Timely exposure of a secret project: Which activities to monitor?

Ben Hermans*[,a], Herbert Hamers[b], Roel Leus[a] and Roy Lindelauf[c]

[a] *Research Center for Operations Research & Business Statistics, KU Leuven, Belgium*
[b] *CentER, Department of Econometrics and Operations Research, and TIAS, Tilburg University, The Netherlands*
[c] *Intelligence & Security, Netherlands Defense Academy, The Netherlands*

Abstract

A defender wants to detect as quickly as possible whether some attacker is secretly conducting a project that could harm the defender. Security services, for example, need to expose a terrorist plot in time to prevent it. The attacker, in turn, schedules his activities so as to remain undiscovered as long as possible. One pressing question for the defender is: which of the project's activities to focus intelligence efforts on? We model the situation as a zero-sum game, establish that a late-start schedule defines a dominant attacker strategy, and describe a dynamic program that yields a Nash equilibrium for the zero-sum game. Through an innovative use of cooperative game theory, we measure the harm reduction thanks to each activity's intelligence effort, obtain insight into what makes intelligence effort more effective, and show how to identify opportunities for further harm reduction. We use a detailed example of a nuclear weapons development project to demonstrate how a careful trade-off between time and ease of detection can reduce the harm significantly.

1 Introduction

In response to the terrorist attacks in Paris early 2015, a number of European countries, notably France and Belgium, decided to deploy soldiers on the streets for guarding against new aggressions. Referring to the field manual of the Belgian operation, Lasoen [24] reports that the objective is "to prevent, deter and defeat threats or aggression by terrorists by providing assistance to the police [...] in order to buy time for the latter to intervene." But in the whole chain of events preceding a terrorist attack, troops on the streets are only able to intervene at the very last step: the attack itself. And even though soldiers successfully prevented an attack in Brussels by shooting a suspected terrorist [35], Lasoen [24] mentions that "the threat should have been neutralized long before defensive action or consequence management is necessary."

Various groups, organizations, and even state actors conduct covert operations that need to be exposed in time to limit the harm they inflict. A hostile state's suspected development of a nuclear weapon has to be discovered as soon as possible to interdict it, a terrorist plot must be detected in time to prevent an attack, and a major batch of synthetic drugs needs to be intercepted before it reaches the market. Even in a corporate environment, the ability to anticipate a competitor's market entrance or new product launch can determine firm survival.

*Ben Hermans is funded by a PhD Fellowship of the Research Foundation – Flanders.

One common feature of these covert operations is that they all consist of different steps that, together, form a project: Harney et al. [18] describe the tasks necessary to produce a nuclear weapon, Stewart [39] structures the events preceding a terrorist attack into a "terrorist attack cycle," and Chiu et al. [10] examine the steps to manufacture illicit drugs. When the project's activities differ in time and ease of detection, a pressing question becomes: which activities to focus intelligence efforts on?

In this paper, we study a defender who wants to detect whether some attacker is secretly conducting a project that could harm the defender. As soon as the defender learns about the project's existence, she can start taking harm-reducing countermeasures, and thus she wants to discover the project as early as possible. The defender can raise the probability for detecting a task through additional *monitoring*: satellites can attempt to register nuclear weapons tests, social workers can try to detect radicalization, or the police can screen the supply of drug precursors. Given that the defender has only a limited budget for monitoring and that the attacker takes into account the defender's decisions when planning his project, the question of which activities to monitor becomes an important yet non-trivial one.

The goal of this paper is twofold. Firstly, we aim to determine how the defender should allocate her limited resources for monitoring. By doing so, we follow up on the research question posed by Kaplan [21] of "what is the best way to allocate intelligence resources across the many competing areas of intelligence concern?" Secondly, we want to evaluate the performance of each task's intelligence effort and explain why performance may differ. Thus, we respond to the suggestion of Kaplan [21] to "develop methods for the performance evaluation of intelligence activity."

To reach our first goal, we introduce a zero-sum game – the *secret project game* – between the defender and attacker; establish that a late-start schedule defines a dominant attacker strategy; and describe a dynamic program that, given the attacker's schedule and the defender's limited budget, identifies the optimal set of tasks to monitor. Although the problem is NP-hard, our pseudo-polynomial-time dynamic program can handle realistically-sized instances, and thus it yields an effective method for identifying a Nash equilibrium for the zero-sum game. To the best of our knowledge, we are the first to model a defender's decision with respect to which activities to monitor in order to expose an attacker's project.

For our second goal, we introduce a cooperative *monitoring game* that captures the reduction in expected harm when monitoring a set of tasks; the Banzhaf value of a task in this game then measures the harm reduction thanks to monitoring a task. By means of an intuitive expression for this Banzhaf value, we show how the defender can quantify the influence of a task's starting time and discovery probability on the harm reduction. Our use of cooperative game theory to obtain insight into an optimization problem appears to be novel.

After reviewing the literature in Section 2, we formalize and determine a Nash equilibrium for a secret project game in Section 3. Next, we define and analyze monitoring games in Section 4 and, in Section 5, we apply our methods to the nuclear weapons development project of Harney et al. [18]. Section 6 concludes and proposes further research.

2 Related work

Our work combines concepts from project management with a probabilistic discovery process inspired by the literature on counter-terrorism and the monitoring decision from the literature on inspection games. Below, we briefly review these domains and position our work.

A number of articles within the field of project management study the problem of planning [32, 33], interdicting [8, 9], or estimating the progress [16, 15] of a secret project. The main

difference with our work is that these articles do not explicitly model the defender's decision of monitoring tasks and do not consider a probabilistic discovery process. Most closely related is the article of Pinker et al. [32], who were the first to study the problem of planning a secret project. They consider an attacker who can use "a combination of deception, task scheduling and crashing" in order to minimize the length of the time between project discovery and completion. The authors assign a detection weight to each task and assume that discovery occurs when the cumulative weight of all initiated tasks exceeds a certain threshold. They show that the problem is NP-hard and describe an integer program to solve it. In [33], the same authors analyze the complexity status for different variants of the problem. Our finding that a late-start schedule defines a dominant attacker strategy reinforces the results of Pinker et al. [32, 33], who have also identified conditions under which a late-start schedule is optimal.

A second strand of related literature contains a wide range of probabilistic models for counter-terrorism. These articles do take into account the stochastic nature of discovery, but, contrary to our work, model the threat as a single event rather than as a project. Atkinson and Wein [2] examine how a government should allocate its resources over the inspection of terror and criminal networks to exploit the finding of Smith et al. [38] that, prior to an attack, terrorists frequently participate in crimes such as theft or procuring explosives. Other articles address problems such as predicting the number of undetected terror threats [19], estimating the duration of a terrorist plot [20], locating terrorists [1, 3], processing intelligence [26, 11], patrolling an area [40, 27, 28], and predicting the goal of a suspected terrorist [42]. In particular, Atkinson et al. [3] consider a searcher who, based on a stream of unreliable intelligence about a target's location, needs to decide whether to engage or to wait for more information. If the searcher waits too long, an attack may occur before having identified the target; if she engages too soon, she might pick the wrong location.

The literature on inspection games, finally, studies an 'inspector' that needs to verify whether some 'inspectee' adheres to certain regulations, as in the follow-up of compliance with an arms control treaty. We refer to Avenhaus et al. [5] and, for the more recent work, von Stengel [43] for an excellent and detailed overview. Our work differs from inspection games in three main aspects. Firstly, inspection games focus on the inspection of one single activity, whereas we study a project consisting of multiple activities. Secondly, we assume that monitoring pertains to structural measures that, contrary to single-period inspections, take place during the entire planning horizon. Thirdly, with the notable exception of Avenhaus and Canty [4], most authors do not take into account the benefit of discovery in an early stage of the attacker's project.

3 Secret project games

In this section, we define the zero-sum game between the defender and attacker (Section 3.1) and show that a late-start schedule describes a dominant strategy for the attacker (Section 3.2). Next, we develop a dynamic program to identify a Nash equilibrium for the zero-sum game (Section 3.3).

3.1 Problem statement

Let $N \coloneqq \{1, \ldots, n\}$ collect the n tasks that the attacker needs to execute to complete his project; examples are the tasks necessary to develop a nuclear weapon, to prepare a terrorist attack, or to manufacture a batch of synthetic drugs. Each task $i \in N$ has a *duration* $d_i \geq 0$ and the strict partial order $A \subset N \times N$ on N models the *precedence constraints*. Here, $(i, j) \in A$ indicates that task j can only start after i's completion and we assume that task n constitutes the project's (unique) final task: $(i, n) \in A$ for each $i \in N \setminus \{n\}$.

The way in which the project harms the defender depends on the specific context: it could reflect the damage in geopolitical power if another state obtains a nuclear weapon, the number of casualties due to a terrorist attack, or the amount of synthetic drugs reaching the market. As soon as the defender learns about the project's existence, however, she can take harm-reducing counteractions such as imposing diplomatic, economic, or military sanctions; or infiltrating/dismantling the terrorist/drug network. The more time the defender has to react, the more effectively she can take counteractions, and the lower the harm will be. Thus, we can model the *harm* by a function $h\colon \mathbb{R}_{\geq 0} \to \mathbb{R}$ that is non-increasing in the length of the time window between project discovery and completion. Pinker et al. [32] call this time window the *exposed time*. The highest possible harm $h(0)$ occurs if the project remains secret until completion. As the exposed time increases, the harm decreases because the defender obtains more time to take counteractions.

The defender discovers the project if she detects one of its tasks, and she can increase the probability of such detection by *monitoring* tasks. Without monitoring, task $i \in N$ exposes the project with probability $p_i^0 \in [0,1]$, which reflects that exposure can still occur through, for example, a whistle-blower that leaks a task's execution or a bomb that accidentally explodes during its production. By actively monitoring the task, in turn, the defender can increase this discovery probability to $p_i^1 \in [p_i^0, 1]$. We denote the complementary probabilities by $q_i^0 \coloneqq 1 - p_i^0$ and $q_i^1 \coloneqq 1 - p_i^1$, and we assume independence between different tasks' discovery probabilities.

Monitoring task $i \in N$ means that the defender pays particularly close attention to the respective task. This increased effort leads to a cost $c_i \in \mathbb{N}$ and the defender has a limited budget $b \in \mathbb{N}$ for monitoring. We assume that if the defender decides to monitor a task, she does so during the entire planning horizon. This seems reasonable whenever monitoring pertains to structural measures; social workers that try to detect radicalization, for example, do this continuously. Other examples include the situation where satellites attempt to register nuclear weapons tests or where the police sets up a task force that is responsible for screening the supply of drug precursors.

Finally, we suppose detection occurs at a task's initiation. Hence, the activity's execution rather than its outcome (e.g. the creation of some artifact) determines the moment of discovery. This reflects a situation where the outcome is relatively easy to conceal: when preparing a small explosive, for instance, the activity of procuring explosive materials seems much more likely to cause discovery than the actual (easily concealable) bomb does. Although we suppose detection occurs at a task's initiation, our model readily generalizes to discovery at a fixed time $f_i \geq 0$ after the initiation of task $i \in N$.

Our assumption that the defender responds as soon as she can link a single task's execution to the project reflects the, often realistic, situation where she cannot afford to wait for more evidence. For example, empirical research based on terrorist plots in the United States between 1980 and 2004 reveals that, on average, authorities only registered 1.7 activities directly related to an attack's preparation before an incident occurred [38].

The above model, summarized by the tuple $\mathcal{S} \coloneqq \big(N, A, (d_i, p_i^0, p_i^1, c_i)_{i \in N}, b, h\big)$, constitutes a *secret project situation*. We initially assume that $\mathcal{S}$ is common knowledge, but show in Section 3.2 that, as long as the objective function is non-increasing in the exposed time, the attacker's optimal strategy depends on the *project network* $(N, A, (d_i)_{i \in N})$ only.

Example. An intelligence agency (the defender) wants to detect whether some terrorist organization (the attacker) is planning to commit a nerve agent attack. Table 1 displays the necessary tasks for doing so and Figure 1 the associated project network. The project network is based on a (fictional) example from Godfrey et al. [16] and we have selected values for the remaining parameter ourselves. An unanticipated attack causes $h(0) = 50$ casualties and with $t \geq 0$ weeks time for trying to infiltrate and neutralize the terrorist network this reduces to $h(t) = \max\{0, 50 - 2t\}$ casualties. Table 1's third and fourth column show that without additional effort, the nerve

i	Description	p_i^0	p_i^1	c_i
1	Assemble team	0.05	0.35	3
2	Produce nerve agent	0.10	0.40	4
3	Develop delivery method	0.05	0.10	2
4	Prepare equipment	0.06	0.20	2
5	Select target	0.01	0.05	1
6	Attack target	0.01	0.99	1

Table 1: Tasks for preparing a nerve agent attack.

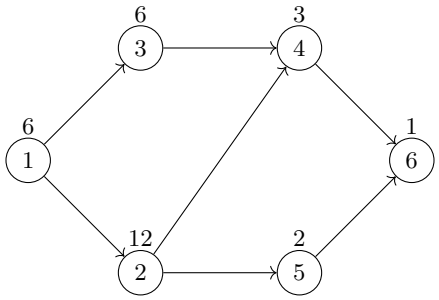

Figure 1: Project network for the nerve agent attack. Nodes represent tasks, arcs precedence constraints, and task durations (in weeks) appear above the nodes.

agent's production can still be detected with a 10% probability (for example through a civilian that signals suspicious behavior) and that the intelligence agency can increase this probability to 40% (for example by monitoring the supply of a crucial ingredient). The fifth column shows the costs of monitoring (in million euros) and we assume a budget of $b = 6$ million euros.

Since the defender needs to decide which set of tasks $S \subseteq N$ to monitor given her limited budget, the set $\mathcal{B} \coloneqq \{S \subseteq N \colon \sum_{i \in S} c_i \leq b\}$ describes her strategy set. The attacker, in turn, chooses when to start which task. Let a *schedule* $\boldsymbol{s} \coloneqq (s_i)_{i \in N}$ specify a starting time $s_i \geq 0$ for each task $i \in N$ and call it *feasible* if it respects the precedence constraints, then the set $\mathcal{F}$ of all such feasible schedules represents the attacker's strategy set. Observe that this definition of a feasible schedule allows tasks to be carried out simultaneously.

For a given schedule $\boldsymbol{s} \in \mathcal{F}$, discovery at the initiation of task $i \in N$ leads to an exposed time

$$\tau_i(\boldsymbol{s}) \coloneqq s_n + d_n - s_i. \tag{1}$$

Let $P_i(\boldsymbol{s})$ collect all tasks $j \in N \setminus \{i\}$ for which $s_j < s_i$, or with $s_j = s_i$ and $j < i$, then the *expected harm* $H(\boldsymbol{s}, S)$ when the attacker initiates his tasks according to schedule $\boldsymbol{s} \in \mathcal{F}$ and the defender monitors tasks $S \subseteq N$ equals

$$H(\boldsymbol{s}, S) = \sum_{i \in N} \left(\prod_{j \in P_i(\boldsymbol{s})} q_j^S \right) p_i^S h\left(\tau_i(\boldsymbol{s})\right) + \left(\prod_{i \in N} q_i^S \right) h\left(0\right). \tag{2}$$

Here, $p_i^S \coloneqq p_i^1$ if $i \in S$ and $p_i^S \coloneqq p_i^0$ if $i \notin S$, and similarly for the complementary probabilities. Term $i \in N$ captures the event that task i exposes the project, combining this event's probability with the resulting harm. The final term reflects the case where the project remains secret until completion.

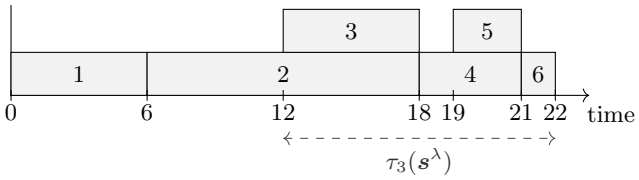

Figure 2: Gantt chart for late-start schedule $\boldsymbol{s}^\lambda$.

Example (Continued)**.** Suppose the attacker initiates each task as late as possible while attaining a completion time of 22 weeks. Figure 2 displays the Gantt chart for this late-start schedule $\boldsymbol{s}^\lambda$. If task 3 exposes the project, then the exposed time equals $\tau_3(\boldsymbol{s}^\lambda) = 22 - 12 = 10$ and the attack would cause $h(10) = 50 - (2 \times 10) = 30$ casualties. Task 3 exposes the project if the defender detects it and if the tasks in $P_3(\boldsymbol{s}^\lambda) = \{1, 2\}$ pass unnoticed; when the defender monitors tasks $S^\star = \{1, 3, 6\}$, this occurs with probability $q_1^1 q_2^0 p_3^1 = 0.0585$. Thus, term $i = 3$ in Equation (2) equals $0.0585 \times 30 = 1.755$. A similar computation for the other terms yields an expected number of $H(\boldsymbol{s}^\lambda, S^\star) = 30.1$ casualties due to the nerve agent attack.

We define a *secret project game* as the two-player zero-sum game $(\mathcal{F}, \mathcal{B}, H)$ between the attacker and defender with payoff function H, and strategy sets $\mathcal{F}$ and $\mathcal{B}$. The attacker chooses a schedule to maximize the expected harm, whereas the defender decides which activities to monitor in order to minimize it.

3.2 Dominant attacker strategy: late-start schedule

A *late-start schedule* $\boldsymbol{s}^\delta := (s_i^\delta)_{i \in N}$ initiates each task as late as possible while attaining a given project completion time $\delta \geq 0$; see e.g. [22]. We obtain its value recursively via $s_n^\delta = \delta - d_n$ and $s_i^\delta = \min\{s_j^\delta - d_i \colon (i,j) \in A\}$ for $i \in N \setminus \{n\}$. The completion time δ reflects the *planned attack time* and, to be feasible, it cannot be smaller than the project's *earliest possible completion time* λ.

Proposition 1 shows that no matter which tasks the defender monitors and no matter which planned attack time the attacker chooses, the resulting late-start schedule maximizes the expected harm. Thus, every feasible late-start schedule constitutes a dominant attacker strategy. The key to this result is that for a given task leading to project discovery, the exposed time is minimal under the late-start schedule (Lemma 1).

Lemma 1. *$\tau_i(\boldsymbol{s}^\delta) \leq \tau_i(\boldsymbol{s})$ for every $\boldsymbol{s} \in \mathcal{F}$, $i \in N$, and $\delta \geq \lambda$.*

Proof. Take arbitrary $\boldsymbol{s} \in \mathcal{F}$ and $\delta \geq \lambda$, and define a schedule $\hat{\boldsymbol{s}}$ by shifting $\boldsymbol{s}$ such that it attains δ as completion time: $\hat{s}_i := s_i + s_n^\delta - s_n$ for each $i \in N$. We obtain that, for each $i \in N$,

$$\tau_i(\boldsymbol{s}) = s_n + d_n - s_i = s_n^\delta + d_n - \hat{s}_i \geq s_n^\delta + d_n - s_i^\delta = \tau_i(\boldsymbol{s}^\delta),$$

where the inequality uses that, by definition of $\hat{\boldsymbol{s}}$ and the late-start schedule, $\hat{s}_i \leq s_i^\delta$. □

Proposition 1. *$H(\boldsymbol{s}^\delta, S) = \max_{\boldsymbol{s} \in \mathcal{F}} H(\boldsymbol{s}, S)$ for every $S \subseteq N$ and $\delta \geq \lambda$.*

Proof. Take arbitrary $\delta \geq \lambda$. It suffices to show that, for every $a \in \mathbb{R}$, the probability that the harm exceeds a is maximal under the late-start schedule. That is, it suffices to show that for every feasible schedule, the harm under the late-start schedule is 'stochastically larger' than the one under that schedule; see e.g. [36]. For $\boldsymbol{s} \in \mathcal{F}$ and $a \in \mathbb{R}$, let

$$N(a \mid \boldsymbol{s}) := \{i \in N \colon h(\tau_i(\boldsymbol{s})) \leq a\} \tag{3}$$

collect the activities whose discovery leads to a harm not larger than a under schedule $\boldsymbol{s}$. The harm exceeds a if and only if all tasks $i \in N(a \mid \boldsymbol{s})$ pass without discovery and, for $S \subseteq N$, this occurs with probability

$$\pi(a \mid \boldsymbol{s}, S) \coloneqq \prod_{i \in N(a|\boldsymbol{s})} q_i^S. \tag{4}$$

Take arbitrary $\boldsymbol{s} \in \mathcal{F}$, $S \subseteq N$, and $a \in \mathbb{R}$. By Equation (3), Lemma 1, and the fact that h is non-increasing in the exposed time, $N(a \mid \boldsymbol{s}^\delta) \subseteq N(a \mid \boldsymbol{s})$. Since $q_i^S \in [0,1]$ for all $i \in N$, Equation (4) then implies that $\pi(a \mid \boldsymbol{s}^\delta, S) \geq \pi(a \mid \boldsymbol{s}, S)$. Thus, for each $a \in \mathbb{R}$, the harm exceeds a with maximal probability under the late-start schedule. □

Proposition 1 holds for every function h that is non-increasing in the exposed time and regardless of the specific values for $(p_i^0, p_i^1, c_i)_{i \in N}$ or b. To determine an optimal strategy, the attacker thus only needs to know that the project network is given by $(N, A, (d_i)_{i \in N})$ and that his objective function is non-increasing in the exposed time. Moreover, since any choice for the planned attack time δ results in the same (maximum) expected harm, the defender can presume that the attacker follows the late-start schedule $\boldsymbol{s}^\lambda$ with a planned attack time equal to the earliest possible completion time λ. In the remainder of this paper, we also assume that the planned attack time equals λ when referring to the late-start schedule.

3.3 Which tasks should the defender monitor?

Given that the attacker follows late-start schedule $\boldsymbol{s}^\lambda$, the defender needs to solve the following optimization problem in order to determine which tasks she should monitor:

$$\min_{S \in \mathcal{B}} H(\boldsymbol{s}^\lambda, S). \tag{Π}$$

Below, we describe a dynamic program that solves this 'Optimization Problem Π' in pseudo-polynomial time, show that the problem is NP-hard, and explain how it yields a Nash equilibrium.

To simplify notation, index the tasks in N such that $s_1^\lambda \leq \ldots \leq s_n^\lambda$; task i is then the i^{th} one to be initiated in the late-start schedule. Moreover, let $h_i \coloneqq h(\tau_i(\boldsymbol{s}^\lambda))$ denote the harm in case discovery occurs at the initiation of task $i \in N$ and write $H(S) \coloneqq H(\boldsymbol{s}^\lambda, S)$ for every $S \subseteq N$.

As the dynamic program's *value function*, define $v_i(r)$ as the minimum expected harm at the start of task $i \in N$ with remaining budget $r \in \{0, \ldots, b\}$; we refer to i and r as the *stage* and *state* respectively. Let $\mathcal{A}_i(r) \coloneqq \{a \in \{0,1\} \colon c_i a \leq r\}$ collect the possible *actions* in stage i and state r, where $a = 1$ indicates that the defender monitors task i. After taking action $a \in \mathcal{A}_i(r)$ in stage i and state r, the defender discovers the project with probability p_i^a, which would then result in harm h_i. With complementary probability q_i^a, she proceeds to the next task's starting time, i.e. to stage $i+1$, with remaining budget $r - c_i a$. We obtain that $v_i(r)$ satisfies the optimality equation

$$v_i(r) = \min_{a \in \mathcal{A}_i(r)} \{p_i^a h_i + q_i^a v_{i+1}(r - c_i a)\} \tag{5}$$

with boundary condition $v_{n+1}(r) \coloneqq h(0)$ for all $r \in \{0, \ldots, b\}$. Since, by definition of $v_i(r)$,

$$v_1(b) = \min_{S \in \mathcal{B}} H(S),$$

Equation (5) defines a dynamic program that solves Optimization Problem Π.

$r\backslash i$	1	2	3	4	5	6
0	43.45	45.43	48.47	49.44	49.92	49.98
1	41.97	43.87	46.74	47.62	47.98	48.02*
2	41.85	43.74	46.60	47.47	47.82*	48.02*
3	31.63*	43.07	45.86*	46.78*	47.82*	48.02*
4	30.61*	36.28*	45.72*	46.66*	47.82*	48.02*
5	30.53*	35.24*	45.11*	46.66*	47.82*	48.02*
6	30.10*	35.16*	44.99*	46.66*	47.82*	48.02*

Table 2: Minimum expected harm $v_i(r)$ for each stage i and state r. An asterisk (*) indicates that monitoring task i is optimal in stage i and state r .

Example (continued)**.** Table 2 shows the result of the dynamic program with boundary condition $h(0) = 50$. The minimum expected number of casualties equals 30.10 and it is optimal to monitor tasks 1, 3, and 6. Since the expected number of casualties without monitoring equals 43.45, the defender expects to save 13.35 lives thanks to her monitoring.

With n stages and $b + 1$ states, the dynamic program runs in time $O(nb)$, which is pseudo-polynomial because of parameter b. Below, we show that Optimization Problem Π is NP-hard, which implies that no polynomial-time algorithm exists unless P = NP; see e.g. [14]. The proof uses a reduction from the NP-complete Partition problem [14] by exploiting the budget constraint in the defender's strategy set $\mathcal{B}$. Before proving the result, we state the following lemma, which can be shown by induction on r (see Appendix A).

Lemma 2. *Let $r \in \mathbb{N}$ and $a_1, \ldots, a_r \in \mathbb{R}$, then*

$$\prod_{i=1}^{r}(1 - a_i) = 1 - \sum_{i=1}^{r} a_i + \sum_{i=1}^{r} \sum_{j=i+1}^{r} a_i a_j \prod_{k=j+1}^{r} (1 - a_k).$$

Proposition 2. *Optimization Problem Π is NP-hard.*

Proof. Consider the Partition problem, where given a finite set U and a positive integer u_i for every $i \in U$ the question is whether there exists a set $S^\star \subseteq U$ with $\sum_{i\in S^\star} u_i = \sum_{i\in U\setminus S^\star} u_i$. For an arbitrary Partition instance $(U, (u_i)_{i\in U})$, we construct a secret project situation $\big(N, A, (d_i, p_i^0, p_i^1, c_i)_{i\in N}, b, h\big)$ as follows. Take $N = U$; choose precedence constraints A such that $(i, i + 1) \in A$ for every $i \in N \setminus \{n\}$; and let $(d_i, p_i^0, p_i^1, c_i) = (1, 0, u_i/M, u_i)$ for all $i \in N$, where $M := (n\bar{u})^2$ and $\bar{u} := \max_{i\in U} u_i$. Next, take $b = \lceil(\sum_{i\in U} u_i)/2\rceil$, and let $h(t) = 0$ for every $t > 0$ and $h(0) = 1$. All terms but the last one in Equation (2) then become zero and, for $S \subseteq N$,

$$H(S) = \prod_{i\in N} q_i^S = \prod_{i\in S} \left(1 - \frac{u_i}{M}\right). \tag{6}$$

Below, we show that the answer to the Partition instance is 'yes' if and only if

$$\min_{S\in\mathcal{B}} H(S) < 1 - \frac{b}{M} + \frac{1}{M}. \tag{7}$$

Since Partition is NP-complete, this reduction would prove that Optimization Problem Π is NP-hard.

Let $S \subseteq N$, expanding Equation (6) by using $a_i \coloneqq u_i/M$ in Lemma 2 then yields that

$$H(S) = 1 - \frac{1}{M}\left(\sum_{i \in S} u_i\right) + \frac{1}{M} Q(S), \tag{8}$$

where

$$Q(S) \coloneqq \sum_{i \in S} \sum_{j \in \sigma(i,S)} \frac{u_i u_j}{M} \prod_{k \in \sigma(j,S)} \left(1 - \frac{u_k}{M}\right) \tag{9}$$

and $\sigma(i, S) \coloneqq S \cap \{i+1, \ldots, n\}$ for each $i \in N$. Our choice for M guarantees that $0 \leq Q(S) < 1$ since for every $i \in S$, $j \in \sigma(i, S)$, and $k \in \sigma(j, S)$ it holds that

$$0 \leq \frac{u_i u_j}{M} \leq \frac{1}{n^2}, \qquad 0 \leq \left(1 - \frac{u_k}{M}\right) \leq 1,$$

and $Q(S)$ contains $|S|(|S|-1)/2 < n^2$ terms.

If the answer to the PARTITION instance is 'yes', then there exists an $S^\star \subseteq N$ for which $\sum_{i \in S^\star} u_i = b$. Since $S^\star \in \mathcal{B}$, Equation (8) yields

$$\min_{S \in \mathcal{B}} H(S) \leq H(S^\star) = 1 - \frac{b}{M} + \frac{1}{M} Q(S^\star) < 1 - \frac{b}{M} + \frac{1}{M}. \tag{10}$$

Conversely, if the answer to the PARTITION instance is 'no', then $\sum_{i \in S} u_i < b$ for each $S \in \mathcal{B}$. Since $b, u_1, \ldots, u_n$ are all integer, this implies $\sum_{i \in S} u_i \leq b - 1$ and, together with Equation (8), we obtain that for every $S \in \mathcal{B}$ it holds that

$$H(S) \geq 1 - \frac{b-1}{M} + \frac{1}{M} Q(S) \geq 1 - \frac{b}{M} + \frac{1}{M}. \tag{11}$$

Thus, the answer to the PARTITION instance is 'yes' if and only if Inequality (7) holds. □

For each $S^\star \in \mathcal{B}$ attaining the minimum in Optimization Problem Π, we have that $(\boldsymbol{s}^\lambda, S^\star)$ describes a Nash equilibrium: given that the attacker follows the late-start schedule, it is optimal for the defender to monitor the tasks in $S^\star$ and, vice versa, given that the defender monitors the tasks in $S^\star$, it is optimal for the attacker to follow the late-start schedule. In spite of the problem's NP-hardness, the dynamic program can easily solve realistically-sized instances (see Section 5) and thus provides an effective method for identifying a Nash equilibrium for a secret project game. This equilibrium is not unique in general, but since we are dealing with a zero-sum game, all Nash equilibria result in the same expected harm; see e.g. [30].

4 The value of monitoring a task

In most practical settings, the defender not only wants to know which tasks she should monitor, she also needs to explain *why* some tasks are included and others are not. For example, intelligence agencies must be able to justify their expenses to the government. Moreover, understanding what exactly renders a task more desirable to monitor is key to identifying avenues for further harm reduction. In this section, we use cooperative game theory to evaluate the performance of each task's intelligence effort and to explain why performance may differ.

4.1 Monitoring games

A *cooperative game* (N, v) is defined by a set N of *players* and a *characteristic function* $v\colon 2^N \to \mathbb{R}$ that assigns to each *coalition* $S \subseteq N$ a real number $v(S)$, with $v(\emptyset) = 0$. Here, $v(S)$ reflects the *payoff* that coalition S can attain and is also called the *worth* of S. For a detailed introduction to cooperative games, see for example [30].

Given a secret project situation, let each activity $i \in N$ correspond to a player, then we say that a cooperative game (N, v) is a *monitoring game* if the worth $v(S)$ equals the decrease in expected harm when monitoring tasks $S \subseteq N$, compared to the situation without monitoring (i.e. when $S = \emptyset$). More formally, we require that

$$v(S) = H(\emptyset) - H(S) \tag{12}$$

for every $S \subseteq N$. Here, the expected harm $H(S) \coloneqq H(\boldsymbol{s}^\lambda, S)$ is as defined in Section 3 and the definition excludes budget and costs because these do not affect the expected harm.

One central theme in cooperative game theory is the question how to distribute the payoff of a coalition amongst its members in a way that reflects each player's contribution. Since the contribution of a player's cooperation in a monitoring game reflects the harm reduction thanks to monitoring the corresponding task, solution concepts from cooperative game theory measure this harm reduction and, as such, the performance of intelligence effort.

S:	$\emptyset$	$\{1\}$	$\{2\}$	$\{1,2\}$	$\{1,3,6\}$	N
$H(S)$:	43.5	31.6	34.8	25.7	30.1	24.3
$v(S)$:	0.0	11.8	8.7	17.8	13.4	19.1

Table 3: Expected harm and worth when monitoring different sets of tasks.

Example (continued)**.** Table 3 shows the expected harm $H(S)$ and the worth $v(S)$ for six of the $2^6 = 64$ possible sets $S \subseteq N$ of tasks to monitor for the monitoring game based on Section 3's example secret project situation. The worth $v(\{1,2\})$ equals $43.45 - 25.68$ and reflects that monitoring tasks 1 and 2 decreases the expected number of casualties by 17.77 compared to the situation without monitoring. Since $v(\{1,2\}) \leq v(\{1\}) + v(\{2\})$, the harm reduction when monitoring both tasks 1 and 2 is less than the sum of the individual reductions; this is because of the possibility that both tasks 1 and 2 would lead to project discovery at their initiation.

Our interpretation of a player deviates from the traditional one in cooperative game theory: we treat each activity *as if* it were an independent player, whereas there is in fact only one player, i.e. the defender, who decides which tasks to monitor. This implies that some traditional game-theoretic concepts such as the core have no real interpretation in our setting. Nevertheless, as will become clear below, modeling the situation as a cooperative game does yield insight into what exactly makes a task more or less desirable to monitor.

4.2 What makes a task desirable to monitor?

For a monitoring game (N, v), the *marginal contribution* of task $i \in N$ given that the defender already monitors $S \subseteq N \setminus \{i\}$ equals

$$m_i(S) \coloneqq v(S \cup \{i\}) - v(S) = H(S) - H(S \cup \{i\}). \tag{13}$$

Thus, $m_i(S)$ captures the expected harm reduction when monitoring task i in addition to the tasks in S. Below, we derive an intuitive expression for the marginal contribution and show (i)

that the harm reduction thanks to monitoring a task decreases as more tasks are being monitored and (ii) that a task is more desirable to monitor if it has an earlier late-start time s_i^λ and a lower ratio of non-discovery probabilities q_i^1/q_i^0.

Let $\rho_k \coloneqq h_{k+1} - h_k$ capture the increase in harm when the defender fails to discover the project at the k^{th} task's initiation and has to wait until the start of task $k+1$, with $k \in \{1, \ldots, n-1\}$. Similarly, $\rho_n \coloneqq h(0) - h_n$ reflects the increase in harm when the defender fails to discover the project at task n's initiation and thus does not discover the project at all. Denoting $P(k) \coloneqq \{1, \ldots, k\}$ for every $k = 1, \ldots, n$ and using $p_i^S = 1 - q_i^S$, we obtain from Equation (2) that

$$H(S) = h_1 + \sum_{k=1}^{n} \left(\prod_{j \in P(k)} q_j^S \right) \rho_k \tag{14}$$

for every $S \subseteq N$. Substituting this into Equation (13) yields, for $i \in N$ and $S \subseteq N \setminus \{i\}$,

$$m_i(S) = \left(q_i^0 - q_i^1\right) \sum_{k=i}^{n} \left(\prod_{j \in P(k) \setminus \{i\}} q_j^S \right) \rho_k. \tag{15}$$

Since $q_i^0 - q_i^1 = p_i^1 - p_i^0$, we can thus interpret this expression for $m_i(S)$ as the increase in task i's discovery probability thanks to monitoring the task times the expected harm occurring after task i's initiation given that i passes undiscovered and the tasks in S are being monitored.

Proposition 3 below shows that the harm reduction thanks to monitoring a task decreases as more tasks are being monitored. Formally, a monitoring game (N, v) is *concave*, see e.g. [30], if $m_i(S) \geq m_i(T)$ for each $i \in N$ and $S \subseteq T \subseteq N \setminus \{i\}$. One managerial implication of this concavity is that if the defender already monitors many tasks, increasing the budget to decrease the harm even further is relatively ineffective.

Proposition 3. *Monitoring games are concave.*

Proof. Let (N, v) be a monitoring game and take arbitrary $i \in N$ and $S \subseteq T \subseteq N \setminus \{i\}$. Since $S \subseteq T$ and $q_j^0 \geq q_j^1$ for all $j \in N$, we obtain that, for each $k = i, \ldots, n$,

$$\prod_{j \in P(k) \setminus \{i\}} q_j^S \geq \prod_{j \in P(k) \setminus \{i\}} q_j^T.$$

Equation (15) and the non-negativity of $q_i^0 - q_i^1$ and ρ_k for all $k = i, \ldots, n$ then yield the result. □

We now show that a task is more desirable to monitor if it has an earlier late-start time s_i^λ and a lower ratio of non-discovery probabilities q_i^1/q_i^0. Let $i, j \in N$ be two tasks with $s_i^\lambda \leq s_j^\lambda$ and suppose the defender is doubting whether to monitor i or j given that she already monitors $S \subseteq N \setminus \{i, j\}$. That is, she wants to know whether $m_i(S) \geq m_j(S)$. Proposition 4 below will show how to decompose the difference in marginal contribution into two quantities $B_{ij}(S)$ and $Q_{ij}(S)$, where the former quantifies the benefit of i's earlier starting time and the latter the difference because of task i and j's possibly different discovery probabilities. Before stating the result, we first elaborate on the interpretation of these two quantities.

Task i is more attractive because of its earlier starting time: the reduction in expected harm that occurs between the initiation of task i and j when monitoring i instead of j equals

$$B_{ij}(S) \coloneqq \left(q_i^0 - q_i^1\right) \sum_{k=i}^{j-1} \left(\prod_{l \in P(k) \setminus \{i\}} q_l^S \right) \rho_k. \tag{16}$$

Here, $q_i^0 - q_i^1$ equals the increase in discovery probability when monitoring task i and the expression's second part reflects the expected harm occurring between the initiation of tasks i and j. Since the tasks' discovery probabilities may differ, monitoring task j could still be preferable: the reduction in expected harm that occurs after task j's initiation when monitoring i instead of j equals

$$Q_{ij}(S) \coloneqq \left(q_i^0 q_j^1 - q_i^1 q_j^0\right) \sum_{k=j}^{n} \left(\prod_{l \in P(k) \setminus \{i,j\}} q_l^S \right) \rho_k. \tag{17}$$

Here, $(q_i^0 q_j^1 - q_i^1 q_j^0)$ equals the difference in discovery probability when monitoring i instead of j and the expression's second part represents the expected harm occurring from task j's initiation until project completion. The next result shows that a comparison of Expressions (16) and (17) determines which task's monitoring produces the highest harm reduction.

Proposition 4. *Let (N, v) be a monitoring game, then*

$$m_i(S) - m_j(S) = B_{ij}(S) + Q_{ij}(S)$$

for each $i, j \in N$ with $s_i^\lambda \leq s_j^\lambda$ and $S \subseteq N \setminus \{i, j\}$.

Proof. Take arbitrary $i, j \in N$ with $s_i^\lambda \leq s_j^\lambda$ and $S \subseteq N \setminus \{i, j\}$. Since $i, j \notin S$, $s_i^\lambda \leq s_j^\lambda$, and

$$(q_i^0 - q_i^1)q_j^0 - (q_j^0 - q_j^1)q_i^0 = q_i^0 q_j^1 - q_i^1 q_j^0,$$

the result follows from computing $m_i(S) - m_j(S)$ by means of Equation (15). □

Thus, Proposition 4 reveals two key drivers that influence the desirability of monitoring a task $i \in N$: the starting time s_i^λ and the ratio q_i^1/q_i^0. The lower s_i^λ and q_i^1/q_i^0, the more attractive monitoring i becomes; the next corollary summarizes this finding.

Corollary 1. *Let (N, v) be a monitoring game and $i, j \in N$ two tasks with q_i^0 and q_j^0 non-zero, then $m_i(S) \geq m_j(S)$ for every $S \subseteq N \setminus \{i, j\}$ if $s_i^\lambda \leq s_j^\lambda$ and $q_i^1/q_i^0 \leq q_j^1/q_j^0$.*

Proof. Take arbitrary $S \subseteq N \setminus \{i, j\}$. Since $B_{ij}(S)$ is always non-negative and $Q_{ij}(S)$ is non-negative if $q_i^0 q_j^1 - q_i^1 q_j^0 \geq 0$, the result follows from Proposition 4. □

Example (continued)**.** From Table 1 and Figure 2, we find that monitoring task 3 always leads to a higher harm reduction than task 5 since $s_3^\lambda = 12 \leq 19 = s_5^\lambda$ and $q_3^1/q_3^0 = 0.95 \leq 0.96 = q_5^1/q_5^0$. For tasks 3 and 4, in turn, the two key drivers conflict since $s_4^\lambda = 18 > s_3^\lambda$ and $q_4^1/q_4^0 = 0.85 < q_3^1/q_3^0$. In the next section, we propose the Banzhaf value as a measure to indicate whether, on average, monitoring task 3 leads to a higher harm reduction than monitoring 4 does.

4.3 The Banzhaf value of monitoring games

The *Banzhaf value* for a player $i \in N$ in a cooperative game (N, v) is defined as [29, 7]

$$\Psi_i(v) \coloneqq \frac{1}{2^{n-1}} \sum_{S \subseteq N \setminus \{i\}} m_i(S). \tag{18}$$

For a monitoring game, it captures the average decrease in expected harm when monitoring task $i \in N$, averaged over all possible $S \subseteq N \setminus \{i\}$. Originally proposed to measure a player's power in a voting game, the Banzhaf value has become one of the most popular methods for measuring players' relative importance [25]. Although the problem of computing the Banzhaf value is #P complete in general [34], the following proposition gives an intuitive and polynomial-time-computable expression for the Banzhaf value of a task in a monitoring game.

Proposition 5. *Let (N, v) be a monitoring game, then*

$$\Psi_i(v) = (q_i^0 - q_i^1) \sum_{k=i}^{n} \left(\prod_{j \in P(k) \setminus \{i\}} \frac{q_j^0 + q_j^1}{2} \right) \rho_k \tag{19}$$

for every task $i \in N$.

Proof. Let $i \in N$ be a task in the monitoring game (N, v). By Equations (15) and (18),

$$\Psi_i(v) = \frac{q_i^0 - q_i^1}{2^{n-1}} \sum_{k=i}^{n} \left(\sum_{S \subseteq N \setminus \{i\}} \prod_{j \in P(k) \setminus \{i\}} q_j^S \right) \rho_k. \tag{20}$$

It can be verified that for each $T \subseteq N$:

$$\sum_{S \subseteq T} \prod_{j \in T} q_j^S = \prod_{j \in T} (q_j^0 + q_j^1). \tag{21}$$

Now take arbitrary $k \in \{i, \ldots, n\}$ and observe that each of the 2^{n-k} choices of $S \subseteq N \setminus \{i\}$ that have the same intersection with $P(k) \setminus \{i\}$ lead to the same value for the quantity

$$\prod_{j \in P(k) \setminus \{i\}} q_j^S.$$

Substituting $P(k) \setminus \{i\}$ for T in Equation (22) then yields

$$\begin{aligned} \sum_{S \subseteq N \setminus \{i\}} \prod_{j \in P(k) \setminus \{i\}} q_j^S &= 2^{n-k} \sum_{S \subseteq P(k) \setminus \{i\}} \prod_{j \in P(k) \setminus \{i\}} q_j^S \\ &= 2^{n-k} \prod_{j \in P(k) \setminus \{i\}} \left(q_j^0 + q_j^1 \right). \end{aligned}$$

In combination with Equation (20), this proves the result. □

i	Description	Banzhaf	Optimal
1	Assemble team	10.04	Yes
2	Produce nerve agent	6.87	-
3	Develop delivery method	0.54	Yes
4	Prepare equipment	0.53	-
5	Select target	0.10	-
6	Attack target	0.92	Yes

Table 4: Banzhaf value as a measure for the decrease in expected harm thanks to monitoring a task.

Example (continued)**.** Table 4 shows that within the optimal set $S^\star = \{1, 3, 6\}$, task 1 is by far the most important one: its monitoring leads to 10.04 fewer casualties on average, whereas for tasks 3 and 6 this is only 0.54 and 0.92 respectively. We observe that tasks 3 and 4 have equal costs (see Table 1) and a very similar Banzhaf value. Hence, if monitoring task 3 suddenly

becomes impossible because, for instance, a crucial informant withdraws from the operation, then the defender knows that task 4 provides a reasonable substitute. The high Banzhaf value of task 2, in turn, signals that if the defender could decrease task 2's monitoring cost or increase the budget, this would reduce the harm considerably. Indeed, one could verify that for $c_2 = 3$ or $b = 7$, the optimal set of tasks to monitor becomes $S^\star = \{1, 2\}$ and the expected harm decreases from 30.10 to 25.68 casualties.

Substituting $(q_j^0 + q_j^1)/2$ for q_j^S in Equation (15) yields Equation (19). This leads to the following intuitive interpretation of the Banzhaf value in a monitoring game: $\Psi_i(v)$ gives the decrease in the expected harm when monitoring task i while all other tasks $j \in N \setminus \{i\}$ are at their average discovery probability. Let $i, j \in N$ be two tasks with $s_i^\lambda \leq s_j^\lambda$, then, in analogy to Equations (16) and (17), the quantity

$$B_{ij} \coloneqq (q_i^0 - q_i^1) \sum_{k=i}^{j-1} \left(\prod_{l \in P(k) \setminus \{i\}} \frac{q_l^0 + q_l^1}{2} \right) \rho_k$$

captures the benefit of i's earlier starting time and

$$Q_{ij} \coloneqq \left(q_i^0 q_j^1 - q_i^1 q_j^0\right) \sum_{k=j}^{n} \left(\prod_{l \in P(k) \setminus \{i,j\}} \frac{q_l^0 + q_l^1}{2} \right) \rho_k$$

reflects the difference because of task i and j their possibly different discovery probabilities. An analogous argument as the one leading to Proposition 4 and Corollary 1 then yields:

Proposition 6. *Let (N, v) be a monitoring game, then for every two tasks $i, j \in N$ with $i < j$:*

$$\Psi_i(v) - \Psi_j(v) = B_{ij} + Q_{ij}.$$

Thus, for q_i^0 and q_j^0 non-zero, $\Psi_i(v) \geq \Psi_j(v)$ if $s_i^\lambda \leq s_j^\lambda$ and $q_i^1/q_i^0 \leq q_j^1/q_j^0$.

Example (continued)**.** Monitoring task 1 instead of 6 decreases the expected harm with $B_{1,6} = 9.86$ because of task 1's earlier starting time and with $Q_{1,6} = -0.74$ because of task 6's lower ratio q_6^1/q_6^0. This distinction reveals that the attractiveness of monitoring task 1 originates from its earlier starting time and the relatively small effect of the difference in discovery probabilities.

In sum, our model for evaluating the performance of a task's intelligence effort leads to multiple insights. Firstly, the Banzhaf value indicates which activities within the optimal set of tasks to monitor are relatively more important. This not only supports all involved parties in justifying their share in the budget, but it also indicates which tasks the defender should pay most attention to. Secondly, the defender can use Proposition 6 to explain why some tasks are relatively more important than others by trading off (i) the starting time and (ii) the relative influence of monitoring on the discovery probability. Thirdly, analyzing tasks excluded from the optimal set of tasks to monitor indicates promising avenues for further harm reduction. An excluded task with a relatively high Banzhaf value, for instance, suggests that its monitoring is currently too expensive and that the expected harm could decrease considerably if the defender found a way to increase the budget or to decrease the task's monitoring cost.

Although in this paper we focus on the Banzhaf value, multiple other allocation indices exist in the rich literature on cooperative games [30]. We have also conducted an analysis of other semivalues [12] and, in particular, the Shapley value [37]. This led to results very similar to those for the Banzhaf value, and since the latter allows for an intuitive expression, we do not include the generalization towards semivalues in this text.

5 Numerical example: developing a first nuclear weapon

Under the Comprehensive Nuclear-Test-Ban Treaty, considerable attention goes to the monitoring of nuclear tests [41]. While these tests are relatively easy to detect, Kemp [23] rightfully points out that "for many security objectives, detecting a nuclear detonation comes too late" and that "[m]ore useful would be an ability to detect nuclear weapon programs well before the achievement of a nuclear device." In this section, we use the nuclear weapons development project of Harney et al. [18] to demonstrate how our methods support defending countries in trading off time and ease of detection when deciding which activities to monitor.

5.1 Project description

Harney et al. [18] describe the tasks necessary to produce "a first small batch of nuclear weapons" together with their durations and precedence constraints. The project – which has also served as an example in [9], [32], and [17] – consists of 124 non-dummy tasks that can be grouped into five main parts as listed in Table 5. The table's third and fourth column give each part's start and end time (in weeks) under the late-start schedule with a deadline equal to the project's minimum makespan of 468 weeks. For a detailed project description, see Appendix B and [18].

Part	Description	Start	End
1	Diversion of yellowcake	116	332
2	Production of uranium hexafluoride from yellowcake	36	332
3	Uranium enrichment	0	356
4	Conversion of highly enriched uranium hexafluoride	200	356
5	Design and construction of the actual weapons	208	468

Table 5: Main parts of the example nuclear weapons development project provided by Harney et al. [18] with their late-start and end times (in weeks).

Since Harney et al. [18] do not specify discovery probabilities or a harm function, we assign values to these parameters ourselves. Given our limited expertise in the domain of nuclear weapons, these values are only illustrative; our main purpose is to show how someone who does possess authentic data can apply our analysis.

We interpret the harm as the probability that the hostile state succeeds in obtaining a nuclear weapon. The higher the exposed time, the more time the defender has to take counteractions, and the lower this probability becomes. Figure 3 displays the three different harm functions that we consider, where the concave harm function captures the situation where it is crucial to discover the project early on (high exposed time), whereas for a convex harm function the harm only becomes significant when the exposed time is relatively small.

To assign the discovery probabilities, we classify each task into one of the three classes listed in Table 6. We use the same distinction between innocent and suspicious tasks as in [32], and we additionally subdivide the suspicious tasks in the first two classes randomly; see Appendix B for details. Figure 4 displays the percentage of tasks initiated in each class as a function of time under the late-start schedule. Most innocent tasks occur relatively early in the project, whereas the suspicious tasks accumulate during the later parts of the project. Thus, the defender faces a trade-off between time and ease of detection.

We consider three different values $b \in \{10, 15, 20\}$ for the budget and assume $c_i = 1$ for each $i \in N$. Under these assumptions, the defender monitors those b tasks that, combined, produce the most harm reduction.

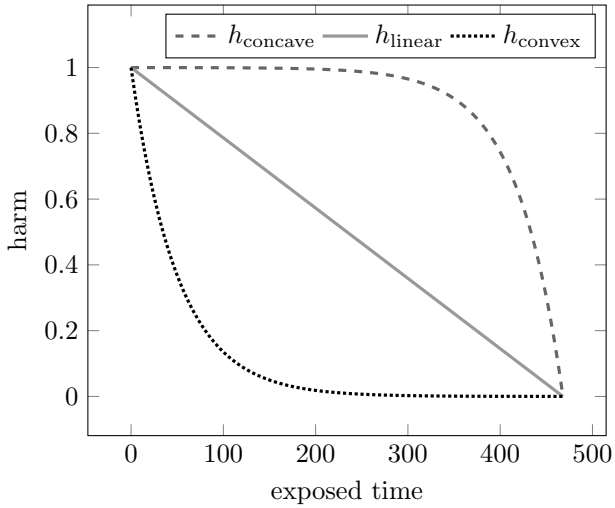

Figure 3: Concave, linear, and convex functional form for the harm function.

Class	Description	p_i^0	p_i^1
1	Suspicious; effective monitoring	0.05	0.25
2	Suspicious; ineffective monitoring	0.05	0.10
3	Innocent	0.00	0.01

Table 6: Description of the classes and the associated discovery probabilities.

All experiments were performed with an Intel Core i7-4790 processor with 3.60 GHz CPU speed and solving an instance took 0.006 seconds on average. This demonstrates that our methods can indeed easily handle realistically-sized instances. We have also performed a sensitivity analysis to examine the effect of changing the discovery probabilities and found that the optimal set of tasks to monitor is relatively insensitive to small perturbations in the discovery probabilities, but that it is important to classify the different activities correctly. We refer to Appendix C for details.

5.2 Analysis of the optimal tasks to monitor

For a linear harm function and a budget $b = 10$, the defender should monitor the ten tasks displayed in Table 7. The vast majority of the tasks are suspicious (class 1-2), but also two innocent tasks (class 3) appear because they occur early in the project. The resulting probability that the hostile state succeeds in obtaining a nuclear weapon equals 40.64%, whereas additional computations show that this expected harm would be 45.04% if the defender monitored the ten earliest-occurring tasks from class 1. This demonstrates that taking into account the timing of an eventual detection can reduce the expected harm significantly. Thus, our quantitative methods corroborate the critique on the Comprehensive Nuclear-Test-Ban Treaty for its focus on the monitoring of – easily detectable, but late occurring – nuclear tests [41].

The Banzhaf value (Table 7, final column) sheds light on which activities within the optimal set of tasks to monitor are most important. Given our interpretation of the harm as the nuclear weapons project's success rate, task 25's Banzhaf value of 0.0235 expresses that additional effort

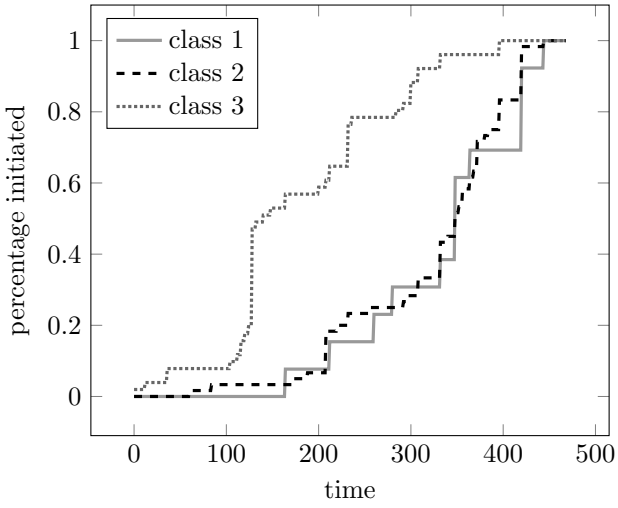

Figure 4: Percentage of tasks initiated in each class as a function of time under late-start schedule.

ID	Description	Class	s_i^λ	Banzhaf
25	Assemble and integrate FP	1	164	0.0235
82	Assemble research devices	2	60	0.0153
83	Test and evaluate research devices	2	84	0.0127
26	Operate FP	1	212	0.0103
110	Integrate components	1	260	0.0050
92	Assemble production devices	2	172	0.0048
78	Design basic AE enrichment device	3	0	0.0041
80	Vortex unit	3	12	0.0039
93	Integrate enrichment cascade	2	188	0.0037
125	Acquire AP site	1	280	0.0034

Table 7: Optimal tasks to monitor for linear harm function and $b = 10$.

for detecting the assembly and integration of a fluoridation plant (FP) reduces the nuclear weapons project's success rate by 2.35 percentage points on average. Thus, task 25 is much more important than task 125, whose monitoring decreases the success rate by only 0.34 percentage points on average.

Since the ten tasks that are optimal to monitor also have the highest Banzhaf value amongst all activities, we conclude that the Banzhaf value measures the harm reduction thanks to monitoring a task quite well. Further numerical experimentation in Appendix C indicates that, while not always exact, the optimal set of tasks to monitor is highly similar to the b activities with the highest Banzhaf value.

The harm function's shape considerably affects the choice of tasks to monitor (Table 8). With a concave harm function, the defender needs to discover the project early on and should focus on the project's earliest-initiated tasks, even if these are mainly innocent: with $b = 10$, the defender monitors six innocent tasks (class 3) and additional budget primarily serves to monitor more innocent tasks. With a convex harm function, in turn, the defender can wait for the initiation of suspicious tasks because the major part of the harm occurs later on in the project: class 1

	Concave			Linear			Convex		
Class	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$
1	2	2	2	4	4	4	8	9	9
2	2	3	3	4	7	11	2	6	11
3	6	10	15	2	4	5	0	0	0

Table 8: Number of tasks included in the optimal set to monitor for each class.

contains eight out of the ten monitored tasks for $b = 10$ and the defender uses additional budget exclusively for monitoring suspicious tasks. With a linear harm function, finally, the trade-off between discovery time and probability is most outspoken and the second class dominates.

h	$b = 10$	$b = 15$	$b = 20$
Concave	0.9126	0.9097	0.9076
Linear	0.4064	0.3924	0.3824
Convex	0.0184	0.0148	0.0121

Table 9: Minimum expected harm.

As Table 9 reveals, the nuclear weapons project's minimum success rate highly depends on the harm function's shape and less so on the budget: with $b = 10$ the success rate is 91.26% for the concave and 1.84% for the convex case, while with $b = 20$, these rates decrease only slightly to 90.76% and 1.21% respectively. Since $h_{\text{concave}}(t) \geq h_{\text{linear}}(t) \geq h_{\text{convex}}(t)$ for each t (see Figure 3), it is not surprising that the same relation holds for the minimum expected harm. Proposition 3, in turn, explains the modest influence of additional budget: the marginal harm reduction diminishes when monitoring more tasks.

6 Discussion

In this paper, we have introduced a new problem setting in which a defender wants to discover whether some attacker is conducting a covert project that would harm the defender. Although reality is too complex to apply our results without modification, our model does capture the key dynamics of the defender's problem setting while being sufficiently tractable to yield insights. As such, our findings provide a quantitative basis that, we hope, will spur further discussions and will lead to better-informed decision making.

We have formalized a novel zero-sum game – the secret project game – and derived a Nash-equilibrium for this game. Despite the problem's NP-hardness, our pseudo-polynomial-time dynamic program can handle realistically-sized instances, and thus it effectively supports the defender in determining how to allocate her limited intelligence resources. Through an innovative use of cooperative game theory, we were able to evaluate the performance of different tasks' intelligence effort and to understand why this performance may differ. In particular, we have introduced and analyzed a cooperative game – the monitoring game – and derived an intuitive and polynomial-time-computable expression for its Banzhaf value as a measure for the harm reduction thanks to monitoring a task.

Our main insights are that (i) the late-start schedule is a dominant attacker strategy, (ii) the marginal contribution of monitoring a task decreases as more tasks are being monitored, and

(iii) a task is more desirable to monitor if it has an earlier late-start time s_i^λ and a lower ratio of non-discovery probabilities q_i^1/q_i^0. Thus, when estimating the progress [16] or duration [20] of a terrorist plot, intelligence services should take into account terrorists' tendency to start activities as late as possible. Additionally, if the defender already monitors many tasks, increasing the budget is relatively ineffective; instead, the defender could employ the budget to improve the response after eventual discovery [9]. The third finding, finally, advocates for not only focusing on those activities that are easiest to detect, but to take into account also the timing of eventual detection. This contrasts with recent efforts in both counter-terrorism [24] and the non-proliferation of nuclear weapons [23], where considerable attention goes to monitoring tasks that are relatively easy to detect but occur late in the project.

This paper is only the first to consider which activities to monitor in order to expose a competitor's project, and still many dimensions of the problem remain unexplored. For example, it would be interesting to enrich the attacker's strategy set by allowing for deception and crashing [32]. Other extensions include allowing for uncertain activity durations, incorporating resource constraints for the attacker, or considering a whole portfolio of projects instead of only a single one. An additional dimension in this latter case would be to superimpose a social network to exploit the relation between individuals that are involved in multiple projects [2].

We have assumed that monitoring pertains to structural measures that take place continuously during the entire planning horizon. This implies that the defender does not need to make any assumption with respect to the start time of the attacker's project. One direction for further research is to include the decision of *when* to monitor an activity, in which case the literature on inspection games could provide a starting point.

Another potential for further research lies in considering a more general discovery process. Building on the model of Pinker et al. [32], one could assign a detection weight to each task and assume that discovery occurs if the total weight of detected tasks exceeds a given threshold. In line with the article of Atkinson et al. [3], an important question then arises: how much evidence should the defender collect before engaging? An additional concern is how to deal with false-positive detections. More generally, detecting a task may increase the general level of alertness, which could then affect other tasks' discovery probabilities [31, 6], and another issue becomes how to update the discovery probabilities. We believe that there is considerable research potential in trying to incorporate such considerations.

References

[1] S. Alpern and T. Lidbetter. Mining coal or finding terrorists: The expanding search paradigm. *Operations Research*, 61(2):265–279, 2013.

[2] M. P. Atkinson and L. M. Wein. An overlapping networks approach to resource allocation for domestic counterterrorism. *Studies in Conflict & Terrorism*, 33(7):618–651, 2010.

[3] M. P. Atkinson, M. Kress, and R.-J. Lange. When is information sufficient for action? Search with unreliable yet informative intelligence. *Operations Research*, 64(2):315–328, 2016.

[4] R. Avenhaus and M. J. Canty. Playing for time: A sequential inspection game. *European Journal of Operational Research*, 167(2):475–492, 2005.

[5] R. Avenhaus, B. von Stengel, and S. Zamir. Inspection games. In R. Aumann and S. Hart, editors, *Handbook of Game Theory with Economic Applications*, volume 3, pages 1947–1987. Elsevier, 2002.

[6] N. Bakshi and E. Pinker. Public warnings in counterterrorism operations: Managing the "cry-wolf" effect when facing a strategic adversary. *Operations Research*, 66(4):977–993, 2018.

[7] J. F. Banzhaf. Weighted voting doesn't work: A mathematical analysis. *Rutgers Law Review*, 19:317–343, 1964.

[8] G. G. Brown, W. M. Carlyle, J. O. Royset, and R. K. Wood. On the complexity of delaying an adversary's project. In *The Next Wave in Computing, Optimization, and Decision Technologies*, volume 29 of *Operations Research/Computer Science Interfaces Series*, pages 3–17. Springer US, 2005.

[9] G. G. Brown, W. M. Carlyle, R. C. Harney, E. M. Skroch, and R. K. Wood. Interdicting a nuclear-weapons project. *Operations Research*, 57(4):866–877, 2009.

[10] Y.-N. Chiu, B. Leclerc, and M. Townsley. Crime script analysis of drug manufacturing in clandestine laboratories: Implications for prevention. *The British Journal of Criminology*, 51(2):355–374, 2011.

[11] N. B. Dimitrov, M. Kress, and Y. Nevo. Finding the needles in the haystack: Efficient intelligence processing. *Journal of the Operational Research Society*, 67(6):801–812, 2016.

[12] P. Dubey, A. Neyman, and R. J. Weber. Value theory without efficiency. *Mathematics of Operations Research*, 6(1):122–128, 1981.

[13] S. E. Elmaghraby. *Activity Networks: Project Planning and Control by Network Models*. John Wiley & Sons, 1977.

[14] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.

[15] G. A. Godfrey and T. L. Mifflin. Likelihood-based optimization of threat operation timeline estimation. In *12th International Conference on Information Fusion*, pages 948–953, July 2009.

[16] G. A. Godfrey, J. Cunningham, and T. Tran. A Bayesian, nonlinear particle filtering approach for tracking the state of terrorist operations. In *IEEE Intelligence and Security Informatics, 2007*, pages 350–355, May 2007.

[17] E. Gutin, D. Kuhn, and W. Wiesemann. Interdiction games on Markovian PERT networks. *Management Science*, 61(5):999–1017, 2014.

[18] R. C. Harney, G. G. Brown, W. M. Carlyle, E. M. Skroch, and R. K. Wood. Anatomy of a project to produce a first nuclear weapon. *Science and Global Security*, 14(2-3):163–182, 2006.

[19] E. H. Kaplan. Terror queues. *Operations Research*, 58(4):773–784, 2010.

[20] E. H. Kaplan. Estimating the duration of Jihadi terror plots in the United States. *Studies in Conflict & Terrorism*, 35(12):880–894, 2012.

[21] E. H. Kaplan. OR forum – Intelligence operations research: The 2010 Philip McCord Morse Lecture. *Operations Research*, 60(6):1297–1309, 2012.

[22] J. E. Kelley. Critical-path planning and scheduling: Mathematical basis. *Operations Research*, 9(3):296–320, 1961.

[23] R. S. Kemp. Environmental detection of clandestine nuclear weapon programs. *Annual Review of Earth and Planetary Sciences*, 44:17–35, 2016.

[24] K. L. Lasoen. War of nerves: The domestic terror threat and the Belgian Army. *Studies in Conflict & Terrorism*, ePub ahead of print February 16, 2018. `https://doi.org/10.1080/1057610X.2018.1431270`.

[25] D. Leech. An empirical comparison of the performance of classical power indices. *Political Studies*, 50(1):1–22, 2002.

[26] K. Y. Lin, M. Kress, and R. Szechtman. Scheduling policies for an antiterrorist surveillance system. *Naval Research Logistics*, 56(2):113–126, 2009.

[27] K. Y. Lin, M. P. Atkinson, and K. D. Glazebrook. Optimal patrol to uncover threats in time when detection is imperfect. *Naval Research Logistics*, 61(8):557–576, 2014.

[28] K. Papadaki, S. Alpern, T. Lidbetter, and A. Morton. Patrolling a border. *Operations Research*, 64(6):1256–1269, 2016.

[29] L. S. Penrose. The elementary statistics of majority voting. *Journal of the Royal Statistical Society*, 109(1):53–57, 1946.

[30] H. Peters. *Game Theory: A Multi-leveled Approach*. Springer, 2008.

[31] E. Pinker. An analysis of short-term responses to threats of terrorism. *Management Science*, 53(6):865–880, 2007.

[32] E. Pinker, J. Szmerekovsky, and V. Tilson. Technical note – Managing a secret project. *Operations Research*, 61(1):65–72, 2013.

[33] E. Pinker, J. Szmerekovsky, and V. Tilson. On the complexity of project scheduling to minimize exposed time. *European Journal of Operational Research*, 237(2):448–453, 2014.

[34] K. Prasad and J. S. Kelly. NP-completeness of some problems concerning voting games. *International Journal of Game Theory*, 19(1):1–9, 1990.

[35] K. Rawlinson, D. Boffey, and J. Rankin. Soldiers shoot suspected terrorist dead at Brussels railway station. *The Guardian*, June 21, 2017. `https://www.theguardian.com/world/2017/jun/20/soldiers-shoot-person-brussels-central-train-station-explosion-belgium`.

[36] S. Ross. *Stochastic Processes*. Wiley, 1982.

[37] L. S. Shapley. A value for n-person games. In A. W. Tucker and H. W. Kuhn, editors, *Contributions to the Theory of Games, Volume II*, pages 307–317. Princeton University Press, 1953.

[38] B. L. Smith, K. R. Damphousse, and P. Roberts. Pre-incident indicators of terrorist incidents: The identification of behavioral, geographic, and temporal patterns of preparatory conduct. Technical report, University of Arkansas, Terrorism Research Center in Fulbright College, 2006.

[39] S. Stewart. Detection points in the terrorist attack cycle. *Security Weekly, Stratfor*, March 1, 2012. `https://worldview.stratfor.com/article/detection-points-terrorist-attack-cycle`.

[40] R. Szechtman, M. Kress, K. Lin, and D. Cfir. Models of sensor operations for border surveillance. *Naval Research Logistics*, 55(1):27–41, 2008.

[41] The Economist. Monitoring nuclear weapons: The nuke detectives. *The Economist*, September 5, 2015. `http://media.economist.com/news/technology-quarterly/21662652-clandestine-weapons-new-ways-detect-covert-nuclear-weapons-are-being-developed`.

[42] J. N. Tsitsiklis and K. Xu. Delay-predictability trade-offs in reaching a secret goal. *Operations Research*, 66(2):587–596, 2018.

[43] B. von Stengel. Recursive inspection games. *Mathematics of Operations Research*, 41(3): 935–952, 2016.

Appendix A Proof of Lemma 2

Lemma 2. *Let $r \in \mathbb{N}$ and $a_1, \ldots, a_r \in \mathbb{R}$, then*

$$\prod_{i=1}^{r}(1-a_i) = 1 - \sum_{i=1}^{r} a_i + \sum_{i=1}^{r}\sum_{j=i+1}^{r} a_i a_j \prod_{k=j+1}^{r}(1-a_k).$$

Proof. The proof goes through induction on r. For $r = 1$, the result is immediate. As the induction hypothesis, suppose the result holds for a given $r \in \mathbb{N}$, then

$$\begin{aligned}
\prod_{i=1}^{r+1}(1-a_i) &= \prod_{i=1}^{r}(1-a_i) - a_{r+1}\prod_{i=1}^{r}(1-a_i) \\
&= 1 - \sum_{i=1}^{r} a_i + \sum_{i=1}^{r}\sum_{j=i+1}^{r} a_i a_j \prod_{k=j+1}^{r}(1-a_k) \\
&\quad - a_{r+1} + a_{r+1}\sum_{i=1}^{r} a_i - \sum_{i=1}^{r}\sum_{j=i+1}^{r} a_i a_j a_{r+1} \prod_{k=j+1}^{r}(1-a_k) \\
&= 1 - \sum_{i=1}^{r+1} a_i + \sum_{i=1}^{r} a_i a_{r+1} + \sum_{i=1}^{r}\sum_{j=i+1}^{r} a_i a_j \prod_{k=j+1}^{r+1}(1-a_k) \\
&= 1 - \sum_{i=1}^{r+1} a_i + \sum_{i=1}^{r+1}\sum_{j=i+1}^{r+1} a_i a_j \prod_{k=j+1}^{r+1}(1-a_k)
\end{aligned}$$

and the result holds for $r+1$ as well. □

Appendix B Detailed description of the nuclear weapons development project

Table 14, included at the end of the appendices, displays the complete output for the numerical example of the nuclear weapons development project that we have described in Section 5. Harney et al. [18] provide three alternative methods for enriching uranium, but we have focused on the so-called 'Aerodynamic enrichment process' (AE) only; the findings for the other two enrichment methods are similar. Since our model does not allow for expediting activities, we only consider the tasks' normal durations. Dummy tasks have been used to transform all precedence constraints to the finish-start type [13]. We refer to Harney et al. [18] for a more detailed description of the project network.

The three functional forms for harm functions displayed in Figure 3 are defined by:

$$\begin{aligned}
h_{\text{concave}}(t) &= 1 - \exp(-0.02(\delta - t)); \\
h_{\text{linear}}(t) &= 1 - t/\delta; \\
h_{\text{convex}}(t) &= \exp(-0.02t).
\end{aligned}$$

Here, δ equals the planned attack time, which we assume to be equal to the minimum makespan of 468 weeks.

To assign each activity to one of the three classes of Table 6, we use the same distinction between innocent and suspicious tasks as in Pinker et al. [32]. Tasks related to the diversion of

	Part				
Class	1	2	3	4	5
1	0	2	0	1	10
2	0	0	6	4	50
3	3	17	10	9	12

Table 10: Number of tasks per part and class.

yellowcake, for example, are considered to be innocent because they could also serve for producing nuclear energy. Most tasks related to the design and construction of the actual weapon, on the other hand, are suspicious because they lack a dual use. Next, we complement this partitioning by randomly subdividing the set of suspicious tasks in another two groups depending on the degree to which monitoring influences the discovery probability: we assign a suspicious task with a 20% chance to the first class and with 80% to the second one. Table 10 contains the resulting number of tasks for each part and class. Next to this 20/80-division, we have also experimented with a 10/90- and 30/70-division, which led to similar findings.

Appendix C Sensitivity analysis for the nuclear weapons development project

Below, we first investigate how sensitive the optimal set of tasks to monitor is to perturbations in the discovery probabilities. Next, we consider the possibility that a task has been misclassified, i.e. assigned to an incorrect class, and examine the optimal set's sensitivity to such misclassification. Finally, we evaluate how well the Banzhaf value measures the harm reduction thanks to monitoring a task by comparing the optimal set of tasks to monitor with the b activities with the highest Banzhaf value (remember that $c_i = 1$ for each $i \in N$). We use the *optimality gap*

$$\varphi(S) \coloneqq \frac{H(S) - H(S^\star)}{H(S^\star)}$$

to assess by how much the expected harm when monitoring $S \subseteq N$ exceeds the minimum expected harm $H(S^\star) = \min_{S \in \mathcal{B}} H(S)$.

Perturbation in the discovery probabilities. To perturb the discovery probabilities, we use the same instance as in Table 14 but multiply each discovery probability with a random factor. That is, for every $i \in N$, we draw two random numbers α and β uniformly form the interval $[l, u]$ and take αp_i^0 and βp_i^1 as task i's new discovery probabilities. We only consider small perturbations with $l, u \in \mathbb{R}$ such that $0 \leq \alpha p_i^0 \leq \beta p_i^1 \leq 1$ for all $\alpha, \beta \in [l, u]$. Let S^o be the optimal set of tasks to monitor for the original, unperturbed, instance. The optimality gap $\varphi(S^o)$ then reflects how well the original solution performs in the perturbed instance and thus indicates how robust the original solution is against small perturbations in the discovery probabilities.

Table 11 shows the average optimality gap resulting from repeating the above procedure 100 times for the different harm functions and budgets. With the average gap never higher than 4.54 percent, the optimum is quite robust against small perturbations; for the concave and linear harm function the gap decreases even further. This robustness is especially reassuring because the discovery probabilities might be hard to estimate accurately in practice.

	Concave			Linear			Convex		
Interval	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$
$[0.95, 1.05]$	0.0000	0.0000	0.0001	0.0000	0.0009	0.0001	0.0005	0.0062	0.0039
$[0.90, 1.10]$	0.0000	0.0000	0.0001	0.0001	0.0021	0.0009	0.0072	0.0186	0.0131
$[0.85, 1.15]$	0.0001	0.0000	0.0002	0.0008	0.0032	0.0018	0.0162	0.0334	0.0218
$[0.80, 1.20]$	0.0001	0.0001	0.0003	0.0017	0.0042	0.0032	0.0260	0.0454	0.0335

Table 11: Average optimality gap for different perturbations in the discovery probabilities.

	Concave			Linear			Convex		
Probability	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$
0.02	0.0019	0.0014	0.0010	0.0277	0.0232	0.0204	0.1079	0.1102	0.0966
0.04	0.0037	0.0028	0.0021	0.0495	0.0435	0.0403	0.1871	0.1924	0.1848
0.06	0.0070	0.0049	0.0030	0.0780	0.0735	0.0671	0.3406	0.3597	0.3338
0.08	0.0085	0.0068	0.0040	0.1141	0.0952	0.0867	0.4585	0.4740	0.4426
0.10	0.0100	0.0070	0.0044	0.1370	0.1167	0.1056	0.5897	0.6329	0.5967

Table 12: Average optimality gap for different probabilities that a task has been misclassified.

Misclassification of activities. In Table 12, we examine how well the original optimal solution performs when a number of tasks have been misclassified. Starting from the instance of Table 14, we randomly decide for each task whether or not to change its class according to the probability given by Table 12's first column. Next, we assign each misclassified task to one of the other two classes with equal probability and compute the optimality gap. Repeating this procedure 100 times and averaging then leads to the displayed numbers.

With the highest average optimality gap equal to 63.3%, misclassifying a task affects the optimal solution significantly more than perturbing the discovery probabilities does. This underlines the importance of paying sufficient attention to a correct classification of the tasks.

Evaluation of the Banzhaf value. Since we have assumed a unit cost $c_i = 1$ for each activity $i \in N$, the defender should monitor those b tasks that, combined, produce the largest harm reduction. Thus, if the Banzhaf value measures the harm reduction thanks to monitoring a task well, the optimal set of tasks to monitor should be sufficiently similar to those b tasks having the highest Banzhaf value. Let $S^\Psi \subseteq N$ contain the b tasks with the highest Banzhaf value, then a small optimality gap $\varphi(S^\Psi)$ corroborates the Banzhaf value as a good measure for the harm reduction thanks to monitoring a task.

Table 13 displays the average optimality gap $\varphi(S^\Psi)$ for nine alternative settings for the discovery probabilities corresponding to each of the three classes of Table 6, averaged over 100 repetitions of the suspicious tasks' random subdivision into the first two classes. For the innocent tasks, we use the same discovery probabilities as in Table 6 and for the suspicious ones we consider $p^0_{\text{susp}} \in \{0.03, 0.05, 0.07\}$ as the exposure probabilities without monitoring. To obtain the discovery probabilities with monitoring, we multiply p^0_{susp} by two for the case of ineffective monitoring and by either 3, 5, or 7 for effective monitoring; we denote this latter probability by p^1_{eff}. Table 6 then corresponds to $p^0_{\text{susp}} = 0.05$ and $p^1_{\text{eff}} = 0.25$.

With an optimality gap never exceeding 5.95%, our computational results suggest that the Banzhaf value constitutes a good measure for the harm reduction by monitoring a task. We have

		Concave			Linear			Convex		
p^0_{susp}	p^1_{eff}	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$	$b = 10$	$b = 15$	$b = 20$
0.03	0.09	0.0000	0.0000	0.0000	0.0008	0.0001	0.0002	0.0051	0.0074	0.0057
0.03	0.15	0.0002	0.0000	0.0000	0.0021	0.0002	0.0006	0.0029	0.0046	0.0050
0.03	0.21	0.0000	0.0000	0.0000	0.0014	0.0018	0.0023	0.0001	0.0009	0.0018
0.05	0.15	0.0000	0.0000	0.0000	0.0015	0.0001	0.0002	0.0163	0.0186	0.0165
0.05	0.25	0.0002	0.0000	0.0000	0.0007	0.0005	0.0010	0.0595	0.0591	0.0336
0.05	0.35	0.0001	0.0000	0.0000	0.0010	0.0009	0.0022	0.0440	0.0402	0.0183
0.07	0.21	0.0000	0.0000	0.0000	0.0005	0.0002	0.0004	0.0076	0.0059	0.0045
0.07	0.35	0.0002	0.0000	0.0000	0.0003	0.0005	0.0023	0.0246	0.0241	0.0196
0.07	0.49	0.0002	0.0000	0.0000	0.0008	0.0006	0.0019	0.0351	0.0290	0.0157

Table 13: Average optimality gap with Banzhaf value for different discovery probabilities.

also experimented with the Shapley value instead of the Banzhaf value and obtained very similar findings.

ID	Description	Part	Class	s_i^λ	$S_1^\star$	Ψ_1	$S_2^\star$	Ψ_2	$S_3^\star$	Ψ_3
3	Design yellowcake plant modifications	1	3	116	$\cdot\star\star$	0.06	$\cdots$	0.18	$\cdots$	0.01
4	Modify yellowcake plant	1	3	164	$\cdots$	0.02	$\cdots$	0.10	$\cdots$	0.01
5	Divert yellowcake	1	3	212	$\cdots$	0.00	$\cdots$	0.04	$\cdots$	0.01
7	Design fluoridation plant (FP)	2	3	36	$\star\star\star$	0.42	$\cdot\star\star$	0.33	$\cdots$	0.01
8	Acquire FP site	2	3	116	$\cdot\star\star$	0.06	$\cdots$	0.18	$\cdots$	0.01
9	Prepare FP site (internal modifications)	2	3	140	$\cdots$	0.03	$\cdots$	0.14	$\cdots$	0.01
11	Stainless steel mixing vessel	2	3	128	$\cdot\cdot\star$	0.05	$\cdots$	0.16	$\cdots$	0.01
12	Distilled water system	2	3	128	$\cdot\cdot\star$	0.05	$\cdots$	0.16	$\cdots$	0.01
13	Nitric acid storage tank	2	3	128	$\cdot\cdot\star$	0.05	$\cdots$	0.16	$\cdots$	0.01
14	Stainless steel boiler	2	3	128	$\cdot\cdot\star$	0.05	$\cdots$	0.16	$\cdots$	0.01
15	Thermal decomposition vessel	2	3	128	$\cdot\cdot\star$	0.05	$\cdots$	0.16	$\cdots$	0.01
16	Drying kiln	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
17	Gas/solid high-temperature reaction vessel	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
18	Hydrogen gas (or ammonia) storage tank	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
19	Stainless steel reaction vessel	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
20	Hydrogen fluoride storage tank	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
21	Gas/solid ultrahigh temperature reaction vessel	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
22	Fluorine storage tank	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
23	Hexafluoride condensing vessel	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
24	Pumps and piping	2	3	128	$\cdots$	0.05	$\cdots$	0.16	$\cdots$	0.01
25	Assemble and integrate FP	2	1	164	$\star\star\star$	0.37	$\star\star\star$	2.35	$\star\star\star$	0.30
26	Operate FP	2	1	212	$\star\star\star$	0.06	$\star\star\star$	1.03	$\star\star\star$	0.26
78	Design basic AE enrichment device	3	3	0	$\star\star\star$	0.94	$\star\star\star$	0.41	$\cdots$	0.01
80	Vortex unit	3	3	12	$\star\star\star$	0.72	$\star\star\star$	0.39	$\cdots$	0.01
81	Pumps and piping	3	3	36	$\star\star\star$	0.42	$\cdot\star\star$	0.33	$\cdots$	0.01
82	Assemble research devices	3	2	60	$\star\star\star$	1.30	$\star\star\star$	1.53	$\star\star\star$	0.08
83	Test and evaluate research devices	3	2	84	$\star\star\star$	0.74	$\star\star\star$	1.27	$\star\star\star$	0.08
84	Design production devices	3	3	104	$\star\star\star$	0.09	$\cdot\cdot\star$	0.20	$\cdots$	0.01
85	Design enrichment cascade	3	3	112	$\star\star\star$	0.07	$\cdots$	0.19	$\cdots$	0.01
86	Design of enrichment plant (EP)	3	3	120	$\cdot\star\star$	0.06	$\cdots$	0.17	$\cdots$	0.01
87	Acquire EP site	3	3	124	$\cdot\star\star$	0.05	$\cdots$	0.17	$\cdots$	0.01
88	Prepare EP site	3	3	148	$\cdots$	0.03	$\cdots$	0.13	$\cdots$	0.01
90	Vortex unit	3	3	164	$\cdots$	0.02	$\cdots$	0.10	$\cdots$	0.01
91	Pumps and piping	3	3	132	$\cdots$	0.04	$\cdots$	0.15	$\cdots$	0.01
92	Assemble production devices	3	2	172	$\cdot\star\star$	0.07	$\star\star\star$	0.48	$\cdot\star\star$	0.07
93	Integrate enrichment cascade	3	2	188	$\cdots$	0.04	$\star\star\star$	0.37	$\cdot\star\star$	0.07
94	Cascade loading	3	2	220	$\cdots$	0.01	$\cdots$	0.21	$\cdots$	0.06

Table 14: Detailed output for nuclear weapons project based on Harney et al. [18]. $S_k^\star$ indicates whether it is optimal to monitor the task and Ψ_k gives the Banzhaf value ($\times 100$). The subscript $k = 1, 2, 3$ refers to the concave, linear, and convex harm function respectively. For $S_k^\star$, an asterisk ($\star$) in position $r = 1, 2, 3$ indicates that it is optimal to monitor the task with budget $b = 10, 15, 20$ respectively.

ID	Description	Part	Class	s_i^λ	$S_1^\star$	Ψ_1	$S_2^\star$	Ψ_2	$S_3^\star$	Ψ_3
95	Produce enriched and depleted material	3	2	332	···	0.00	···	0.02	···	0.02
97	Design metal plant (MP)	4	3	200	···	0.00	···	0.06	···	0.01
98	Acquire MP site	4	3	212	···	0.00	···	0.04	···	0.01
99	Prepare MP site	4	3	236	···	0.00	···	0.03	···	0.01
101	Gas-phase reactor with particulate collection	4	2	232	···	0.01	···	0.18	···	0.06
102	Hydrogen storage tank	4	3	232	···	0.00	···	0.03	···	0.01
103	Metallurgical furnace	4	3	232	···	0.00	···	0.03	···	0.01
104	Hafnia crucibles	4	3	232	···	0.00	···	0.03	···	0.01
106	Gas-phase reactor with particulate collection	4	2	232	···	0.01	···	0.18	···	0.06
107	Hydrogen storage tank	4	3	232	···	0.00	···	0.03	···	0.01
108	Metallurgical furnace	4	3	232	···	0.00	···	0.03	···	0.01
109	Hafnia crucibles	4	3	232	···	0.00	···	0.03	···	0.01
110	Integrate components	4	1	260	···	0.01	⋆⋆⋆	0.50	⋆⋆⋆	0.21
111	Produce natural uranium metal	4	2	292	···	0.00	···	0.06	···	0.04
112	Produce depleted/enriched uranium metal	4	2	332	···	0.00	···	0.02	···	0.02
115	Gun	5	2	208	···	0.02	·⋆⋆	0.25	·⋆⋆	0.06
116	Propellant	5	3	208	···	0.00	···	0.05	···	0.01
118	Fissionable receiver	5	2	208	···	0.02	·⋆⋆	0.25	·⋆⋆	0.06
119	Fissionable projectile	5	2	208	···	0.02	··⋆	0.25	··⋆	0.06
120	Tamper	5	2	208	···	0.02	·⋆⋆	0.25	··⋆	0.06
121	Initiator	5	2	208	···	0.02	··⋆	0.25	··⋆	0.06
122	Safety and arming devices	5	2	208	···	0.02	··⋆	0.25	··⋆	0.06
123	Fuse	5	2	208	···	0.02	··⋆	0.25	··⋆	0.06
124	Design weapon assembly plant (AP)	5	2	256	···	0.00	···	0.12	···	0.05
125	Acquire AP site	5	1	280	···	0.01	⋆⋆⋆	0.34	⋆⋆⋆	0.18
126	Prepare AP site	5	2	296	···	0.00	···	0.05	···	0.04
128	Large-diameter precision lathe	5	3	284	···	0.00	···	0.01	···	0.01
129	Inert-gas environment precision milling machine	5	3	300	···	0.00	···	0.01	···	0.01
130	Metallurgical furnace	5	3	292	···	0.00	···	0.01	···	0.01
131	Hafnia crucibles	5	3	300	···	0.00	···	0.01	···	0.01
132	Inert-gas environment casting system	5	3	300	···	0.00	···	0.01	···	0.01
134	High-strength steel cylinder	5	3	308	···	0.00	···	0.01	···	0.01
135	Double-base propellant powder	5	3	308	···	0.00	···	0.01	···	0.01
136	Polonium	5	2	308	···	0.00	···	0.04	···	0.03
137	Beryllium powder	5	2	308	···	0.00	···	0.04	···	0.03
138	Detonator and explosive train components	5	2	308	···	0.00	···	0.04	···	0.03
140	Gun barrel	5	2	332	···	0.00	···	0.02	···	0.02
141	Breech mechanism	5	2	332	···	0.00	···	0.02	···	0.02
142	Cast uranium components	5	2	340	···	0.00	···	0.01	···	0.02
143	Cast uranium tamper	5	2	348	···	0.00	···	0.01	···	0.01
144	Machine uranium receiver	5	1	348	···	0.00	···	0.03	⋆⋆⋆	0.06
145	Machine uranium projectile	5	2	348	···	0.00	···	0.01	···	0.01
146	Machine uranium tamper	5	2	348	···	0.00	···	0.01	···	0.01
147	Initiator	5	2	348	···	0.00	···	0.01	···	0.01
148	Propellant charge	5	1	348	···	0.00	···	0.03	⋆⋆⋆	0.06
149	Detonator and explosive train	5	1	348	···	0.00	···	0.03	⋆⋆⋆	0.06
150	Assemble research devices (natural uranium prototype)	5	2	352	···	0.00	···	0.01	···	0.01
152	High-strength steel cylinder	5	3	332	···	0.00	···	0.00	···	0.00
153	Double-base propellant powder	5	3	332	···	0.00	···	0.00	···	0.00

Table 14: Detailed output for nuclear weapons project based on Harney et al. [18] (continued). $S_k^\star$ indicates whether it is optimal to monitor the task and Ψ_k gives the Banzhaf value ($\times 100$). The subscript $k = 1, 2, 3$ refers to the concave, linear, and convex harm function respectively. For $S_k^\star$, an asterisk ($\star$) in position $r = 1, 2, 3$ indicates that it is optimal to monitor the task with budget $b = 10, 15, 20$ respectively.

ID	Description	Part	Class	s_i^λ	$S_1^\star$	Ψ_1	$S_2^\star$	Ψ_2	$S_3^\star$	Ψ_3
154	Polonium	5	2	332	$\cdots$	0.00	$\cdots$	0.02	$\cdots$	0.02
155	Beryllium powder	5	1	332	$\cdots$	0.00	$\cdots$	0.07	$\star\star\star$	0.08
156	Detonator and explosive train components	5	2	332	$\cdots$	0.00	$\cdots$	0.02	$\cdots$	0.02
158	Gun barrel	5	2	356	$\cdots$	0.00	$\cdots$	0.01	$\cdots$	0.01
159	Breech mechanism	5	2	356	$\cdots$	0.00	$\cdots$	0.01	$\cdots$	0.01
160	Casting of enriched uranium components	5	2	356	$\cdots$	0.00	$\cdots$	0.01	$\cdots$	0.01
161	Cast depleted uranium tamper	5	2	372	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
162	Machine enriched uranium receiver	5	1	364	$\cdots$	0.00	$\cdots$	0.02	$\cdot\star\star$	0.04
163	Machine enriched uranium projectile	5	2	364	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
164	Machine depleted uranium tamper	5	2	364	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
165	Initiator	5	2	372	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
166	Propellant charge	5	2	372	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
167	Detonator and explosive train	5	2	372	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
168	Assemble research devices (enriched uranium prototype)	5	2	368	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
170	Verify critical mass	5	2	384	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
171	Verify gun velocity	5	2	372	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
172	Delivery vehicle compatibility mock-up	5	2	396	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
173	Test full-scale device (not required)	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
174	Finalize production-weapon design	5	2	380	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
176	High-strength steel cylinder	5	3	396	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
177	Double-base propellant powder	5	3	396	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
178	Polonium	5	2	396	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
179	Beryllium powder	5	2	396	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
180	Detonator and explosive train components	5	2	396	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
182	Gun barrel	5	1	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
183	Breech mechanism	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
184	Cast enriched uranium components	5	1	444	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
185	Cast depleted uranium tamper	5	2	396	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
186	Machine enriched uranium receiver	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
187	Machine enriched uranium projectile	5	1	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
188	Machine depleted uranium tamper	5	1	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.01
189	Initiator	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
190	Propellant charge	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
191	Detonator and explosive train	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
192	Fuse	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
193	Safety and arming device	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
194	Weapon case and structure	5	2	420	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00
195	Assemble weapon components	5	2	444	$\cdots$	0.00	$\cdots$	0.00	$\cdots$	0.00

Table 14: Detailed output for nuclear weapons project based on Harney et al. [18] (continued). $S_k^\star$ indicates whether it is optimal to monitor the task and Ψ_k gives the Banzhaf value ($\times 100$). The subscript $k = 1, 2, 3$ refers to the concave, linear, and convex harm function respectively. For $S_k^\star$, an asterisk ($\star$) in position $r = 1, 2, 3$ indicates that it is optimal to monitor the task with budget $b = 10, 15, 20$ respectively.