CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Counterterrorism planning by multi-objective multi-agent reinforcement learning
Authors	Prestwich, Steven;Dogan, Vedat;O'Sullivan, Barry
Publication date	2025
Original Citation	Prestwich, S., Dogan, V. and O'Sullivan, B. (2025) 'Counterterrorism planning by multi-objective multi-agent reinforcement learning', in Arabnia, H. R., Deligiannidis, L., Shenavarmasouleh, F., Amirian, S., Ghareh Mohammadi, F. (eds) Computational Science and Computational Intelligence. CSCI 2024. Communications in Computer and Information Science, 2510, pp. 51-63. Cham: Springer. https://doi.org/10.1007/978-3-031-94956-2_4
Type of publication	Conference item;book-chapter
Link to publisher's version	10.1007/978-3-031-94956-2_4
Rights	© 2025, The Authors, under exclusive license to Springer Nature Switzerland AG. This version of the paper has been accepted for publication, after peer review (when applicable) and is subject to Springer Nature's AM terms of use, but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record is available online at: https://doi.org/10.1007/978-3-031-94956-2_4
Download date	2026-11-15 15:03:20
Item downloaded from	https://hdl.handle.net/10468/17864



UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

Counterterrorism Planning by Multi-Objective Multi-Agent Reinforcement Learning

Steven Prestwich*[0000−0002−6218−9158], Vedat Dogan[0000−0002−3807−5425], and Barry O'Sullivan[0000−0002−0090−2085]

Insight SFI Research Centre for Data Analytics
School of Computer Science & IT, University College Cork, Ireland
{steven.prestwich, vedat.dogan, barry.osullivan}@insight-centre.org

Abstract. In areas including counterterrorism, security, diplomacy and supply chain optimisation, an analyst must make decisions under assumptions about the risks posed by an adversary. Research fields including operations research, decision theory, game theory, influence diagrams and adversarial risk analysis provide a rich variety of methods to model and solve such problems. Reinforcement learning (RL) is also an approach to sequential decision making that has been applied to specific problems involving risk. We propose multi-objective multi-agent RL (MOMARL) as a general-purpose approach to risk analysis. Using a MOMARL solver we model and solve variants of a problem in counterterrorism, including a notoriously difficult problem class: pessimistic bilevel optimisation under uncertainty.

Keywords: Counterterrorism · Reinforcement Learning · Multi-Objective · Multi-Agent.

1 Introduction

In application areas such as counterterrorism, security, diplomacy and supply chain optimisation, an analyst must reason under uncertainty and incomplete information, and make decisions under assumptions about the risks posed by other decision makers. Research fields such as operations research, game theory, influence diagrams and adversarial risk analysis provide methods to model and solve such problems, as well as traditional risk analysis.

However, few researchers are expert in all these areas. When faced with a new application a researcher might make simplifying assumptions or approximations in order to fit it into a familiar framework. For example a stochastic programmer might model an adversary as a chance variable; an integer or constraint programmer might model a chance variable as an expectation; and a game theorist might assume complete information. Another drawback of the fragmentation of approaches is that some applications do not fit into any existing framework.

* Corresponding Author.

This problem has motivated the study of hybrid problems such as bi-level multi-objective stochastic integer linear programming [5], but it is easy to conceive of novel hybrid problems that have not yet been studied.

Reinforcement learning (RL) is an approach to sequential decision making that has been applied to specific problems involving risk. RL has been proposed as a unifying approach to sequential decision making under uncertainty [11], and we propose multi-objective multi-agent RL (MOMARL) as a general-purpose approach to risk analysis. To illustrate our approach, we model and solve variants of a counterterrorism problem from the literature using MOMARL.

The rest of this paper is organized as follows. In Section 2 we explain the counterterrorism problem and the motivation behind the proposed framework. In Section 3 we model the problem and describe the MOMARL solver used to solve it. Section 4 presents several new variants of the problem, with empirical results in each case. Finally, Section 5 concludes the paper by discussing limitations and future works.

2 The problem

We consider a small but interesting problem in counterterrorism taken from [9], who used it to illustrate the benefit of applying game theory as opposed to classical risk analysis. The problem involves a defender planning for a possible terrorist attack on a railway station, by two different types of attacker with different payoffs. One type (T1) prefers to attack infrastructure and not humans, while the other type (T2) prefers to target humans. The attacker has a choice of striking during rush hour when the station is crowded, or at another time when few humans are present.

The defender must invest limited resources into making different types of attack less desirable to the attackers. The defender moves first by investing resources, then the attacker reacts in full knowledge of the defender's actions. This is an example of a *defender-attacker game* in game theory. Both players in the game are assumed to act rationally to optimise their own utilities.

Attacker and defender payoff tables are shown in Table 1. H1 denotes a human target and H2 non-human; R1 denotes rush hour and R2 non-rush hour; T1 and T2 denotes the two types of attacker; and D denotes the defender. The payoffs reflect the objectives of the players, who each aim to maximise their payoffs.

T1	H1	H2
R1	-2	1
R2	-1	2

T2	H1	H2
R1	3	2
R2	2	1

D	H1	H2
R1	-6	-3
R2	-3	-1

Table 1. Attacker and defender payoffs

The defender has a budget of up to 5 defence units to invest in the four scenarios R1H1, R1H2, R2H1 and R2H2, with a cost of 0.2 for each invested unit.

Some or all of these units might not be invested. The consequence of invested units to the attacker is to reduce utility: if U units are invested in a scenario (such as R1H1), U is subtracted from the payoff for that scenario. Following [9] we represent an investment plan by a list of 4 numbers (d_1, d_2, d_3, d_4) corresponding to investment in the scenarios (R1H2, R1H1, R2H1, R2H2). Three cases are considered: pure T1, pure T2 and a T1/T2 mixture. In the latter case we assign a probability of 0.5 to each possibility, and the defender must maximise an *expected* utility.

There are two additional complications. Firstly, if a player has two or more equal-best actions, he should randomly choose between them. This is a form of mixed strategy (from game theory) or stochastic policy (from RL). For example if the defender uses an investment plan $(0, 2, 0, 0)$ then the T2 payoff for scenario R1H1 is modified from 3 to 1, leaving two equal-best options R1H2 and R2H1 each with payoff 2. Secondly, if the modified payoffs are all negative or zero then there is no benefit for the attacker, so no attack should take place. We shall treat these two complications as constraints.

The problem can be seen as one of bilevel optimisation, which involves two decision-makers or agents: a *leader* who makes initial decisions and must optimise an objective, and a *follower* who reacts and must optimise another objective that might be the same, directly opposed or orthogonal to the leader objective. Both must also satisfy constraints, and the optimality of the follower's solution is a constraint of the leader's optimisation problem. In our application the defender is the leader while the attacker is the follower. In the T1/T2 case the problem is one of bilevel optimisation under uncertainty, as the defender must make a plan without knowing the type of the attacker, which is represented by a chance variable.

Many applications are naturally bilevel, for example in hierarchical decision making within an organisation, farming, homeland security, logistics, interdiction, design, supply chain management and hyperparameter tuning. These problems are therefore of great interest in game theory, operations research and artificial intelligence. They are known to be strongly NP-hard, and even evaluating a solution for optimality is NP-hard. Special cases (linear, continuous, differentiable, convex) can be solved efficiently by reduction to a single-level problem, but the general case is intractable. Most research focuses on continuous bilevel optimisation, sometimes with *discrete* leader or follower, but less often on discrete leader *and* follower. For such problems we must often resort to nested evolutionary algorithms, or evolutionary algorithms paired with other methods such as integer linear programming, which are computationally expensive [1]. A recent survey on methods and applications can be found in [15] and a survey of bilevel optimisation under uncertainty is given in [3].

3 A MOMARL approach

We shall apply reinforcement learning (RL) to the problems, specifically multi-objective multi-agent RL (MOMARL). Surprisingly, very little work has been

reported on applying RL to bilevel optimisation, though its success on multi-agent decision problems (such as learning to play Chess and Go) seems to make it a natural candidate and *risk-averse* RL is an active research area (see for example [4]). RL is not mentioned in the bilevel surveys of [3, 15].

3.1 A solver

We use a MOMARL solver called InfProg [13] based on infinite-step SARSA with random tile coding. It is a prototype designed for complex nonlinear optimisation problems involving decision and chance variables, with each decision variable belonging to an agent (decision maker). There may be multiple agents, and each agent may have multiple objectives. There are additional features (correlated chance variables, observations) that are not relevant here. It has been applied to a 2-stage stochastic program, a chance-constrained program, bilevel and trilevel problems, and a multi-objective influence diagram. For this paper we modify the solver in two ways:

- To handle stochastic policies, which were not considered in [13]. In a given state, if there are 2 (or more) equal-best actions then the solver chooses randomly between them. The quality of an action is estimated by a floating-point value in a lookup table, and as a general principle floating point numbers should not be tested for equality, so we consider them to be essentially equal if they differ by no more than a small amount (we use 0.00001).
- The original solver used L1 weighted metric scalarisation to reduce multiple objectives to single objectives, but for this work we use linear scalarisation which requires fewer hyperparameters.

An advantage of using such a solver is that solving a complex problem is reduced to a modelling problem, with no need to consider algorithmic details. The class of problem that can be modelled and solved by InfProg is called a *(discrete) influence program* (IP) [13] which can be represented by a tuple $\langle \mathcal{V}, \mathcal{D}, \mathcal{A}, \mathcal{U}, \mathcal{L}, \mathcal{O}, \mathcal{P} \rangle$ where:

- $\mathcal{V} = \langle v_1, \ldots, v_n \rangle$ is an ordered list of variables v_i;
- $\mathcal{D} = \langle D_1, \ldots, D_n \rangle$ is a list of their corresponding finite domains D_i of possible values, typically ranges of integers or sets of symbolic names;
- $\mathcal{A} = \langle A_1, \ldots, A_n \rangle$ is a list of their corresponding *agents* (decision-makers) A_i, which have symbolic names (in **bold**);
- $\mathcal{U}$ is a function mapping a total variable assignment to a utility vector for each agent;
- $\mathcal{L}$ is a set of directed links $v_i \rightarrow v_j$ $(i < j)$ between variables;
- $\mathcal{O}$ is a set of *observations*, where an observation is an assignment $v_i = d_j$ of a value $d_j \in D_i$ to a chance variable x_i;
- $\mathcal{P}$ is a function assigning a probability to each chance variable v_j assignment given an assignment for each variable v_i such that $v_i \rightarrow v_j \in \mathcal{L}$: it is typically represented by a (conditional) probability table.

Each variable is associated with one agent, with chance variables associated with a dummy agent called **chance**. The links $\mathcal{L}$ define which variables are visible to later decisions and chance variable distributions. A utility is a function mapping a total variable assignment (one value per variable) to a real value. The utilities are programmable, requiring the user to write a small function to compute utilities from a total variable assignment. Each non-**chance** agent has at least one utility. The aim of an IP is to find a policy for each agent that Pareto-optimises its utilities in the context of observations and inter-variable visibility.

3.2 Modelling the problem

To model the defender's choices we use 4 variables d_1, d_2, d_3, d_4 each taking integer values in the range 0–5. These values refer to investment in scenarios R1H2, R1H1, R2H1 and R2H2 respectively. To model the type of attacker we use a binary chance variable t which is 0 for T1 and 1 for T2: the probability distribution can be adjusted to obtain pure T1, pure T2 or the 50–50 mixed case. To model the attacker's choice of attack we use an integer decision variable a which takes the value 0 for no attack, 1 for the R1H2 attack case, 2 for R1H1, 3 for R2H1 and 4 for R2H2. Hence $\mathcal{V} = \langle d_1, d_2, d_3, d_4, t, a \rangle$. The domains are $\mathcal{D} = \langle [0,5], [0,5], [0,5], [0,5], [0,1], [0,1] \rangle$. The corresponding agents are $\mathcal{A} = \langle$**D, D, D, D, chance, A**$\rangle$ where **D** is the defender and **A** the attacker. All decision variables can see all previous variables, so $\mathcal{L} = \langle d_i \to d_j,\ d_i \to a,\ t \to a \rangle$ $(1 \leq i < j \leq 4)$. There are no Bayesian network-style observations so $\mathcal{O} = \emptyset$. The function $\mathcal{P}$ defines a discrete uniform distribution over $\{0, 1\}$ for chance variable a.

We have not yet defined the utility $\mathcal{U}$, which in optimisation terms represents the objective to be optimised, or in RL terms the reward. The defender has a budget of up to 5 defence units, so we require a constraint

$$\sum_{i=1}^{4} d_i \leq 5$$

There are various ways of handling constraints in RL, and in [13] a multi-objective approach was used: the leader and follower each had an additional objective to minimise the expected number of constraint violations to 0. Here we use a *repair* approach that gave better results in experiments: when calculating the agents' rewards, if a constraint is violated then the decisions are modified so that it is satisfied. The modification should be deterministic so that it can be recreated after training ends. We use a simple procedure: if the sum of the d_i exceeds 5 then we subtract 1 from each variable in turn, round-robin style, until the constraint is satisfied.

A constraint is also required for the other complication mentioned earlier: that if the best modified payoff is negative or zero then there is no benefit for the attacker, so no attack should take place. We also handle this by repair: if the modified payoff is negative or zero and $a = 1$ then we set $a = 0$.

The use of repair creates a form of symmetry in the model, which wastes some runtime effort: several infeasible plans learned by RL map to the same feasible plan, but each case must be learned independently. However, the symmetry is not excessive.

We can now define the utility $\mathcal{U}$. InfProg allows programmable utilities, which are computed at the end of each RL episode when all variables have been assigned values. For this application the d_i values are first adjusted to represent a feasible investment, as described above. The payoff tables and investments are then used to compute defender and attacker payoffs. If the attacker payoff is not strictly positive then the value of a is set to 0 (if it was 1). If $a = 0$ then there is no attack and the payoffs are adjusted accordingly.

To summarise the model: the defender agent **D** makes a series of 4 decisions d_1, d_2, d_3, d_4 on how many units to invest in each of 4 scenarios, then the type of attacker is chosen according to the probability distribution of **chance** variable t, then an attack is chosen by the attacker agent **A** by assigning a value to variable a. The latter can be based on the defender's investments and on the attacker type. There are two constraints to be satisfied, which are handled by a repair mechanism.

4 Experiments

The above IP model corresponds to the problem described in [9] but in this section we shall also define variants of the problem, and apply the modified InfProg solver to them. It was run 10 times on each problem with 1M episodes per run, taking just over 1 second per run, and the best solution reported in each case.

4.1 Original problem

First we consider the original problem from [9].

T1 attacker The known optimal plan is $(1, 0, 0, 2)$ with defender score -0.6 and attacker score 0 (hence no attack), which we shall denote by $(-0.6, 0)$. This plan reduces all attacker payoffs to zero or less, and it is found by the solver.

T2 attacker The known optimal plan is $(1, 3, 1, 0)$ with scores $(-3.33, 1)$. This plan modifies all T2 payoffs to be 1 apart from the R1H1 case which is 0, so the T2 attacker will randomly choose from attacks R1H2, R2H1 and R2H2 with expected score 1. Hence the expected defender payoff is the mean of -3, -3 and -1, which is approximately -2.33. 5 units have been invested, costing 5×0.2 so the total defender score is -3.33. The solver finds this plan.

Mixed attacker The optimal plan reported in [9] is $(1, 3, 0, 0)$ with scores $(-2.8, 2)$. However, our solver finds the plan $(0, 2, 0, 0)$ with improved scores $(-2.4, 2)$. In the T1 case our plan leaves R2H2 with the best attacker score of 2 and defender score -1.4. In the T2 case the plan makes R1H2 and R2H1 the equal best attacks, with attacker score 2 and defender score -3.4 in both cases. Hence the expected scores of this plan are -2.4 for the defender and 2 for the attacker.

4.2 Pessimistic variant

A notoriously hard form of bilevel optimisation is *pessimistic* [8]. When the follower has multiple optimal solutions, or in the multi-objective case a Pareto front of solutions, it is convenient to assume that the chosen solution optimises the leader objective. Optimistic problems are commonly modelled simply because they are much easier to solve than pessimistic ones: see for example [6]. The optimistic assumption is of course unrealistic, especially in the multi-objective case, though it can be partly justified by arguing that there might be limited cooperation between the players to the extent that the follower altruistically chooses an optimal solution that also benefits the leader, or that the leader may be able to make small side payments that bias the follower's objective in his favor. However, a pessimistic approach is more valuable to a risk-averse leader: if we assume that the follower will choose the worst optimal solution from the leader's point of view, then we know that the leader solution will be optimal for all follower choices.

This is a particularly interesting variant in which we assume that the attacker will break ties between equal-best options by making the worst choice from the defender's point of view: a risk-averse approach by the defender. In the mixed T1/T2 case we have a pessimistic bilevel problem under uncertainty.

The IP model can incorporate pessimism via a multi-objective approach: add a new objective to the follower, which is the negation of the leader objective. This new objective is used to break ties among optimal follower solutions, instead of choosing randomly as in the original problem. In our linear scalarisation approach we use a coefficient of 1 for the attacker objective and 0.1 for the tie-breaking objective.

T1 attacker We find the same plan as for the original problem: $(1, 0, 0, 2)$ with the same scores $(-0.6, 0)$.

T2 attacker We find plan $(0, 2, 0, 0)$ with scores $(-3.4, 2)$. This is worse for the defender than the original problem, as might be expected. The plan makes the equal best T2 attacks R1H2 and R2H1, each with the same attacker payoff of 2. They also have the same defender payoff of -3 so either can be chosen, and the defender score is -3.4 including the investment cost of 0.4.

Mixed attacker We find plan $(0, 2, 0, 0)$ with scores $(-2.4, 2)$ as in the original problem.

4.3 Optimistic variant

The pessimistic model can be adapted to an optimistic model simply by changing the sign of the new objective, using the leader objective instead of its negation.

T1 attacker We find the same plan as in the original problem: $(1, 0, 0, 2)$ with scores $(-0.6, 0)$.

T2 attacker We find plan $(1, 2, 1, 0)$ with scores $(-1.8, 1)$, which is better for the defender than the original problem. The plan makes all attacker payoffs equal to 1. Optimism then chooses the best from the defender point of view, which is R2H2 with scores $(-1.8, 1)$.

Mixed attacker We again find plan $(1, 2, 1, 0)$ but with scores $(-1.8, 1.5)$, which is better for the defender than the original problem. In the T1 case R2H2 has best payoffs $(-1.8, 2)$. In the T2 case we already showed that the scores are $(-1.8, 1)$.

4.4 Strong-weak variant

As a compromise between the optimistic and pessimistic approaches, in an α-strong-weak model a parameter $\alpha \in [0, 1]$ controls the leader's level of conservatism, where α is the probability that the follower chooses the optimistic solution; otherwise with probability $1 - \alpha$ the pessimistic solution is chosen. For any fixed α the problem of finding an optimal solution is strongly NP-hard even in the case of linear bilevel optimisation, and when optimal optimistic and pessimistic solutions are known [7].

We can model this variant by introducing a new discrete **chance** variable we shall call o. It can be placed anywhere in the IP variable list as it only affects the objective, and we arbitrarily place it at the end of the variable list. It is invisible to any decision variable so no new links are added to $\mathcal{L}$. Its domain is $[0, 1]$ with 0 indicating pessimistic and 1 optimistic, with probability distribution $(\alpha, 1 - \alpha)$. We choose $\alpha = 0.5$ for our experiment. The new objective is modified by multiplying it by $2o - 1$ so that the follower is pessimistic if $\alpha = 0$ and optimistic if $\alpha = 1$.

T1 attacker We find plan $(1, 0, 0, 2)$ with scores $(-0.6, 0)$, as in the pessimistic and optimistic cases.

T2 attacker We find plan $(1, 3, 1, 0)$ with scores $(-3.33, 1)$: the optimal plan for the original problem, lying between the optimistic and pessimistic cases.

Mixed attacker We find plan $(0, 2, 0, 0)$ with scores $(-2.4, 2)$. This plan is optimal for the original problem, and lies between the original and pessimistic cases but not the optimistic case.

4.5 Simultaneous game (I)

A further possible complication is *partial knowledge*, for example in Bayesian games, where the follower might not know some of the leader's decisions. This might arise if the attacker's surveillance is incomplete or unreliable. In the case where neither player knows *any* of the others' decisions, we have a *simultaneous game* in which both must make decisions independently; our previous variants were *sequential games*.

If we remove all links $d_i \rightarrow a$ from $\mathcal{L}$ from the model of the original problem, then the attacker does not know the defender's actions and must formulate an attack plan that does not depend on them. This makes the attacker's task harder so we expect a worse or equal attacker score, and consequently a better or equal defender score.

T1 attacker We find plan $(1, 0, 0, 2)$ with scores $(-0.6, 0)$ as in the original problem.

T2 attacker We find plan $(2, 0, 0, 0)$ with scores $(-1.4, 1)$. This is better for the defender and no worse for the attacker than the original plan $(1, 3, 1, 0)$ with scores $-3.33, 1$.

Mixed attacker We find plan $(1, 0, 2, 0)$ with scores $(-1.6, 1.5)$, which as expected is worse for the attacker and better for the defender, whether the original problem has optimal plan $(1, 3, 0, 0)$ with scores $(-2.8, 2)$ (according to [9]) or $(0, 2, 0, 0)$ with scores $(-2.4, 2)$ (according to our result).

4.6 Simultaneous game (II)

If we remove the link $t \rightarrow a$ from $\mathcal{L}$ from the model of Section 4.5 then the attacker does not know what type he is. This could model a situation in which the defender believes that the attack has been planned by a high-ranking terrorist T who is not sure which agent A will carry it out, so that T must formulate an attack plan that works well whichever A is used.

T1 attacker We find plan $(1, 0, 0, 2)$ with scores $(-0.6, 0)$ as before.

T2 attacker We find plan $(2, 0, 0, 0)$ with scores $(-1.4, 1)$ as before.

Mixed attacker We find plan $(0, 0, 1, 0)$ with scores $(-1.2, 1.5)$ which is better for the defender and no worse for the attacker than the previous partial information variant (Section 4.5).

4.7 Adversarial risk analysis

In adversarial risk analysis (ARA) [2] an analyst expresses beliefs about an adversary's utilities, capabilities, probabilities and type of strategic thinking. The analyst then forms an optimal plan based on these beliefs, using subjective probability distributions to model unknowns. ARA has incorporated a game theoretic approach to handling incomplete information called *level-k reasoning* [10, 16] which assumes that players play strategically but with bounded rationality. It has recently been used in adversarial risk analysis for modelling problems in counterterrorism [14] and other fields. A simple example of level-k thinking was solved using MOMARL in [13], and we now devise an ARA variant of the counterterrorism problem in the same spirit.

Suppose that we have used the original approach in the past, but have now secretly acquired much greater resources. Given this changed situation, our new aim is to prevent *any* attack while minimising expenditure. We could plan on the basis that the attacker assumes that we have only 5 resource units as before, and make a new improved plan based on that belief.

In addition to the defender **D** and the attacker **A** (plus **chance**) we will also use a hypothetical defender **D'** who behaves as **D** did in the past (in Section 4.1). **D'** has decision variables d'_1, d'_2, d'_3, d'_4 with the same domains as the d_i. The variable list is now $\mathcal{V} = \langle d'_1, d'_2, d'_3, d'_4, d_1, d_2, d_3, d_4, t, a \rangle$ where a, t are as before. $\mathcal{O}, \mathcal{P}$ are also as before, but the links are now $\mathcal{L} = \langle d_i \to d_j,\ d'_i \to d'_j,\ d'_i \to a,\ d'_i \to d_k,\ t \to a \rangle$ $(1 \leq i, j, k \leq 4,\ i < j)$: the attacker sees the decisions made by the hypothetical defender, but not those made by the real defender. The new agent **D'** uses the old **D** utility.

The new version of **D** uses a different utility. The aim is to invest just enough resources so that the investment is always at least as large as the payoff, nullifying the effects of an attack no matter which attack is used (recall that the attacker might choose randomly between more than one attack). **D** also minimises total investment via a multiobjective approach, using linear scalarisation as in Section 4.2 and elsewhere.

T1 attacker As in Section 4.1, **D'** uses plan $(1, 0, 0, 2)$ and scores -0.6, while **A** does not attack and scores 0, so **D** need not invest any units and scores $(1, 0)$: the 1 indicates that the defender payoff is covered with probability 1.0, while the 0 is the investment cost: an improvement over the previous value of -0.6. This case illustrates the advantage of adversarial risk analysis.

Of course, in the future the attacker will no longer believe that the defender behaves as before, so the defender must update his beliefs about the attacker in order to formulate a new plan.

T2 attacker Again as in Section 4.1 **D'** uses plan $(1, 3, 1, 0)$ and scores -3.33, while **A** stochastically uses plans R1H2, R2H1 and R2H2 and scores 1. Now **D** must invest enough units to cover the defender payoff for each plan: $-3, -3, -1$ respectively. So **D** uses the plan $(3, 0, 3, 1)$ and scores $(1, 7)$: again we can ignore the 1, and 7 is the total investment.

Mixed attacker Again as in Section 4.1 **D'** uses plan $(0, 2, 0, 0)$ and scores -2.4, while **A** stochastically uses plans R1H2 and R2H1 and scores 2. As in the T2 case, **D'** uses plan $(3, 0, 3, 1)$ and scores $(1, 7)$.

4.8 Summary

The results are summarised in Table 2. (Note that for the ARA case we report the **D** defender result but not the hypothetical **D'** defender, and we omit the probability component of its two objectives.) With the MOMARL solver we found the known optimal strategies for two cases, and an improved strategy for the third case. On new variants of the problem we showed the potential of applying RL to risk analysis.

Though we do not know whether most of the plans we found are optimal, they follow expected patterns: pessimism (non-strictly) decreases the defender score; optimism increases the defender score; making information unavailable to the attacker decreases the attacker score and increases the defender score. Furthermore, we showed that applying MOMARL to an ARA model can lead to improved results: the T1 case allows the defender to avoid any costs. The ARA approach can not be compared to the others using the T2 and mixed cases, as it assumes greater resources; however, we include it to illustrate that this type of model can be handled by MOMARL.

	T1 attacker		T2 attacker		mixed attacker	
	plan	score	plan	score	plan	score
original	$(1, 0, 0, 2)$	$(-0.6, 0)$	$(1, 3, 1, 0)$	$(-3.33, 1)$	$(1, 3, 0, 0)$	$(-2.8, 2)$
pessimistic	$(1, 0, 0, 2)$	$(-0.6, 0)$	$(0, 2, 0, 0)$	$(-3.4, 2)$	$(0, 2, 0, 0)$	$(-2.4, 2)$
optimistic	$(1, 0, 0, 2)$	$(-0.6, 0)$	$(1, 2, 1, 0)$	$(-1.8, 1)$	$(1, 2, 1, 0)$	$(-1.8, 1.5)$
strong-weak	$(1, 0, 0, 2)$	$(-0.6, 0)$	$(1, 3, 1, 0)$	$(-3.33, 1)$	$(0, 2, 0, 0)$	$(-2.4, 2)$
simultaneous (I)	$(1, 0, 0, 2)$	$(-0.6, 0)$	$(2, 0, 0, 0)$	$(-1.4, 1)$	$(1, 0, 2, 0)$	$(-1.6, 1.5)$
simultaneous (II)	$(1, 0, 0, 2)$	$(-0.6, 0)$	$(2, 0, 0, 0)$	$(-1.4, 1)$	$(0, 0, 1, 0)$	$(-1.2, 1.5)$
ARA	$(0, 0, 0, 0)$	$(0, 0)$	$(3, 0, 3, 1)$	$(7, 1)$	$(3, 0, 3, 1)$	$(7, 2)$

Table 2. Summary of results

5 Conclusion

We investigated several variations on a risk analysis problem from counterterrorism, applying an existing solver based on MOMARL. The modelling task was

quite straightforward, demonstrating the potential of MOMARL for a range of approaches to risk analysis. In particular, it provides a way of modeling and solving a particularly hard type of problem: pessimistic bilevel optimisation under uncertainty.

It should be mentioned that our approach did not always lead to an optimal plan. Using an alternative model (not shown) in which constraints are handled by a multiobjective approach (see [13] and an earlier work [12]), on the pure T2 version of the original problem InfProg almost always finds the near-optimal plan $(0, 2, 0, 0)$ with defender score -3.4, missing the optimal plan with defender score -3.33. Replacing linear scalarisation by Chebyshev scalarisation did not help, nor did longer runs. This result motivated the use of repair for constraint handling. More generally, it shows that the prototype solver InfProg should be replaced by something more powerful, and we are currently working on a deep RL method. However, the problem might be caused by the scalarisation approach to multiobjective optimisation. These issues will be addressed in future work: our aim in this paper is to show the convenience and flexibility of a MOMARL approach to various forms of bilevel optimisation, especially applied to problems of risk analysis.

Acknowledgments

This material is based upon works supported by the Science Foundation Ireland under Grant No. 12/RC/2289-P2 which is co-funded under the European Regional Development Fund. We would also like to acknowledge the support of the EU H2020 ICT48 project TAILOR under contract #952215.

References

1. M. Antoniou, G. Papa. Solving Pessimistic Bilevel Optimisation Problems With Evolutionary Algorithms. *Proceedings of the 14th International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control*, 2021.
2. D. Banks, V. Gallego, R. Naveiro, D. R. Insua. Adversarial Risk Analysis: An Overview. *WIREs Computational Statistics* **14**(1), 2020.
3. Y. Beck, I. Ljubić, M. Schmidt. A Survey On Bilevel Optimization Under Uncertainty. *European Journal of Operational Research* **311**:401–426, 2023.
4. L. Bisi, D. Santambrogio, F. Sandrelli, A. Tirinzoni, B. D. Ziebart, M. Restelli. Risk-Averse Policy Optimization Via Risk-Neutral Policy Optimization. *Artificial Intelligence* **311**, 2022.
5. M. M. K. Elshafei, M. S. El-Sherberry. Interactive Bi-level Multiobjective Stochastic Integer Linear Programming Problem. *Trends in Applied Sciences Research* **3**(2):154–164, 2008.
6. T. Kis, A. Kovács, Csaba Mészáros. On Optimistic and Pessimistic Bilevel Optimization Models for Demand Response Management. *Energies* 14(8), 2021.
7. T. Lagos, O. A. Prokopyev. On Complexity of Finding Strong-Weak Solutions in Bilevel Linear Programming. *Operations Research Letters* **51**(6):612–617, Elsevier, 2023.

8. J. Liu, Y. Fan, Z. Chen. Pessimistic Bilevel Optimization: A Survey. *International Journal of Computational Intelligence Systems* **11**(1):725–736, 2018.
9. S. Meng, M. Wiens, F. Schultmann. A Game-Theoretic Approach to Assess Adversarial Risks. *WIT Transactions on Information and Communication Technologies*, 2014.
10. R. Nagel. Unraveling in Guessing Games: an Experimental Study. *Amer. Econ. Rev.* **85**(5):1313–1326, 1995.
11. W. B. Powell. Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions. Wiley, 2022.
12. S. D. Prestwich. Solving Mixed Influence Diagrams by Reinforcement Learning. *Proceedings of the 9th International Conference on Machine Learning, Optimization, and Data Science*, 2023.
13. S. D. Prestwich. Solving Complex Optimisation Problems by Machine Learning. *AppliedMath* **4**(3):908–926, MDPI, 2024.
14. C. Rothschild, L. Albert, S. Guikema. Adversarial Risk Analysis with Incomplete Information: A Level-k Approach. *Risk Analysis* **32**(7):1219–31, 2011.
15. A. Sinha, P. Malo,K. Deb. A Review on Bilevel Optimization: From Classical to Evolutionary Approaches and Applications. *IEEE Transactions on Evolutionary Computation* **22**:276–295, 2017.
16. D. O. Stahl, P. W. Wilson. On Players' Models of Other Players: Theory and Experimental Evidence. *Games Econ. Behav.* **10**(1):218–254, 1995.