Models for Solving Multi-Agent Decision Problems

Jerry Aunula

School of Science

Thesis submitted for examination for the degree of Master of Science in Technology.
Espoo 7.10.2023

Supervisor

Prof. Ahti Salo

Advisor

D.Sc. (Tech.) Juho Roponen

Aalto University
School of Science

Copyright © 2023 Jerry Aunula

The document can be stored and made available to the public on the open internet pages of Aalto University.
All other rights are reserved.

Aalto University
School of Science

Aalto University, P.O. BOX 11000, 00076 AALTO
www.aalto.fi
Abstract of the master's thesis

Author Jerry Aunula		
Title Models for Solving Multi-Agent Decision Problems		
Degree programme Mathematics and Operations Research		
Major Systems and Operations Research		**Code of major** SCI3055
Supervisor Prof. Ahti Salo		
Advisor D.Sc. (Tech.) Juho Roponen		
Date 7.10.2023	**Number of pages** 53+10	**Language** English

Abstract

Several ways of solving multi-agent decision problems, e.g., complete information game theory and adversarial risk analysis, have been proposed in the literature. Most approaches, however, cannot accommodate constraints spanning the entire problem or situations in which earlier decisions cannot be recalled when making later ones. Decision Programming is a methodology that helps address these limitations; however, it has thus far only been used to solve single-agent decision problems. This thesis aims to expand Decision Programming to accommodate multi-agent decision problems.

Our approach first transforms the multi-agent problem into several single-agent problems, one for each actor, that are represented with an influence diagram. We solve these problems with the level-k approach, updating the initial distribution for the decisions of each actor iteratively. For each problem at each level, we use Decision Programming to solve the single-agent problems. This process is repeated until we arrive at a convergent solution.

We apply our methodology to two examples, one in critical infrastructure protection and the other in border security. In the first example, our methodology produced results very similar to a traditional adversarial risk analysis method that used node removals. Furthermore, we solved an extension of the first example and one example concerning border security, which both included overarching constraints, non-perfect strategists, and multiple decision alternatives. Overall, our solutions created beneficial results for protecting critical infrastructure and constructing border security portfolios.

Keywords influence diagram, decision programming, adversarial risk analysis, optimization

Aalto University
School of Science

Aalto-yliopisto, PL 11000, 00076 AALTO
www.aalto.fi
Diplomityön tiivistelmä

Tekijä Jerry Aunula

Työn nimi Malleja useita päätöksentekijöitä sisältävien ongelmien ratkaisemiseksi

Koulutusohjelma Mathematics and Operations Research

Pääaine Systems and Operations Research **Pääaineen koodi** SCI3055

Työn valvoja Prof. Ahti Salo

Työn ohjaaja TkT Juho Roponen

Päivämäärä 7.10.2023 **Sivumäärä** 53+10 **Kieli** Englanti

Tiivistelmä

Kirjallisuudessa on esitetty useita tapoja ratkaista usean päätöksentekijän päätösongelmia, kuten complete information -peliteoria ja vastakkainasettelullinen riskianalyysi. Näillä metodeilla ei voi kuitenkaan ratkaista ongelmia, joissa on koko ongelman kattavia rajoitusehtoja tai tilanteita, joissa aikaisempia päätöksiä ei tiedetä tehtäessä myöhempiä päätöksiä. Decision Programming -metodologia kehitettiin ratkaisemaan päätösongelmia, joissa esiintyy näitä haasteita. Tätä metodologiaa on kuitenkin käytetty vain yhden päätöksentekijän päätösongelmien ratkaisuun. Tässä diplomityössä Decision Programming -metodologiaa kehitetään siten, että sillä voidaan ratkaista usean päätöksentekijän päätösongelmia.

Työssä esitetty menetelmä luo ensin usean päätöksentekijän päätösongelmasta yhden päätösongelman jokaiselle päätöksentekijälle. Nämä ongelmat ratkaistaan level-k -viitekehyksellä päivittäen jokaisen päätöksentekijän päätöksille luotua alkujakaumaa iteratiivisesti. Jokaisen tason jokainen päätösongelma ratkaistaan Decision Programming -metodologialla. Menetelmää toistetaan kunnes tasojen tulokset konvergoituvat.

Kehitettyä menetelmää sovelletaan kahteen esimerkkiin, joista ensimmäinen liittyy kriittisen infrastruktuurin suojaamiseen ja toinen rajaturvallisuuteen. Menetelmä antaa näistä ensimmäiseen samankaltaisia tuloksia kuin perinteisempi vastakkainasettelullinen riskianalyysi. Lisäksi työ esittelee ratkaisut jatkokehitetylle versiolle infrastruktuuri-esimerkistä ja rajaturvallisuutta kuvaavasta esimerkistä. Näistä molemmat sisälsivät ongelman kattavia rajoitusehtoja, rajoittunutta rationaalisuutta sekä useita päätösvaihtoehtoja. Saadut tulokset ovat hyödyllisiä infrastruktuurien suojelemiseen sekä rajaturvallisuusportfolioiden kehittämiseen.

Avainsanat vaikutuskaavio, decision programming, vastakkainasettelullinen riskianalyysi, optimointi

Preface

I want to thank my supervisor, Ahti Salo, and advisor, Juho Roponen, for their continual support throughout this thesis project. Furthermore, I want to thank my friends and family for helping me keep smiling despite some difficult times during my studies.

Espoo, 7.10.2023

Jerry Aunula

Contents

1 Introduction

Influence diagrams are a method for compactly representing the essential aspects of decision problems (Olmsted, 1984; Howard and Matheson, 2005). Influence diagrams consist of chance, decision, and utility nodes, with directed arcs between the nodes showing the information relationships of random events, decisions, and outcomes, respectively. Influence diagrams are commonly solved by iteratively eliminating chance and decision nodes from the diagram, possibly using arc reversals (Shachter, 1986, 1988).

Influence diagrams, however, have traditionally been used to solve decision problems in which there is only a single decision-maker. Nevertheless, many problem contexts require an approach concerning multiple decision-makers who are either collaborating or competing. The approach that has received the most recognition in this area is complete information game theory, proposed by Von Neumann and Morgenstern (1947), which was adapted to problems representable by influence diagrams by Koller and Milch (2003). However, the main weakness in this approach is the assumption that decision-makers are perfect strategists, which is unreasonably restrictive (Kahneman and Tversky, 1979). Moreover, assuming complete information decreases accuracy in most real-life problems, as the adversaries in the decision problem can only estimate the utility functions of the other decision-makers.

Rios Insua et al. (2009), among others, presented Adversarial Risk Analysis (ARA) to address these issues. In ARA, uncertainties about the utility functions and probability distributions of other decision-makers are modeled via Bayesian conditional probabilities. The approaches using ARA to solve multi-agent influence diagrams have relied mainly on the aforementioned standard solution methods of influence diagrams (Shachter, 1986). However, the weakness of these solution methods is that they cannot accommodate problems in which earlier decisions cannot be recalled when making later ones, nor problems in which some constraints span the entire problem in that they include states from several nodes of the influence diagram.

For influence diagrams with one decision-maker, Salo et al. (2022) developed Decision Programming, which solves the decision problem by transforming it into a Mixed Integer Linear Programming (MILP) formulation. This approach is able to solve a wide variety of influence diagrams with different types of restrictions, such as overarching constraints and problems where earlier decisions cannot be recalled while making later ones and solving them to optimality. To date, however, this approach has only been used in decision problems with a single decision-maker.

This thesis aims to develop a generalized solution method for solving multi-agent decision problems. Conceptually, we approach the subject similarly to Adversarial Risk Analysis, modeling our knowledge of the opponents' probabilities, utilities, and strategic intelligence as uncertain. Specifically, we use the level-k thinking approach (Stahl and Wilson, 1994). In this approach, all actors at level 0 first solve their view of the multi-agent decision problem without considering what the other actors will do. Subsequently, at each level k, each actor solves their problem assuming the other actors make decisions as if they were level $k-1$ decision-makers. In this approach, we solve the problems of each actor on each level with Decision Programming, allowing us to solve general multi-agent decision problems to optimality with limited memory, problem-spanning constraints, and non-regularity.

The benefits of our approach are four-fold. First, each actor may model their uncertainty about the other actors, not needing to make unrealistic assumptions about common knowledge or perfect strategy. Second, this approach accommodates a wide range of problem and constraint types, which gives flexibility in supporting decision-making. Third, our approach is intuitive: each actor thinks about the likely decisions of other decision-makers and uses this knowledge to optimize their own strategy. Finally, our approach is flexible in that we can solve several versions of the original problem, using various assumptions about, e.g., the likely actions of the opponents and the overall problem structure. Ultimately, we can combine these answers to make as robust a decision recommendation as possible.

This thesis is structured as follows. Section 2 reviews earlier approaches and summarizes the key properties of the Adversarial Risk Analysis and Decision Programming methodologies. Section 3 develops the mathematical basis for solving general multi-agent decision problems with Decision Programming. Section 4 shows illustrative computational examples demonstrating the strengths of our approach. Finally, section 5 concludes the thesis and proposes topics for further research.

2 Background

2.1 Multi-Agent Decision Problems

The mainstream approach for problems involving multiple decision makers is game theory with complete information (Von Neumann and Morgenstern, 1947). It has also been adapted to problems directly representable by influence diagrams by Koller and Milch (2003). Complete information game theory usually aims to find Nash equilibrium solutions, defined as strategies where no decision-maker can improve their own utility without provoking changes to the strategies of other decision-makers. This approach assumes that all decision-makers are perfect strategists, which is, however, to some degree unrealistic: humans are only boundedly rational and prone to many cognitive biases that affect decision-making (Kahneman and Tversky, 1979; Kahneman, 2003). Furthermore, it is common for game-theoretic treatments to assume common knowledge, which is an unreasonably restrictive assumption. For instance, in counterterrorism and cybersecurity topics, the defender rarely knows their adversaries' knowledge and motivations (Rios Insua et al., 2021). Moreover, significant empirical evidence suggests that common game-theoretic approaches are poor at predicting the behavior of real decision-makers (Camerer, 2010; Gintis, 2014).

Adversarial Risk Analysis has been developed to address the issues mentioned above (Rios Insua et al., 2009). ARA uses Bayesian probability distributions for the utility functions and probabilities of the other actors, thus rectifying the restrictive approach of common knowledge used in some game theoretic approaches. It also allows changing the actors' utility functions to accommodate non-traditional utility functions, for example, those suggested by prospect theory (Kahneman and Tversky, 1979; Kahneman, 2003).

ARA distinguishes three types of uncertainty: aleatory, epistemic, and concept (Banks et al., 2015). Aleatory uncertainty refers to the uncertainties of outcomes conditioned by the different possible choices. Epistemic uncertainty concerns the uncertainty of one actor about their opponents' utility functions and other parameters. Finally, concept uncertainty refers to the uncertainty concerning the strategic thinking type of the opponent. Aleatory uncertainty can be included in the problem via chance nodes, posing no difficulties. On the other hand, epistemic uncertainty can be included, e.g., by considering types (Harsanyi, 1967). Thus, actors can view other actors as represented by one of several possible types characterized by different probability and utility distributions. This convention rectifies the common knowledge assumption discussed above.

To address concept uncertainty, ARA proposes four different strategic thinking types, i.e., problem-solving approaches: non-strategic, level-k thinking, game theoretic, and mirroring (Banks et al., 2015). Non-strategic thinking means that the opponents optimize only their own utility functions without considering the opponents' actions. Level-k thinking refers to a layered approach to thinking: "I think that you think that I think that...". Here, k is the highest level of thinking the algorithm accounts for. A level-0 thinker is the same as a non-strategic opponent, as referred to above. A game-theoretic approach is finding Nash equilibrium solutions, assuming the actors are perfect strategists with common knowledge. Finally, the mirroring approach addresses the potentially infinite regress of level-k thinking by assuming that all adversaries model their opponents in the same way, which makes it possible to determine the solution via a mirroring argument. This approach assumes that each opponent's problem structures and solution methods are identical.

These strategic thinking types are proposed to address concept uncertainty and give options to the standard game-theoretic solution methods. For example, it has been shown that the level-k thinking approach (Stahl and Wilson, 1994) produces more accurate predictions for decision-makers' behavior in real-life circumstances (Nagel, 1995). In addition, ARA has been successfully used to solve decision problems in application areas such as Borel games (Banks et al., 2011), piracy (Sevillano et al., 2012), counterterrorism (Rios and Rios Insua, 2012), and cybersecurity (Rios Insua et al., 2021).

ARA approaches commonly use the node elimination approach, for example, in (Rios and Rios Insua, 2012), as developed by Shachter (1986), to solve the multi-agent influence diagrams. However, this approach runs into the same issues as the single decision-maker alternative: it cannot accommodate problems with limited memory, problem-spanning constraints, or multiple value nodes.

2.2 Single-Agent Decision Problems

Since the '80s, influence diagrams have been widely used to visualize and structure decision problems (Olmsted, 1984). As a result, several ways of solving problems representable by influence diagrams have been proposed. Most prominently, Shachter (1986) showed how nodes of the influence diagram can be deleted iteratively, possibly after arc reversals, to arrive at an optimum solution. Additionally, Tatman and Shachter (1990) represent the influence diagram as a corresponding decision tree, which can be solved with dynamic programming. Figure 1 shows an example of a single-agent influence diagram.

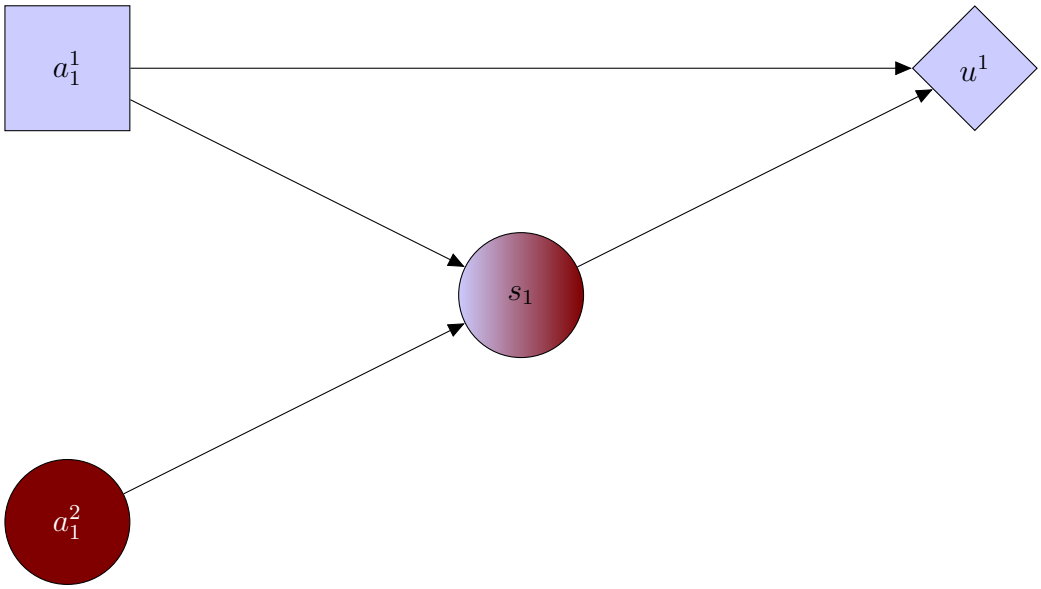

Figure 1: An example of a single-agent influence diagram.

Specically, an influence diagram $G = (N, A)$ is an acyclic, directed graph, comprising of nodes $N = C \cup D \cup V$ and arcs $A = \{(i, j) \,|\, i, j \in N\}$. There are three types of nodes: chance nodes C, representing uncertain outcomes for chance events; decision nodes D, corresponding to decision alternatives for the decision-maker; and value nodes V, which express the consequences resulting from the outcomes of the other nodes over some utility function. Furthermore, to fully represent the decision problem, the specification of an influence diagram includes the states of nodes, conditional probability distributions for the chance nodes, and utility functions. In Figure 1, $C = \{a_1^2, s_1\}$, $D = \{a_1^1\}$, and $V = \{u^1\}$. The arcs $A = \{(i, j) \,|\, i, j \in N\}$ represent dependencies between these nodes. For $j \in C, D, V$, the arcs show how earlier realizations of nodes affect the probability distributions and states of chance events, the decisions to be made, and the utility to be gained, respectively. For example, the arc between nodes s_1 and u^1 shows that the realization of node s_1 affects the utility gained in node u^1.

Despite extensive adoption, these approaches by Shachter (1986) and Tatman and Shachter (1990) cannot solve all kinds of decision problems. For example, Limited Memory Influence Diagrams (LIMIDs), in which earlier decisions are not known when making later ones cannot be solved with the aforementioned methods. Furthermore, issues arise with constraints that span the entire problem base, such as conditional value at risk (CVaR). In stochastic programming, Zhou et al. (2013) and Lauritzen and Nilsson (2001) have proposed some approaches to handle LIMIDs,

which, however, assume a singular value node. Yuan et al. (2010) propose a branch-and-bound algorithm for solving LIMIDs, which assumes the influence diagram to be regular (having a single path traversing all decision nodes).

Decision Programming (Salo et al., 2022) is a general method of solving influence diagrams, which may include several of the issues mentioned above. Decision Programming restructures the influence diagram into a mixed integer linear programming formulation, which is solvable by commercial optimization problem solvers. Decision Programming can represent and subsequently solve problems with multiple value nodes, problem-spanning constraints, limited memory, and non-regularity. It only assumes the decision alternatives must be discrete and finite in number. Here, we summarize the Decision Programming framework, following the notation used by Salo et al. (2022) with some computational enhancements.

We define the information set of a node j as the nodes from which there is an arc to j, i.e., $I(j) = \{i \in N \,|\, (i,j) \in A\}$. In Figure 1, $I(s_1) = \{a_1^1, a_1^2\}$. Here, we assume a singular value node; this framework can be extended to multiple value nodes. Furthermore, we can index the nodes of the influence diagram as $C \cup D = \{1, 2, ..., n\}$ and $V = \{n+1\}$. Each node j has a finite number of discrete states $s_j \in S_j$, which represent the possible realizations of that node. A path $s = (s_1, s_2, ..., s_n)$ is a sequence of states $s_j \in S_j$ of all $j \in C \cup D$. Similarly, we denote a subpath s_J, where $J \subseteq C \cup D$, as a path of states s_j, where $j \in J$. Let us denote a decision strategy, which maps each information state of a decision node into a decision, by $Z = (Z_1, ..., Z_n) \in \mathbb{Z}$, where $Z_j : s_{I(j)} \mapsto s_j$ and $\mathbb{Z}$ is the set of all decision strategies. We call a decision strategy Z compatible with a path s if $Z_j(s_{I(j)}) = s_j \; \forall j \in D$; in other words, if the decision strategy maps each information state of node j to a corresponding state $s_j \in s$ in the path. We denote this compatibility by a binary variable $z(s_j \,|\, s_{I(j)}) \in \{0, 1\}$. More specifically, $Z_j(s_{I(j)}) = s_j \iff z(s_j \,|\, s_{I(j)}) = 1$.

Furthermore, let $\mathcal{U}(s)$ be the utility associated with the consequences implied by the path s; $\mathcal{U}$ is constructed to map to the unit interval. Finally, let us define $\overline{C}$, $\overline{D}$, $\overline{V}$, which are unions of sets C, D, and V and their information sets:

$$\begin{aligned}
\overline{C} &= C \cup \{k \in C \cup D \,|\, \exists\, j \in C : k \in I(j)\} \\
\overline{D} &= D \cup \{k \in C \cup D \,|\, \exists\, j \in D : k \in I(j)\} \\
\overline{V} &= V \cup \{k \in C \cup D \,|\, \exists\, j \in V : k \in I(j)\}.
\end{aligned}$$

We denote $s_{\overline{C}} \in S_{\overline{C}}$ as a subpath that includes all the nodes from $\overline{C}$, but no others.

Subpath $s_{\overline{D}}$ is defined analogously. Let $x(s_{\overline{D}}) \in \{0,1\} = \prod_j z(s_j \mid s_{I(j)})$, denoting whether the decision strategy is compatible with the whole path. Furthermore, let us denote the probability of subpath $s_{\overline{C}}$ as

$$p(s_{\overline{C}}) = \prod_{i \in \overline{C}} \mathbb{P}(X_i = s_i \mid X_{I(i)} = s_{I(i)}),$$

where X_j is a random variable representing the uncertain state of node j. Thus, we arrive at the formulation

$$\max_{Z \in \mathbb{Z}} \sum_{s \in S} x(s_{\overline{D}}) p(s_{\overline{C}}) \mathcal{U}(s_{\overline{V}}) \tag{1}$$

$$\text{subject to} \sum_{s_j \in S_j} z(s_j \mid s_{I(j)}) = 1 \qquad \forall j \in D, s_{I(j)} \in S_{I(j)} \tag{2}$$

$$x(s_{\overline{D}}) \leq \frac{1}{|D|} \sum_{j \in D} z(s_j | s_{I(j)}), \qquad \forall s_{\overline{D}} \in S_{\overline{D}} \tag{3}$$

$$z(s_j \mid s_{I(j)}) \in \{0,1\}, \qquad \forall j \in D, s_j \in S_j, s_{I(j)} \in S_{I(j)} \tag{4}$$

$$x(s_{\overline{D}}) \in \{0,1\}, \qquad \forall s_{\overline{D}} \in S_{\overline{D}}. \tag{5}$$

Here, (1) is the expected utility, constraints (2) and (3) enforce decision strategy compatibility, and (4) and (5) give the range for variables x and z.

3 Methodological Development

We begin by analyzing the strengths and weaknesses of ARA's four strategic thinking types presented in Section 2.1: complete information game theory, mirroring, non-strategic thinking, and level-k thinking. Although complete information game theory is widely used in multi-agent decision theory and capable of quickly calculating equilibrium solutions, it does not accurately reflect decision-making, mainly because of its perfect rationality assumption (Kahneman and Tversky, 1979; Kahneman, 2003). On the other hand, mirroring partly shares the strengths of complete information game theory, but it, in turn, assumes that the actors share the same problem structure. Level-k thinking is computationally more demanding, but it has been shown to produce accurate decision-making predictions (Stahl and Wilson, 1994). It is also versatile, as it can accommodate different levels of intelligence and strategic skill. Furthermore, non-strategic thinking directly corresponds to level-0 thinking in the level-k thinking framework.

Based on the above, most real-life problems can be modeled via the level-k thinking approach. The advantage of this approach is that if all the actors are modeled as level-k thinkers, the resulting methodology is applicable to any general multi-agent decision problem with several actors, non-common knowledge, problem-spanning constraints, and simultaneous and sequential decisions.

Let us have m actors, which we denote by a^l, $l = 1, ..., m$. We shall be supporting the decision-making of actor a^1. Let k_l denote the strategic level of thinking of actor a^l. To optimize the strategy of actor a^1 with respect to the other actors, let

$$k_1 = \max_{l \neq 1}\{k_l\} + 1; \tag{6}$$

thus, we assume actor a^1 thinks at least one level deeper than the other actors. Figure 2 offers an example of a multi-agent influence diagram. Here, the actors are a^1 and a^2. Nodes representing decisions and utilities, which are actor-specific, are colored in blue for a^1 and red for a^2. Furthermore, chance nodes, which are common for both actors, are colored in a blue-red gradient. Both actors have a single decision; actor a^1 decides a^1_1, and actor a^2 decides a^2_1. Furthermore, s_1 is a chance node, and u^1 and u^2 are utility nodes. If we assume the level of thinking of actor a^2 to be $k_2 = 3$, Equation (6) gives $k_1 = 4$. The levels of thinking k_l of the actors can depend on various factors, including cognitive ability, personality traits, and situational factors. In reality, the level of thinking rarely exceeds four (Gill and Prowse, 2016). Uncertainty about the levels of the other actors can be accommodated by weighing

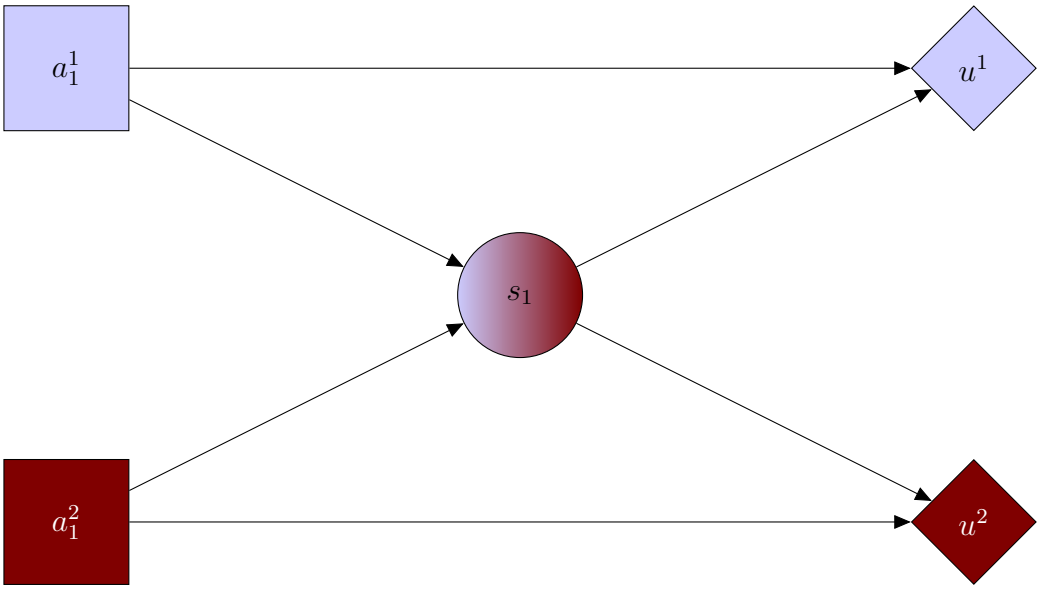

Figure 2: An example of a multi-agent influence diagram.

the solutions of different levels to arrive at a recommendation, as our methodology nevertheless computes these solutions. In addition to Equation 6, the maximum level of thinking can also be chosen as one that achieves convergence in the actors' optimal decisions. In this thesis, we use $k_l = 4\ \forall l = 1, ..., m$, as we aim to study the convergence of our examples thoroughly.

Additionally, we need to estimate the parameters u^l and $p^l(\cdot|a^j)$; thus, the utility functions and probabilities all actors $l, j = 1, ..., m$ assign to their chance nodes, conditioned on the actions of the other actors (or, more specifically, what actor a^1 thinks the other actors assign to themselves and actor 1). These would include, for example, how actor a^1 perceives the probability distribution $p(s_1|a_1^1, a_1^2)$,, i.e., how the random outcome s_1 depends on her decision a_1^1 and actor a^2's decision a_1^2.

Next, we construct the influence diagrams of the actors' problems by transforming the other actors' decision nodes into chance nodes, removing their utility and barren nodes (nodes without successors), and finally considering the possible asymmetry in the problem structure (e.g., private information). These influence diagrams and their corresponding single-agent decision problems are denoted by $G^l, l = 1, ...m$. Additionally, although not shown in the influence diagrams, the problems G^l may include the probabilistic characterization of the type of actors (Harsanyi, 1967). This is included to accommodate uncertainty, as actors do not know the other actors' precise utility functions and probabilities. For example, the previously shown Figure 1 in Section 2.2 shows the influence diagram G^1 for the decision problem represented

by Figure 2. The type of actor a^2, as described above, denoted by a_0^2, would be a chance node with arcs from it to actor 2's other decisions, namely a_1^2.

Let index k denote the different levels of thinking to be solved, ranging from 0 to k_1. Then, for each actor a^l, let us denote their decision strategy by Z^{lk} at level k, as defined in Section 2.2. Furthermore, let C_j^l denote the set of chance nodes of the influence diagram G^l, which correspond to the decisions of actor $a^j, j \neq l$.

We begin from the 0^{th} level of thinking. Here, we construct an initial distribution for the decision strategies of all the actors, Z^{li}. Then, each actor solves their decision problem assuming the strategy of actor l for each $l = 1, ..., m$ is Z^{li}. As the 0th level does not depend on the actions of the other actors, this initial distribution can be constructed in several ways. These include maximum entropy, which assigns an equal probability to all actions; maximin, which maximizes the minimum utility the actor can hope to achieve; and minimax regret, which minimizes the maximum difference in utility between the actor's choice and the optimal choice. In this thesis, we use the maximum entropy initial distribution, as it is usually computationally the least demanding. Furthermore, the iterative nature of the level-k approach decreases the impact of the initial distribution's accuracy.

At the end of the k^{th} level, for each actor l, we substitute the solved decision strategy Z^{lk} into C_j^l for each $j = 1, ..., m, j \neq l$; thus, the decisions actor a^l makes become degenerate chance nodes (only one realization for each information set with probability one) in the influence diagrams of all the other actors. In the $(k+1)^{th}$ level, we solve the updated problems G^l for each actor, optimizing the next decision strategy, $Z^{l(k+1)}$. If, however, the level $k+1$ exceeds the maximum level of thinking k_l of a given actor l, the decision strategy for that actor stays unchanged from that point on.

Finally, after iterating through the k_1 levels, the decision strategy Z^{1k_1} is the newest strategy for supporting actor a^1. This is an equilibrium solution if, for each $l = 1, ..., m$, $Z^{lk_1} = Z^{l(k_1-1)}$, i.e., the decision strategy stays unchanged for the last two iterations. The steps of the algorithm are summarized below. The computational complexity of this algorithm is proportional to $t \cdot k \cdot m$, where t is the time to solve a corresponding single decision-maker decision problem, k is the number of levels of thinking, and m is the number of actors.

1. Calculate $k_1 = \max_{l \neq 1}\{k_l\}+1$, or decide k_l based on convergence considerations.
2. For all actors $l = 1, ..., m$: construct G^l, u^l, $p^l(\cdot|a^j)$, Z^{li} for all $l = 1, ..., m$ actors. Initialize $k = 0$.
3. For all actors $l = 1, ..., m$: If $k < k_l$, solve G^l and substitute the solved Z^{lk} into C_l^j for each $j = 1, ..., m$. Else, $Z^{lk} = Z^{lk-1}$.
4. Move to next level $k = k + 1$.
5. Repeat steps three and four until $k = k_1$.
6. Return, for each actor, Z^{lk_1} and whether $Z^{lk_1} = Z^{l(k_1-1)}$.

4 Examples

4.1 Critical Infrastructure Protection

4.1.1 Problem Description

Our first example is on Critical Infrastructure Protection (CIP), described by González-Ortega et al. (2019). We first solve the original problem using our methodology, as laid out in Section 3, and then extend the example by adding resource constraints and more granular choices.

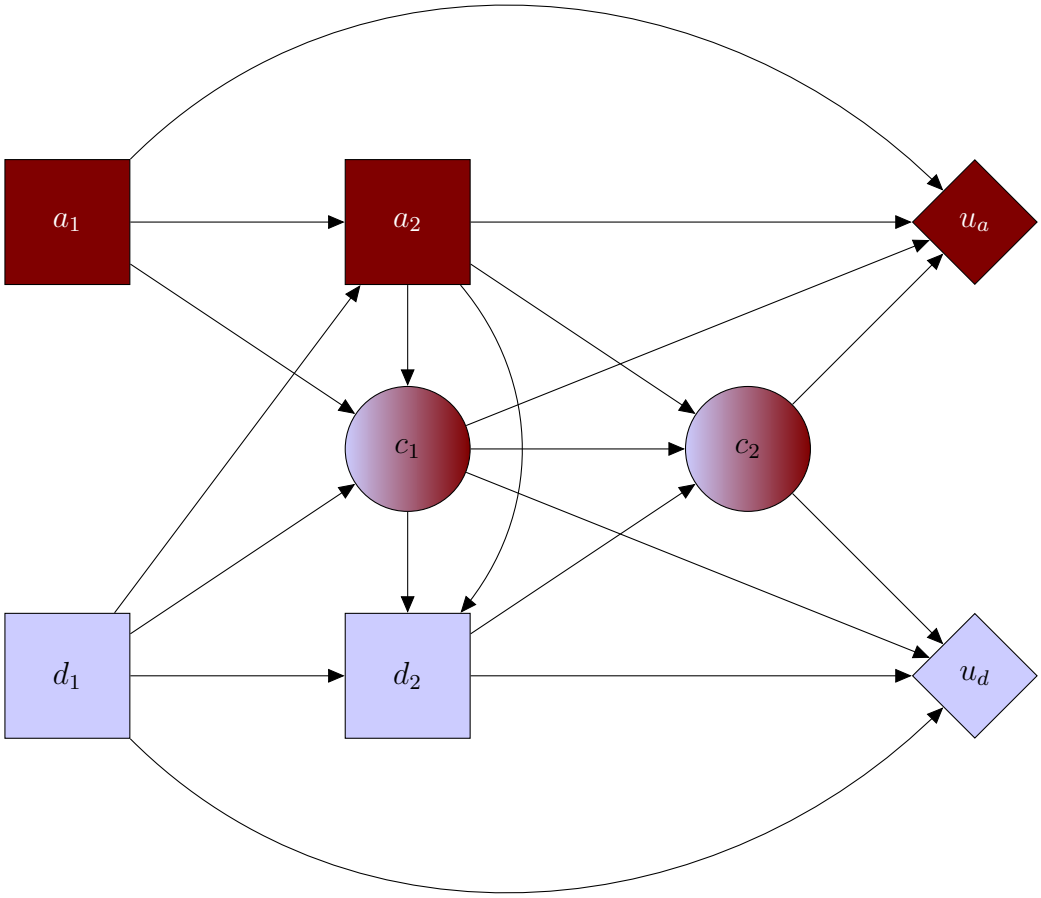

Figure 3: CIP influence diagram.

		Interpretation	State space
Node	a_1	Infiltration	0, 1 (No, Yes)
	d_1	Reinforcement	0, 1 (No, Yes)
	a_2	Attack	0, 1 (No, Yes)
	c_1	Service shortage	0, 0.5, 1 (in days)
	d_2	Recovery measures	0, 1 (No, Yes)
	c_2	Shortage reduction	0, 0.25, 0.5, 0.75, 1 (in days)
	u_a	Apollo's utility	
	u_d	Daphne's utility	

Table 1: Nodes, their interpretations, and their state spaces in the original CIP problem.

The CIP problem has two actors: Daphne and Apollo. Daphne represents a governmental entity aiming to protect a critical infrastructure via reinforcement and recovery measures. Apollo represents a terrorist organization aiming to disrupt Daphne's operations by attacking the infrastructure using intelligence gained by infiltration. The influence diagram for this problem is in Figure 3. Here, Daphne's decisions are denoted by d_i, colored in blue, and Apollo's by a_i, colored in red. The nodes of the influence diagram, their interpretations, and state spaces are summarized in Table 1.

First, Apollo and Daphne make decisions a_1 and d_1 without information about each other's actions. Decision a_1 is whether or not to infiltrate the infrastructure to gain intelligence (yes/no), and d_1 is the decision to reinforce the infrastructure's security (yes/no). Apollo's second decision a_2 is whether or not to attack the infrastructure (yes/no). The consequences of this attack, measured by days of service shortage (0/0.5/1), are represented by chance node c_1. These consequences depend on decisions a_1, d_1, and a_2.

After observing the service shortage c_1, Daphne decides d_2 (yes/no) whether or not to implement recovery measures to decrease the shortage observed in c_1. The consequences of these recovery measures are represented by chance node c_2, whose states indicate the decrease in shortage (0/0.25/0.5/0.75/1) in days. Finally, utility nodes u_a and u_d represent the utilities of Apollo and Daphne, respectively. Because we are analyzing the problem from Daphne's perspective, we start with her influence diagram, as seen in Figure 4.

Apollo's attack a_2 results in the infrastructure shortage, chance node c_1, the probabilities of which are as follows. $P(c_1 = 0|a_2 = 0) = 1$; the infrastructure

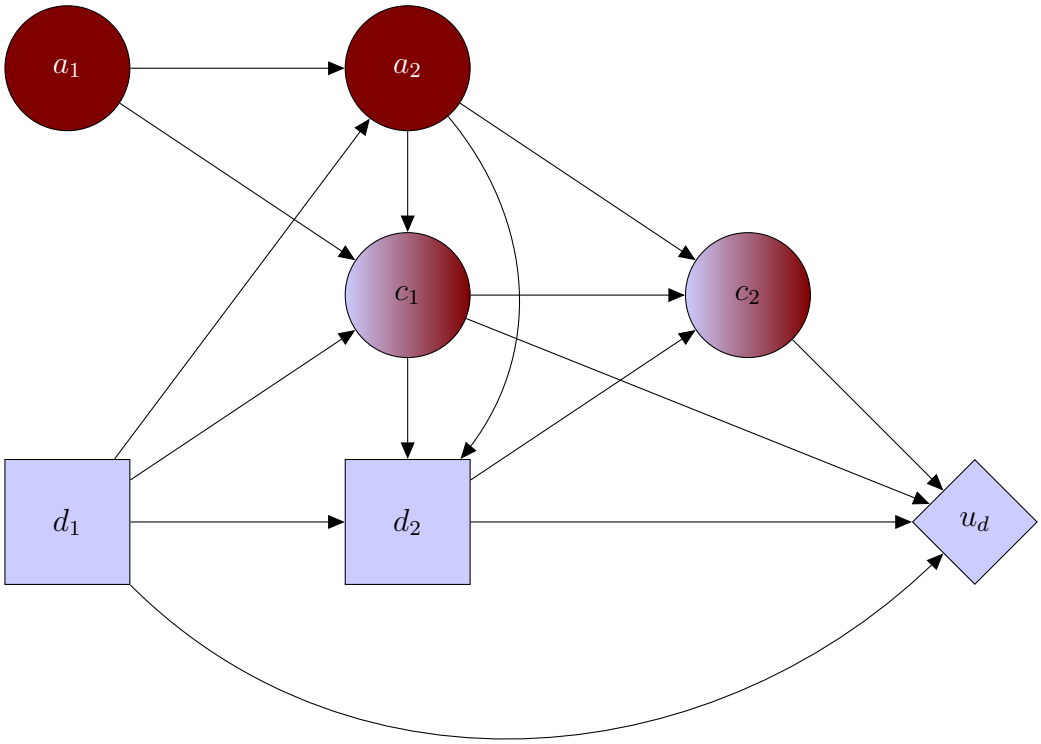

Figure 4: Daphne's influence diagram.

has no service shortage if Apollo does not attack. Furthermore, Table 2 shows the probability distribution $p(c_1|a_2 = 1)$, while Figure 5 shows c_1's cumulative distribution function, based on the values of Table 2. Furthermore, Table 3 shows the probability distribution for chance node c_2. As there is no reduction in shortage ($c_2 = 0$) when no recovery measures are implemented ($d_2 = 0$), and no reduction is needed when Apollo does not attack ($a_2 = 0$), the table only shows cases where $a_2 = d_2 = 1$. The corresponding cumulative distribution is in Figure 6.

		(Reinforce, Inflitrate) = (d_1, a_1)			
		(0,0)	(0,1)	(1,0)	(1,1)
Shortage - c_1	0	0.3	0.15	0.40	0.25
	$\frac{1}{2}$	0.45	0.55	0.40	0.50
	1	0.25	0.30	0.20	0.25

Table 2: Daphne's probability distribution $p(c_1|a_2 = 1)$ (González-Ortega et al., 2019).

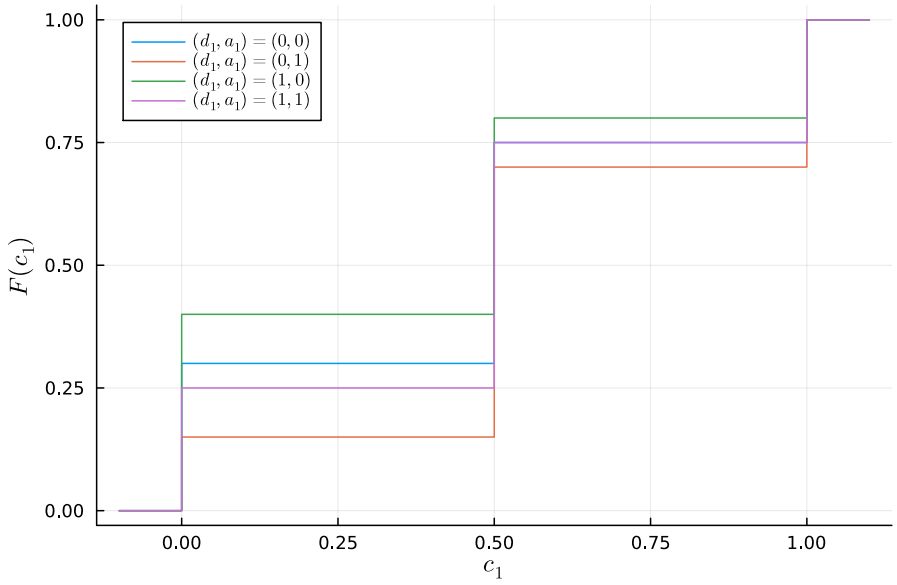

Figure 5: Plot of service shortage c_1's cumulative distribution function $F(c_1)$ for different decision combinations of reinforcement measures d_1 and infiltration a_1.

		Shortage - c_1		
		0	$\frac{1}{2}$	1
Shortage reduction - c_2	0	1.00	0.20	0.10
	$\frac{1}{4}$	0.00	0.50	0.15
	$\frac{1}{2}$	0.00	0.30	0.30
	$\frac{3}{4}$	0.00	0.00	0.25
	1	0.00	0.00	0.20

Table 3: Daphne's probability distribution $p(c_2|a_2 = 1, d_2 = 1)$ (González-Ortega et al., 2019).

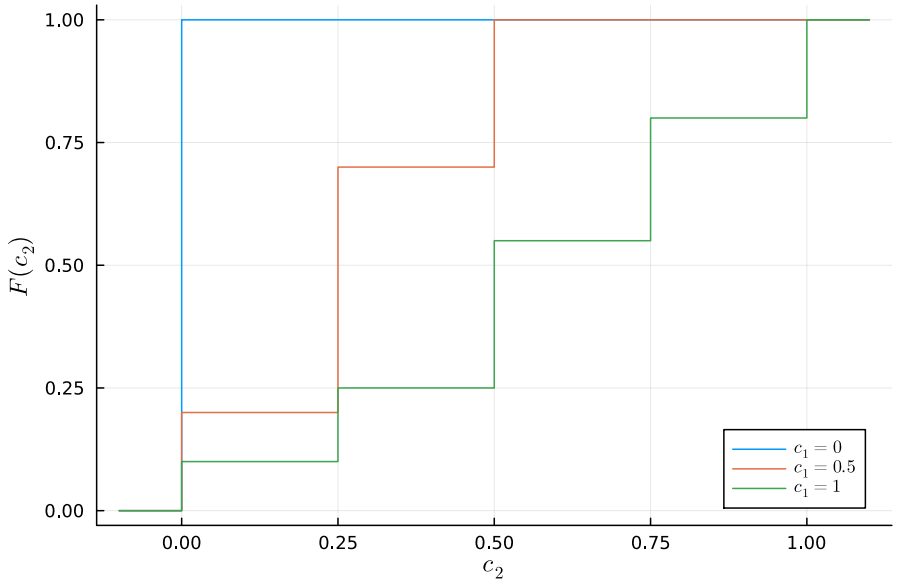

Figure 6: Plot of shortage reduction c_2's cumulative distribution function $F(c_2)$ for different realizations of service shortage c_1.

Finally, we assess Daphne's utility function. Security reinforcement costs $m_{d,1} = 5M€$; relief budget is $m_{d,2} = 10M€$; and each shortage day costs $m_{d,3} = 40M€$. These three costs constitute Daphne's value function

$$v_d(d_1, d_2, c_1, c_2) = -m_{d,1}d_1 - m_{d,2}d_2 - m_{d,3}(c_1 - c_2).$$

We assume that Daphne is constant risk averse in that her utility function is

$$u_d(d_1, d_2, c_1, c_2) = 1 - \exp\left[-\lambda_d(v_d + c_d)\right],$$

where $\lambda_d = 0.06$ is Daphne's risk aversion coefficient and $c_d = 55$ is an adjusting constant to normalize the utility function to the unit interval, i.e., $0 \leq u_d \leq 1$. Here, all the parameters concerning probabilities and utility functions are the same as in the original example description by González-Ortega et al. (2019).

We next focus on Apollo's problem, or more specifically, Daphne's perspective of Apollo's problem. Figure 7 shows Apollo's influence diagram. Daphne is uncertain about Apollo's precise probabilities and utilities; however, she assumes they have the same problem structure, i.e., Apollo's influence diagram is that of Figure 7. Daphne believes Apollo's probabilities to be similar to hers, with a small uncertainty around them. González-Ortega et al. (2019) use the Dirichlet distributions to account for this uncertainty. We, however, define different types (Harsanyi, 1967) that Apollo may have, incorporating a similar uncertainty structure as that of Dirichlet distributions.

We aim to see whether our solution converges to a similar one as in (González-Ortega et al., 2019), even with a different approach to modeling Daphne's uncertainty.

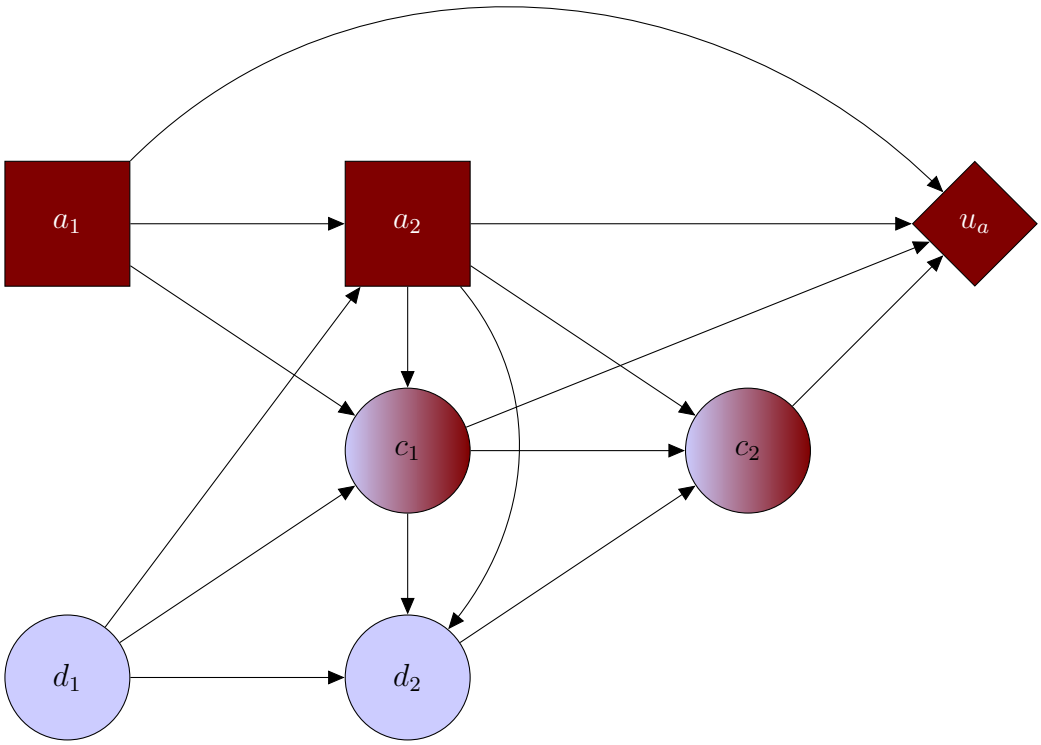

Figure 7: Apollo's influence diagram.

We give three possible types to Apollo: $a_0 = 1$, 2, and 3. Type $a_0 = 1$ has the same probabilities as Daphne, and his utility function is the average utility function in (González-Ortega et al., 2019), specified explicitly later. Type $a_0 = 2$ has more expensive decisions and is more risk-prone. Additionally, he has probabilities more in favor of outcomes most favorable for Apollo. Type $a_0 = 3$ has less expensive decisions, is less risk-prone, and has probabilities more in favor of favorable outcomes for Daphne. These types can be considered a chance node a_0, with $P(a_0 = 1) = P(a_0 = 2) = P(a_0 = 3) = 1/3$, affecting Apollo's decisions on infiltrating (a_1) and attacking (a_2) the infrastructure. The reasons for not using types for Daphne are two-fold. First, the effect of introducing types for Daphne is not that large, as it can be argued that it is easier for an outsider to infer the utilities of governments than those of terrorists. Second, the original example (González-Ortega et al., 2019) does not incorporate uncertainties for Daphne's distributions or utilities.

The probabilities of Apollo's types are calculated according to equations (7)-(10). Here, α_1^i is the Dirichlet parameter for outcome $c_i = 1$, α_{sum}^i is the sum of the Dirichlet

parameters for all outcomes of c_i, P^i_{old} are the probabilities of Apollo's type $a_0 = 1$, and P^i_{new} are the probabilities for Apollo's types $a_0 = 2$ and $a_0 = 3$. These Dirichlet parameters are enumerated in (González-Ortega et al., 2019). The probabilities of all other outcomes are scaled down such that their ratio stays constant. Table 4 shows an example of the probability distribution of service shortage c_1 for Apollo's type $a_0 = 2$, and all of the probabilities, as calculated with Equations (7)-(10), are tabulated in Appendix A.1 in Tables A1, A2 and A3.

$$P^1_{new}(c_1 = 1) = P^1_{old}(c_1 = 1) + \sqrt{\frac{(\alpha^1_1/\alpha^1_{sum})(1 - \alpha^1_1/\alpha^1_{sum})}{\alpha^1_{sum} + 1}}, \quad \text{when } a_0 = 2 \tag{7}$$

$$P^1_{new}(c_1 = 1) = P^1_{old}(c_1 = 1) - \sqrt{\frac{(\alpha^1_1/\alpha^1_{sum})(1 - \alpha^1_1/\alpha^1_{sum})}{\alpha^1_{sum} + 1}}, \quad \text{when } a_0 = 3 \tag{8}$$

$$P^2_{new}(c_2 = 1) = P^2_{old}(c_2 = 1) + \sqrt{\frac{(\alpha^2_1/\alpha^2_{sum})(1 - \alpha^2_1/\alpha^2_{sum})}{\alpha^2_{sum} + 1}}, \quad \text{when } a_0 = 2 \tag{9}$$

$$P^2_{new}(c_2 = 1) = P^2_{old}(c_2 = 1) - \sqrt{\frac{(\alpha^2_1/\alpha^2_{sum})(1 - \alpha^2_1/\alpha^2_{sum})}{\alpha^2_{sum} + 1}}, \quad \text{when } a_0 = 3 \tag{10}$$

		(Reinforce, Inflitrate) = (d_1, a_1)			
		(0,0)	(0,1)	(1,0)	(1,1)
Shortage - c_1	0	0.29	0.15	0.38	0.24
	$\frac{1}{2}$	0.43	0.53	0.38	0.48
	1	0.28	0.32	0.23	0.27

Table 4: Type 2 Apollo's probability distribution $p(c_1|a_2 = 1)$.

Finally, we assess Apollo's utility function. Daphne estimates the infiltration to cost type 1 Apollo $m_{a,1} = 1M€$, attacking to cost $m_{a,2} = 5M€$, and each day of service shortage to profit him $m_{a,3} = 35M€$. These parameters give the value function

$$v_a = -m_{a,1}a_1 - m_{a,2}a_2 + m_{a,3}(c_1 - c_2).$$

Assuming Apollo to be constant risk-prone, we get the corresponding utility function

$$\exp\left[\lambda_a(v_a + c_a)\right],$$

where $c_a = -\max v_a$ adjusts the utility function to the unit interval. The parameters for the utility functions for Apollo's three types are in Table 5. Correspondingly, Apollo's utility functions are in Figure 8.

		Type (a_0)		
		1	2	3
Parameters	$m_{a,1}$	1	2	0.5
	$m_{a,2}$	5	7	4
	$m_{a,3}$	35	35	35
	λ_a	0.06	0.07	0.05

Table 5: Parameters for the utility functions of Apollo's types.

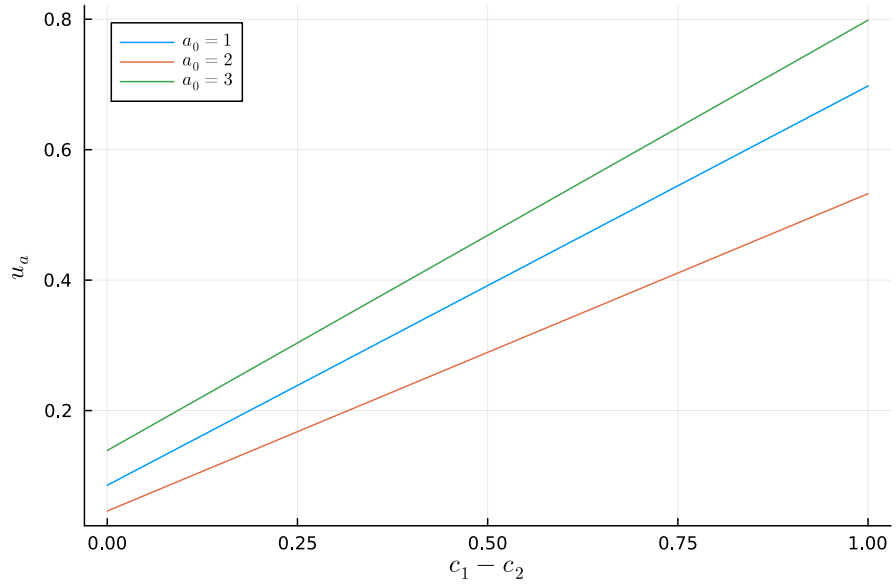

Figure 8: Apollo's utility functions with $a_1 = a_2 = 1$.

4.1.2 Solution

Based on the above problem description, we solve the CIP problem using the solution methodology presented in Section 3. Thus, we start from the 0^{th} level of thinking, using the maximum entropy initial distribution: $p(k_i) = 1/n(k)$, where k_i is a particular state of node k, and $n(k)$ is the number of states of node k. In other words, we assign an equal probability to all possible states of all nodes to construct an initial distribution for both Daphne's and Apollo's decisions. We solve until the 4^{th} level of thinking to arrive at an overall decision recommendation for Daphne, as this is a common upper bound for the level of thinking in real-life examples (Gill and Prowse, 2016), and probably a suitably high level for our methodology to achieve convergence. We used Julia as our programming platform and HiGHS as our solver (Huangfu and Hall, 2018).

Apollo's optimal strategy at level 4 for decisions to infiltrate (a_1) and attack (a_2) are in Tables 6 and 7. Thus, Apollo's optimal strategy is to infiltrate ($a_1 = 1$) the

infrastructure with types 1 and 3 and not infiltrate ($a_1 = 0$) with type 2, regardless of whether Daphne reinforces or not. Then, decision $a_2 = 1$ is always optimal; Apollo always attacks the infrastructure. For the sake of completeness, the tables also show strategies that are not included in the optimal solution. These have been reasoned by using the strategies utilized by Apollo in lower levels of thinking. However, the cells relating to the equilibrium strategy are colored in blue.

		Type = a_0		
		1	2	3
Reinforce = d_1	0	1	0	1
	1	1	0	1

Table 6: Apollo's optimal strategy for the infiltration decision a_1.

		Type = a_0		
		1	2	3
(Reinforce, Infiltrate) = (d_1, a_1)	(0,0)	0	1	0
	(0,1)	1	0	1
	(1,0)	0	1	0
	(1,1)	1	0	1

Table 7: Apollo's optimal strategy for the decision to attack a_2.

Daphne's optimal strategy is not to reinforce the infrastructure ($d_1 = 0$); as this decision is first in the influence diagram, its strategy is binary. Furthermore, the strategy for the recovery measures d_2 is in Table 8; Daphne mitigates the shortage ($d_1 = 1$) if and only if Apollo attacks and achieves full shortage to the infrastructure.

		(Reinforce, Attack) = (d_1, a_2)	
		(0,0)	(0,1)
Shortage = c_1	0	0	0
	1/2	0	0
	1	0	1

Table 8: Daphne's optimal strategy for the recovery measures d_2.

Tables 9 and 10 show how Daphne's and Apollo's expected decisions and utility vary as functions of the level of thinking k. Comparing our solution to that of González-Ortega et al. (2019), our optimal strategy for the reinforcement decision d_1 is different ($d_1 = 0$ compared to $d_1 = 1$), but the same for the recovery measures d_2. Furthermore, González-Ortega et al. (2019) achieved an optimal utility of 0.900 for Daphne, while ours is 0.860. Both of these differences can be explained by our

differing approach to modeling Daphne's uncertainty about Apollo's probabilities. First, the expected utilities for Daphne with the infiltration decisions $d_1 = 1$ and $d_1 = 0$ differ only by 0.4%, which explains how a slight change in parametrization can change this decision. Second, our approach underestimates the magnitude of Daphne's uncertainty about Apollo; we have $P(a_2 = 1) = 1$, while González-Ortega et al. (2019) has $P(a_2 = 1) \in [0.41, 0.66]$ depending on decisions a_1 and d_1. This results in Apollo always attacking in our approach, decreasing Daphne's expected utility. We see that convergence is achieved on the first level of thinking, indicating an equilibrium solution. We perform a rudimentary sensitivity analysis to analyze further how the optimal decision strategies vary as we change initial distributions and utility functions.

		$\mathbb{E}[P(d_1 = 0)]$	$\mathbb{E}[P(d_2 = 0)]$	$\mathbb{E}(u_d)$
Level k	0	1.000	0.863	0.913
	1	1.000	0.717	0.860
	2	1.000	0.717	0.860
	3	1.000	0.717	0.860
	4	1.000	0.717	0.860

Table 9: Expected values for Daphne's decisions and utility for different levels of thinking.

		$\mathbb{E}[P(a_1 = 0)]$	$\mathbb{E}[P(a_2 = 0)]$	$\mathbb{E}(u_a)$
Level k	0	0.333	0.000	0.175
	1	1.000	0.000	0.160
	2	1.000	0.000	0.160
	3	1.000	0.000	0.160
	4	1.000	0.000	0.160

Table 10: Expected values for Apollo's decisions and utility for different levels of thinking.

We first solved the problem with two different initial distributions of minimum entropy: $(a_1, a_2) = (0, 0)$ and $(1, 1)$ for Daphne's problem, and correspondingly $(d_1, d_2) = (0, 0)$ and $(1, 1)$ for Apollo's problem. The optimum solutions stayed unchanged for both initial distributions, suggesting the equilibrium described above might be the only one. Furthermore, we tested how much parameters $m_{d,3}$ and $m_{a,3}$ had to be changed to alter the optimum solutions of Daphne and Apollo, respectively. We varied these parameters only, as the parameters of the problems are meaningful only in relation to each other; the absolute cost of shortage does not affect the optimal strategy, only its cost relative to the cost of reinforcing and mitigating. Starting

with Daphne, the stable region for parameter $m_{d,3}$ was $m_{d,3} \in [19.3, 42.4]$; thus, this parameter, keeping all other things equal, needed to be lowered below 19.3 or raised above 42.4 to change the optimum solution. The corresponding stable region for Apollo was $m_{a,3} \in [18.2, 47.3]$. Comparing these to the original values of $m_{d,3} = 40$ and $m_{a,3} = 35$, the stable region as percentage changes of the original values are

$$\Delta m_{d,3} \in [-51.2\%, +6.0\%]$$
$$\Delta m_{a,3} \in [-48.0\%, +35.1\%].$$

Thus, the problem's parameters are relatively stable for Apollo but not for Daphne, in the direction of valuing lower shortages more than saving resources on mitigation. However, a 6 percent increase only makes Daphne mitigate a half-day shortage instead of not mitigating it, so the difference is not significant. All in all, the problem parameters and solution equilibrium seem to be stable.

4.1.3 Extended Problem Description

Next, we extend the original Critical Infrastructure Protection problem statement in two ways. First, we further granularize Daphne's and Apollo's possible choices, allowing for a wider variety of strategies. Specifically, we add a third alternative to each of the four decisions of infiltration, reinforcement, attack, and recovery measures (a_1, d_1, a_2, and d_2), changing the state space of each of these nodes to $\{0, 1/2, 1\} = \{\text{No, Some, Yes}\}$. The costs of these decisions are in Table 11. For this extended version, we denote the functions corresponding to the previous section's parameters $m_{i,j}$ as $M_{i,j}$; for example, from Table 11, $M_{d,2}(1/2) = 7$. Furthermore, the nodes and state spaces of the extended CIP problem are summarized in Table 12.

		Decision alternatives		
		0	1/2	1
Decision	d_1	0	3	5
	d_2	0	7	10
Type 1	a_1	0	0.8	1
Type 2	a_1	0	1.6	2
Type 3	a_1	0	0.4	0.5
Type 1	a_2	0	2	5
Type 2	a_2	0	3	7
Type 3	a_2	0	1.5	4

Table 11: Costs ($M€$) of each decision.

		Interpretation	State space
Node	a_1	Infiltration	0, 1/2, 1 (No, Some, Yes)
	d_1	Reinforcement	0, 1/2, 1 (No, Some, Yes)
	a_2	Attack	0, 1/2, 1 (No, Some, Yes)
	c_1	Service shortage	0, 0.5, 1 (in days)
	d_2	Recovery measures	0, 1/2, 1 (No, Some, Yes)
	c_2	Shortage reduction	0, 0.25, 0.5, 0.75, 1 (in days)
	u_a	Apollo's utility	
	u_d	Daphne's utility	

Table 12: Nodes, their interpretations, and their state spaces in the extended CIP problem.

These costs have been chosen on the assumption that the cost growth of investments is usually concave (marginal cost of investment decreases with each added investment), as each decision investment generally consists of both a fixed cost and a variable cost. The convex growth (marginal cost of investment increases with each added investment) of the decision to attack a_2 is based on the assumption of Apollo's limited resources and the need for further investments on his part to accommodate a full-scale attack.

Second, we add overarching budget constraints to both Daphne and Apollo. These constraints are $m_{a,max} = 5, 7, 4M€$ for types $a_0 = 1$, $a_0 = 2$ and $a_0 = 3$, respectively. Similarly, $m_{d,max} = 13M€$. In other words, $m_a = M_{a,1}(a_1) + M_{a,2}(a_2) \leq m_{a,max}$ for Apollo, and $m_d = M_{d,1}(d_1) + M_{d,2}(d_2) \leq m_{d,max}$. These costs have been chosen to provide a meaningful constraint for the possible decisions of both actors. After these extensions, the new parametrizations and utility functions of Apollo and Daphne are set out below.

Daphne has similar probabilities as in the original version of the CIP problem, accommodating the changes brought by adding more states to decisions to reinforce and mitigate d_1 and d_2. The probabilities of states 1/2 are calculated as the averages of corresponding probabilities of states 0 and 1. Thus, for example,

$$\begin{aligned} &P(c_1 = 1 | d_1 = 1/2, a_1 = 1, a_2 = 1) \\ &= \frac{1}{2} \left(P(c_1 = 1 | d_1 = 0, a_1 = 1, a_2 = 1) + P(c_1 = 1 | d_1 = 1, a_1 = 1, a_2 = 1) \right). \end{aligned}$$

This linearity assumption is justified because the inherent non-linearity in this problem is transferred into the decision costs. Thus, the states 1/2 are defined such

that the probabilities can be calculated using averages, showing the non-linearity only in the costs, which are laid out in Table 11. Table 13 is an example of the probability distribution $p(c_1|a_1 = 1, a_2 = 1)$; the rest of the probabilities can be found in Appendix A.1, in Tables A1, A2 and A3. Figure 9 shows the corresponding cumulative distribution function of node c_1.

		Reinforce $= d_1$		
		0	1/2	1
Shortage - c_1	0	0.15	0.20	0.25
	$\frac{1}{2}$	0.55	0.53	0.50
	1	0.3	0.28	0.25

Table 13: Daphne's probability distribution $p(c_1|a_1 = 1, a_2 = 1)$.

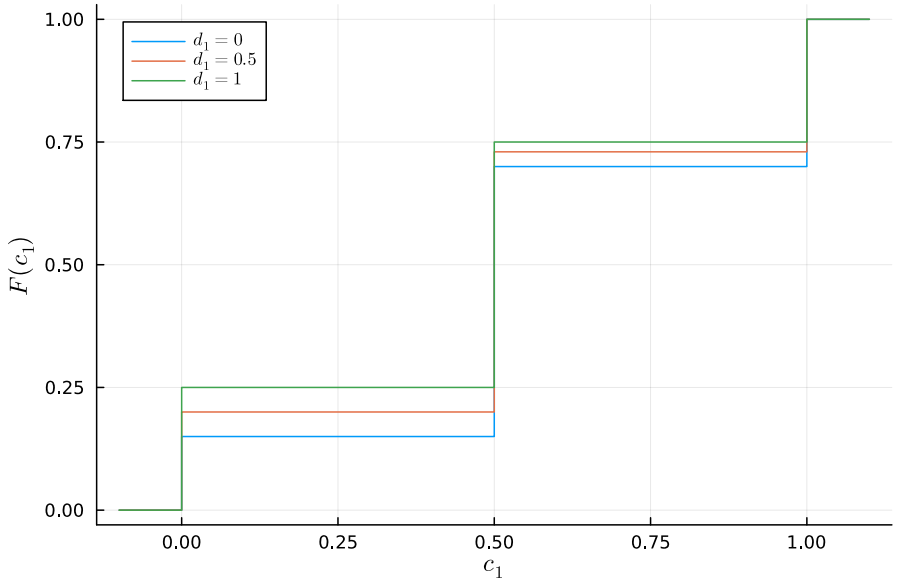

Figure 9: Plot of service shortage c_1's cumulative distribution function $F(c_1)$.

Daphne's value function is

$$v_d(d_1, d_2, c_1, c_2) = -M_{d,1}(d_1) - M_{d,2}(d_2) - m_{d,3}(c_1 - c_2),$$

where $M_{d,1}(d_1)$ and $M_{d,2}(d_2)$ are given by Table 11, and $m_{d,3}$ is defined similarly as in Section 4.1.1. This value function gives Daphne the utility function

$$u_d(d_1, d_2, c_1, c_2) = 1 - \exp\left[-\lambda_d(v_d + c_d)\right],$$

where $\lambda_d = 0.06$ is the risk aversion coefficient, and c_d is a normalizing constant to restrict u_d to the unit interval.

We continue using similar types for Apollo as in Section 4.1.1. The tabulated probabilities for all different states of the service shortage c_1 and decrease in shortage c_2 for Apollo's three types are in Appendix A.1, in Tables A1, A2 and A3. Apollo's value and utility functions are

$$v_a(a_1, a_2, c_1, c_2) = -M_{a,1}(a_1) - M_{a,2}(a_2) + m_{a,3}(c_1 - c_2)$$
$$u_a(a_1, a_2, c_1, c_2) = \exp\left[\lambda_a(v_a + c_a)\right],$$

where c_a are normalizing constants.

4.1.4 Extended Solution

We solve the extended problem statement similarly as in Section 4.1.2. We also use the maximum entropy principle in constructing the initial probability distribution. Tables 14, 15, and 16, show Daphne's and Apollo's optimal decisions. In the equilibrium, the optimal reinforcement decision d_1 for Daphne is no reinforcement ($d_1 = 0$). Apollo then infiltrates fully for Types 1 and 3 ($a_1 = 1$) and does not infiltrate for Type 2 ($a_1 = 0$). Then, Apollo's optimal strategy is always to perform a half-attack ($a_2 = 1/2$), and Daphne never mitigates the shortage ($d_2 = 0$). Furthermore, Tables 17 and 18 show Daphne's and Apollo's expected decisions and utilities for the levels of thinking $k = 0, ..., 4$.

		Shortage $= c_1$		
		0	1/2	1
(Reinforce, Attack) $= (d_1, a_2)$	(0,0)	0	-	-
	(0,1/2)	0	0	0
	(0,1)	0	0	1

Table 14: Daphne's optimal strategy for decision to mitigate d_2.

		Type $= a_0$		
		1	2	3
Reinforce $= d_1$	0	1	0	1
	1/2	1/2	0	1/2
	1	1/2	0	1/2

Table 15: Apollo's optimal strategy for decision to infiltrate a_1.

Several observations can be made from these tables. First, Apollo's most likely decisions are $a_1 = 1$ and $a_2 = 1/2$, differing from their optimums in Section 4.1.2.

		Type = a_0		
		1	2	3
(Reinforce, Infiltrate) = (d_1, a_1)	(0,0)	0	1/2	0
	(0,1/2)	1	1	1
	(0,1)	1/2	1/2	1/2
	(1/2,0)	0	1	1
	(1/2,1/2)	1	1	1
	(1/2,1)	-	-	-
	(1,0)	1/2	1	1/2
	(1,1/2)	1	1	1
	(1,1)	-	-	-

Table 16: Apollo's optimal strategy for decision to attack a_2.

The difference in the decision to attack a_2 can be partly explained by the fact that the cost of a partial attack is relatively low compared to a full attack. The infiltration decision a_1 probably differs as a result, as infiltration lends better support for a partly made attack. Second, Daphne's optimal decisions are neither to reinforce nor to mitigate ($d_1 = d_2 = 0$); the decision not to mitigate results from the decreased expected utility of mitigating a partly made attack. Finally, the decisions converge on the third iteration, which makes this an equilibrium solution.

		$\mathbb{E}[P(d_1)]$			$\mathbb{E}[P(d_2)]$			$\mathbb{E}[u_d]$
		0	1/2	1	0	1/2	1	-
Level k	0	1.000	0.000	0.000	0.908	0.000	0.092	0.906
	1	1.000	0.000	0.000	0.733	0.000	0.267	0.866
	2	1.000	0.000	0.000	1.000	0.000	0.000	0.889
	3	1.000	0.000	0.000	1.000	0.000	0.000	0.889
	4	1.000	0.000	0.000	1.000	0.000	0.000	0.889

Table 17: Expected values for Daphne's decisions and utility for different levels of thinking.

		$\mathbb{E}[P(a_1)]$			$\mathbb{E}[P(a_2)]$			$\mathbb{E}[u_a]$
		0	1/2	1	0	1/2	0	-
Level k	0	0.333	0.667	0.000	0.000	0.000	1.000	0.170
	1	0.333	0.000	0.667	0.000	1.000	0.000	0.185
	2	0.333	0.000	0.667	0.000	1.000	0.000	0.185
	3	0.333	0.000	0.667	0.000	1.000	0.000	0.185
	4	0.333	0.000	0.667	0.000	1.000	0.000	0.185

Table 18: Expected values for Apollo's decisions and utility for different levels of thinking.

A similar sensitivity analysis as in the original problem, changing the initial distributions to $(a_1, a_2) = (d_1, d_2) = (1, 0, 0)$ or $(0, 0, 1)$ did not change the optimum solutions for either actor. Likewise, removing the budget constraints did not change the optimum solutions, as these constrained neither optimum. The stable regions for variables $m_{d,3}$ and $m_{a,3}$ are $m_{d,3} \in [-, 161.0]$ and $m_{a,3} \in [18.5, 72.9]$, presented as percentage changes below. Here, $m_{d,3}$ has no lower bound, as Daphne already decides to neither reinforce nor mitigate. As was in the original problem statement, the solutions of the extended problem statement are relatively stable.

$$\Delta m_{d,3} \in [-, +302.5\%]$$
$$\Delta m_{a,3} \in [-47.1\%, +108.3\%].$$

4.2 Border Security

4.2.1 Problem Description

Our second example is in border security. We study an adversarial situation between two countries. Although there is no armed conflict between these parties, one country, which we call the "attacking" one, may try to influence and disrupt the operations of the other country. One possible way to achieve this is to systematically send illegal immigrants across the border of the "defending" country. The attacking country aims to achieve disruption in two ways. First, sending immigrants strains the defending country's border security resources. Second, if they successfully cross the border, these immigrants can be used for various criminal activities within the defending country, such as an illicit workforce contributing to a gray economy, causing long-term economic drawbacks.

The motivation for this example is, in part, the strained security situation in Europe. There have been reports of Belarus illegally sending immigrants to Poland (Foreign Policy, 2023), and the Finnish government is debating the construction of a border fence to counteract a similar possibility from the Russian border (Finnish Border Guard, 2023). We aim to construct our example to be general enough to accommodate possibilities for analyzing such situations.

The mathematical representation of the problem has two actors: Daphne, representing the border control of a defending country, and Apollo, an operator inside an attacking country. Apollo aims to send immigrants across the border unnoticed, while Daphne aims to create an efficient border security portfolio to counteract this. The influence diagram for this bi-agent decision problem is in Figure 10. We support the decision-making of Daphne. In this example, Daphne's decisions are denoted by d_i, while Apollo's by a_i. A summary of the nodes and state spaces of the problem is in Table 19.

First, Daphne decides on the resources to be used on the border security portfolio (d_1): what types of equipment to use and how much to invest in them. Furthermore, she decides (d_2) on the strategy for controlling the border, i.e., how to place passive equipment and move active equipment and personnel. Moreover, Apollo decides (a_1) how much resources to spend on the reconnaissance of Daphne's border. More specifically, Apollo aims to investigate the types and locations of passive border control equipment and the movement of active equipment and border patrol personnel. Daphne's decisions for resources of portfolio d_1 and placement and strategies of equipment d_2 are made without knowledge of Apollo's decision to reconnaissance a_1,

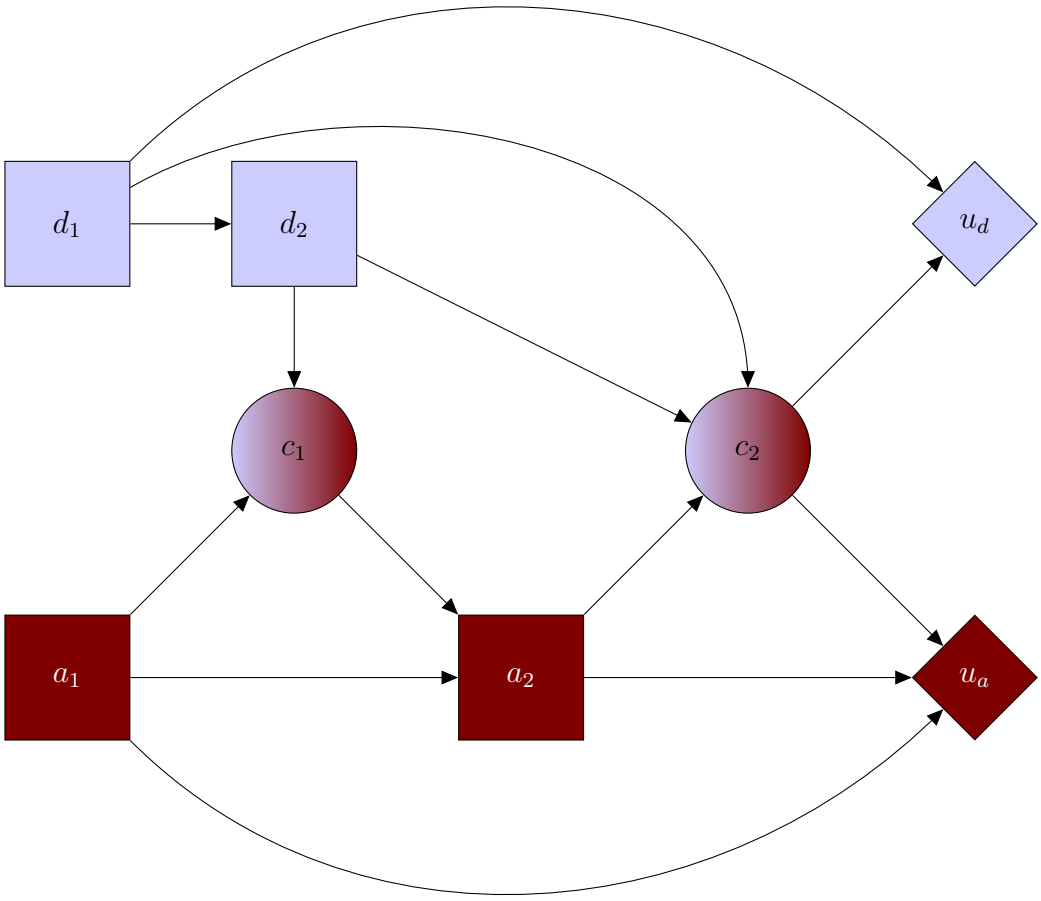

Figure 10: Border Security Influence Diagram.

and vice versa. Decisions a_1 and d_2 result in a random outcome c_1, representing the information Apollo can gather.

Based on this information, Apollo decides (a_2) on how many immigrants to send across the border and their border crossing strategy. The leftover resources from reconnaissance a_1 also constrain decision a_2. The border security portfolio d_1, Daphne's strategy d_2, and Apollo's strategy a_2 result in c_2, the outcome of the attack.

Finally, Daphne's utility u_d is based on the resources used for the border security portfolio and the outcome of Apollo's border crossing. Apollo's utility u_a depends on the resources used for reconnaissance and attacking and the attack's outcome.

As we aim to solve this example through several levels of the level-k thinking framework, we use the concept of types for the uncertainty both actors have about

Node		Interpretation	State space
d_1	$d_{1,f}$	Resources used on fence	0, 1/2, 1 (Low, Medium, High)
	$d_{1,c}$	Resources used on cameras	0, 1/2, 1 (Low, Medium, High)
	$d_{1,u}$	Resources used on drones	0, 1/2, 1 (Low, Medium, High)
	$d_{1,p}$	Resources used on border patrol	0, 1/2, 1 (Low, Medium, High)
d_2	$d_{2,f}$	Placement of fence	0, 1/3, 2/3 (Fraction of the distance from the border to the main road)
	$d_{2,c}$	Placement of cameras	0, 1/3, 2/3 (Fraction of the distance from the border to the main road)
	$d_{2,u}$	Strategies of drones	0, 1 (0: Active patrolling, 1: Passive patrolling)
	$d_{2,p}$	Strategies of border patrols	0, 1 (0: Active patrolling, 1: Passive patrolling)
a_1		Resources used on reconnaissance	0, 1/3, 2/3, 1 (Low, Medium-Low, Medium-High, High)
c_1		Outcome of reconnaissance	0, 1, 2, 3, 4 (Levels of knowledge gained)
a_2	$a_{2,n}$	Number of immigrant groups sent across the border	1, 2, 4, 8 (Low, Medium-Low, Medium-High, High)
	$a_{2,s}$	Strategy of the immigrant groups	0, 1, 2, 3, 4
c_2	$c_{2,o}$	Number of immigrant groups observed	[0, 8]
	$c_{2,i}$	Number of immigrant groups intercepted	[0, 8]
u_a		Apollo's utility	
u_d		Daphne's utility	

Table 19: Nodes, their interpretations, and their state spaces for the border security problem.

the other's probability and utility functions (Harsanyi, 1967). We assume that both actors have three possible types; d_0 =1, 2, or 3 and a_0 =1, 2, or 3. We assume the types to have the same problem structures and possible decision strategies but different costs and probability and utility function parameters. Appendix A.2 shows how these costs and parameters vary as a function of types d_0 and a_0. In the rest of this section, we present the example without referring to these types.

We now describe more precisely the geographical problem environment and the

state spaces of the nodes. The geographical environment of the problem is an area near the border, limited on one side by the border and on the other by a main road, shown as a sketch in Figure 11. Here, x is the width of the area examined, and y is the average distance from the border to the main road. Immigrants are sent from the border to cross the border area into the main road; if they arrive on the main road, they are considered to have successfully escaped, as Apollo is assumed to have prepared a transport for the escaped immigrants to safety.

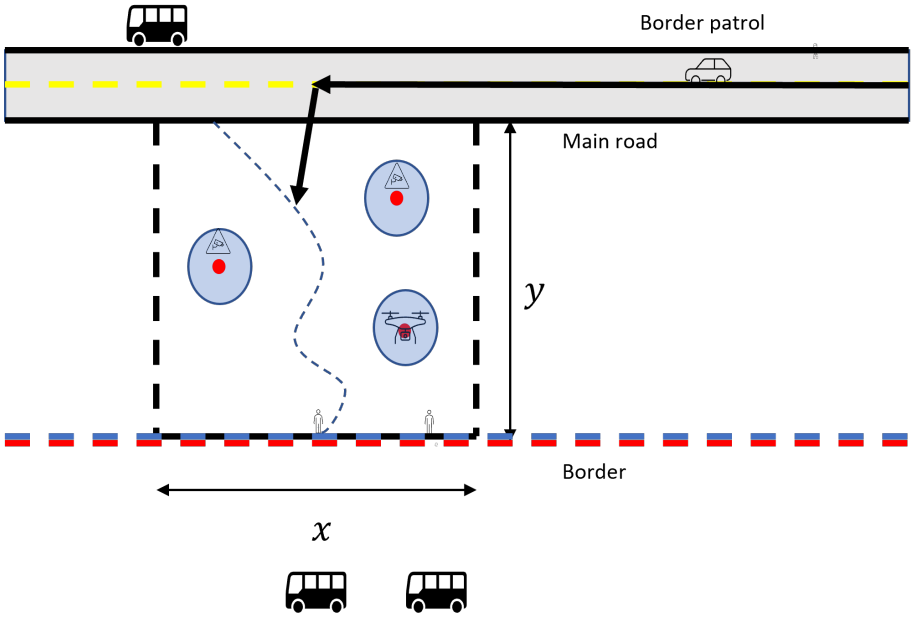

Figure 11: Border Environment. Adopted from (Saranpää et al., 2023).

As we consider several possible border environments, which are neither similar nor homogenous, we examine several scenarios. First, we distinguish three border environments: natural forest, near a border crossing point, and inhabited areas. Furthermore, we differentiate between winter and summer, as the circumstances in Europe can differ significantly due to weather. The combinations of border environments and seasons then define six different scenarios. First, we solve these scenarios individually to gain insight into the possible optimal portfolios for each scenario. Second, we analyze the optimal portfolios for each scenario qualitatively to develop rules of thumb for constructing border security portfolios in general cases.

Daphne's influence diagram is in Figure 12, and Apollo's is in Figure 13. The first decision of Daphne denoted d_1 represents the resources used on the border security portfolio. Daphne can use three types of equipment: a fence with integrated surveillance systems, cameras, and drones. These pieces of equipment are chosen based

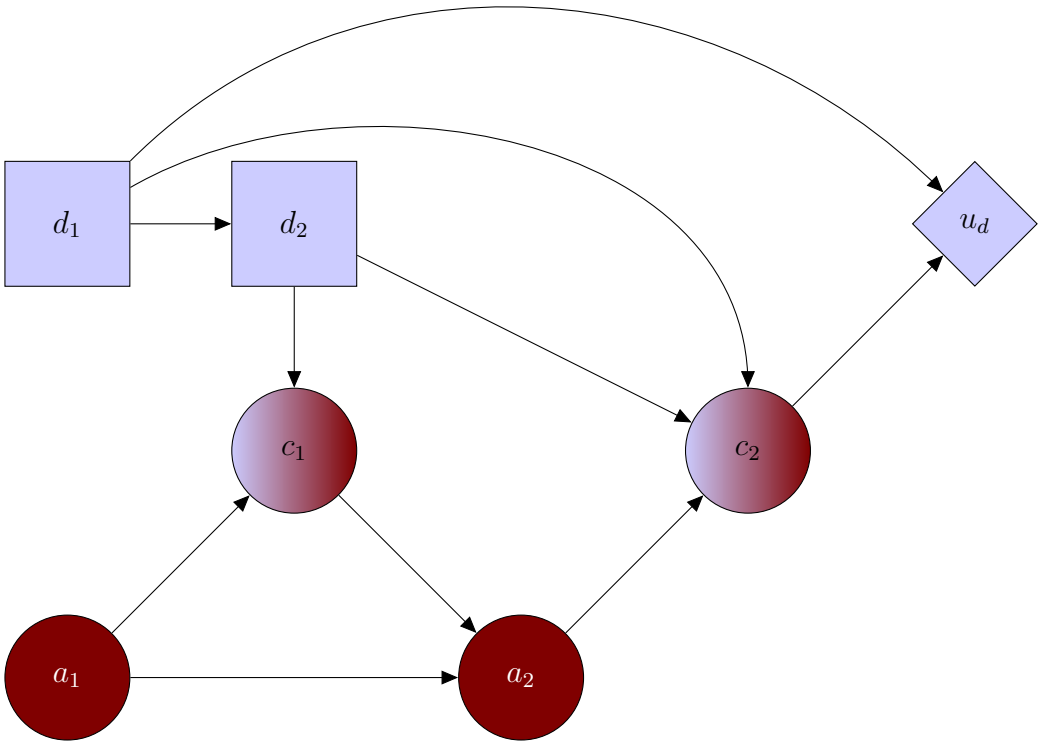

Figure 12: Daphne's Influence Diagram.

on being realistic options to be used in border security but offering reasonable variety to produce meaningful results. In addition, Daphne decides how much personnel resources are devoted to border security. Decision d_1 is, in theory, continuous: Daphne decides how much resources (i.e., money) to devote to each of the four possibilities as mentioned above: $d_1 = (d_{1,f} \;\; d_{1,c} \;\; d_{1,u} \;\; d_{1,p})$ for the resources used on the fence $d_{1,f}$, cameras $d_{1,c}$, drones (UAV's) $d_{1,u}$, and personnel $d_{1,p}$. We, however, limit the number of possibilities: $d_{1,i} \in \{\text{Low, Medium, High}\} = \{0, 1/2, 1\}$, for $i \in \{f, c, u, p\}$. These choices are further limited by a total maximum resource constraint of $b_{d,max}$, i.e., $b_{d,1} \cdot (d_{1,f} + d_{1,c} + d_{1,u} + d_{1,p}) \leq b_{d,max}$, where $b_{d,1}$ is a constant representing the true cost of the normalized decision alternatives in relation to the budget constraint. In the larger context, these investments are best thought of as additional investments to an existing border security portfolio - for example, it is unrealistic to think Daphne has no border security without the added investments in this problem.

Second, Daphne makes decision $d_2 = (d_{2,f} \;\; d_{2,c} \;\; d_{2,u} \;\; d_{2,p})$ on how to place fences and cameras and what strategies to use for drones and personnel. As the number of possible placements is combinatorially huge, we employ several simplifying heuristics

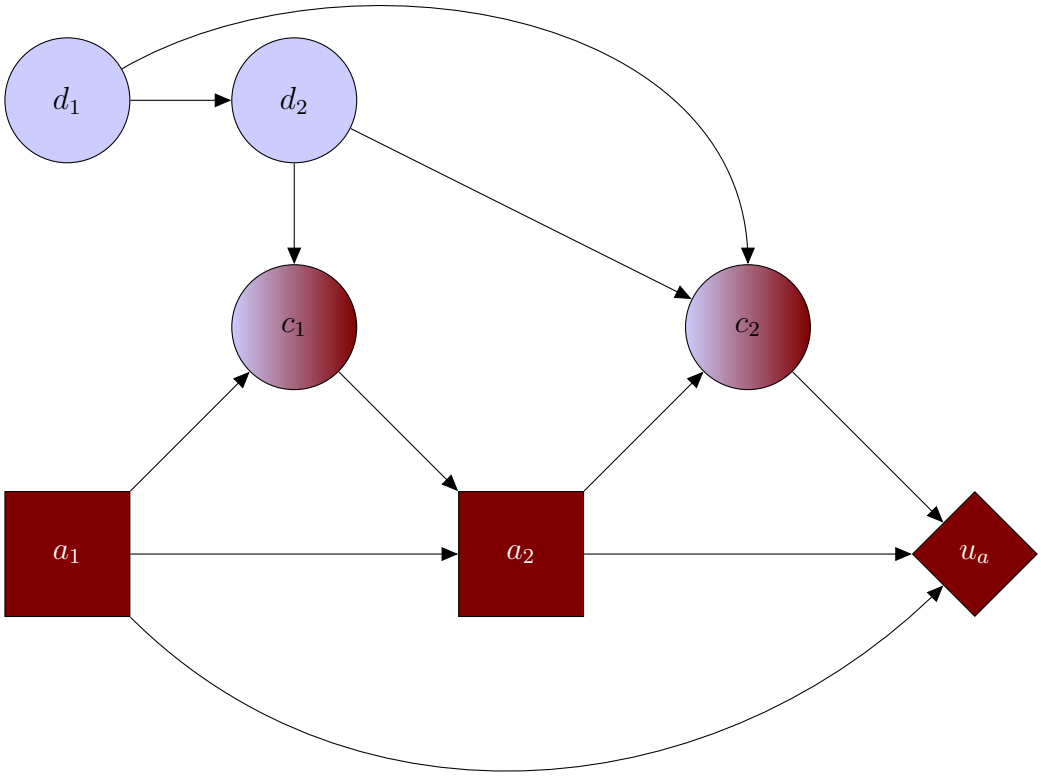

Figure 13: Apollo's Influence Diagram.

to increase the problem's solubility. First, all fence is placed, without gaps, along the border starting from the left side of the border environment (without loss of generality). This heuristic is chosen to optimize the use of the fence in hindering the movement of immigrants. Thus, we only present three possible choices for the location of the fence away from the border: $d_{2,f} = 0$; the fence is next to the border; $d_{2,f} = 1/3$, the fence is a third of the distance from the border to the road; and $d_{2,f} = 2/3$, the fence is two-thirds of the distance from the border to the road. Furthermore, as our problem is simplified in that it does not consider the strategies Apollo's immigrants might employ to evade cameras, and we are looking for equilibrium solutions, we place all cameras in equidistant intervals in a row along the border. We present the same options for the placement of the cameras away from the border as for the fence: $d_{2,c} \in \{0, 1/3, 2/3\}$. We offer two different strategies for deploying drones: $d_{2,u} = 0$ is when drones follow the immigrant groups first noticed by cameras, providing the border security personnel with updated information of their location, and $d_{2,u} = 1$,

when drones patrol the border area, aiming to detect as many border crossings as possible. Finally, the border security personnel can catch the immigrant groups, one by one, while traveling towards the road ($d_{2,p} = 0$), or patrol the main road in hopes of catching immigrants arriving on the road ($d_{2,p} = 1$).

Apollos choice $a_1 \in \{\text{Low, Medium-Low, Medium-High, High}\} = \{0, 1/3, 2/3, 1\}$ concerns gathering intelligence from Daphne. This choice represents the amount of resources used for reconnaissance. Next, node $c_1 \in \{0, 1, 2, 3, 4\}$ represents the outcome of Apollo's reconnaissance. The outcome is divided into five levels, each representing the intelligence that Apollo can gather. Each level $k + 1$ includes the information in level k and the additional information gained in that level. The first level, $c_1 = 0$, means Apollo only knows the total resources used for border security; as this is public data, it is easily available. $c_1 = 1$ adds the information on the location of fences, $c_1 = 2$ the placement of cameras, $c_1 = 3$ strategies of border security personnel, and $c_1 = 4$ the strategies of drones.

The decreasing marginal utility of border security equipment is represented by the logistic function $f(r, x)$, defined in Equation (11), where $r = (r_1, ..., r_n)$ and $x = (x_1, ..., x_n)$. See Figure 14 for the function graph of Equation (11) for the real numbers $x = x_1 = t \in [-5, 5]$ and $r = r_1 = 1$. We use the logistic function to represent the probability of gaining a piece of intelligence, observing an immigrant group, or catching one, for chance nodes $c_1, c_{2,o}$ and $c_{2,i}$, respectively. Here, x represents the states of decision and chance nodes that affect the probability distributions, while r represents the relative effect of each state. For example, increasing the resources used for reconnaissance a_1 increases the probability of gathering more intelligence c_1. However, the effect of added resources is smaller in relation to previous investments, as is often the case in real-life investments (decreasing marginal utility).

$$f(r, x) = \frac{1}{1 + e^{-\sum_i r_i x_i}}. \tag{11}$$

The probability distribution of chance node c_1 is represented by the probability function

$$\begin{aligned} p(c_1|a_1, d_2) &= \binom{4}{c_1} q_1^{c_1} (1 - q_1)^{(4-c_1)}, \\ \text{where} \quad q_1 &= f(r_1, x_1), \\ r_1 &= (r_{1,a,1} \;\; r_{1,d,2,c} \;\; r_{1,d,2,u} \;\; r_{1,d,2,p}), \\ \text{and} \quad x_1 &= (a_1 \;\; 1 - d_{2,c} \;\; d_{2,u} \;\; d_{2,p}). \end{aligned}$$

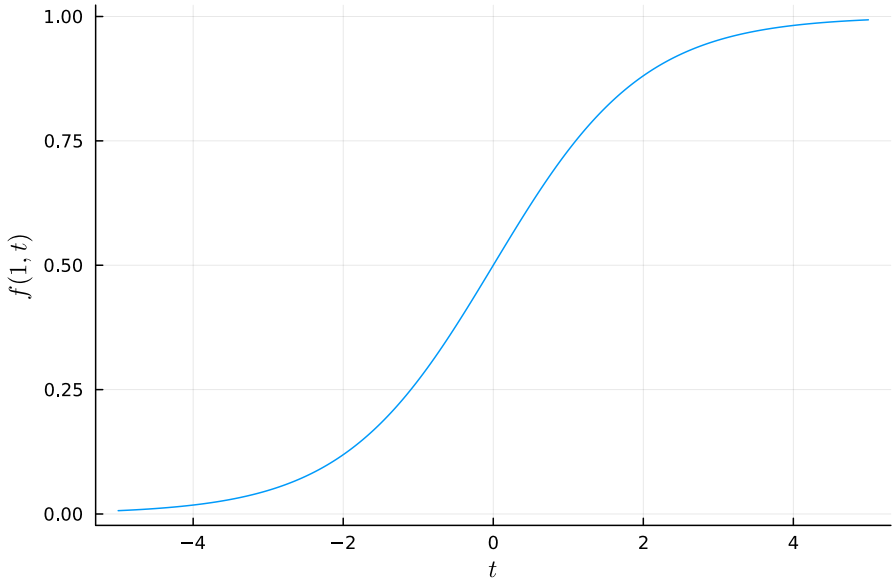

Figure 14: Logistic function with $r = r_1 = 1$ and $x = x_1 = t$

The function $p(c_1|a_1, d_2)$ is a binomial function portraying the quality and amount of intelligence Apollo can gather. The binomial function is natural in this context; for example, decision a_1 (resources used on intelligence) can be thought of as how many scouts Apollo can send to the border. Moreover, each scout sent has a certain probability of gathering a piece of intelligence, e.g., the location of one camera. Aggregating the individual probabilities of gathering intelligence leads to a binomial function. Here, 4 is the maximum value of chance node c_1. Furthermore, $r_{1,a,1}$ represents how much the resources used for reconnaissance a_1 affect the gathered intelligence, i.e., how much $r \cdot x$ is increased for each value of a_1. Parameters $r_{1,d,2,u}$, $r_{1,d,2,p}$, and $r_{1,d,2,c}$ are defined similarly - Apollo can gather intelligence more easily if the value of $d_{2,u}$ and $d_{2,p}$ is one, i.e., border patrols and drones have a set patrolling route, and when cameras are closer to the border. Figure 15 shows an example graph of the probability function with $d_2 = (0, 0, 0, 0)$ and varying a_1. This graph shows how increasing the resources used on reconnaissance a_1 increases the probability of gathering better intelligence.

Next, Apollo decides a_2 on sending immigrants across the border. Here, $a_2 = (a_{2,n}, a_{2,s})$, where $a_{2,n} \in \{\text{Low, Medium-Low, Medium-High, High}\} = \{1, 2, 4, 8\}$ is the total number of immigrant groups Apollo sends across the border, and $a_{2,s} \in \{0, 1, 2, 3, 4\}$ is their border crossing strategy. In our example, $a_{2,s} = c_1$; the information of the outcome of reconnaissance c_1 is passed unchanged onto node $a_{2,s}$. This function approximates the real-life interaction, where Apollo would change

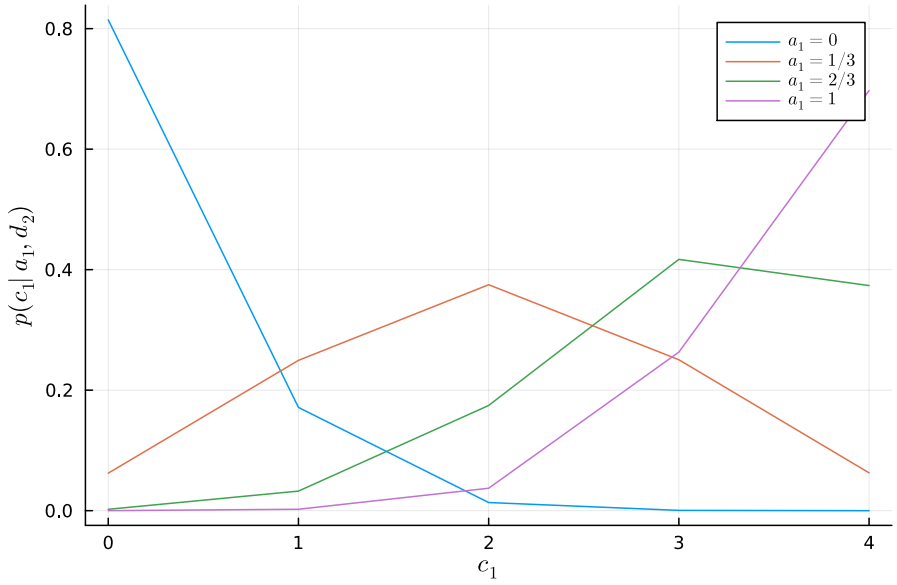

Figure 15: Plot of $p(c_1|a_1, d_2)$ with $d_2 = (0, 0, 0, 0)$ with different a_1

the immigrants' border crossing routes based on information gathered from reconnaissance. The costs for each a_1 and $a_{2,n}$ above are modeled by the constants $b_{a,1}$ and $b_{a,2}$, respectively. Apollo has a total maximum resource constraint of $b_{a,max}$, i.e., $b_{a,1} \cdot a_1 + b_{a,2} \cdot a_{2,n} \leq b_{a,max}$. Node $c_2 = (c_{2,o}, c_{2,i})$ represents the attack's outcome. Outcomes of $c_{2,o}$ and $c_{2,i}$ represent the number of immigrant groups the border security has observed and intercepted, respectively. The probability distribution of the result of attack c_2, assuming $a_{2,n} \geq c_{2,o} \geq c_{2,i}$ (maximum number observed is the amount sent, and maximum number intercepted is the number observed), is

$$
\begin{aligned}
p(c_{2,o}|d_1, d_2, a_2) &= \binom{a_{2,n}}{c_{2,o}} q_{2,o}^{c_{2,o}} (1 - q_{2,o})^{(a_{2,n} - c_{2,o})}, \\
\text{where} \quad q_{2,o} &= f(r_{2,o}, x_{2,o}), \\
r_{2,o} &= (r_{o,a,2,s} \;\; r_{o,a,2} \;\; r_{o,d,1,f} \;\; r_{o,d,1,c} \;\; r_{o,d,1,u} \;\; r_{o,d,1,p}), \\
\text{and} \quad x_{2,o} &= (a_{2,s} \;\; a_{2,n} \;\; d_{1,f} \;\; d_{1,c} \;\; d_{1,u} \cdot d_{2,u} \;\; d_{1,p} \cdot d_{2,p}) \\
p(c_{2,i}|d_1, d_2, a_2, c_{2,o}) &= \binom{c_{2,o}}{c_{2,i}} q_{2,i}^{c_{2,i}} (1 - q_{2,i})^{(c_{2,o} - c_{2,i})}, \\
\text{where} \quad q_{2,i} &= f(r_{2,i}, x_{2,i}), \\
r_{2,i} &= (r_{i,a,2,n} \;\; r_{i,d,1,f} \;\; r_{i,d,2,f} \;\; r_{i,d,2,c} \;\; r_{i,d,1,u} \;\; r_{i,d,1,p}), \\
\text{and} \quad x_{2,i} &= (a_{2,n} \;\; d_{1,f} \;\; 1 - d_{2,f} \;\; 1 - d_{2,c} \;\; d_{1,u} \cdot (1 - d_{2,u}) \;\; d_{1,p} \cdot (1 - d_{2,p})).
\end{aligned}
$$

Table 20 shows Daphne's parameters for the probability functions $p(c_1|a_1, d_2)$,

$p(c_{2,o}|d_1, d_2, a_2)$, and $p(c_{2,i}|d_1, d_2, a_2, c_{2,o})$ for the Summer, Forest scenario. The rest of the probabilities are in Appendix A.2 in Table A6.

	Probability distribution					
	$p(c_1\|a_1, d_2)$		$p(c_{2,o}\|d_1, d_2, a_2)$		$p(c_{2,i}\|d_1, d_2, a_2, c_{2,o})$	
Summer, Forest	$r_{1,a,1}$	$+3.0$	$r_{o,a,2,s}$	-0.8	$r_{i,a,2,n}$	$-1.0 \cdot 10^{-3}$
	$r_{1,d,2,c}$	$+1.0 \cdot 10^{-2}$	$r_{o,a,2,n}$	$-5.0 \cdot 10^{-6}$	$r_{i,d,1,f}$	$+4.0 \cdot 10^{-1}$
	$r_{1,d,2,u}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,f}$	$+1.2 \cdot 10^{-2}$	$r_{i,d,2,f}$	$+1.0 \cdot 10^{-4}$
	$r_{1,d,2,p}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,c}$	$+5$	$r_{i,d,2,c}$	$+1.0 \cdot 10^{-2}$
			$r_{o,d,1,u}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,u}$	$+2.4$
			$r_{o,d,1,p}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,p}$	$+2.4$

Table 20: Daphne's parameters for probability distributions $p(c_1|a_1, d_2)$, $p(c_{2,o}|d_1, d_2, a_2)$, and $p(c_{2,i}|d_1, d_2, a_2, c_{2,o})$ for the Summer, Forest scenario.

As the purpose of this example is to explore the solution process, the input parameters are not analyzed in depth. More specifically, we aim to illustrate how a border security problem can be solved using our methodology rather than proposing absolute results for varying border security situations. Rather, the parameters here are examples and should vary for each different situation to which this example is adapted. There is, however, previous simulation work done on the border situation between Finland and Russia (Saranpää et al., 2023). The parameters used here in the scenario Summer, Forest are mainly based on the results of this work. The rest of the parameters are determined by a combination of an estimated effectiveness measure for each type of equipment or strategy and their cost. For example, fences with integrated surveillance systems and cameras are equally effective in observing immigrants. However, cameras are significantly cheaper, explaining the difference between parameters $r_{o,d,1,f}$ and $r_{o,d,1,c}$. Figures 16 and 17 show the probability functions of $p(c_{2,o}|d_1, d_2, a_2)$ and $p(c_{2,i}|d_1, d_2, a_2, c_{2,o})$ for certain states of the decision and chance variables.

Finally, Daphne's utility depends on the states of nodes d_1 and c_2. Daphne's value function is

$$v_d = -B_d - k_{d,1} \cdot (a_{2,n} - c_{2,o}) - k_{d,2} \cdot (a_{2,n} - c_{2,i}),$$

where $B_d = b_{d,1} \cdot (d_{1,f} + d_{1,c} + d_{1,u} + d_{1,p})$ represents the total resources used, and $k_{d,1}$ and $k_{d,2}$ represent the lost utility of not observing or catching a single immigrant group crossing the border, respectively. As Daphne is constant risk averse, her utility

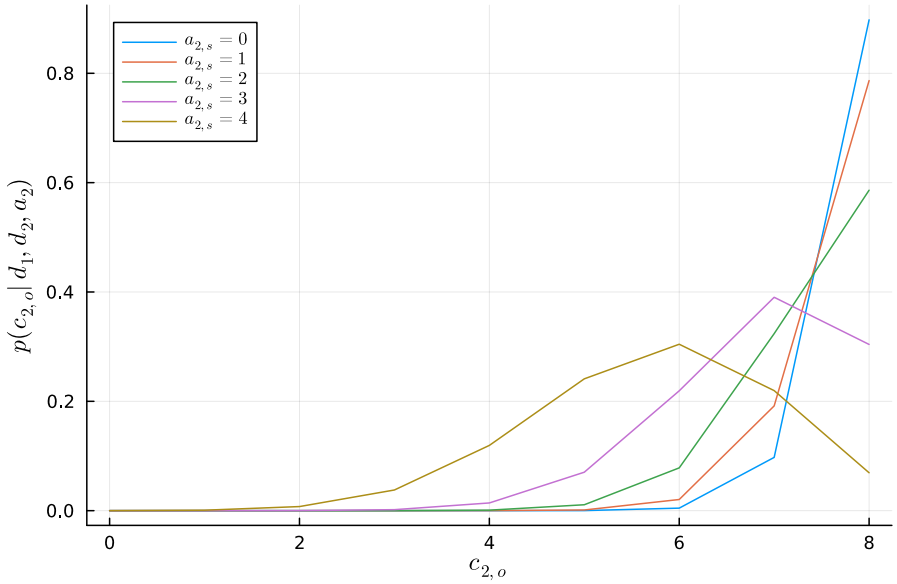

Figure 16: Plot of $p(c_{2,o}|d_1, d_2, a_2)$ with $d_1 = (0, 1, 0, 0), d_2 = (0, 0, 0, 0)$, $a_{2,n} = 8$, and varying $a_{2,s}$.

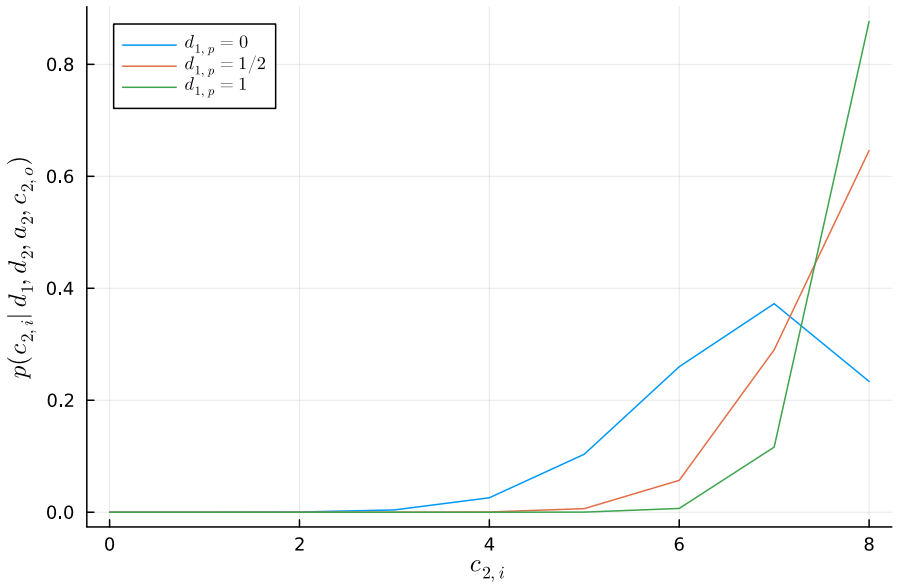

Figure 17: Plot of $p(c_{2,i}|d_1, d_2, a_2, c_{2,o})$ with $d_1 = (0, 0, 1, 1), d_2 = (1, 1, 0, 0)$, $a_{2,n} = c_{2,o} = 8$, and varying $d_{1,p}$.

function is of the form

$$u_d = 1 - \exp\left[-\lambda_d(v_d + c_{d,1})/c_{d,2}\right],$$

where λ_d is Daphne's risk aversion coefficient, and $c_{d,1}$ and $c_{d,2}$ are normalizing

constants. Our approach is somewhat different than that of Section 4.1.1. Here, constant $c_{d,1} = \max |v_d|$ restricts the value function to be positive, while $c_{d,2} = \max(v_d + c_{d,1})$ restricts it to the unit interval. Then, we can determine Daphne's normalized risk aversion coefficient λ_d. Studies show varying results for the risk aversion coefficients for real-life decision-makers; Friend and Blume (1975) propose the risk aversion coefficient to be larger than one, and possibly larger than two, while Hansen and Singleton (1983) suggests it is at least two and maybe over three. Based on these two varying results, we use three different risk aversion coefficients, $\lambda_d \in \{1, 2, 3\}$, which are enumerated along with the rest of the type-specific parameters in Appendix A.2 in Table A4 for Daphne and A5 for Apollo.

Apollo's value function, on the other hand, is

$$v_a = -B_a + k_{a,1} \cdot (a_{2,n} - c_{2,o}) + k_{a,2} \cdot (a_{2,n} - c_{2,i}),$$

where $B_a = b_{a,1} \cdot a_1 + b_{a,2} \cdot a_{2,n}$ are the total resources used by Apollo, and $k_{a,1}$ and $k_{a,2}$ portray the gained utility of successfully sending an immigrant unobserved or uncaught across the border, respectively. This value function results in the utility function

$$u_a = \exp\left[\lambda_a (v_a + c_{a,1})/c_{a,2}\right],$$

where $c_{a,1}$ and $c_{a,2}$ are normalizing constants defined similarly as in Daphne's utility function.

4.2.2 Solution

To solve the border security problem, we use the maximum entropy principle for constructing the initial distribution. We assume the maximum level of thinking for both Daphne and Apollo to be $k = 4$. Furthermore, as this example was large, we used the approximation algorithm for Decision Programming to prune zero probability paths out of the problem (Aunula, 2021). Table 21 shows Daphne's expected value of optimal decisions (d_1 and d_2) for the six different scenarios defined in 4.2.1. The full results for each type are in A.2, in Tables A7 and A8 for Daphne and Apollo, respectively.

The optimum strategy for Apollo is relatively simple; depending on the scenario and type, it is optimal to decide either $a_1 = 2/3$ or $a_1 = 1$, i.e., letting the resources used on reconnaissance be either Medium-High or High. After this, regardless of the outcome of gained intelligence c_1, it is optimal to send as many immigrant groups

Scenario	$\mathbb{E}[d_1]$				$\mathbb{E}[d_2]$			
	$d_{1,f}$	$d_{1,c}$	$d_{1,u}$	$d_{1,p}$	$d_{2,f}$	$d_{2,c}$	$d_{2,u}$	$d_{2,p}$
Summer, Forest	0.00	1.00	0.00	0.67	0.00	0.00	0.00	0.00
Winter, Forest	0.00	1.00	0.00	0.83	0.00	0.00	0.00	0.00
Summer, Near crossing	0.00	1.00	0.00	0.50	0.00	0.00	0.00	0.00
Winter, Near crossing	0.00	1.00	0.00	0.50	0.00	0.00	0.00	0.00
Summer, Inhabited area	0.00	0.50	0.00	0.67	0.00	0.00	0.00	0.00
Winter, Inhabited area	0.00	1.00	0.17	0.67	0.00	0.00	0.00	0.00

Table 21: Daphne's expected value of optimal decisions for the six scenarios.

$(a_{2,n})$ as possible while still satisfying the budget constraint. Furthermore, several observations can be made from Table 21 and Daphne's strategies to counteract Apollo's aggressive attacks. First, Daphne's optimal placements d_2 for all equipment are similar for all scenarios; cameras and fences should be placed close to the border, and drones and border patrol personnel should move actively. Second, regarding the resources used on the border security portfolio d_1, Daphne's primary sources of investment should be cameras and border patrol personnel. Drones and fences should not be invested in. These results make sense, as cameras are more cost-efficient than fences with integrated surveillance systems in observing immigrants (Saranpää et al., 2023). Furthermore, border patrol personnel are a better source of investment than drones, as border patrols can both observe and catch immigrants, while drones can only observe.

Tables 22 and 23 show the expected decisions and utilities as a function of the level of thinking k for Daphne and Apollo, respectively, for the Summer, Forest scenario. These tables show that the decisions converge on the third level of thinking for Daphne and the second level for Apollo. Although the utilities still change somewhat on the fourth and final level of thinking, these will also converge as the decisions of neither Daphne nor Apollo will change. This change in utility for Apollo results from the fact that his solution on the third level employs Daphne's solution on the second level, which is different from her third-level solution.

To further examine the stability of our solution, we performed three types of sensitivity analyses, the full results of which are in Appendix A.2, in Tables A9

		$\mathbb{E}[d_1]$				$\mathbb{E}[d_2]$				$\mathbb{E}[u_d]$
		$d_{1,f}$	$d_{1,c}$	$d_{1,u}$	$d_{1,p}$	$d_{2,f}$	$d_{2,c}$	$d_{2,u}$	$d_{2,p}$	-
Level k	0	0.000	0.667	0.500	0.000	0.000	0.000	0.000	0.000	0.192
	1	0.000	1.000	0.833	0.000	0.000	0.000	0.000	0.000	0.236
	2	0.000	1.000	0.833	0.000	0.000	0.000	0.000	0.000	0.235
	3	0.000	1.000	0.667	0.000	0.000	0.000	0.000	0.000	0.239
	4	0.000	1.000	0.667	0.000	0.000	0.000	0.000	0.000	0.239

Table 22: Expected values for Daphne's decisions and utility for different levels of thinking for the Summer, Forest scenario.

		$\mathbb{E}[P(a_1)]$				$\mathbb{E}[P(a_{2,n})]$				$\mathbb{E}[u_a]$
		0	1/3	2/3	1	1	2	4	8	-
Level k	0	0.333	0.000	0.667	0.000	0.000	0.333	0.000	0.667	0.711
	1	0.667	0.000	0.333	0.000	0.000	0.000	0.333	0.667	0.548
	2	0.000	0.000	0.333	0.667	0.000	0.333	0.333	0.333	0.161
	3	0.000	0.000	0.333	0.667	0.000	0.333	0.333	0.333	0.161
	4	0.000	0.000	0.333	0.667	0.000	0.333	0.333	0.333	0.191

Table 23: Expected values for Apollo's decisions and utility for different levels of thinking for the Summer, Forest scenario.

and A10. First, we solved the problem with minimum entropy initial distributions, $P[d_1 = d_2 = (1,0,0,0)] = 1$ and $P[d_1 = d_2 = (0,0,0,1)] = 1$ for Apollo, and $P[a_1 = 0, a_{2,n} = 1] = 1$ and $P[a_1 = 1, a_{2,n} = 8] = 1$ for Daphne. Second, we changed the parameters of the problem; $(k_{a,1}, k_{a,2}) \to 1.2 \cdot (k_{a,1}, k_{a,2})$ and $(k_{a,1}, k_{a,2}) \to 0.8 \cdot (k_{a,1}, k_{a,2})$ for Apollo, and $(k_{d,1}, k_{d,2}) \to 1.2 \cdot (k_{d,1}, k_{d,2})$ and $(k_{d,1}, k_{d,2}) \to 0.8 \cdot (k_{d,1}, k_{d,2})$ for Daphne. Finally, we removed the budget constraints $B_{a,max}$ and $B_{d,max}$ for Apollo and Daphne, respectively.

For both actors, the solution did not change when the initial distributions had minimum entropy, which suggests our equilibrium solution is the global optimum for the problem. Furthermore, increasing and decreasing the problem parameters changed the optimum for a few scenarios, but these differences were minor. For Daphne, removing the budget constraints did not affect decisions, while for Apollo, removing the constraint led to maximum investment in both reconnaissance a_1 and attacking a_2. Overall, our optimum solution seems stable to small changes in the problem's parameters.

Even in the case of Apollo's budget constraints being removed, Daphne's optimal strategy stayed relatively the same; cameras and border patrol personnel are the best sources of investment, in addition to drones, in some rare cases. Moreover, if border patrol investments were more expensive than indicated by our parameters,

drone investments could be efficient in maximizing the number of immigrant groups a single border patrol group can catch, as indicated by the results of Saranpää et al. (2023). All in all, this example can be used flexibly with different assumptions about scenarios and parameters to create valuable results for border security investments for varying scenarios.

5 Conclusions and Discussion

Widely employed solution methods for solving multi-agent decision problems, among them complete information game theory and adversarial risk analysis, are not suitable for solving decision problems with limited memory or overarching constraints (Kahneman and Tversky, 1979; Kahneman, 2003; Salo et al., 2022). This thesis aimed to combine Decision Programming (Salo et al., 2022), a solution method that accommodates these challenges for single-agent decision problems, with the methods mentioned above for solving multi-agent decision problems. We used the level-k approach, in which the multi-agent decision problem is transformed into corresponding single-agent decision problems, separately for each actor, and solved iteratively to arrive at a convergent solution. We use the Decision Programming framework at each phase to solve each single-agent problem. Our methodology thus solves decision problems with several actors, simultaneous and sequential decisions, imperfect strategists and information, and overarching constraints.

We used our methodology to solve two multi-agent decision problems: the first in critical infrastructure protection and the other in border security. In the critical infrastructure protection example, our methodology generated similar results as the original method examining this problem (González-Ortega et al., 2019). The methodology we developed converged in less than three iterations to a stable solution and, in almost all cases, was not sensitive to variations in the model parameters. We also extended the critical infrastructure protection example by introducing more granular choices and budget constraints, which our methodology solved to optimality. We furthermore solved an example in border security. Here, despite a more complicated problem structure, we were able to solve this example to optimality, creating useful results regarding border security. The method also performed well in this problem, both in convergence and sensitivity analyses, similarly to the critical infrastructure protection example.

The methodological basis of our framework allows it to solve several other types of problems as those presented in the examples. First, the number of decision-makers is not limited to two; our method can solve problems with several decision-makers. Furthermore, these decision-makers can be collaborating, competing, or anything in between as long as they have quantifiable utility functions. Finally, the problems may include limited information about earlier decisions, multiple value nodes, and arbitrary problem-spanning constraints.

Although the results our methodology generated are promising, it has limitations that can be addressed in further research. First, in this thesis, it was assumed that

the decision problem can be accurately modeled via the level-k thinking method. However, this may not always hold for the strategies of the decision-makers. The issue could be addressed by examining the basis of level-k thinking more closely and seeing if other types of strategic thinkers can be modeled via this method. Second, we do not research problems in which the problem structure is uncertain to the actors or in which utility functions include only a qualitative preference relation. Further research could adapt our method to these kinds of uncertain circumstances. Third, our methodology requires that the decision-makers' strategy cannot be mixed, i.e., include a randomized choice among decision alternatives. This results from the structure of the Decision Programming framework. Further research could advance Decision Programming to incorporate mixed strategies. Fourth, we only used the maximum entropy assumption as our initial distribution. More research could be done on the influence of this initial distribution on convergence. Finally, further research could test our methodology for a wider variety of problems to uncover more information on how the methodology works in different contexts and problem structures.

References

Jerry Aunula. An Algorithm for Generating Most Probable Paths in Decision Programming. Bachelor's thesis. 2021. URL https://sal.aalto.fi/publications/pdf-files//theses/bac/taun21_public.pdf.

David Banks, Francesca Petralia, and Shouqiang Wang. Adversarial risk analysis: Borel games. *Applied Stochastic Models in Business and Industry*, 27(2):72–86, 2011.

David L Banks, Jesus M Rios Aliaga, and David Ríos Insua. *Adversarial risk analysis*. CRC Press, 2015.

Colin F Camerer. Behavioural game theory. *Behavioural and Experimental Economics*, pages 42–50, 2010.

Finnish Border Guard. The eastern border barrier fence. https://raja.fi/en/the-eastern-border-barrier-fence, 2023. Accessed: 20.09.2023.

Foreign Policy. The hottest forest in the world. https://foreignpolicy.com/2023/08/13/poland-belarus-border-migration-crisis-wagner-group-russia-ukraine-election/, 2023. Accessed: 20.09.2023.

Irwin Friend and Marshall E Blume. The demand for risky assets. *The American Economic Review*, 65(5):900–922, 1975.

David Gill and Victoria Prowse. Cognitive ability, character skills, and learning to play equilibrium: A level-k analysis. *Journal of Political Economy*, 124(6): 1619–1676, 2016.

Herbert Gintis. *The Bounds of Reason: Game Theory and the Unification of the Behavioral Sciences-Revised Edition*. Princeton University Press, 2014.

Jorge González-Ortega, David Ríos Insua, and Javier Cano. Adversarial risk analysis for bi-agent influence diagrams: An algorithmic approach. *European Journal of Operational Research*, 273(3):1085–1096, 2019.

Lars Peter Hansen and Kenneth J Singleton. Stochastic consumption, risk aversion, and the temporal behavior of asset returns. *Journal of Political Economy*, 91(2): 249–265, 1983.

John C Harsanyi. Games with incomplete information played by "Bayesian" players, I–III part I. the basic model. *Management Science*, 14(3):159–182, 1967.

Ronald A Howard and James E Matheson. Influence diagrams. *Decision Analysis*, 2 (3):127–143, 2005.

Qi Huangfu and JA Julian Hall. Parallelizing the dual revised simplex method. *Mathematical Programming Computation*, 10(1):119–142, 2018.

Daniel Kahneman. Maps of bounded rationality: Psychology for behavioral economics. *American Economic Review*, 93(5):1449–1475, 2003.

Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263–292, 1979.

Daphne Koller and Brian Milch. Multi-agent influence diagrams for representing and solving games. *Games and Economic Behavior*, 45(1):181–221, 2003.

Steffen L Lauritzen and Dennis Nilsson. Representing and solving decision problems with limited information. *Management Science*, 47(9):1235–1251, 2001.

Rosemarie Nagel. Unraveling in guessing games: An experimental study. *The American Economic Review*, 85(5):1313–1326, 1995.

Scott Mostyn Olmsted. *On Representing and Solving Decision Problems*. PhD thesis, Stanford University, 1984.

Jesus Rios and David Rios Insua. Adversarial risk analysis for counterterrorism modeling. *Risk Analysis*, 32(5):894–915, 2012.

David Rios Insua, Jesus Rios, and David Banks. Adversarial risk analysis. *Journal of the American Statistical Association*, 104(486):841–854, 2009.

David Rios Insua, Aitor Couce-Vieira, Jose A Rubio, Wolter Pieters, Katsiaryna Labunets, and Daniel G. Rasines. An adversarial risk analysis framework for cybersecurity. *Risk Analysis*, 41(1):16–36, 2021.

Ahti Salo, Juho Andelmin, and Fabricio Oliveira. Decision programming for mixed-integer multi-stage optimization under uncertainty. *European Journal of Operational Research*, 299(2):550–565, 2022.

Juho Saranpää, Jerry Aunula, Markus Virtanen, Juho Ponkkonen, and Mikko Murhu. Assesment of design options for border control. https://sal.aalto.fi/files/teaching/ms-e2177/2023/1.%20FDF%20-%20Final%20report%20-%202023.pdf, 2023. Accessed: 20.09.2023.

Juan Carlos Sevillano, David Rios Insua, and Jesus Rios. Adversarial risk analysis: The Somali pirates case. *Decision Analysis*, 9(2):86–95, 2012.

Ross D Shachter. Evaluating influence diagrams. *Operations Research*, 34(6): 871–882, 1986.

Ross D Shachter. Probabilistic inference and influence diagrams. *Operations Research*, 36(4):589–604, 1988.

Dale O Stahl and Paul W Wilson. Experimental evidence on players' models of other players. *Journal of Economic Behavior & Organization*, 25(3):309–327, 1994.

Joseph A Tatman and Ross D Shachter. Dynamic programming and influence diagrams. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(2):365–379, 1990.

John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior, 2nd rev.* Princeton University Press, 1947.

Changhe Yuan, Xiaojian Wu, and Eric Hansen. Solving multistage influence diagrams using branchand-bound search. In *Proceedings of the Twentysixth Conference on Uncertainty in Artificial Intelligence (UAI'10)*, pages 691–700, 2010.

Yang Zhou, Guo H Huang, and Boting Yang. Water resources management under multi-parameter interactions: A factorial multi-stage stochastic programming approach. *Omega*, 41(3):559–573, 2013.

A Appendix

A.1 Critical Infrastructure Protection

The tabulated probability distributions for chance nodes c_1 (infrastructure downtime) and c_2 (reduction in downtime) of the problem in Section 4.1.3 are presented below in Tables A1-A3. However, the probabilities of the instances also found in the original problem description, in Section 4.1.1, are unchanged. Thus, these original probabilities can be read from the tables below; the corresponding rows are colored blue. Furthermore, as is described in the problem statements, Type 1 Apollo's probabilities correspond precisely to those of Daphne and can be read from Daphne's columns.

			Shortage = c_1								
			Daphne			Type 2 Apollo			Type 3 Apollo		
d_1	a_1	a_2	0	1/2	1	0	1/2	1	0	1/2	1
0	0	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
0	1/2	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
0	1	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
1/2	0	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
1/2	1/2	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
1/2	1	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
1	0	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
1	1/2	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
1	1	0	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
0	0	1/2	0.65	0.23	0.13	0.64	0.22	0.14	0.66	0.23	0.11
0	1/2	1/2	0.61	0.25	0.14	0.61	0.24	0.15	0.62	0.26	0.13
0	1	1/2	0.58	0.28	0.15	0.57	0.27	0.16	0.58	0.28	0.14
1/2	0	1/2	0.68	0.21	0.11	0.67	0.20	0.13	0.68	0.22	0.10
1/2	1/2	1/2	0.64	0.24	0.13	0.63	0.23	0.14	0.64	0.25	0.11
1/2	1	1/2	0.60	0.26	0.14	0.60	0.25	0.15	0.60	0.27	0.13
1	0	1/2	0.70	0.20	0.10	0.69	0.19	0.12	0.71	0.21	0.08
1	1/2	1/2	0.66	0.23	0.11	0.66	0.22	0.13	0.67	0.23	0.10
1	1	1/2	0.63	0.25	0.13	0.62	0.24	0.14	0.63	0.26	0.11
0	0	1	0.30	0.45	0.25	0.29	0.43	0.28	0.31	0.47	0.22
0	1/2	1	0.23	0.50	0.28	0.22	0.48	0.30	0.23	0.52	0.25
0	1	1	0.15	0.55	0.30	0.15	0.53	0.32	0.15	0.57	0.28
1/2	0	1	0.35	0.43	0.23	0.34	0.41	0.26	0.36	0.44	0.19
1/2	1/2	1	0.28	0.48	0.25	0.27	0.46	0.28	0.28	0.49	0.22
1/2	1	1	0.20	0.53	0.28	0.19	0.51	0.30	0.21	0.54	0.25
1	0	1	0.40	0.40	0.20	0.38	0.38	0.23	0.42	0.42	0.17
1	1/2	1	0.33	0.45	0.23	0.31	0.43	0.25	0.34	0.47	0.20
1	1	1	0.25	0.50	0.25	0.24	0.48	0.27	0.26	0.52	0.23

Table A1: Probability distributions $p(c_1|d_1, a_1, a_2)$. Probabilities relating to the original problem statement are colored in blue.

			Shortage reduction = c_2				
			Daphne				
a_2	c_1	d_2	0	1/4	1/2	3/4	1
0	0	0	1.00	0.00	0.00	0.00	0.00
0	1/2	0	1.00	0.00	0.00	0.00	0.00
0	1	0	1.00	0.00	0.00	0.00	0.00
1/2	0	0	1.00	0.00	0.00	0.00	0.00
1/2	1/2	0	1.00	0.00	0.00	0.00	0.00
1/2	1	0	1.00	0.00	0.00	0.00	0.00
1	0	0	1.00	0.00	0.00	0.00	0.00
1	1/2	0	1.00	0.00	0.00	0.00	0.00
1	1	0	1.00	0.00	0.00	0.00	0.00
0	0	1/2	1.00	0.00	0.00	0.00	0.00
0	1/2	1/2	1.00	0.00	0.00	0.00	0.00
0	1	1/2	1.00	0.00	0.00	0.00	0.00
1/2	0	1/2	1.00	0.00	0.00	0.00	0.00
1/2	1/2	1/2	0.89	0.02	0.04	0.03	0.03
1/2	1	1/2	0.78	0.04	0.08	0.06	0.05
1	0	1/2	1.00	0.00	0.00	0.00	0.00
1	1/2	1/2	0.78	0.04	0.08	0.06	0.05
1	1	1/2	0.55	0.08	0.15	0.13	0.10
0	0	1	1.00	0.00	0.00	0.00	0.00
0	1/2	1	1.00	0.00	0.00	0.00	0.00
0	1	1	1.00	0.00	0.00	0.00	0.00
1/2	0	1	1.00	0.00	0.00	0.00	0.00
1/2	1/2	1	0.60	0.25	0.15	0.00	0.00
1/2	1	1	0.55	0.08	0.15	0.13	0.1
1	0	1	1.00	0.00	0.00	0.00	0.00
1	1/2	1	0.20	0.50	0.30	0.00	0.00
1	1	1	0.10	0.15	0.30	0.25	0.20

Table A2: Probability distributions $p(c_2|a_2, c_1, d_2)$ for Daphne and Type 1 Apollo. Probabilities relating to the original problem statement are colored in blue.

			Shortage reduction = c_2									
			Type 2 Apollo					Type 3 Apollo				
a_2	c_1	d_2	0	1/4	1/2	3/4	1	0	1/4	1/2	3/4	1
0	0	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0	1/2	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0	1	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1/2	0	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1/2	1/2	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1/2	1	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1	0	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1	1/2	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1	1	0	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0	0	1/2	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0	1/2	1/2	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0	1	1/2	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1/2	0	1/2	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1/2	1/2	1/2	0.89	0.02	0.04	0.03	0.02	0.89	0.02	0.04	0.03	0.03
1/2	1	1/2	0.78	0.04	0.07	0.06	0.05	0.77	0.04	0.08	0.06	0.05
1	0	1/2	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1	1/2	1/2	0.78	0.04	0.07	0.06	0.05	0.77	0.04	0.08	0.06	0.05
1	1	1/2	0.56	0.07	0.15	0.12	0.10	0.54	0.08	0.15	0.13	0.10
0	0	1	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0	1/2	1	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0	1	1	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1/2	0	1	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1/2	1/2	1	0.61	0.24	0.15	0.00	0.00	0.59	0.26	0.15	0.00	0.00
1/2	1	1	0.56	0.07	0.15	0.12	0.10	0.54	0.08	0.15	0.13	0.10
1	0	1	1.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
1	1/2	1	0.22	0.49	0.29	0.00	0.00	0.18	0.51	0.31	0.00	0.00
1	1	1	0.11	0.15	0.30	0.25	0.20	0.09	0.15	0.30	0.25	0.20

Table A3: Probability distributions $p(c_2|a_2, c_1, d_2)$ for Type 2 and Type 3 Apollo. Probabilities relating to the original problem statement are colored in blue.

A.2 Border Security

Tables A4 and A5 contain the type-specific general parameters, and Table A6 shows the parameters for the probability distributions of all six scenarios for Daphne's type 1. The probabilities for Daphne's types 2 and 3 are the tabulated probabilities multiplied by 0.60 and 1.40, respectively, while the multipliers in relation to Daphne's Type 1 for Apollo's types 1, 2, and 3 are 1.00, 0.80, and 1.20, respectively. In addition, Tables A7 and A8 show the optimal decisions for Daphne and Apollo, while Tables A9 and A10 show the results of sensitivity analyses, more specifically laid out in Section 4.2.1.

		Parameters				
		$b_{d,max}$ (k€)	$b_{d,1}$ (k€)	$k_{d,1}$ (k€)	$k_{d,2}$ (k€)	λ_d
Daphne's type	1	250	100	20	100	2.0
	2	200	100	10	60	1.0
	3	300	100	40	150	3.0

Table A4: Parameters for Daphne's types.

		Parameters					
		$b_{a,max}$ (k€)	$b_{a,1}$ (k€)	$b_{a,2}$ (k€)	$k_{a,1}$ (k€)	$k_{a,2}$ (k€)	λ_a
Apollo's type	1	80	30	10	4	20	2.0
	2	160	30	10	6	40	4.0
	3	40	30	10	2	12	1.0

Table A5: Parameters for Apollo's types.

	Probability distribution					
	$p(c_1\|a_1, d_2)$		$p(c_{2,o}\|d_1, d_2, a_2)$		$p(c_{2,i}\|c_{2,o}, d_1, d_2, a_2)$	
Summer, Forest	$r_{1,a,1}$	$+3.0$	$r_{o,a,2,s}$	-0.8	$r_{i,a,2,n}$	$-1.0 \cdot 10^{-3}$
	$r_{1,d,2,c}$	$+1.0 \cdot 10^{-2}$	$r_{o,a,2,n}$	$-5.0 \cdot 10^{-6}$	$r_{i,d,1,f}$	$+4.0 \cdot 10^{-1}$
	$r_{1,d,2,u}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,f}$	$+1.2 \cdot 10^{-2}$	$r_{i,d,2,f}$	$+1.0 \cdot 10^{-4}$
	$r_{1,d,2,p}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,c}$	$+5$	$r_{i,d,2,c}$	$+1.0 \cdot 10^{-2}$
			$r_{o,d,1,u}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,u}$	$+2.4$
			$r_{o,d,1,p}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,p}$	$+2.4$
Winter, Forest	$r_{1,a,1}$	$+2.4$	$r_{o,a,2,s}$	-0.8	$r_{i,a,2,n}$	$-1.0 \cdot 10^{-3}$
	$r_{1,d,2,c}$	$+1.0 \cdot 10^{-2}$	$r_{o,a,2,n}$	$-4.0 \cdot 10^{-6}$	$r_{i,d,1,f}$	$+4.0 \cdot 10^{-1}$
	$r_{1,d,2,u}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,f}$	$+1.2 \cdot 10^{-2}$	$r_{i,d,2,f}$	$+1.0 \cdot 10^{-4}$
	$r_{1,d,2,p}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,c}$	$+5$	$r_{i,d,2,c}$	$+1.0 \cdot 10^{-2}$
			$r_{o,d,1,u}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,u}$	$+2.9$
			$r_{o,d,1,p}$	$+5.8 \cdot 10^{-2}$	$r_{i,d,1,p}$	$+3.6$
Summer, Near crossing	$r_{1,a,1}$	$+4.5$	$r_{o,a,2,s}$	-0.8	$r_{i,a,2,n}$	$-1.0 \cdot 10^{-3}$
	$r_{1,d,2,c}$	$+1.0 \cdot 10^{-2}$	$r_{o,a,2,n}$	$-5.0 \cdot 10^{-6}$	$r_{i,d,1,f}$	$+4.0 \cdot 10^{-1}$
	$r_{1,d,2,u}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,f}$	$+1.2 \cdot 10^{-2}$	$r_{i,d,2,f}$	$+1.0 \cdot 10^{-4}$
	$r_{1,d,2,p}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,c}$	$+5$	$r_{i,d,2,c}$	$+1.0 \cdot 10^{-2}$
			$r_{o,d,1,u}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,u}$	$+2.4$
			$r_{o,d,1,p}$	$+9.6 \cdot 10^{-2}$	$r_{i,d,1,p}$	$+4.8$
Winter, Near crossing	$r_{1,a,1}$	$+3.6$	$r_{o,a,2,s}$	-0.8	$r_{i,a,2,n}$	$-1.0 \cdot 10^{-3}$
	$r_{1,d,2,c}$	$+1.0 \cdot 10^{-2}$	$r_{o,a,2,n}$	$-4.0 \cdot 10^{-6}$	$r_{i,d,1,f}$	$+4.0 \cdot 10^{-1}$
	$r_{1,d,2,u}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,f}$	$+1.2 \cdot 10^{-2}$	$r_{i,d,2,f}$	$+1.0 \cdot 10^{-4}$
	$r_{1,d,2,p}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,c}$	$+5$	$r_{i,d,2,c}$	$+1.0 \cdot 10^{-2}$
			$r_{o,d,1,u}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,u}$	$+2.9$
			$r_{o,d,1,p}$	$+1.2 \cdot 10^{-1}$	$r_{i,d,1,p}$	$+7.2$
Summer, Inhabited area	$r_{1,a,1}$	$+6.0$	$r_{o,a,2,s}$	$-6.4 \cdot 10^{-1}$	$r_{i,a,2,n}$	$-1.0 \cdot 10^{-3}$
	$r_{1,d,2,c}$	$+1.0 \cdot 10^{-2}$	$r_{o,a,2,n}$	$-5.0 \cdot 10^{-6}$	$r_{i,d,1,f}$	$+8.0 \cdot 10^{-1}$
	$r_{1,d,2,u}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,f}$	$+3.6 \cdot 10^{-2}$	$r_{i,d,2,f}$	$+1.0 \cdot 10^{-4}$
	$r_{1,d,2,p}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,c}$	$+10$	$r_{i,d,2,c}$	$+1.0 \cdot 10^{-2}$
			$r_{o,d,1,u}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,u}$	$+1.7$
			$r_{o,d,1,p}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,p}$	$+1.7$
Winter, Inhabited area	$r_{1,a,1}$	$+6.0$	$r_{o,a,2,s}$	$-6.4 \cdot 10^{-1}$	$r_{i,a,2,n}$	$-1.0 \cdot 10^{-3}$
	$r_{1,d,2,c}$	$+1.0 \cdot 10^{-2}$	$r_{o,a,2,n}$	$-5.0 \cdot 10^{-6}$	$r_{i,d,1,f}$	$+8.0 \cdot 10^{-1}$
	$r_{1,d,2,u}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,f}$	$+3.0 \cdot 10^{-2}$	$r_{i,d,2,f}$	$+1.0 \cdot 10^{-4}$
	$r_{1,d,2,p}$	$+1.0 \cdot 10^{-1}$	$r_{o,d,1,c}$	$+10$	$r_{i,d,2,c}$	$+1.0 \cdot 10^{-2}$
			$r_{o,d,1,u}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,u}$	$+1.7$
			$r_{o,d,1,p}$	$+4.8 \cdot 10^{-2}$	$r_{i,d,1,p}$	$+1.7$

Table A6: Type 1 Daphne's parameters for probability distributions $p(c_1|a_1, d_2)$, $p(c_{2,o}|d_1, d_2, a_2)$, and $p(c_{2,i}|c_{2,o}, d_1, d_2, a_2)$ for all the scenarios.

Type and scenario	d_1				d_2			
	$d_{1,f}$	$d_{1,c}$	$d_{1,u}$	$d_{1,p}$	$d_{2,f}$	$d_{2,c}$	$d_{2,u}$	$d_{2,p}$
Type 1 - Summer, Forest	0	1	0	1	0	0	0	0
Type 1 - Winter, Forest	0	1	0	1	0	0	0	0
Type 1 - Summer, Near crossing	0	1	0	1/2	0	0	0	0
Type 1 - Winter, Near crossing	0	1	0	1/2	0	0	0	0
Type 1 - Summer, Inhabited area	0	1/2	0	1	0	0	0	0
Type 1 - Winter, Inhabited area	0	1	0	1	0	0	0	0
Type 2 - Summer, Forest	0	1	0	0	0	0	0	0
Type 2 - Winter, Forest	0	1	0	1/2	0	0	0	0
Type 2 - Summer, Near crossing	0	1	0	1/2	0	0	0	0
Type 2 - Winter, Near crossing	0	1	0	1/2	0	0	0	0
Type 2 - Summer, Inhabited area	0	1/2	0	0	0	0	0	0
Type 2 - Winter, Inhabited area	0	1	0	0	0	0	0	0
Type 3 - Summer, Forest	0	1	0	1	0	0	0	0
Type 3 - Winter, Forest	0	1	0	1	0	0	0	0
Type 3 - Summer, Near crossing	0	1	0	1/2	0	0	0	0
Type 3 - Winter, Near crossing	0	1	0	1/2	0	0	0	0
Type 3 - Summer, Inhabited area	0	1/2	0	1	0	0	0	0
Type 3 - Winter, Inhabited area	0	1	1/2	1	0	0	0	0

Table A7: Daphne's optimal decisions for the six scenarios.

Scenario	a_1	a_2				
		$c_1 = 0$	$c_1 = 1$	$c_1 = 2$	$c_1 = 3$	$c_1 = 4$
Type 1 - Summer, Forest	1	4	4	4	4	4
Type 1 - Winter, Forest	1	4	4	4	4	4
Type 1 - Summer, Near crossing	2/3	4	4	4	4	4
Type 1 - Winter, Near crossing	1	4	4	4	4	4
Type 1 - Summer, Inhabited area	2/3	4	4	4	4	4
Type 1 - Winter, Inhabited area	2/3	4	4	4	4	4
Type 2 - Summer, Forest	1	8	8	8	8	8
Type 2 - Winter, Forest	1	8	8	8	8	8
Type 2 - Summer, Near crossing	1	8	8	8	8	8
Type 2 - Winter, Near crossing	1	8	8	8	8	8
Type 2 - Summer, Inhabited area	2/3	8	8	8	8	8
Type 2 - Winter, Inhabited area	2/3	8	8	8	8	8
Type 3 - Summer, Forest	2/3	2	2	2	2	2
Type 3 - Winter, Forest	2/3	2	2	2	2	2
Type 3 - Summer, Near crossing	2/3	2	2	2	2	2
Type 3 - Winter, Near crossing	2/3	2	2	2	2	2
Type 3 - Summer, Inhabited area	2/3	2	2	2	2	2
Type 3 - Winter, Inhabited area	2/3	2	2	2	2	2

Table A8: Apollo's optimal decisions for the six scenarios.

	SF	WF	SC	WC	SI	WI
$P[a_1 = 0, a_{2,n} = 1] = 1$						
$P[a_1 = 1, a_{2,n} = 8] = 1$						
$(k_{d,1}, k_{d,2}) \rightarrow 0.8 \cdot (k_{d,1}, k_{d,2})$		x			x	
$(k_{d,1}, k_{d,2}) \rightarrow 1.2 \cdot (k_{d,1}, k_{d,2})$		x				x
Remove $B_{d,max}$						

Table A9: Sensitivity analyses for Daphne. A cell is red and marked with an x if the result of the decision problem changed as a result of the parameter change and green if it did not. Legend: SF = Summer, Forest; WF = Winter, Forest; SC = Summer, Near crossing; WC = Winter, Near crossing; SI = Summer, Inhabited area; WC = Winter, Inhabited area.

	SF	WF	SC	WC	SI	WI
$P[d_1 = d_2 = (1, 0, 0, 0)] = 1$						
$P[d_1 = d_2 = (0, 0, 0, 1)] = 1$						
$(k_{a,1}, k_{a,2}) \rightarrow 0.8 \cdot (k_{a,1}, k_{a,2})$			x			
$(k_{a,1}, k_{a,2}) \rightarrow 1.2 \cdot (k_{a,1}, k_{a,2})$						
Remove $B_{a,max}$	x	x	x	x	x	x

Table A10: Sensitivity analyses for Apollo. A cell is red and marked with an x if the result of the decision problem changed as a result of the parameter change and green if it did not. Legend: SF = Summer, Forest; WF = Winter, Forest; SC = Summer, Near crossing; WC = Winter, Near crossing; SI = Summer, Inhabited area; WC = Winter, Inhabited area.