7-2019

A scalable approach to joint cyber insurance and Security-as-a-Service provisioning in cloud computing

Jonathan David CHASE
Singapore Management University, jdchase@smu.edu.sg

Dusit NIYATO

Ping WANG

Sivadon CHAISIRI

Ryan K. L. KO

Published in IEEE Transactions on Dependable and Secure Computing, 2019, 16 (4), 565-579.
DOI: 10.1109/TDSC.2017.2703626

A Scalable Approach to Joint Cyber Insurance and Security-as-a-Service Provisioning in Cloud Computing

Jonathan Chase, Dusit Niyato, Ping Wang, Sivadon Chaisiri, Ryan K L Ko

Abstract—As computing services are increasingly cloud-based, corporations are investing in cloud-based security measures. The Security-as-a-Service (SECaaS) paradigm allows customers to outsource security to the cloud, through the payment of a subscription fee. However, no security system is bulletproof, and even one successful attack can result in the loss of data and revenue worth millions of dollars. To guard against this eventuality, customers may also purchase cyber insurance to receive recompense in the case of loss. To achieve cost effectiveness, it is necessary to balance provisioning of security and insurance, even when future costs and risks are uncertain. To this end, we introduce a stochastic optimization model to optimally provision security and insurance services in the cloud. Since the model we design is a mixed integer problem, we also introduce a partial Lagrange multiplier algorithm that takes advantage of the total unimodularity property to find the solution in polynomial time. We also apply sensitivity analysis to find the exact tolerance of decision variables to parameter changes. We show the effectiveness of these techniques using numerical results based on real attack data to demonstrate a realistic testing environment, and find that security and insurance are interdependent.

Index Terms—Cloud computing, cyber insurance, security as a service, partial Lagrange multiplier method, sensitivity analysis.

1 Introduction

In a 2014 report by McAfee it was estimated that financial losses due to cyber risks were between USD 300 billion and USD 1 trillion a year [1]. The Identity Theft Resource Center's 2016 data breach category summary found that as of November, there were 873 recorded breaches in the US with over 29 million records exposed [2]. With the number of cyber attacks growing, successful attacks are now a question of 'when', not 'if'. Yet, the Monetary Authority of Singapore observed that cyber insurance adoption for small and medium-sized enterprises (SMEs) is less than 10% in Asia, despite 42% of the world's Internet users living in the region [3]. 90% of all cyber insurance is purchased by US companies, whilst in the UK, only 2% of companies have specialist cyber insurance. The market, therefore, has considerable room for further growth, with PricewaterhouseCoopers (PwC) estimating that annual premiums could grow USD 5 billion by 2018 and exceed USD 7.5 billion by 2020 [4].

Outsourcing computation to the cloud is now common practice, and it is not surprising that the Security-as-a-Service (SECaaS) paradigm has arisen to counter the growing level of cyber threats. Thus, an application may guard against attacks by provisioning security services from providers such as McAfee [5] and Trend Micro [6]. These services may take various forms, such as secure data storage, identity and access management (IAM), and intrusion detection services to screen incoming traffic [7]. These services can be provisioned in a similar manner to other cloud services, either through advance subscription or dynamically through on-demand options. SECaaS allows customers to reduce their security overheads while maintaining a high level of protection.

Despite the variety of security options available, it is inevitable that they will eventually be circumvented. Cyber insurance is used to provide explicit cover in the event that malicious activity leads to financial loss. Insurance coverage may be first- or third-party with first-party insurance covering eventualities such as theft of money and digital assets, business interruption, and cyber extortion. Third-party insurance may cover problems such as privacy breaches, loss of third-party data (e.g. user account information), and public relations expenses [8]. Major insurers, such as Allianz [8] or QBE [9] offer cyber insurance policies that cover a range of first- and third-party risk. Cyber insurance is an important and growing field, but carries some unique features that makes it challenging. For example, in 2013, millions of credit card details were stolen from Target [10]. The extent of a cyber attack can be difficult to assess, and Target revised its estimate of stolen user records from 40 million to over 70 million. Additionally, the costs of the attack were not measured in the literal value of the data stolen, but must be calculated from a combination of factors, such as damage to business (sales dropped 3-4% over the previous year), settlement costs with credit card companies, management of public relations damage, and legal fees. By 2016, costs directly attributable to the breach neared USD 300 million, but the damages were offset by USD 90 million in cyber insurance payouts [11]. Therefore, it is clear that

J. Chase, D. Niyato, and P. Wang are with the School of Computer Science and Engineering, Nanyang Technological University. S. Chaisiri and R. K. L. Ko are with CROW, University of Waikato.

This research was supported by STRATUS (Security Technologies Returning Accountability, Trust and User-centric Services in the Cloud) (https://stratus.org.nz), a science investment project funded by the New Zealand Ministry of Business, Innovation and Employment (MBIE).

for the majority of companies, whose data breaches will be of lesser magnitude, cyber insurance is a crucial addition to their security plans.

As the Target case shows, damages are costly, indemnity payouts can be expensive, and it is in an insurance company's best interests to encourage customers to also provision adequate security, rather than only buying cyber insurance. [12] introduced a combined security and cyber insurance provisioning scheme that balances the need of a customer to invest in security to reduce the likelihood of a successful cyber attack, whilst also acknowledging the inevitability of cyber damage. The paper employed stochastic optimization to account for the uncertainty of attacks to avoid both overprovisioning and underprovisioning of services. We significantly extend the work in that paper to consider multiple time periods and uncertain costs of security and insurance premiums. We also improve the scalability of the solution through a partial Lagrange multiplier method, perform an analytical sensitivity analysis, and base our result on parameters derived from real data. Our contributions are summarized as follows:

- We devise a stochastic optimization for a customer to jointly provision security services and buy cyber insurance premiums across multiple time periods. We account for uncertainty in traffic quantities and attack frequency, as well as future uncertainty of security service prices and insurance premiums.
- Due to the tractability problems of integer programming, we introduce a partial Lagrange multiplier algorithm to find the optimal solution in, at worst, polynomial time. We provide proofs of convergence and scalability.
- We perform a sensitivity analysis, which provides precise values for solution tolerance to parameter change. We then demonstrate the effectiveness of our methods through evaluation of an example scenario, based on analysis of real attack data to provide realistic parameter settings.

The paper is organised as follows: Section 2 outlines the related work, giving background and context on cyber insurance and SECaaS provisioning. Section 3 outlines the system model for our combined security and insurance environment, and the stochastic optimization formulation is provided in Section 4. To improve the runtime performance of our solution, we introduce the partial Lagrange multiplier algorithm in Section 5, and give proofs of its performance in Section 6. The technique used in our sensitivity analysis is presented in Section 7. We give comprehensive practical results in Section 8 demonstrating the effectiveness of our ideas.

2 Related Work

There are two aspects to the system model which we propose in this paper. The first is the problem of security service allocation, and the second is cyber insurance provisioning. Research in this area primarily addresses the problem of security allocation, the setting of cyber insurance parameters, and whether there is a symbiotic relationship between Internet security and cyber insurance.

The notion of Security-as-a-Service (SECaaS) was introduced in [13], where it was proposed as a way of securing cloud-based data through encryption and distribution of data. [14] addresses the problem of selecting cloud service providers (CSP) with security considerations as a priority. The authors propose a framework to manage risk through a combination of technology, processes, and people. [15] considers allocation of resources in a parallel computing context with the security overhead considered for both heterogeneous and homogeneous systems. [16] similarly divides security services by priority to optimize processing requirements in a mobile cloud context. Considering real-time systems that are security-critical, [17] provisions security services to optimize performance, where the scheduling of jobs is combined with the allocation of security services. [18] takes an approach that is both security-aware and budget-aware.

[19] introduces the idea of firewall-style SECaaS providers, similarly [20] introduces an API called FlowTap to provide a security policy enforcement and monitoring infrastructure for network traffic. This is shown to be important as, whilst users can install security software in a virtual environment, they have no control over the network traffic in the cloud. We consider this approach for the SECaaS providers in this paper, which focuses on network traffic analysis.

The field of cyber insurance contains a number of unique problems compared to other fields of insurance. Information asymmetry is a particular problem, where companies are reluctant to share full details of their security provisioning with insurers [21]. Further, the frequency of breaches is difficult to predict [22] and security systems are often interdependent, making it difficult to assess the vulnerability of a system [23]. This correlation extends to different insured entities, where the vulnerability of one client may impact another, due to the homogeneity of modern computer systems and lack of geographical variation. Thus, a vulnerability in one client's software package installation may be mirrored in many others, and may allow the infection of another client through the first's system. A scarcity of data compounds the problem, making it challenging to accurately assess the risk level and determine the correctness of that assessment [24]. Unlike other fields of insurance, it is even challenging to determine when an attack has actually taken place, as many are lengthy and leave liability unclear [25]. The quantification of damage is difficult to define, due to the intangibility of losses. For the sake of this paper, we make assumptions that the probability distribution of risk is known, and damages incurred can be correctly assessed. This area of cyber insurance requires further work to develop that is beyond the scope of this paper.

Cyber insurance is primarily concerned with the problem of risk transfer - where the cost from risk is transferred to the insurer. However, risk management and assessment is also a notable part of the literature, attempting to address the issues described above. Risk management is achieved through the establishment of methodologies and best practices that are adapted from normal business practices, with consideration for IT-specific measures [26]. ISO/IEC 27001 is an example of a standard for IT security risk management [27], showing that solutions are primarily based on

business practices, rather than technology. An important aspect of risk transfer in cyber insurance is the specification of the premium. The research in this area is primarily divided between independent security and interdependent security. In this paper we focus on the independent case, where clients are independent of one another. The interdependent case is more complex and is beyond the scope of this paper.

[28] presents a mathematical model and experimental results to show that cyber insurance encourages good security practices, one of the key aspects of cyber insurance. In many cases, there are both security-related and non-security-related IT risks, and these are not always easy to separate. [29] proposes a model of cyber insurance called Aegis where users choose to take a proportion of liability on themselves. When insurance is compulsory, this is shown to be preferable. [30] looks at non-life cyber insurance, that is, insurance that covers repeated events rather than insuring against a single event, such as a major cyber attack. Using a Markov chain to model attack scenarios, the probability of damages/survival can be established, and premiums are set accordingly. [12] considers the problem of allocating non-life insurance alongside SECaaS services, with a stochastic optimization model finding the optimum allocation of security and insurance as a joint decision to minimize the customer's cost. We use this model as the basis for the work in this paper, in which we focus on non-life insurance in an independent-security context. We must assume that the liabilities can be determined, and that the probability distribution for attacks is known. Since real cyber attacks cannot be assumed to conform to typical probability distributions, we require a realistic custom distribution, and so our numerical results are based on parameters derived from an analysis of cyber attack data. Whilst it can be argued that security-conscious system design and business processes are the most effective defence against cyber threats, the rapidly changing nature of cyber attacks means that traditional preventative measures are still necessary. Further, there is precedent for the use of statistical analysis in protection, for example in IDS (Intrusion Detection System) implementations to identify attack behaviours [31] [32]. In this paper we aim to demonstrate a connection between cyber insurance and SECaaS provisioning, as properly designed insurance contracts can incentivize security investment. [33] shows that cyber insurance does not intrinsically encourage security investment, but in this paper we add the stipulation that if packets are not assessed by security services, they are not liable for insurance coverage, and demonstrate that in this case, cyber insurance encourages investment in security. Whilst this is a strong assumption, it is a reasonable and practical one, as, at the very least, network traffic should pass through a firewall, and it is common practice for insurance companies to add exclusions to policies to protect themselves from reckless clients.

3 System Model

We consider a SECaaS framework similar to the one used in [12], containing a customer who uses applications, which receive Internet traffic in the form of packets. These packets are scanned by services from SECaaS providers, provisioned by a subscription management process (SMP). In the event that harmful packets elude security, cyber insurers, subscribed to by an insurance management process (IMP), provide compensation for damages incurred. In this section we outline the components of the system as illustrated in Fig. 1. The notations for the system model elements are given in Table 1.

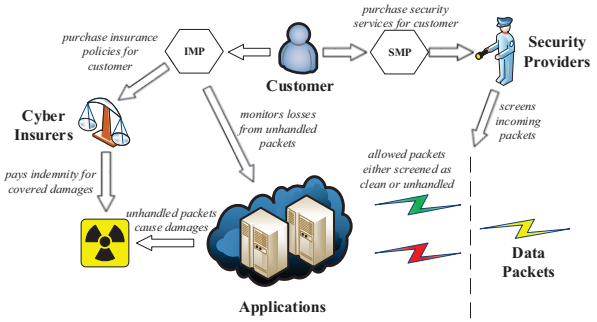


Fig. 1: Illustration of interaction between system components.

3.1 Security-as-a-Service Model

In our security model, a customer runs applications (denoted by set $\mathcal{A}$) that we assume to be Internet-accessible, either on a cloud service such as Amazon EC2, or in-house servers managed by the customer. Applications receive data packets in accordance with their operating purpose, e.g. email data or financial transactions. Legitimate packets are called safe packets, while packets used in cyber attacks are called unsafe packets. Unsafe packets are deemed handled if they are correctly detected by security services, or unhandled if they are not successfully processed (for example if they are undetected). These unhandled packets will cause damage, which incurs costs to the customer. In reality, security threats may take various forms, and malicious packets may represent various types of threat, including malware and backdoor exploits, as well as the damages caused being manifested in various ways (such as business interruption caused by malware, data breach caused by backdoor exploitation). Since this paper examines the relationship between SECaaS and cyber insurance, we focus on those threats that can be represented by malicious packets. However, the model may be extended and applied to consider other security flaws and the probability of their exploitation, as well as the nature of the damage caused, and the duration of the damage. This extension is left for future work. Security services are provisioned and charged on a per-packet basis. Security providers (denoted by set $\mathcal{S}$) can offer customers prepaid plans tailored to each application (denoted by set $\mathcal{P}_{as}$) which have a particular per-packet price, duration, and maximum number of packets. Prepaid services are provisioned in advance, with services utilized according to the incoming traffic at the present time. In the event that the prepaid services are not sufficient, further services can be provisioned under an on-demand pricing scheme. On-demand allocation can be done dynamically, at a per-packet usage price that is higher than that of the prepaid scheme.

3.2 Cyber Insurance Model

Customers can purchase cyber insurance products in the form of insurance policies (denoted by set $\mathcal{L}$) from insurance companies (denoted by set $\mathcal{I}$) that compensate them for incidents and disasters such as data breach, data corruption, and business interruption. The IMP purchases insurance policies, which includes the premium, types of risks covered, indemnity value, and policy duration. These insurance policies can be packaged with security services, to cover damages caused by undetected unsafe packets. The price for insurance purchased in advance is charged at a rate known as a 'present premium'. However, the insurer may adjust charges for the same policy at a future time, which is obtained in an on-demand fashion, known as a 'future premium'. Insurance covers an application and the coverage of an application can only be purchased from one insurer at a given time. Further, we assume that damages can be accurately determined. This may be in various forms, such as a 'ransom' paid [34], or the cost of legal fees and court settlements, as in the case of Target's 2013 data breach [11]. We assume that both first- and third-party damages are covered by the insurance, and that each unhandled packet for an application has the same damage cost and insurance coverage. These assumptions simplify the model with removal of each assumption left for future work.

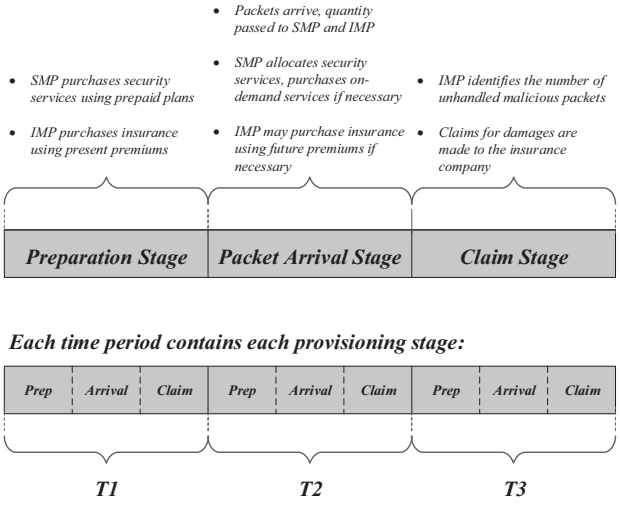


Fig. 2: In each time period, there are three provisioning stages. This figure illustrates an example case with 3 time periods, however this can be freely expanded to n time periods.

3.3 Decision Stages

Our system model takes place across multiple time periods (e.g. one hour or one day), as illustrated in Fig. 2. The set of all time periods is defined as $\mathcal{T} = \{t_1, t_2, t_3, \ldots, t_n\}$, where n denotes the number of time periods. Security service reservation and cyber insurance policies may last for multiple time periods. For example, a user may purchase prepaid security services for two time periods, allowing them to use those services in the following time period at no additional cost. Let t denote the current time period, then $\mathcal{F}_{pt}$ denotes the set of previous time periods (including the current time period) in which security plan p could have been purchased and still be active. Let $|p|$ denote the length of plan p, then $\mathcal{F}_{pt} = \{\max(1, t-|p|+1), \ldots, t\}$. $\mathcal{F}_{lt}$ denotes the equivalent set for insurance policies, where l denotes the policy. Each individual time period consists of three decision stages. The first stage is the 'preparation stage'. In this stage, the SMP and IMP make initial provisioning choices to provision prepaid security services and buy insurance policies using the present premium. The second stage is the 'packet arrival stage', where the number of incoming packets is realized. The number of required prepaid services and any additional on-demand services are provisioned, and additional insurance can be purchased using the future premium. The third and final stage is the 'claim stage', where the number of unhandled packets is realized and the damage costs and per-packet indemnity are calculated. It is assumed that insurance claims for unhandled packets are approved and there are no attempts to defraud the insurer.

3.4 Uncertainty Parameters

Our model contains a number of uncertain parameters, which are not known in advance. For example in the preparation stage, the number of arriving packets at an application, and the corresponding number of unhandled packets, are unknown, and are represented by random parameters. Likewise, insurance premium costs and SECaaS prices may be uncertain, as they can be adjusted by providers, so these are also treated as random parameters (since the pricing strategy is opaque to the customer). In the preparation stage, the prepaid security and present premium prices are known. In the packet arrival stage, the number of arriving packets and the future premiums are known. In the claim stage, the number of unhandled packets is known.

An unrealized circumstance is known as a scenario, which is called a realization when it occurs at a particular time. Let $\Omega^{\dagger}$ represent the set of scenarios in the preparation stage. Then, given realization $\omega^{\dagger}$, $\Omega^{\ddagger}(\omega^{\dagger})$ represents the set of possible scenarios in the packet arrival stage. Likewise, given the pair of realizations $(\omega^{\dagger}, \omega^{\ddagger})$, $\Omega^{\ddagger\ddagger}(\omega^{\dagger}, \omega^{\ddagger})$ represents the set of possible scenarios in the claim stage. The terms $\pi_{\omega^{\dagger}}$, $\pi_{\omega^{\ddagger}}$, and $\pi_{\omega^{\ddagger\ddagger}}$ represent the corresponding probabilities of the scenarios, and scenario sets of each provisioning stage have finite support and have probabilities summing to 1. The scenario probabilities of a time period t are independent from those of other time periods, so the probabilities of scenarios in period t are not affected by events in period $t-1$.

3.5 Decision Variables

The solution of a stochastic optimization formulation is known as a decision, and is represented by a set of values assigned to the decision variables.

3.5.1 Preparation Stage Variables

The following are the decision variables for the preparation stage. (1) defines the domain of prepaid security services, where $X^{\dagger}_{aspt\omega^{\dagger}}$ denotes prepaid services from provider s,

covering application a under security plan p, and (2) defines the domain of $W^{\dagger}_{asiltω^{\dagger}}$, which denotes the binary decision to buy insurance policy l using the present premium from insurer i to cover application a as protected by security provider s. These are evaluated over time stage t, and therefore must be considered for each scenario $\omega^{\dagger}$.

$$X^{\dagger}_{asptω^{\dagger}} \in \mathbb{N}_0, \forall a \in \mathcal{A}, s \in \mathcal{S}, p \in \mathcal{P}_{as}, t \in \mathcal{T}, \omega^{\dagger} \in \Omega^{\dagger}, \tag{1}$$

$$W^{\dagger}_{asiltω^{\dagger}} \in \{0,1\}, \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, l \in \mathcal{L}, t \in \mathcal{T}, \omega^{\dagger} \in \Omega^{\dagger}. \tag{2}$$

3.5.2 Packet Arrival Stage Variables

The variables for the packet arrival stage are defined as follows:

$$W^{\ddagger}_{asiltω^{\ddagger}} \in \{0,1\}, \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, l \in \mathcal{L}, t \in \mathcal{T}, \omega^{\ddagger} \in \Omega^{\ddagger}(\omega^{\dagger}), \tag{3}$$

$$Y_{asptω^{\ddagger}} \in \mathbb{N}_0, \forall a \in \mathcal{A}, s \in \mathcal{S}, p \in \mathcal{P}_{as}, t \in \mathcal{T}, \omega^{\ddagger} \in \Omega^{\ddagger}(\omega^{\dagger}), \tag{4}$$

$$Z_{astω^{\ddagger}} \in \mathbb{N}_0, \forall a \in \mathcal{A}, s \in \mathcal{S}, t \in \mathcal{T}, \omega^{\ddagger} \in \Omega^{\ddagger}(\omega^{\dagger}). \tag{5}$$

$W^{\ddagger}_{asiltω^{\ddagger}}$ in (3) represents the choice to purchase insurance using the future premium, as an alternative to $W^{\dagger}_{asiltω^{\dagger}}$, subject to scenario $\omega^{\ddagger}$ in time t. $Y_{asptω^{\ddagger}}$ in (4) indicates the number of prepaid services to be executed and $Z_{astω^{\ddagger}}$ in (5) is the number of on-demand services that additionally need to be provisioned to cover any shortfall.

3.5.3 Claim Variables

The variables for the claim stage are defined as follows:

$$C^{ω^{\ddagger}}_{asitω^{\ddagger\ddagger}} \in \mathbb{N}_0, \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, t \in \mathcal{T}, \omega^{\ddagger} \in \Omega^{\ddagger}, \omega^{\ddagger\ddagger} \in \Omega^{\ddagger\ddagger}(\omega^{\dagger}, \omega^{\ddagger}), \tag{6}$$

$$C^{\ddagger ω^{\ddagger}}_{asitω^{\ddagger\ddagger}} \in \mathbb{N}_0, \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, t \in \mathcal{T}, \omega^{\ddagger} \in \Omega^{\ddagger}, \omega^{\ddagger\ddagger} \in \Omega^{\ddagger\ddagger}(\omega^{\dagger}, \omega^{\ddagger}). \tag{7}$$

$C^{ω^{\ddagger}}_{asitω^{\ddagger\ddagger}}$, whose domain is defined in (6), is the number of unhandled packets claimed for that have insurance coverage paid for by a present premium whilst $C^{\ddagger ω^{\ddagger}}_{asitω^{\ddagger\ddagger}}$, defined in (7), is the number of unhandled packets claimed for that have coverage paid for by a future premium. This is under the realization of unhandled packets in scenario $\omega^{\ddagger\ddagger} \in \Omega^{\ddagger\ddagger}(\omega^{\dagger}, \omega^{\ddagger})$.

4 Problem Formulation

The stochastic optimization problem can be represented as a nested formulation with multi-stage recourse. The solutions for both the IMP and the SMP can be obtained using the following sequence of equations:

$$\min \mathbb{E}_{\Omega^{\dagger}}\Big[\mathscr{Q}^{\dagger}(X^{\dagger}, W^{\dagger}, \omega^{\dagger}) + \mathbb{E}_{\Omega^{\ddagger}}\Big[\mathscr{Q}^{\ddagger}(Y, Z, W^{\ddagger}, \omega^{\dagger}, \omega^{\ddagger}) + \mathbb{E}_{\Omega^{\ddagger\ddagger}}[\mathscr{Q}^{\ddagger\ddagger}(C, C^{\ddagger}, \omega^{\dagger}, \omega^{\ddagger}, \omega^{\ddagger\ddagger})]\Big]\Big]. \tag{8}$$

The objective function in (8) minimizes the total cost of both security service and insurance allocation across all time periods in the set $\mathcal{T}$, and thus it is only necessary to solve the optimization once for all $t \in \mathcal{T}$. The problem formulation is nested, as the expected value of each decision stage is affected by the expected value of the following stages. For example, the expected value of the packet arrival stage is determined, not only by the packet arrival stage variables, but also by the expected value of the claim stage. Therefore, the decisions made in the preparation stage and packet arrival stage are made in consideration of the effect those decisions will have on the expected value of the subsequent stages. This allows decisions to be made in the present that will have the best outcome in the future. The function $\mathscr{Q}^{\dagger}(\cdot)$ is an optimization problem to minimize the cost of the preparation stage across all time periods, and is defined as follows:

$$\mathscr{Q}^{\dagger}(X^{\dagger}, W^{\dagger}, \omega^{\dagger}) = \min\Big[\mathscr{R}^{\dagger}(\omega^{\dagger})\Big] \tag{9}$$

$$\text{where: } \mathscr{R}^{\dagger}(\omega^{\dagger}) = \sum_{a\in\mathcal{A}}\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\Bigg(\sum_{p\in\mathcal{P}_{as}} k^{\dagger}_{asptω^{\dagger}} X^{\dagger}_{asptω^{\dagger}} + \sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} f^{\dagger}_{asiltω^{\dagger}} W^{\dagger}_{asiltω^{\dagger}}\Bigg), \tag{10}$$

$$\text{s.t. } (1), (2),$$

$$\sum_{p\in\mathcal{P}_{as}} X^{\dagger}_{asptω^{\dagger}} \le e_{asp}, \forall a \in \mathcal{A}, s \in \mathcal{S}, t \in \mathcal{T}, \tag{11}$$

$$\sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} W^{\dagger}_{asiltω^{\dagger}} \le 1, \forall a \in \mathcal{A}, s \in \mathcal{S}, t \in \mathcal{T}. \tag{12}$$

The objective function in (9) is to minimize the value of the function $\mathscr{R}^{\dagger}(\cdot)$, which denotes the cost of the preparation stage in all time periods, where $k^{\dagger}_{asptω^{\dagger}}$ gives the prepaid security service cost, and $f^{\dagger}_{asiltω^{\dagger}}$ is the present insurance premium price. The constraint in (11) ensures that the number of prepaid services does not exceed the packet limit of provider s, for plan p, denoted by e_{asp}. Similarly, the constraint in (12) ensures that each application is only covered by one insurer at a time. Given fixed values of the decision variables, $X^{\dagger}_{asptω^{\dagger}}$ and $W^{\dagger}_{asiltω^{\dagger}}$, we optimize the cost of the packet arrival stage as follows:

$$\mathscr{Q}^{\ddagger}(Y, Z, W^{\ddagger}, \omega^{\dagger}, \omega^{\ddagger}) = \min\Big[\mathscr{R}^{\ddagger}(\omega^{\dagger}, \omega^{\ddagger})\Big] \tag{13}$$

$$\text{where: } \mathscr{R}^{\ddagger}(\omega^{\dagger}, \omega^{\ddagger}) = \sum_{a\in\mathcal{A}}\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\Bigg(k^{\ddagger}_{astω^{\ddagger}} Z_{astω^{\ddagger}} + \sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} f^{\ddagger}_{asiltω^{\ddagger}} W^{\ddagger}_{asiltω^{\ddagger}}\Bigg), \tag{14}$$

$$\text{s.t. } (3) - (5),$$

$$Y_{asptω^{\ddagger}} \le \sum_{\hat{t}\in\mathcal{F}_{pt}} X^{\dagger}_{aspt\hat{ω}^{\dagger}}, \forall a \in \mathcal{A}, s \in \mathcal{S}, p \in \mathcal{P}_{as}, t \in \mathcal{T}, \tag{15}$$

$$n_{aω^{\ddagger}} \le \sum_{s\in\mathcal{S}}\Bigg(Z_{astω^{\ddagger}} + \sum_{p\in\mathcal{P}_{as}} Y_{asptω^{\ddagger}}\Bigg), \forall a \in \mathcal{A}, t \in \mathcal{T}, \tag{16}$$

TABLE 1: List of Key Notations.

Symbol	Definition
$\mathcal{A}$	Set of all user applications, while $a \in \mathcal{A}$ denotes the application index.
$\mathcal{S}$	Set of all SECaaS providers, while $s \in \mathcal{S}$ denotes the provider index.
$\mathcal{T}$	Set of all time periods, while $t \in \mathcal{T}$ denotes the time period index.
$\mathcal{P}_{as}$	Set of all security plans, while $p \in \mathcal{P}_{as}$ denotes the plan index.
$\mathcal{I}$	Set of all cyber insurers, while $i \in \mathcal{I}$ denotes the insurer index.
$\mathcal{L}$	Set of all insurer policies while, $l \in \mathcal{L}$ denotes the policy index.
$\Omega^\dagger, \Omega^\ddagger(\omega^\dagger), \Omega^{\ddagger\ddagger}(\omega^\dagger, \omega^\ddagger)$	Set of all scenarios for preparation stage, packet arrival stage, and claim stage with each set dependent on the scenario choice from the previous stage, while $\omega^\dagger, \omega^\ddagger, \omega^{\ddagger\ddagger}$ are the corresponding scenario indices.
$k^\dagger_{asp t\omega^\dagger}, k^\ddagger_{ast\omega^\ddagger}$	The cost of prepaid security services and on-demand under scenarios $\omega^\dagger$, and $\omega^\ddagger$, respectively.
$f^\dagger_{asilt\omega^\dagger}, f^\ddagger_{asilt\omega^\ddagger}$	Present insurance premium and future premium under scenarios $\omega^\dagger$, and $\omega^\ddagger$, respectively.
e_{asp}	Limit of prepaid packets for application a set by provider s under plan p.
$n_{a\omega^\ddagger}$	The total number of packets arriving at application a in scenario $\omega_\ddagger$.
u_a	The damages cost caused per packet .
$m^{\omega^\ddagger}_{as\omega^{\ddagger\ddagger}}$	The number of packets unhandled by provider s arriving at application a.
h_{asi}	The indemnity payable for unhandled packets by insurer i.
g_{asi}	The number of packets covered by insurer i for security service s.
$X^\dagger_{asp t\omega^\dagger}$	Decision variable for prepaid security services in the preparation stage.
$W^\dagger_{asilt\omega^\dagger}, W^\ddagger_{asilt\omega^\ddagger}$	Decision variables for present and future insurance premiums, respectively.
$Y_{asp t\omega^\ddagger}, Z_{ast\omega^\ddagger}$	Decision variables for security services to be utilized under the prepaid and on-demand schemes, respectively.
$C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}}, C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}}$	Decision variables for number of packets claimed for under present and future premiums, respectively.

$$\sum_{l\in\mathcal{L}}(W^\dagger_{asilt\omega^\dagger} + W^\ddagger_{asilt\omega^\ddagger}) \leq 1, \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, t \in \mathcal{T}, \tag{17}$$

$$\sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} W^\ddagger_{asilt\omega^\ddagger} \leq 1, \forall a \in \mathcal{A}, s \in \mathcal{S}, t \in \mathcal{T}. \tag{18}$$

The optimization problem in (14) minimizes the cost of the packet arrival stage. $k^\ddagger_{ast\omega^\ddagger}$ gives the on-demand security cost and $f^\ddagger_{asilt\omega^\ddagger}$ is the future insurance premium cost. The constraint in (15) ensures that utilized security does not exceed the prepaid provisioning. The constraint in (16) ensures that the mixture of prepaid and on-demand security services are sufficient to cover incoming packets $n_{a\omega^\ddagger}$. The constraints in (17) and (18) are similar to that in (12), ensuring that only one insurer is bought from under present or future premiums per application. Finally, we also minimize the cost of the claim stage. Assuming fixed values of the decision variables $W^\ddagger_{asilt\omega^\ddagger}$, $Y_{asp t\omega^\ddagger}$ and $Z_{ast\omega^\ddagger}$, we can define the function $\mathscr{Q}^{\ddagger\ddagger}(\cdot)$ as a minimization problem in the following way:

$$\mathscr{Q}^{\ddagger\ddagger}(C, C^\ddagger, \omega^\dagger, \omega^\ddagger, \omega^{\ddagger\ddagger}) = \min\left[\mathscr{R}^{\ddagger\ddagger}(\omega^\dagger, \omega^\ddagger, \omega^{\ddagger\ddagger})\right] \tag{19}$$

$$\text{where: } \mathscr{R}^{\ddagger\ddagger}(\omega^\dagger, \omega^\ddagger, \omega^{\ddagger\ddagger}) = \sum_{a\in\mathcal{A}}\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\left(u_a m^{\omega^\ddagger}_{as\omega^{\ddagger\ddagger}} - \sum_{i\in\mathcal{I}} h_{asi}\left(C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} + C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}}\right)\right), \tag{20}$$

$$\text{s.t. } (6)-(7),$$

$$C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} \leq \sum_{l\in\mathcal{L}}\sum_{\hat{t}\in\mathcal{F}_{lt}} g_{asi} W^\dagger_{asil\hat{t}\omega^\dagger}, \quad \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, t \in \mathcal{T}, \tag{21}$$

$$C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} \leq \sum_{l\in\mathcal{L}}\sum_{\hat{t}\in\mathcal{F}_{lt}} g_{asi} W^\ddagger_{asil\hat{t}\omega^\ddagger}, \quad \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, t \in \mathcal{T}, \tag{22}$$

$$C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} + C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} \leq Z_{ast\omega^\ddagger} + \sum_{p\in\mathcal{P}_{as}} Y_{asp t\omega^\ddagger}, \quad \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, t \in \mathcal{T}, \tag{23}$$

$$C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} + C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} \leq m^{\omega^\ddagger}_{as\omega^{\ddagger\ddagger}}, \forall a \in \mathcal{A}, s \in \mathcal{S}, i \in \mathcal{I}, t \in \mathcal{T}. \tag{24}$$

The damages per packet are denoted by u_a, with the number of penetrating packets given by $m^{\omega^\ddagger}_{as\omega^{\ddagger\ddagger}}$. This cost is offset by h_{asi} the per-packet indemnity. The claims are governed by the constraints in (21) and (22), which ensure that the numbers of packets claimed do not exceed those covered by the insurer, who can insure number of packets, g_{asi}, per policy. The constraint in (23) ensures that the packets were processed by security (a requirement for the insurance claims). The constraint in (24) ensures that they are not greater than the number of actual malicious packets.

Having defined the problem, we must now solve it. When a stochastic optimization problem has finite support, it is possible to create a deterministic equivalent formulation, which can be solved as a mixed integer problem using an established technique, such as the branch-and-bound algorithm (B&B) [35]. In our case, it is reasonable to assume that this criterion holds, and so the following equation provides the equivalent objective function:

$$\min \sum_{\omega^\dagger\in\Omega^\dagger} \pi_{\omega^\dagger}\left(\mathscr{R}^\dagger(\omega^\dagger) + \sum_{\omega^\ddagger\in\Omega^\ddagger(\omega^\dagger)} \pi_{\omega^\ddagger}\left(\mathscr{R}^\ddagger(\omega^\dagger, \omega^\ddagger) + \sum_{\omega^{\ddagger\ddagger}\in\Omega^{\ddagger\ddagger}(\omega^\dagger,\omega^\ddagger)} \pi_{\omega^{\ddagger\ddagger}}\mathscr{R}^{\ddagger\ddagger}(\omega^\dagger, \omega^\ddagger, \omega^{\ddagger\ddagger})\right)\right). \tag{25}$$

Given that the only modification required for the constraints is that they are evaluated over the scenario sets, we include them in Appendix A, due to space limits. We can therefore find the optimal solution to our problem, although in real world terms, the optimality of the solution is dependent on the quality of the chosen parameters and the accuracy with which the model reflects reality.

5 Partial Lagrange Multiplier Method

The mixed integer programming deterministic equivalent formulation is an effective way of finding the optimum solution to the stochastic optimization, but suffers from tractability problems, as we would normally use a technique such as branch-and-bound, which has exponential computational complexity [35]. However, if we can convert our problem to a linear program (LP) without loss of precision, we can solve it in polynomial time. In this section we introduce the partial Lagrange multiplier method [36], which solves a sequence of LPs with at worst a polynomial number of steps.

5.1 Total Unimodularity of Constraint Matrix

To solve the tractability problem, we take advantage of the Total Unimodularity (TU) property [37]. If the set of constraints in (11)-(24) are considered as a matrix $\boldsymbol{M}$ and vector $\vec{\boldsymbol{b}}$, as in standard LP notation (we give an example of standard notation in (22)), then if $\vec{\boldsymbol{b}}$ is all-integer (which in our case it is), and $\boldsymbol{M}$ satisfies TU, we can relax all integer variables to linear ones and solve as an LP in polynomial time while guaranteeing integer variable solutions.

In our case, however, $\boldsymbol{M}$ only partially satisfies TU, which we can establish according to Theorem 3 in [38]. The matrix $\boldsymbol{M}$ is trimmed by removing the constraints that obviously violate TU, for example, those with coefficients not equal to -1, 0, or 1. The resultant matrix, $\boldsymbol{M'}$ can then be divided into two disjoint subsets, one containing the rows corresponding to constraints (12), (15), (17), and (24), and the other containing (11), (16), and (18). It is clear from these constraints that the conditions for unimodularity given in [38] are satisfied for the matrix, $\boldsymbol{M'}$, and that it will also hold for every square, non-singular submatrix, which is the criterion for unimodularity to be 'total'. Therefore, total unimodularity is satisfied for $\boldsymbol{M'}$. The remaining constraints, which do not satisfy TU, mean that it is necessary to still use an inefficient method, therefore we resort to the partial Lagrange multiplier method, which is described as follows.

5.2 Partial Lagrange Multiplier

The full Lagrange multiplier method is an established technique for solving convex optimization problems using linear constraints. In the vector notation of standard form, the problems are written as follows:

$$\min_{\vec{\mathbf{x}}} f_0(\vec{\mathbf{x}}) \;\Big|\; \begin{matrix} f_i(\vec{\mathbf{x}}) \leq 0,\ i = 1, \ldots, m; \\ h_i(\vec{\mathbf{x}}) = 0,\ i = 1, \ldots, p, \end{matrix} \tag{26}$$

where $\vec{\mathbf{x}} \in \mathbb{R}^n$. The constraints in (26) can be moved into the objective function weighted by Lagrange multipliers, as in the following equation:

$$L(\vec{\mathbf{x}}, \vec{\boldsymbol{\lambda}}, \vec{\mathbf{v}}) = f_0(\vec{\mathbf{x}}) + \sum_{i=1}^{m} \lambda_i f_i(\vec{\mathbf{x}}) + \sum_{i=1}^{p} v_i h_i(\vec{\mathbf{x}}), \tag{27}$$

where $\vec{\boldsymbol{\lambda}} \in \mathbb{R}^m$ and $\vec{\mathbf{v}} \in \mathbb{R}^p$ are the Lagrange multipliers. The dual function, expressed in $\vec{\boldsymbol{\lambda}}$ and $\vec{\mathbf{v}}$ is therefore:

$$g(\vec{\boldsymbol{\lambda}}, \vec{\mathbf{v}}) = \min_{\vec{\mathbf{x}} \in \mathbb{R}^n} L(\vec{\mathbf{x}}, \vec{\boldsymbol{\lambda}}, \vec{\mathbf{v}}), \tag{28}$$

and the corresponding dual problem can be written:

$$\max_{\vec{\boldsymbol{\lambda}}, \vec{\mathbf{v}}} g(\vec{\boldsymbol{\lambda}}, \vec{\mathbf{v}}) \;\Big|\; \vec{\boldsymbol{\lambda}} \geq 0. \tag{29}$$

When used for convex optimization problems, the dual problem can be solved for $\vec{\boldsymbol{\lambda}}$ and $\vec{\mathbf{v}}$ via the subgradient method. The corresponding values found by the dual problem for $\vec{\mathbf{x}}$ are also the optimal values for the primal problem, due to the convexity of the primal problem. The partial Lagrange multiplier method for linear programs adopts a similar strategy, with the constraints that do not satisfy TU being moved into the objective function with the set of $\vec{\boldsymbol{\lambda}}$ Lagrange multipliers. The remaining, TU-satisfying, constraints remain as constraints in the problem, allowing it to be solved as an LP. Therefore, applying this to the objective function in (25) along with the remaining constraints in (21)-(23), we get the following dual function in terms of $\vec{\boldsymbol{\lambda}}$:

$$\begin{aligned} g(\vec{\boldsymbol{\lambda}}) = \sum_{\omega^\dagger} \pi_{\omega^\dagger} \Bigg(& \mathscr{R}^\dagger(\omega^\dagger) + \sum_{\omega^\ddagger} \pi_{\omega^\ddagger} \Bigg(\mathscr{R}^\ddagger(\omega^\dagger, \omega^\ddagger) \\ & + \sum_{\omega^{\ddagger\ddagger}} \pi_{\omega^{\ddagger\ddagger}} \mathscr{R}^{\ddagger\ddagger}(\omega^\dagger, \omega^\ddagger, \omega^{\ddagger\ddagger}) \Bigg) \Bigg) \\ & + \sum_{a,s,i,t,\omega^\dagger,\omega^\ddagger,\omega^{\ddagger\ddagger}} \lambda^1_{asit\omega^\dagger\omega^\ddagger\omega^{\ddagger\ddagger}} \Big(C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} \\ & \qquad - \sum_{l,\hat{t} \in \mathcal{F}_{lt}} g_{asi} W^\dagger_{asil\hat{t}\omega^\dagger} \Big) \\ & + \sum_{a,s,i,t,\omega^\dagger,\omega^\ddagger,\omega^{\ddagger\ddagger}} \lambda^2_{asit\omega^\dagger\omega^\ddagger\omega^{\ddagger\ddagger}} \Big(C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} \\ & \qquad - \sum_{l,\hat{t} \in \mathcal{F}_{lt}} g_{asi} W^\ddagger_{asil\hat{t}\omega^\ddagger} \Big) \\ & + \sum_{a,s,i,t,\omega^\dagger,\omega^\ddagger,\omega^{\ddagger\ddagger}} \lambda^3_{asit\omega^\dagger\omega^\ddagger\omega^{\ddagger\ddagger}} \Big((C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} + C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}}) \\ & \qquad - (Z_{ast\omega^\ddagger} + \sum_{p \in \mathcal{P}_{as}} Y_{aspt\omega^\ddagger}) \Big). \end{aligned} \tag{30}$$

In this formulation, we use an abbreviated notation for the sake of brevity and readability, where $\sum_{a,s}$ is equivalent to writing $\sum_{a \in \mathcal{A}} \sum_{s \in \mathcal{S}}$.

5.3 Solution to the Partial Lagrange Multiplier

To solve our optimization problem using the partial Lagrange multiplier method, we iteratively minimize the dual problem in terms of $\vec{\boldsymbol{\lambda}}$, given by the objective function in (30):

$$\min g(\vec{\boldsymbol{\lambda}}), \tag{31}$$

subject to the TU-satisfying constraints in (11)-(12), (15)-(18), and (24). We also relax the variable range constraints to be linear intervals.

5.3.1 Dual Problem Solution

Since the outlined dual problem can be solved as an LP with variables guaranteed to give integer values, we can solve it in polynomial time using the Interior Point Method (IPM) [35]. In each iteration of the algorithm, we solve the

dual problem, and use the new values of the primal variables to determine the value of the original primal objective function. This value is then used to calculate the Lagrange multipliers for the next iteration of the algorithm.

5.3.2 Updating Lagrange Multipliers

The behaviour of the algorithm is governed by the updated values of the Lagrange multipliers. This calculation is governed by the stepsize α_k for a given step k. The Lagrange multipliers are updated by adding the old value to the gradient of the dual function weighted by the step size, or 0, whichever is larger:

$$\vec{\boldsymbol{\lambda}}^{(k+1)} = \vec{\boldsymbol{\lambda}}^{(k)} + \alpha_k \cdot \nabla g(\vec{\boldsymbol{\lambda}}^{(k)}). \tag{32}$$

The stepsize, α_k can be calculated in a variety of ways, for example, as a constant value, $1/k$, or $1/\sqrt{k}$. The ideal value will depend on the problem being solved, with a calculation dependent on k giving a decreasing stepsize as k increases, providing greater precision. A constant value may converge more rapidly, but may also result in oscillation between two fixed values, of which one is the optimum. The function $\nabla g(\vec{\boldsymbol{\lambda}}^{(k)})$ is the derivative of the dual function with respect to $\vec{\boldsymbol{\lambda}}$ and, when derived from (30) is as follows:

$$\begin{aligned}\nabla g(\vec{\boldsymbol{\lambda}}) = &\sum_{a,s,i,t,\omega^\dagger,\omega^\ddagger,\omega^{\ddagger\ddagger}} \Big(C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} - \sum_{l,\hat{t}\in\mathcal{F}_{lt}} g_{asi} W^{\dagger}_{asil\hat{t}\omega^\dagger}\Big) \\ &+ \sum_{a,s,i,t,\omega^\dagger,\omega^\ddagger,\omega^{\ddagger\ddagger}} \Big(C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} - \sum_{l,\hat{t}\in\mathcal{F}_{lt}} g_{asi} W^{\ddagger}_{asil\hat{t}\omega^\ddagger}\Big) \\ &+ \sum_{a,s,i,t,\omega^\dagger,\omega^\ddagger,\omega^{\ddagger\ddagger}} \Big((C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} + C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}}) \\ &\qquad - (Z_{ast\omega^\ddagger} + \sum_{p\in\mathcal{P}_{as}} Y_{aspt\omega^\ddagger})\Big). \end{aligned} \tag{33}$$

5.3.3 Stopping Criterion

The partial Lagrange multiplier method will always converge to the optimal solution, as we prove in Section 6. To determine when the solution has been reached, we use the stopping criterion. If the best value for the primal objective does not improve by more than the marginal value ϵ and Φ iterations have passed since the best value was found, then the algorithm terminates. If ϵ is sufficiently small and Φ is sufficiently large, the best value on termination would be guaranteed to be the optimum. However, the user may choose values for ϵ and Φ that suit their desired tradeoff between accuracy and runtime. The pseudocode for the partial Lagrange multiplier algorithm outlined in this section, with the described stopping criterion, is given in Algorithm 1.

Algorithm 1 Partial Lagrange Multiplier Method

Input : Dual problem $g(\vec{\boldsymbol{\lambda}})$, Φ
Output: Optimal solution for $X^\dagger$, $W^\dagger$, $W^\ddagger$, Y, Z, C, $C^\ddagger$
Initialize $g^{(0)}_{best} = -\infty$;
Set the iteration index: $k = 1, k_x = 0$;
Initialize $\vec{\boldsymbol{\lambda}}$ values ≥ 0, upper bounds given in (34) ;
do
  Find optimal solution for $g(\vec{\boldsymbol{\lambda}})$, given in Eq. (30) ;
  Compute solution of primal problem ;
  if $g^{(k)}_{best} - g^{(k-1)}_{best} \geq \epsilon$ **then** $k_x = k$;
  $k = k + 1$;
  Compute step size α_k;
  Update Lagrange multipliers using Eq. (32) and (33);
while $k - k_x \leq \Phi$;

6 Theoretical Performance Analysis

In this section we demonstrate analytically the convergence and scalability of the partial Lagrange multiplier method.

6.1 Convergence Analysis

To prove the convergence of the algorithm, we need to prove three elements. Firstly, we demonstrate that the dual function is concave. Secondly, since the function is concave, we can demonstrate that the algorithm will converge to the optimum by iteratively updating $\vec{\boldsymbol{\lambda}}$. Finally we show that we can guarantee strong duality, thus the solution to the dual problem found by the algorithm, is also the optimal solution to the corresponding primal problem.

6.1.1 Proof of Convergence

To prove the convergence of the algorithm, we first prove its concavity. This proof is given in Appendix B due to space constraints. Since we have proved that $g(\vec{\boldsymbol{\lambda}})$ is concave, we can assume that $\nabla g(\vec{\boldsymbol{\lambda}}) \leq G$, where G is a value that we define in (35) and (36). Further, each individual Lagrange multiplier value, denoted by λ_i, is bounded above, as well as by the defined bound of $\lambda_i \geq 0$. We can derive the upper bounds of $\vec{\boldsymbol{\lambda}}$ from Eq. (30), giving the following limits:

$$\begin{aligned} \lambda^1_{asit\omega^\dagger\omega^\ddagger\omega^{\ddagger\ddagger}} &\leq \frac{\pi_{\omega^\dagger} f^\dagger_{asil t\omega^\dagger}}{g_{asi}} \\ \lambda^2_{asit\omega^\dagger\omega^\ddagger\omega^{\ddagger\ddagger}} &\leq \frac{\pi_{\omega^\ddagger} f^\ddagger_{asil t\omega^\ddagger}}{g_{asi}} \\ \lambda^3_{asit\omega^\dagger\omega^\ddagger\omega^{\ddagger\ddagger}} &\leq \pi_{\omega^\ddagger} k^\ddagger_{as\omega^\ddagger}. \end{aligned} \tag{34}$$

Given these upper bounds, we can say $||\vec{\boldsymbol{\lambda}}^{(1)} - \vec{\boldsymbol{\lambda}}^*||_2 \leq R$, where $\vec{\boldsymbol{\lambda}}^{(1)}$ denotes the Lagrange multipliers in the first step, $\vec{\boldsymbol{\lambda}}^*$ denotes the optimal values for the Lagrange multipliers, and R is a value that we will define in Section 6.2. With this established, we can write the convergence proof from [36], which we provide in Appendix C for reference. This proof is a known result from convex optimization, and thus we know that $g^{(k)}_{best}$, which denotes the best solution value for $g(\vec{\boldsymbol{\lambda}})$ at step k, will converge to $g(\vec{\boldsymbol{\lambda}}^*)$, the optimal solution of the dual problem. For the optimal version of the stepsize calculation $\alpha_k = (R/G)/\sqrt{k}$, the number of iterations required to do so will be bounded above by $K \leq (RG/\epsilon)^2$,

where ϵ is the error margin covered under Section 5.3.3 on the stopping criterion.

6.1.2 Strong Duality

We have shown that the partial Lagrange multiplier algorithm will converge to an optimal solution of the dual problem. However, we must also show that this solution is optimal for the original primal problem. Since we have satisfied the property of total unimodularity, we are guaranteed an integer solution. By the Slater's constraint qualification, if the optimization problem is convex, and there is at least one solution that is strictly feasible (which it is not difficult to find), then the solution to the Lagrange dual problem is equivalent to the primal solution [39]. Therefore if $g(\vec{\boldsymbol{\lambda}}^*)$ is the optimal solution to the dual problem, the corresponding values of the decision variables are the optimal solution to the original mixed integer optimization problem.

6.2 Scalability Analysis

We have proved that the partial Lagrange multiplier method will converge within a maximum of $K \leq (RG/\epsilon)^2$ iterations. We have already established that, using vector notation, $||\vec{\boldsymbol{\lambda}}^{(1)} - \vec{\boldsymbol{\lambda}}^*||_2 \leq R$. Given the upper bounds on the Lagrange multipliers in (34), we can reasonably assume that, given the probability multipliers and larger denominators giving upper bounds ≤ 1, $R = |\vec{\boldsymbol{\lambda}}|$, where $|\vec{\boldsymbol{\lambda}}|$ denotes the total number of Lagrange multipliers. Additionally, we have also given the bound $||\nabla g(\vec{\boldsymbol{\lambda}})||_2 \leq G$, with $g(\vec{\boldsymbol{\lambda}})$ given in (33). If the expressions $(C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} - \sum_{l,\hat{t}\in\mathcal{F}_{lt}} g_{asi} W^{\dagger}_{asil\hat{t}\omega^{\dagger}}) \geq 0$, $(C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} - \sum_{l,\hat{t}\in\mathcal{F}_{lt}} g_{asi} W^{\ddagger}_{asil\hat{t}\omega^{\ddagger}}) \geq 0$, and $(C^{\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} + C^{\ddagger\omega^\ddagger}_{asit\omega^{\ddagger\ddagger}} - (Z_{ast\omega^\ddagger} + \sum_{p\in\mathcal{P}_{as}} Y_{aspt\omega^\ddagger})) \geq 0$, then:

$$||\nabla g(\vec{\boldsymbol{\lambda}})||_2 \leq ||\vec{\mathbf{c}}||_2 = G, \tag{35}$$

where $\vec{\mathbf{c}}$ is the vector representation of the claim variables. Conversely, if the expressions given above from (33) are less than 0, then we have:

$$||\nabla g(\vec{\boldsymbol{\lambda}})||_2 \leq ||\mathbf{W}\vec{\mathbf{u}}||_2 = G, \tag{36}$$

where $\vec{\mathbf{u}}$ is the vector representation of the security utilization and insurance provisioning variables, while $\mathbf{W}$ is the matrix representation of their coefficients in the above equations. Substituting these values for R and G into the inequality $K \leq (RG/\epsilon)^2$, we have a function that is bounded by a linear function of the problem size. The solution to the dual problem is the main point of complexity in the partial Lagrange multiplier algorithm, which can be solved in polynomial time using the IPM [35]. Therefore we have a worst case of polynomial execution time for the partial Lagrange multiplier algorithm.

7 Sensitivity Analysis

Stochastic optimization finds the optimal solution to a programming problem given uncertain parameters. In [40], an analytical sensitivity analysis was introduced to find the precise breakpoints for parameters at which the corresponding variable allocations change. This technique is specifically for linear programs. Therefore, we use the best result from the partial Lagrange multiplier algorithm, fixing the values of the Lagrange multipliers at their optimal values. We define the variable γ to denote the offset value for the cost coefficient under examination. Two linear programming problems can then be formulated to give γ_1 and γ_2, which are the lower and upper bounds of γ, within a linear interval. We give the standard form of the two optimization problems in (37) and (38), where $e_j = 1$ if j is the index of the c_j under analysis, else $e_j = 0$.

$$\gamma_1 = \min\{\gamma : \mathbf{A}^\top\vec{\mathbf{y}}+\vec{\mathbf{s}} = \vec{\mathbf{c}}+\gamma e_j, \vec{\mathbf{b}}^\top\vec{\mathbf{y}} = \vec{\mathbf{c}}^\top\vec{\mathbf{x}}^*+\gamma x^*_j, \vec{\mathbf{s}} \geq 0\}, \tag{37}$$

$$\gamma_2 = \max\{\gamma : \mathbf{A}^\top\vec{\mathbf{y}}+\vec{\mathbf{s}} = \vec{\mathbf{c}}+\gamma e_j, \vec{\mathbf{b}}^\top\vec{\mathbf{y}} = \vec{\mathbf{c}}^\top\vec{\mathbf{x}}^*+\gamma x^*_j, \vec{\mathbf{s}} \geq 0\}. \tag{38}$$

We give full details of the linear optimization problems to be solved in Appendix D, along with further explanation of the sensitivity analysis technique from [40] as it applies to our problem.

8 Performance Evaluation

8.1 Parameter Settings

To test the performance of our optimization formulation, we consider a scenario across two time periods with three SECaaS providers (i.e. $s1$, $s2$, and $s3$) and two cyber insurers (i.e. $i1$ and $i2$). Each time period lasts for one day, with all traffic to the user's applications hosted in the cloud scanned by a SECaaS provider before reaching the user's application. Each SECaaS provider offers two prepaid plans (i.e. $p1$ and $p2$), with one lasting for one time period, the other lasting for two. Provider $s1$ offers services in units of 5 packets for $p1$ and 8 packets for $p2$ with each unit costing \$0.10 and \$0.15, respectively. Provider $s2$ offers units of 15 and 20 at \$0.25 and \$0.30. The damages for a packet that is undetected by the security services is priced at \$1.5 per packet. To simulate the possibility of packets not being processed by security, we include an additional 'dummy' SECaaS provider, $s3$, with no cost and no insurance coverage. The user may purchase policies from either of the two insurers, who each offer two premiums (i.e. $q1$ and $q2$), with $q1$ lasting for one time period, and $q2$ lasting for two, and being priced just below twice the price of $q1$. Values for service prices are synthesized based on cloud and security providers, such as Amazon EC2 [41] and Trend Micro [6]. Packet damages and indemnity are synthesized from estimated damages caused by historic cyber attacks and example insurance premiums for small scale businesses, scaled according to our incoming packet data [42] [43]. Packet demand levels and detection probabilities are generated from real honeypot data provided by the University of Waikato's Cyber Security Lab [44]. Honeypots positioned in Singapore, Sao Pǎolo, Brazil, and San Jose, USA, collected packet data over a number of days. Using this data, we employed the Snort IDS [45] with two different rulesets. The more complete ruleset provided the benchmark for the number of malicious packets, with the other ruleset giving the proportion of missed packets and the corresponding missed-detection probabilities for each scenario. This analysis provides us

with realistic values for cyber attacks and provides more informative numerical results.

8.2 Numerical Results

The deterministic equivalent of the problem formulation was encoded in a GAMS script and solved using the CPLEX solver [46]. The tests were designed to examine the performance of the stochastic optimization formulation, and explore the relationship between security provisioning and insurance coverage.

8.2.1 Influence of prepaid security on cost

In Fig. 3 we set the number of prepaid SECaaS services provisioned from each security provider and measure the impact on the expected cost and provisioning quantities across all scenarios. The upper plot measures the total used services. As the provisioned prepaid services increase, the number of on-demand services required decreases. This is the expected result, but compares well with the lower plot, which measures the change in prepaid services against total cost. The total cost curve is convex and shows that there is an ideal prepaid point where the number of prepaid services mitigates the damages of both overprovisioning and underprovisioning. Thus the optimal solution requires some on-demand services when demand is high, as the increased expense does not outweigh the wasted expense from overprovisioning at low demand that more prepaid services would incur. This shows the usefulness of stochastic optimization, as our formulation finds this optimal point.

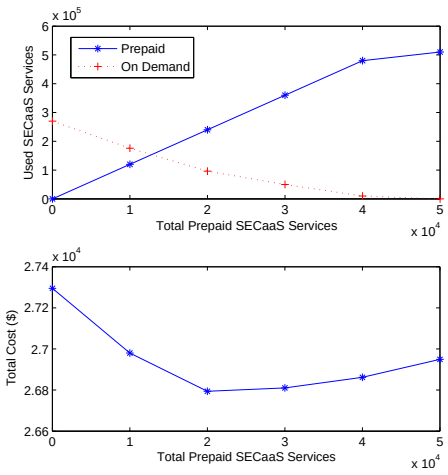


Fig. 3: Impact of prepaid SecaaS services on service utilization and total cost.

8.2.2 Impact of packet damages and indemnity on security provisioning

One of the key challenges in cyber insurance is the accurate estimation of damages caused by cyber attacks. In Fig. 4 we investigate the effect of changing the damage values assigned to individual packets on security provisioning. The model allows for some incoming packets to not be covered by the security services. We then measure the decrease in the number of ignored packets as the damage per malicious unhandled packet. The result is clear - the damages caused by inadequate security necessitate the increasing of security provisioning, and cyber insurance is not a sufficient mitigation. This also shows the importance of insurers correctly assessing the expected damages caused by cyber attacks so as to optimally set their insurance policies.

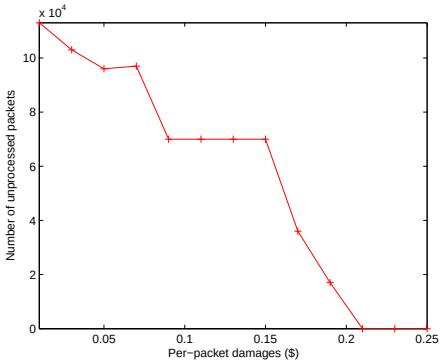


Fig. 4: Impact of packet damages on security provisioning measured in unprocessed packets. As damages increase in cost, it becomes more costly to leave packets unprocessed, and so security provisioning is increased to ensure more packets are processed.

We also examine the influence that the indemnity values (set by the insurers) have on security provisioning. As before, we measure the quantity of unprocessed packets as the indemnity changes. This is shown in Fig. 5. In this graph we consider three different packet damage values, with each showing that the number of unprocessed packets decreases as the indemnity rises. This is because packets unprocessed by the security services are not covered by the insurance policy. Thus, higher indemnity values create an incentive for users to employ comprehensive security coverage. The height of the curves shows that this is complementary to the per-packet damage costs, as the greater the damage the lower the curve, and the faster it descends. Therefore we conclude that cyber insurance can be used to incentivize security provisioning.

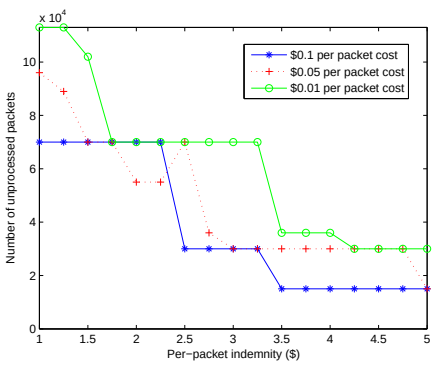


Fig. 5: Impact of packet indemnity on security provisioning measured in unprocessed packets. Higher indemnity provides an incentive to increase security provisioning so that packets are not left unprocessed.

8.2.3 Comparison of solution methods with varying attack probability

To justify our use of the stochastic optimization approach, we compare the performance of the stochastic optimization to various alternative methods under varying attack probabilities. This will highlight the importance of accurate security provisioning and insurance provisioning. In Fig. 6 we provide a comparison between the methods of stochastic optimization, expected value function, and no-insurance, and also include a fourth result, with no security (and therefore no insurance), to demonstrate the high cost of ignoring cyber security. The expected value function method purchases prepaid security services sufficient to process the mean number of packet arrivals as specified by the probability distribution. The four methods are compared across three different probability distributions - normal, uniform, and the one derived from our real data. The expected value function gives a good result but is inferior to stochastic optimization as it does not consider future impact, with the order of the bar heights following the expected order.

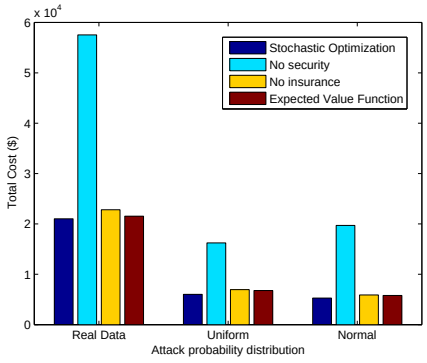


Fig. 6: Comparison of alternative solution methods with varying attack probability distributions.

8.2.4 Influence of budget on provisioning

In the real world, a user will have a restricted budget to spend on security services and insurance. To examine the priorities of the user, we introduced a budget limit which acts as a maximum value for the objective function solution. In Fig. 7 we provide a comparison between the amount of security provisioning and the number of insurance claims. As the budget rises, the security provisioning increases until it reaches a maximum to correspond with the maximum incoming traffic, showing that security is the top priority. The increase in packet claims, however, shows that not only is the insurance provision increased, but using the prediction from stochastic optimization, the user can increase their security knowing that the indemnity benefits from covered packets will offset the security costs. Thus we see a symbiotic relationship between security and insurance provisioning where the insurance coverage provides an incentive for the security provisioning, which in turn limits the costs from packet damages.

8.2.5 Influence of security capacity on total cost

Each security plan from a provider has a maximum capacity of prepaid SECaaS packets that can be provisioned. In Fig. 8

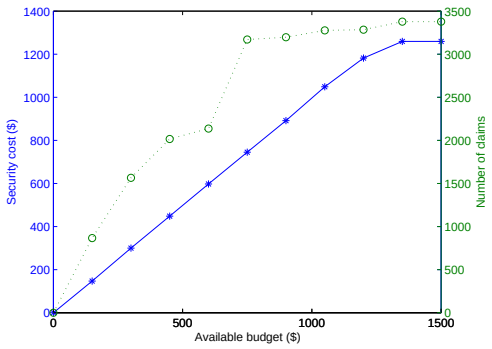


Fig. 7: Exploration of the relationship between security and insurance provisioning under budget constraints.

we vary the prepaid capacity of the security providers, and measure the impact on the total cost. As expected, with more limited prepaid options, costs are higher, as the number of packets that must be processed by on-demand services increases. This problem is more severe initially, as the number of on-demand services must be increased. The impact lessens non-linearly, as the prepaid plans reach a capacity sufficient to handle most demand without on-demand services, and further increases have limited impact. The number of on-demand services required decreases and so security costs also decrease. To provide a comparison we add results using the mean number of incoming packets and a curve with no insurance provision. The stochastic optimization performs best, as expected, but the mean value result offers worse performance, because it is inflexible and results in underprovisioning, making it worse than the no-insurance option, due to the cost penalty from excessive on-demand security.

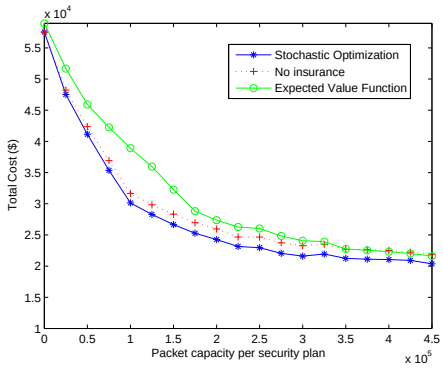


Fig. 8: Exploration of the effect security capacity has on total cost.

8.2.6 Benefits of joint optimization

The premise of our formulation is that optimizing the security provisioning and cyber insurance jointly is preferable to handling the two separately, due to the influence that the two services have on each other. We have demonstrated this connection in our previous results. Nonetheless, to justify our premise we implemented an alternative method that first, independently, provisions security, before making

cyber insurance decisions. Again, we vary the security capacity, to show how the solutions perform. The result, given in Fig. 9 shows that the joint solution gives clear financial benefits over the separate approach. Thus, the symbiotic nature of security and insurance is demonstrated in cost benefits.

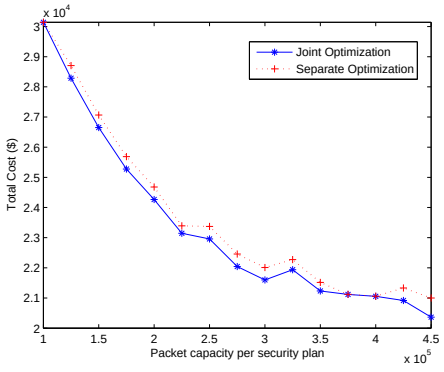


Fig. 9: Demonstration of the benefits of joint optimization.

8.2.7 Performance of partial Lagrange multiplier method

The key purpose of introducing the partial Lagrange multiplier method is to achieve improved scalability over the conventional branch-and-bound (B&B) method for solving mixed integer programming problems. In Fig. 10 we introduce a reduced size version of our test case and vary the number of time periods, allowing us to increase the problem size linearly. We compare the execution time for a standard B&B algorithm with the execution time for the partial Lagrange multiplier method and find that the increase in execution time matches the expectation from our analytical results. The B&B curve increases at a rapid rate, showing that for a large problem, the memory and processing time requirements would rapidly become impractical. By contrast, using a stepsize of $\alpha = 1$, we find that the partial Lagrange multiplier method converges quickly and only increases at a very slow rate, and linearly, as the problem size grows. This result supports our claims that the partial Lagrange multiplier method offers much improved scalability over traditional mixed integer program solution methods.

8.2.8 Convergence Behaviour

In Fig. 11 we demonstrate the manner in which the partial Lagrange multiplier method converges to the solution, testing the performance on the same problem as the previous result. We observe that the convergence is extremely rapid, as the initial value is high, but the solution drops quickly to near the optimum, and finds the optimal value on the third iteration. We find that this performance remains the same independent of the initial value of the Lagrange multipliers, or the value of the stepsize - the optimal value is still found within three steps. After the optimal value is found, the solution alternates between this value and a slightly higher one, until the algorithm terminates. This is within the expected behaviour of the algorithm, which specifies termination if the algorithm does not improve within a predetermined number of iterations. In this case, we terminate

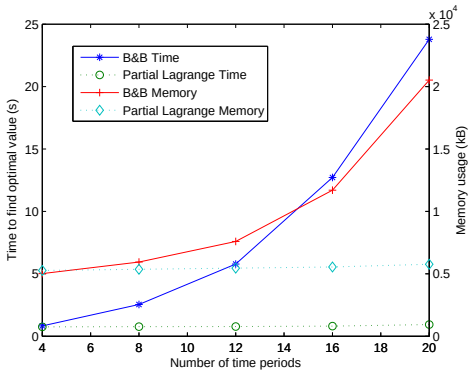


Fig. 10: Comparison of execution time and memory usage of partial Lagrange multiplier and branch-and-bound (B&B) methods as problem size increases.

after ten iterations, with the best value being the solution to the original integer problem. The speed of convergence is therefore dependent on the structure of the problem, and the tightness of the constraints (i.e. those that satisfy total unimodularity). If the constraints are tight, within the parameters set, then the solution can be found faster than the polynomial worst case, as we find here.

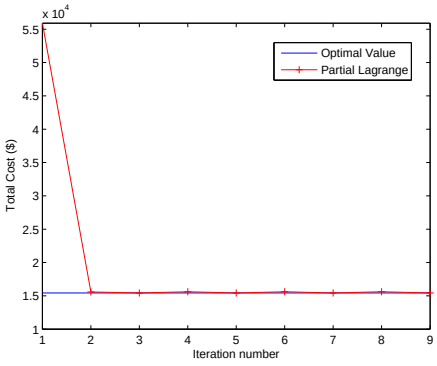


Fig. 11: Example case of speed of convergence of the partial Lagrange algorithm to the optimal value.

8.2.9 Sensitivity Analysis of Prepaid SECaaS Provisioning

We use parameter settings that allow the customer to choose security plans of varying lengths, with a choice of service provider. To understand how pricing influences the user's provisioning decisions, we perform an analytical sensitivity analysis on the variable instance $X^{\dagger}_{1,1,1,1,1}$. The sensitivity analysis formulation, using the final step of the reduced-size partial Lagrange multiplier algorithm result, gives precise values for the change in cost at which the customer changes their prepaid security allocation. The point of change is called a 'breakpoint', and we show this in Fig. 12. As the price increases, the value of $X^{\dagger}_{1,1,1,1,1}$ decreases, as expected, with prepaid security provisioning redistributed to the alternative plan. At each point on the graph, a threshold is reached, causing the customer to change their purchasing choices. The breakpoints are primarily determined by the cost of the alternative. However, since $X^{\dagger}$ is a prepaid

service variable, it must also consider the increased cost against the on-demand cost. Thus, an increased cost of this variable may not immediately result in a drop to 0, if the on-demand cost is still high, rather the customer prefers to change plan. The primary benefit of this approach over a trial-and-error approach, where the cost values must be manually changed and the solution recalculated, is that we can guarantee that we find the exact point at which the cost changes, and minimize the number of tests required to find those points. To demonstrate this, a second line is plotted, showing a set of reasonable values for cost variation, and the corresponding variable allocation. Whilst the various levels of provisioning are found, the exact breakpoints are not, and further measurements would be required to find them. Service providers may employ this sensitivity analysis method to set competitive prices, as they can see exactly how to price their services such that the customer chooses them over the competition, without offering a lower price than necessary. To examine the choice between providers, we take the result of the partial Lagrange multiplier method for the full-size problem in Fig. 13, and vary the cost of plan $p2$ from provider $s1$ across two time stages. We plot the breakpoints and measure the allocation for plans $p1$ and $p2$ from providers $s1$ and $s2$. As the cost increases, the customer prefers to switch provider, up to the capacity limit of that provider, rather than choose a short-term plan. As the price increases further, the customer switches to the short-term plan to cover the remaining demand. Therefore, it is better to choose a long-term plan from any provider, as they are more cost-efficient than short-term plans, with stochastic optimization allowing us to minimize the long-term costs.

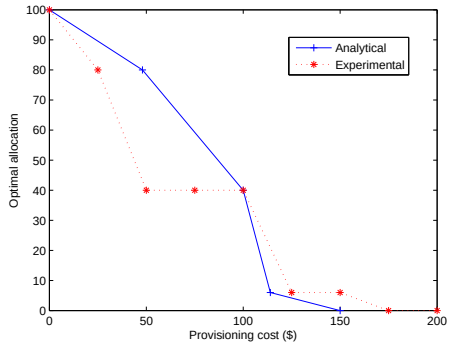


Fig. 12: Cost breakpoints of the variable $X^{\dagger}_{1,1,1,1,1,1}$, comparing analytical and experimental approaches to sensitivity analysis.

8.2.10 Results conclusions

From the above numerical results, we can see clearly that the stochastic optimization provides benefits as it allows for variation in future demand and attack probabilities, as well as the corresponding damages and indemnity values. It also offers clues as to how those values should be optimally set, as we see the relationship between security provisioning and cyber insurance, as good insurance encourages good security, and vice versa. Further, we demonstrate the significantly better scalability of the partial Lagrange multiplier method over traditional branch-and-bound for MIP solving.

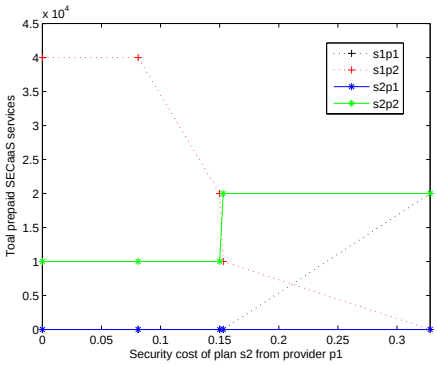


Fig. 13: Cost breakpoints of the variable $X^{\dagger}_{1,1,2,1,1,1}$, for the full-size problem showing the choices made between plans and providers.

Finally, we implement our analytical sensitivity analysis, and show that it is more efficient and precise than an experimental approach.

9 Conclusion

In this paper we have presented a combined approach to security and cyber insurance provisioning in the cloud. Using a stochastic optimization, we have presented a method of optimally provisioning both services in the face of uncertainty regarding future pricing, incoming traffic and cyber attacks. Since our optimization involves solving an integer programming problem, we present the partial Lagrange multiplier method, which exploits the total unimodularity property to guarantee integer solutions, while relaxing the problem to a linear programming problem. This problem is solved iteratively using a subgradient method, which we prove converges to the optimal solution in at worst polynomial time. Using the solution produced by the algorithm, we apply an analytical sensitivity analysis approach that gives precise sensitivity values for individual parameters. Finally we provide an experimental evaluation of our contributions using realistic traffic and attack data derived by running real traffic data through an Intrusion Detection System. The main challenge of cyber insurance is the number of assumptions that must be made, for example, the ability to detect cyber attacks, establish accurate damages, and successfully make insurance claims. Future extensions could consider the interaction of applications, where the security performance of one part of the system can impact the security of other parts. We have introduced real honeypot data, but future extensions could consider more extensive data to produce more accurate options. Accuracy of data could further be extended through the implementation of systems to update parameters on a daily or weekly basis, to improve future decisions.

References

[1] McAfee, "Net Losses: Estimating the Global Cost of Cybercrime," Center for Strategic and International Studies, Economic Impact of Cybercrime II, Jun. 2014.

[2] (2016) Identity theft resource center data breach reports. [Online]. Available: http://www.idtheftcenter.org/2016databreaches.html

[3] (2016) A bold approach to cyber risk management. [Online]. Available: http://www.mas.gov.sg/News-and-Publications/Speeches-and-Monetary-Policy-Statements/Speeches/2016/A-Bold-Approach-to-Cyber-Risk-Management.aspx

[4] (2016) Insurance 2020 beyond: Reaping the dividends of cyber resilience. [Online]. Available: http://www.pwc.com/gx/en/industries/financial-services/insurance/publications/insurance-2020-cyber.html

[5] (2016) Mcafee security-as-a-service solutions. [Online]. Available: https://www.mcafeeasap.com/MarketingContent/Products/ProductsLanding.aspx

[6] (2016) Deep security as a service. [Online]. Available: http://www.trendmicro.com/us/business/saas/deep-security-as-a-service/#usage-based-pricing

[7] B. Delamore and R. K. L. Ko, "Chapter 9 - security as a service (secaas)an overview," in *The Cloud Security Ecosystem*, R. K. L. Ko and K.-K. R. Choo, Eds. Boston: Syngress, 2015, pp. 187 – 203. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128015957000094

[8] (2016) Allianz cyber protect. [Online]. Available: http://www.agcs.allianz.com/services/financial-lines/cyber-insurance/

[9] (2016) Cyber and data security. [Online]. Available: http://www.qbeeurope.com/professional-financial/cyber-liability.asp

[10] M. Clark. (2014) Timeline of target's data breach and aftermath: How cybertheft snowballed for the giant retailer. [Online]. Available: http://www.ibtimes.com/timeline-targets-data-breach-aftermath-how-cybertheft-snowballed-giant-retailer-1580056

[11] C. A. Newman. (2016) Targets cyber insurance: A $100 million policy vs. $300 million (so far) in costs. [Online]. Available: http://datasecuritylaw.com/blog/targets-cyber-insurance-a-100-million-policy-vs-300-million-so-far-in-costs/

[12] S. Chaisiri, R. K. L. Ko, and D. Niyato, "A joint optimization approach to security-as-a-service allocation and cyber insurance management," in *Trustcom/BigDataSE/ISPA, 2015 IEEE*, Aug 2015, pp. 426–433.

[13] C. P. Ram and G. Sreenivaasan, "Security as a service (sass): Securing user data by coprocessor and distributing the data," in *Trendz in Information Sciences Computing(TISC2010)*, Dec 2010, pp. 152–155.

[14] C. Tang and J. Liu, "Selecting a trusted cloud service provider for your saas program," *Computers Security*, vol. 50, pp. 60 – 73, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404815000139

[15] T. Xie and X. Qin, "Security-aware resource allocation for real-time parallel jobs on homogeneous and heterogeneous clusters," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 5, pp. 682–697, May 2008.

[16] H. Liang, D. Huang, L. X. Cai, X. Shen, and D. Peng, "Resource allocation for security services in mobile cloud computing," in *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, April 2011, pp. 191–195.

[17] M. Lin, L. Xu, L. T. Yang, X. Qin, N. Zheng, Z. Wu, and M. Qiu, "Static security optimization for real-time systems," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 1, pp. 22–37, Feb 2009.

[18] L. Zeng, B. Veeravalli, and X. Li, "Saba: A security-aware and budget-aware workflow scheduling strategy in clouds," *Journal of Parallel and Distributed Computing*, vol. 75, pp. 141 – 151, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0743731514001658

[19] F. A. Guenane, "Network security management using a novel firewall cloud-based service," in *2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS)*, July 2015, pp. 1–6.

[20] Y. Sun, S. Nanda, and T. Jaeger, "Security-as-a-service for microservices-based cloud applications," in *2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom)*, Nov 2015, pp. 50–57.

[21] ENISA, "Incentives and barriers of the cyber insurance market in Europe," Tech. Rep., Jun. 2012. [Online]. Available: https://www.enisa.europa.eu/publications/incentives-and-barriers-of-the-cyber-insurance-market-in-europe

[22] B. Filkins, "Quantifying Risk: Closing the Chasm Between Cybersecurity and Cyber Insurance," SANS Institute InfoSec Reading Room, Tech. Rep., Mar. 2016. [Online]. Available: https://www.sans.org/reading-room/whitepapers/analyst/quantifying-risk-closing-chasm-cybersecurity-cyber-insurance-36770

[23] R. Anderson, R. Böhme, R. Clayton, and T. Moore, "Security Economics and the Internal Market," ENISA, Tech. Rep., Jan. 2008. [Online]. Available: https://www.enisa.europa.eu/publications/archive/economics-sec/

[24] C. Biener, M. Eling, and J. H. Wirfs, "Insurability of Cyber Risk: An Empirical Analysis," University of St. Gallen, School of Finance, Working Papers on Finance 1503, Jan. 2015. [Online]. Available: https://ideas.repec.org/p/usg/sfwpfi/201503.html

[25] T. Bandyopadhyay, "Organizational adoption of cyber insurance instruments in it security risk management: A modeling approach," in *Southern Association for Information Systems (SAIS) 2012 Proceedings*, 2012. [Online]. Available: http://aisel.aisnet.org/sais2012

[26] (2006) Security risk management guide. [Online]. Available: http://trygstad.rice.iit.edu:8000/Books/TheSecurityRiskManagementGuide-Microsoft.pdf

[27] "Information technology – security techniques – information security management systems – requirements," ISO/IEC 27001, Standard, Jan. 2013.

[28] W. J. Y. J. P. Kesan, R. P. Majuca, "The Economic Case for Cyberinsurance," University of Illinois College of Law, Law and Economics Working Papers Series LE04-004, Nov. 2009.

[29] R. Pal, L. Golubchik, and K. Psounis, *Aegis A Novel Cyber-Insurance Model*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 131–150. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-25280-8_12

[30] C. Barracchini and M. E. Addessi, "Cyber risk and insurance coverage: An actuarial multistate approach," *Review of Economics Finance*, vol. 4, pp. 57–69, 2014. [Online]. Available: http://EconPapers.repec.org/RePEc:bap:journl:140105

[31] A. A. Waskita, H. Suhartanto, P. D. Persadha, and L. T. Handoko, "A simple statistical analysis approach for intrusion detection system," *CoRR*, vol. abs/1405.7268, 2014. [Online]. Available: http://arxiv.org/abs/1405.7268

[32] (2010) Statistical-based intrusion detection. [Online]. Available: https://www.symantec.com/connect/articles/statistical-based-intrusion-detection

[33] R. Pal, L. Golubchik, K. Psounis, and P. Hui, "Will cyber-insurance improve network security? A market analysis," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, April 2014, pp. 235–243.

[34] R. Winton. (2016) Hollywood hospital pays $17,000 in bitcoin to hackers; fbi investigating. [Online]. Available: http://www.latimes.com/business/technology/la-me-ln-hollywood-hospital-bitcoin-20160217-story.html

[35] C. P. Gomes, "Structure, duality, and randomization: Common themes in AI and OR." in *AAAI/IAAI*, H. A. Kautz and B. W. Porter, Eds. AAAI Press / The MIT Press, 2000, pp. 1152–1158. [Online]. Available: http://dblp.uni-trier.de/db/conf/aaai/aaai2000.html#Gomes00

[36] Z. Cao, H. Guo, J. Zhang, D. Niyato, and U. Fastenrath, "Improving the efficiency of stochastic vehicle routing: A partial Lagrange multiplier method," *Vehicular Technology, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[37] E. Santos, Jr. and E. S. Santos, "Polynomial solvability of cost-based abduction," *Artif. Intell.*, vol. 86, no. 1, pp. 157–170, Sep. 1996. [Online]. Available: http://dx.doi.org/10.1016/0004-3702(96)00016-1

[38] A. J. Hoffman and J. B. Kruskal, *Integral Boundary Points of Convex Polyhedra*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 49–76. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-68279-0_3

[39] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[40] B. Jansen, J. de Jong, C. Roos, and T. Terlaky, "Sensitivity analysis in linear programming: Just be careful!" *European Journal of Operational Research*, vol. 101, no. 1, pp. 15 – 28, 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221796001725

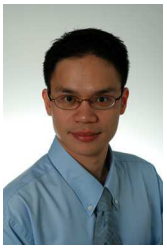
[41] (2016) Amazon EC2 pricing. [Online]. Available: http://aws.amazon.com/ec2/pricing/

[42] (2015) 2015 Ponemon Institute cost of cyber crime study. [Online]. Available: http://www8.hp.com/us/en/software-solutions/ponemon-cyber-security-report/

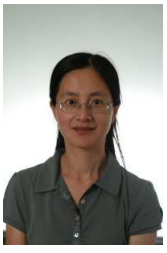
[43] C. Marciano, "How much does cyber/data breach insurance cost?" Jun. 2016. [Online]. Available: https:

//databreachinsurancequote.com/cyber-insurance/cyber-insurance-data-breach-insurance-premiums/
[44] (2016) Cybersecurity Researchers of Waikato. [Online]. Available: https://crow.org.nz/
[45] (2016) Snort IDS. [Online]. Available: https://www.snort.org/
[46] (2017) GAMS - The Solver Manuals. [Online]. Available: http://www.gams.com/latest/docs/solvers/allsolvers.pdf

Jonathan Chase is currently a PhD Student in the School of Computer Science and Engineering, at the Nanyang Technological University, Singapore. He received an M.Eng. degree in the Department of Computer Science from the University of Warwick, UK in 2011. His research interests include optimization in cloud computing, mobile application migration, and cyber security and insurance.



Dusit Niyato is currently an Associate Professor in the School of Computer Science and Engineering, at the Nanyang Technological University, Singapore. He received a Ph.D. in Electrical and Computer Engineering from the University of Manitoba, Canada in 2008. His research interests are in the area of the optimization of wireless communication and mobile cloud computing, smart grid systems, and green radio communications.



Ping Wang received a Ph.D. degree in electrical engineering from the University of Waterloo, Canada in 2008. She is currently an Assistant Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. Her current research interests include resource allocation in multimedia wireless networks, cloud computing, and smart grid.

Sivadon Chaisiri received the PhD degree in computer engineering from the Nanyang Technological University, Singapore in 2013. He is currently a Research Fellow at the University of Waikatos Cyber Security Lab where he is conducting cyber security research in a cloud security project named STRATUS funded by the Ministry of Business, Innovation and Employment, New Zealand. He was the Principal Investigator of a mobile security project funded by InternetNZ and a privacy management project funded by the Office of Privacy Commissioner, New Zealand. His research interests include cyber security economics, context-aware security, cloud security, and stochastic programming. He is a member of the IEEE.

Associate Professor Ryan Ko is Director of the New Zealand Institute for Security and Crime Science, and the Head of the Cyber Security Researchers of Waikato (https://crow.org.nz) at the University of Waikato, New Zealand. He is principal investigator of the NZD12.2 mil STRATUS (https://stratus.org.nz) research project (2014-2020) funded by the NZ Ministry of Business Innovation and Employment. His cyber security research aims to return control of data to users through cloud data provenance (tracking and reconstruction), practical homomorphic encryption and situation awareness. He received his PhD and BEng (Hons) from Nanyang Technological University. He is a Fellow of the Cloud Security Alliance, life member of the ACM, and member of the Royal Society of New Zealand, and IEEE.