ell_w\} \) into \( I \) subsets, \( \{S_i\}_{i=1}^I \), where

\[ \sum_{j \in S_i} r_j \leq \beta + w \quad i = 1, \ldots, I \]

and

\[ \sum_{j \in S_i} r_j \geq \beta \quad i = 1, \ldots, I - 1. \] (12)

Notice that the last bound is not required for \( S_I \). The LP relaxation had at most \( 1 - A \) units of capacity available for the \( x \) variables. Installing only the subservices in one of the sets \( S_i \) will leave at least \( 1 - \beta - w \) units of capacity available. The \( x \)-variable values from the LP relaxation solution corresponding to subservices in \( S_i \) may be scaled down, if necessary, to use a total of no more than \( 1 - \beta - w \) units of capacity in each scenario.

For each \( i = 1, \ldots, I \) generate the integer feasible solution \( (z_i^{H_i}, x_i^{H_i}) \) for which \( z_j^{H_i} = 1 \) for \( j \in S_i \), and \( z_j^{H_i} = 0 \) for all \( j \notin S_i \). Set \( x_j^{H_i} = \gamma x_j^{LP} \) for \( j \in S_i \), \( k = 1, \ldots, n \) and \( x_j^{H_i} = 0 \) for all other \( jk \) with

\[ \gamma = \begin{cases} \frac{1 - \beta - w}{1 - A} & \text{if } \beta + w \geq A, \\ 1 & \text{otherwise.} \end{cases} \] (13)
Now the objective value of the solution \((z^H, x^H)\) is

\[
\pi^H_i = \sum_{j \in S_i} q_j E_k[x^H_{jk}] = \gamma \sum_{j \in S_i} q_j E_k[x_{jk}]^{LP}
\]

and it follows that

\[
\pi_1^{LP} = \sum_{i=1}^{I} \sum_{j \in S_i} q_j E_k[x_{jk}]^{LP} = \frac{1}{\gamma} \sum_{i=1}^{I} \pi^H_i \leq \frac{I}{\gamma} \pi^{OPT}
\]

Observe, that the size of \(I\) may be bound using (12) with the following construction.

\[
1 \geq \sum_{j=1}^{n} r_j \geq \sum_{j=1}^{l_w} r_j = \sum_{i=1}^{I} \sum_{j \in S_i} r_j \geq (I - 1)\beta.
\]

This means that \(I \leq 1 + 1/\beta\) leading to the following bound

\[
\pi_2^{LP} \leq \frac{\beta + 1}{\beta \gamma} \pi^{OPT}.
\]

where \(\gamma\) is given by (13).

For bounding \(\pi_2^{LP}\) consider installing each subservice \(j = \ell_w + 1, \ldots, \ell\) individually. Note that from the definition of \(A\), and since \(r_j \geq w\) for \(j = \ell_w + 1, \ldots, \ell\),

\[
A = \sum_{j=1}^{\ell} r_j z_{j}^{LP} \geq \sum_{j=\ell_w+1}^{\ell} r_j z_{j}^{LP} \geq w \sum_{j=\ell_w+1}^{\ell} z_{j}^{LP}
\]

Thus,

\[
\sum_{j=\ell_w+1}^{\ell} z_{j}^{LP} \leq \frac{A}{w}.
\]

The solution obtained by installing just subservice \(j\) from among subservices \(\ell_w + 1, \ldots, \ell\) has an objective value of no more than \(q_j E_k[\delta_{jk}]\).

From the demand constraint it follows that \(E_k[x_{jk}^{LP}] \leq E_k[\delta_{jk}] z_j^{LP}\). By Assumption 1, this leads to the following bound.

\[
\pi_2^{LP} = \sum_{j=\ell_w+1}^{\ell} q_j E_k[x_{jk}^{LP}]\]
Combining (11), (16), and (17) gives

\[ \pi^{LP} \leq \left(1 + \frac{\beta + 1}{\beta \gamma} + \frac{A}{w}\right) \pi^{OPT} \]  

where \( \gamma \) is given by (13) and \( w, \beta \in (0, 1) \) may be chosen with \( w + \beta < 1 \). This leads to the following theorem.

**Theorem 5** Under the assumption that \( \sum_{j=1}^{n} r_j \leq s \)

\[ \pi^{LP} \leq (5 + 2\sqrt{3})\pi^{OPT}. \]

**PROOF.** Scaling the problem so that \( s = 1 \) leaves the ratio unchanged. The bound (18) is used.

The choice of \( w \) and \( \beta \) is based on the value of \( A \) (which depends on the LP solution). When \( A < \frac{1}{2} \) take \( w = 1 - \frac{1}{2}\sqrt{3} \) and \( \beta = -\frac{1}{2} + \frac{1}{2}\sqrt{3} \) and when \( A \geq \frac{1}{2} \) take \( w = \beta = \frac{1}{2}A \). For both cases \( w + \beta \geq A \) so that \( \gamma = \frac{1-\beta-w}{1-A} \). For the former case the bound (18) leads to

\[ \pi^{LP} \leq \left(1 + \frac{2(1+\sqrt{3})(1-A)}{-1+\sqrt{3}} + \frac{A}{1-\frac{1}{2}\sqrt{3}}\right) \pi^{OPT} \]

\[ = \left(1 + (1+\sqrt{3})^2(1-A) + 4(1+\frac{1}{2}\sqrt{3})A\right) \pi^{OPT} = (5 + 2\sqrt{3})\pi^{OPT} \]

while in the latter case (18) leads to the bound

\[ \pi^{LP} \leq \left(4 + \frac{2}{A}\right) \pi^{OPT} \leq 8\pi^{OPT} \leq (5 + 2\sqrt{3})\pi^{OPT}. \]

\[ \Box \]

We can show that in case \( A \leq \frac{1}{2} \) there is no better choice of \( w \) and \( \beta \) in this analysis. In case \( A > \frac{1}{2} \) a better choice of \( w \) and \( \beta \) seems possible though, so that in that case the analysis could lead to a slightly better constant bound.
3.2 A round and partition heuristic with constant worst-case ratio

Based on the previous LP bound a round and partition heuristic (RP) is developed with a worst case performance ratio bounded above by $5 + 2\sqrt{3}$.

Consider the class of heuristics that, given $S \subseteq \{1, \ldots, m\}$, produce the solution $(z^S, x^S)$ with objective value $\pi^S$, by setting $z^S_j = 1$ if $j \in S$ or $z^S_j = 0$ if $j \notin S$ and choosing $x^S$ to maximize the LP created by fixing $z$ to $z^S$ in (SSNP). Guided by the previous section, we will generate many such solutions by partitioning the set of services.

The two constants $w$ and $\beta$ of the previous subsection are chosen as in Theorem 5. That is, when $A < \frac{1}{2}$ choose $w = 1 - \frac{1}{2}\sqrt{3}$ and $\beta = -\frac{1}{2} + \frac{1}{2}\sqrt{3}$ and when $A \geq \frac{1}{2}$ choose $w = \beta = \frac{1}{2}A$. Regarding the remark following Theorem 5 in the previous subsection in case $A > \frac{1}{2}$ also here better choices of $w$ and $\beta$ seem possible.

Let $(z^{LP}, x^{LP})$ be a basic optimal LP relaxation solution with the optimal solution value given by (10). From this solution we generate a partition $\{W, Z, B, T_1, \ldots, T_K\}$ for some $K$ of $\{1, \ldots, m\}$.

- $W = \{j | z^{LP}_j = 1\}$
- $Z = \{j | z^{LP}_j = 0\}$
- $B = \{j | 0 < z^{LP}_j < 1, r_j > w\}$

The remaining subservices with $z^{LP}_j > 0$ and $r_j \leq w$ are partitioned into the sets $T_1, \ldots, T_K$ in the following way. Consider these subservices in arbitrary order. Start by filling the set $T_1$ with the first subservices until addition of the next subservice will raise the sum of the installation requirements above $w + \beta$. That subservice will be the first one to go into the set $T_2$. Continue in the same way filling the set $T_2$ and so on until the last set $T_K$ is constituted by the last few items. Thus, the sets $T_1, \ldots, T_K$ have the properties

- $\sum_{j \in T_i} r_j \in [\beta, \beta + w]$ for $i = 1, \ldots, K - 1$
- $\sum_{j \in T_K} r_j \leq \beta + w$

The partition generation takes $O(m)$ time once the LP solution is known.

The round and partition heuristic then chooses a solution $(x^S, z^S)$ where $S$ is: $W$, one of the sets $T_i$, or a single element of $B$. That is, the round and
partition heuristic solution, \((z^{RP}, x^{RP})\), is given by

\[
(z^{RP}, x^{RP}) = \operatorname{argmax} \left\{ \pi^S \mid S \in \{W, T_1, \ldots, T_K\} \cup \bigcup_{j \in B} \{ \{j\} \} \right\}.
\]

Let \(\pi^{RP}\) be the solution value of the round and partition heuristic.

**Theorem 6** The round and partition heuristic has a worst case performance ratio of

\[
\frac{\pi^{OPT}}{\pi^{RP}} \leq (5 + 2\sqrt{3}).
\]

**PROOF.** This follows almost immediately from the proof of the bound for the LP-relaxation in Section 3.1 taking the \(\{S_i\}_{i=1}^L\) as \(\{T_i\}_{i=1}^K\). The \(w\) and \(\beta\) values used above are the same as in the proof.

Notice that \(\pi^W = \pi^{LPR}\) and for any \(j \in B\) \(\pi^{(j)} = q_j E_k[\delta_{jk}]\). Also, for each \(i \in \{1, \ldots, K\}\) \(\pi^{T_i} = \pi^{H_i}\). With this, from the definition of the heuristic,

\[
\pi^{RP} \geq \pi^{LP}_0, \quad \pi^{RP} \geq \pi^{H_i}; \quad \forall i = 1, \ldots, K \quad \text{and} \quad \pi^{RP} \geq q_j E_k[\delta_{jk}]; \quad \forall j \in B.
\]

From this \(\pi^{OPT}\) may be replaced by \(\pi^{RP}\) in (14) and (17). Following this through to the proof of the LP bound in Theorem 5 gives

\[
\pi^{OPT} \leq \pi^{LP} \leq (5 + 2\sqrt{3})\pi^{RP}.
\]

\(\square\)

### 4 Computational complexity

This section gives evidence that the above results are interesting in the sense that one cannot hope to arrive at the optimal solution of (SSNP) in polynomial time. As indicated in the introduction the deterministic counterpart of the problem admits a fully polynomial approximation scheme for its solution. Here we show that this is unlikely to be achievable for (SSNP) by proving that it is strongly NP-hard.

**Theorem 7** The stochastic single node service provision problem is strongly NP-hard.
PROOF. The natural recognition version of this problem obtained by introducing a number and asking if there is a feasible solution giving profit at least that number is obviously in NP, since the representation of the probabilistic input in scenarios allows the formulation of a deterministic equivalent mixed-integer programming problem. To see that the recognition version is strongly NP-Complete consider a reduction from the well-known strongly NP-Complete vertex cover problem (see [4]):

Given a graph $G = (V, E)$ with $|V|$ vertices and $|E|$ edges and a constant $K$, does there exist a subset $V'$ of the vertices, such that each edge in $E$ is incident to at least one vertex in $V'$, and such that $|V'| \leq K$?

For every vertex introduce a subservice with installation requirement $\alpha = \frac{1}{\sqrt[|E|]}$. For every edge introduce a scenario with demand 1 for the two subservices incident to it and demand 0 for all other subservices. Let $q_j = |E|$ $\forall j$, and let all scenarios have a probability $\frac{1}{|E|}$ of occurring. Then the expected profit from meeting one unit of demand in a single scenario is 1. Take $K \alpha + 1$ as capacity of the node in (SSNP). The question is whether there is a solution to this instance of (SSNP) with total expected profit at least $|E|$.

This transformation is obviously polynomial. In case there exists a vertex cover of size at most $K$ then there is a service provision with total expected profit at least $|E|$. Install the subservices corresponding to the vertices in the vertex cover. Then for each scenario (edge) at least one of the subservices with demand 1 is installed. The total capacity used by the installation of the subservices is at most $K \alpha$ leaving at least capacity 1 to fill with the demands for each scenario.

The other direction is a bit more complicated. Suppose there does not exist a vertex cover of size $K$ or less. Then installing all subservices corresponding to a vertex cover would use node capacity strictly greater than $K \alpha$ leaving strictly less than 1 for meeting demand in each of the $|E|$ scenarios, making a total expected profit of at least $|E|$ unattainable. Installing any set of subservices of size $L < K$ would leave $(K - L) \alpha + 1$ node capacity for meeting demand in each scenario. However, at least one edge will remain uncovered, implying that there is at least one scenario in which both subservices with a positive demand are not installed. With at most $|E| - 1$ scenarios the expected profit will be at most $(|E| - 1)((K - L) \alpha + 1) \leq (|E| - 1)(K \alpha + 1) = (|E| - 1)(\frac{1}{|E|} + 1) < |E|$. □

In case the number of scenarios is fixed a dynamic programming algorithm shows that the problem can be solved in pseudo-polynomial time. We argued in the introduction that this problem is not only of academic interest, but reflects a plausible real-world situation. For this it is assumed that all problem
parameters are integers.

**Theorem 8** *The stochastic single node service provision problem with a fixed number of scenarios can be solved in pseudo-polynomial time.*

**PROOF.** Consider the following DP that has the subservices as its stages. A state, \( S \in \mathbb{Z}_+^m \), gives the capacity used in each scenario. Define \( f_j(S) \) as the maximum profit that can be achieved from scenario capacities \( S = (S_1, \ldots, S_m) \) using the subservices \( 1, \ldots, j \). Each \( S_k \) may take a value between 0 and \( s \) so there are at most \((s + 1)^m\) states per stage. There are two types of transitions in every stage, either the subservice is not installed, or it is installed and some demand is met. There are fewer than \( s + 1 \) possible choices concerning the demand to meet in each scenario, and overall there are then fewer than \((s + 1)^m\) different feasible decisions in a state. The initial settings are

\[
 f_0(S) = \begin{cases} 
 0 & \text{if } 0 \leq S_i \leq s, \quad \forall i = 1, \ldots, m \\
 -\infty & \text{otherwise.}
\end{cases}
\]

The recurrence is given by

\[
f_j(S_1, \ldots, S_m) = \max_{0 \leq x_k \leq \delta_{jk}} \left\{ f_{j-1}(S_1 - r_j - x_1, \ldots, S_m - r_j - x_m) + q_j \sum_{k=1}^{m} p_k x_k, f_{j-1}(S_1, \ldots, S_m) \right\}.
\]

From each state there are at most \((s + 1)^m + 1\) possible transitions, at each stage there are at most \((s + 1)^m\) states and there are \( n \) stages. The running time of the DP is therefore at most \( O(ns^{2m}) \), which implies the theorem. \( \Box \)

Thus, the conclusion is that the problem with a fixed number of scenarios is not strongly NP-hard. This suggests also the existence of a polynomial approximation scheme for the problem, a nice subject for future investigations. That this subclass of problems is still NP-hard is implied by the NP-hardness of the deterministic counterpart of the problem which has been proved in [2].
5 Conclusions

This paper considered a service provision problem on a distributed processing telecommunication network, under uncertain demand for the services. It was shown that the natural stochastic integer programming model is strongly NP-hard. It is worthwhile to stress this as its deterministic counterpart having the same number of binary decision variables is weakly NP-hard. Thus, the complexity of the problem increases by introducing stochasticity, even if it only means adding continuous decision variables for each scenario of the problem.

Because of the strong NP-hardness, approximation algorithms were studied for this problem. A first algorithm based on the LP relaxation of the deterministic equivalent of the stochastic problem has worst-case performance ratio equal to the minimum of the number of services and the number of scenarios that describe the stochastic demand plus one. The second algorithm has a constant worst-case performance ratio for a more restricted class of problems. The assumption defining this subclass is, however, satisfied for many reasonable practical problem situations.

Moreover, the variable bound on the performance ratio of the first algorithm is not as bad as it may seem at first sight because (as indicated in the introduction) the number of scenarios may actually be small in our telecommunication application. In a situation with a small number of scenarios one might alternatively think of using the dynamic programming formulation of Section 4. However, it should be noted that if precision is required and the resource capacity and the resource requirements are large then the pseudo-polynomial nature of the method leads to excessive computation times.

Of course it would be interesting to test the performance of the algorithms in this paper on real-life data. However, the telecommunication company has only recently started gathering data that may be used for this problem. Thus, empirical testing of the methods is postponed until enough data are available. Then it will also become clear if the methods are fast enough to be useful in a nearly on-line problem situation or whether faster methods, possibly giving worse performance guarantees, should be sought.

References


