Optimal location of launchers in a missile defence system

N. Azzollini [a], B. Babu [a], A. Reade [a] and J. Rodda [a]

[a]*Mathematics Clinic, UniSA STEM, University of South Australia, Mawson Lakes SA 5095*

Abstract: A commonly recognised problem in defence is defending against incoming missile threats, and determining how a variety of platforms and assets must work collaboratively in order achieve mission success.

A particular problem is the optimal placement of sensors and surface-to-air missile launchers around a high-value asset. The primary objective in this instance would be to defend the high-value asset with the greatest likelihood of success. The missile defence systems considered include sensors and missile launchers, which are used to detect and interdict enemy missiles, respectively.

Our problem is to optimise the arrangement of a sensor and missile launcher to defend a high-value asset against threats from any possible direction. We devise an objective function which measures the probability of failing to intercept at all possible intercept points across all possible paths of an incoming missile. We use a simple simulation to calculate the objective score for a given arrangement of sensors and missile launchers. We optimise the arrangement using a Nelder-Mead optimisation method, which calls the simulator with candidate arrangements. Nelder-Mead was used because it does not require derivatives, is easy to implement and can handle multidimensional non-linear optimisation.

We tested our optimiser and simulator using 6 different test cases where we optimised the missile launcher position for various threat scenarios. Some of the resulting optimal missile launcher positions were closer to the high value asset than we anticipated. We found, for example, that if the threat was entirely from one direction then the optimal missile launcher position is far from the asset it is protecting. But if even a small probability of attack, such as 0.0001, occurs from the opposite direction then the missile launcher would move back closer to the asset.

This project is supported by the Commonwealth of Australia as represented by the Defence Science and Technology Group of the Department of Defence. The work for this project was conducted as a part of the University of South Australia's Mathematics Clinic for 2021. The Mathematics Clinic is a year-long sponsored research project undertaken by final-year undergraduates studying mathematics, and offers in-depth research experience in real world mathematics where the project is posed from industry and supported by academic advisors.

Keywords: *Missile defence, facility location, optimisation*

1 Introduction

Missile defence systems are used to defend high-value assets against incoming threats. The performance of a missile defence system will depend on the location of the sensing systems and defending missile launchers. Our specific problem is to optimise the arrangement of threat sensors and surface-to-air missile launchers around a single high-value asset.

Australia's Defence Science and Technology Group (DSTG) has high-fidelity models for simulating missile defence scenarios. Our focus was to develop an objective function and optimisation method that could be used with these models to optimise the arrangement of sensors and missile launchers.

In this paper we:

1. describe our mathematical model of a simple missile defence system (Section 2)
2. formulate an objective function that quantifies the performance of an arrangement of sensors and missile launchers around a high value asset (Section 3)
3. describe our approach to simulating and optimising the arrangements (Section 4)
4. give example results for various defence scenarios (Section 5).

2 Problem Formulation

We consider a simple problem with one high value asset to be protected against one threat missile. The missile defence system has one sensor for detecting threat missiles, and one missile launcher that can launch multiple defending missiles at the threat.

Figure 1 shows an example. The HVU is a *High Value Unit* such as a defended cargo vessel. The *sensor* is typically a radar, used to detect and track enemy missiles. The green region around the sensor indicates the sensor range. *Effectors* are devices used to intercept enemy threat missiles; in our case we consider surface-to-air missiles launched from a fixed location. The yellow circle represents the effector range. The missile system defends against a single *threat missile* that targets the HVU.

We do not know exactly which direction the threat will come from. Our arrangement of the sensor and effector must be able to defend against threats from any possible direction.

The sequence of events in a missile defence scenario would be: initial detection of the incoming threat missile by the sensor, establishing a track on the threat, predicting where effectors can intercept the threat, launching effectors, assessing the engagement, and re-engaging if required.

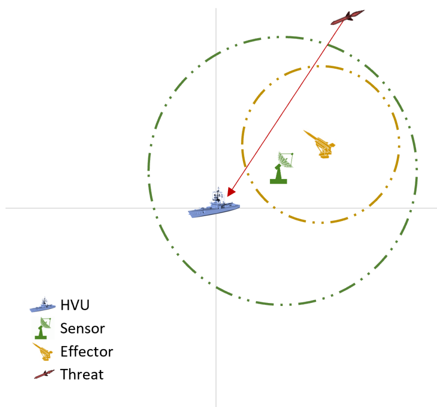

Figure 1. Plan view of a missile defence scenario

For our simple models, we will assume that:

- we can model defence scenarios in 2 dimensions, looking down from above
- the HVU, sensor and effector launcher do not move during any engagement
- the threat missile travels in a straight line and at constant speed towards the HVU
- we can model the unknown approach direction of the threat missile by directions with given probabilities
- the sensor has a large range (100 km) and can detect threats from any direction
- effectors travel in a straight line at constant speed towards the threat
- we launch effectors one at a time
- if an effector misses the threat, there is a short delay before we can launch another effector.

These assumptions may appear to limit the relevance of our results to real-world missile defence scenarios. However, our optimisation approach will be general enough that it can work with higher fidelity models. Further work could be implemented with parameter values such as altitude and radar horizon to make the problem 3 dimensional and hence more relevant.

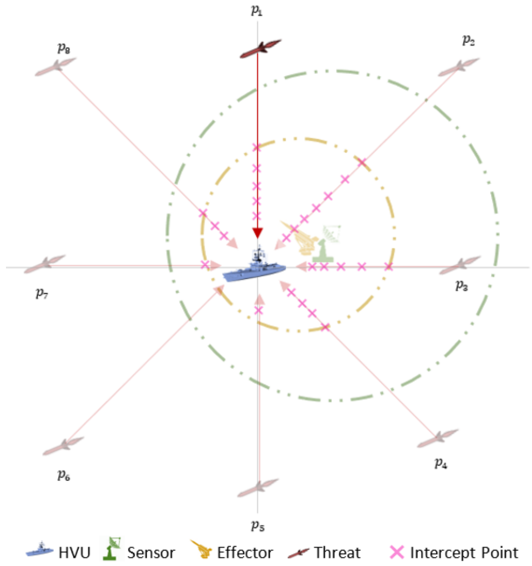

Figure 2. Example missile defence scenario with 8 possible approach paths

3 Scoring Metric

In order to determine the optimal arrangement of the sensor and effector launcher about the HVU, we must be able to quantify the performance of any possible arrangement. We have developed a scoring metric that calculates the probability of failing to neutralise an enemy missile. First, we consider that the incoming enemy missile could approach along any one of N possible paths, each with a given probability. Figure 2 shows an example with eight possible approach paths, with probabilities $p_1, p_2, ..., p_8$.

Figure 3 is an engagement timeline that shows the sequence of events that would occur if the missile were to approach along a particular path. In this particular case, the effector and the sensor are both located at the HVU. The horizontal axis represents time, while the vertical axis represents the distance the threat is from the HVU (and sensor and effector).

The threat missile is first detected when $t = 0$, at the detection range of the sensor. The first attempted intercept occurs at the effector range, though this is not always possible if the sensor range is close to the effector range. Each attempted intercept has a probability of failure $f = 0.4$. If we miss, there is a time delay between the failed first intercept and the second effector launch. At the second intercept point, the accumulated probability of failure is $0.4^2 = 0.16$.

To evaluate the number of possible intercepts along a path, the process continues until the threat missile reaches the 'Collateral Damage Range'—any intercept beyond this point will inflict significant damage on the HVU—or flies beyond the effector range. We can see that the probability of failure at the fifth and final intercept attempt is $0.4^5 = 0.01024$. This is the probability that we miss all possible intercept attempts along the path.

Ultimately, we want to minimise the probability of *failing to intercept at all possible intercept points*, across all possible paths. This is calculated using the objective function

$$P_{\text{fail}} = \sum_i p_i f^{n_i}. \tag{1}$$

where p_i is the probability that the threat missile is on path i, with $\sum_i p_i = 1$; f is the probability of a single defending effector failing to neutralise the threat missile; and n_i is the number of intercept attempts possible along path i. The probability of successfully defending the asset is

$$P_{\text{success}} = 1 - P_{\text{fail}}. \tag{2}$$

The objective value given by (1) will have step changes in it as the number n_i of possible intercepts on a path i changes from one integer to another. A smooth objective function will be easier to optimise, and so

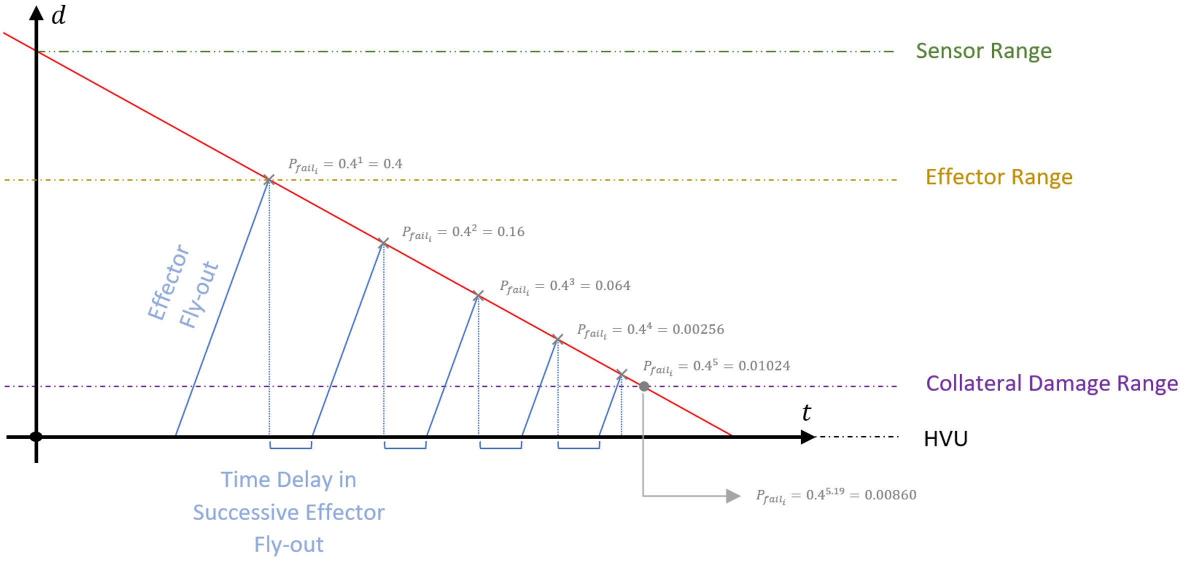

Figure 3. Engagement timeline

we let n_i be a continuous variable. In the example shown in Figure 3, the threat enters the collateral damage range somewhere between attempt 5 and attempt 6. We use interpolation to determine $n_1 \approx 5.19$ and so the probability of failure is $0.4^{5.19} \approx 0.00860$.

4 Simulating and Optimising Arrangements

We developed a simple simulator that, for a given arrangement of sensors and effector launchers, can calculate the number of possible intercepts along each possible approach path, and hence calculate the score for that arrangement (see Algorithm 1). With multiple effector launchers, the simulator chooses the launcher that will give the earliest intercept time for the next launch.

Algorithm 1: Simulator psuedocode

Input: Arrangement
Output: Score and visualisation
read arrangement data;
$s \leftarrow 0$;
foreach *path i* **do**
  calculate the threat missile path;
  calculate the first intercept;
  $n_i \leftarrow 0$;
  while *intercept is possible* **do**
    $n_i \leftarrow n_i + 1$;
    move the threat to the intercept point;
    calculate the next intercept point;
  end
  interpolate n_i beyond the last intercept;
  $s \leftarrow s + p_i f^{n_i}$;
end
write s;
create visualisation;

Ultimately, our simple simulator can be replaced with DTSG's high-fidelity simulators.

For the experiments described in the next section, the sensor range is considerably larger than the effector range, so we fix the sensor at the HVU location $(0, 0)$ and search for an effector launcher position (x, y) that minimises the score. The optimiser objective function calls the simulator with a candidate launcher position, and the simulator returns the corresponding score.

The optimisation method must work without having access to derivatives of the objective function, and should not require too many objective function evaluations since each objective function evaluation requires a simulation.

We used the `scipy.optimize.minimize` function from the Python `SciPy` package (SciPy 2021) to optimise the arrangement, using the Nelder-Mead solver.

5 Examples

To test our optimiser and simulator, we generated a series of test cases for which we defined the expected output, and compared this to the output from the software.

Table 1 shows optimal effector launcher locations for various scenarios with one HVU, a single sensor centred on the HVU, and a single effector launcher. The sensor has a range of 100 km. Threat missiles approach at 400 m/s. Defending missiles travel at 600 m/s and have a range of 40 km. These values were suggested by our DSTG Liaisons. Each scenario has a single threat with specified possible threat paths, with the angles (in degrees clockwise from north) and probabilities shown in the table. These threat paths were chosen for each scenario to test specific aspects of the optimisation. For example, with threats coming from north, south, east and west with equal probability, we expect the effector to be located at the HVU. Effector launcher positions are in metres relative to the HVU. Each attempt to intercept a threat has a 0.4 probability of failure. The last column is the number of simulations performed by the Nelder-Mead algorithm to find the optimal launcher position.

Table 1. Optimal locations for a single effector launcher

Scenario	Threat paths (θ, P)	Launcher position	Score	Simulations
1	(0, 0.25), (90, 0.25), (180, 0.25), (270, 0.25)	(0, 0)	8.32×10^{-4}	112
2	(0, 1)	(-9, 33 341)	4.93×10^{-5}	136
3	(0, 0.9), (180, 0.1)	(-19, 1022)	6.66×10^{-4}	133
4	(0, 0.9999), (180, 0.0001)	(-26, 33 341)	7.69×10^{-5}	141
5	(0, 0.5), (90, 0.5)	(248, 237)	7.66×10^{-4}	113
6	(190, 0.1), (210, 0.3), (280, 0.3), (290, 0.1), (350, 0.2)	(-84, 27)	8.14×10^{-4}	115

Figure 4 shows the optimal arrangement for each of the scenarios in Table 1. In each diagram, the HVU and sensor are located at the centre, with the effector launcher positioned where the optimiser suggests. The sensor range is indicated by the green ring, the effector range by the yellow ring, and the threat paths by the red lines. The intercept points are depicted by the black points.

In Scenario 1 the threat directions are uniformly spaced around the HVU. The optimal effector position found by the optimiser is at the HVU, as we would expect based on the symmetry of the problem. The optimal arrangement is shown in Figure 4(a).

Scenario 2 has only one threat direction. We expected the effector to be shifted towards the approach direction so that the effector range covers as much of the approach path as possible. Perhaps surprisingly, the effector is moved far enough north that the northern effector range almost coincides with the sensor range, as seen in Figure 4(b). This means that the first intercept cannot occur at the effector range, because it takes time for an effector to fly out after detection. The launcher fires effectors at the threat as it approaches from the north, then turns around and fires effectors at the threat as it recedes to the south. For Scenario 2 we also plotted the sequence of effector launcher locations generated by the Nelder-Mead method (Figure 5). The path starts at $(0, 0)$, and finishes at $(-9, 33\,341)$.

Scenario 3 has approach probabilities $p(0°) = 0.9$ and $p(180°) = 0.1$. The effector position shifts north by 1 km, which is not as much as we expected as we initially thought the effector would be shifted further north like Scenario 2. But shifting the effector further north would reduce the number of possible intercepts to the south, resulting in an increased probability of failure for that path. In this scenario, there are 8.27 intercepts on the northern approach, and 6.76 intercepts on the southern approach, as shown in Figure 4(c). As the path probability on $\theta = 0°$ approaches 1, the effector will be positioned further in this direction. Scenario 4 verifies this. As we increase $p(0°)$ to 0.9999, the optimal effector position is shifted to 17 km north of the HVU, as shown in Figure 4(d).

Scenario 5 has equally likely approach paths from the north and east. Although we might expect the effector to move in the north-east direction, the optimal effector location is close to the HVU (Figure 4(e)). We see this result because if we move the effector north, the number of intercepts on the eastern approach path will be reduced. Similarly, if we move the effector east then the number of intercepts on the northern approach path will be reduced. Moving the effector diagonally north-east would reduce the number of intercepts on both paths. The optimal launcher position is about (250, 250). The scores for (0, 0) and (500, 500) are both worse.

Finally, Scenario 6 was devised in order to investigate how the effector position changes when the threats approach from a variety of different bearings, all to the west. Figure 4(f) shows how the effector moves only 84 metres west.

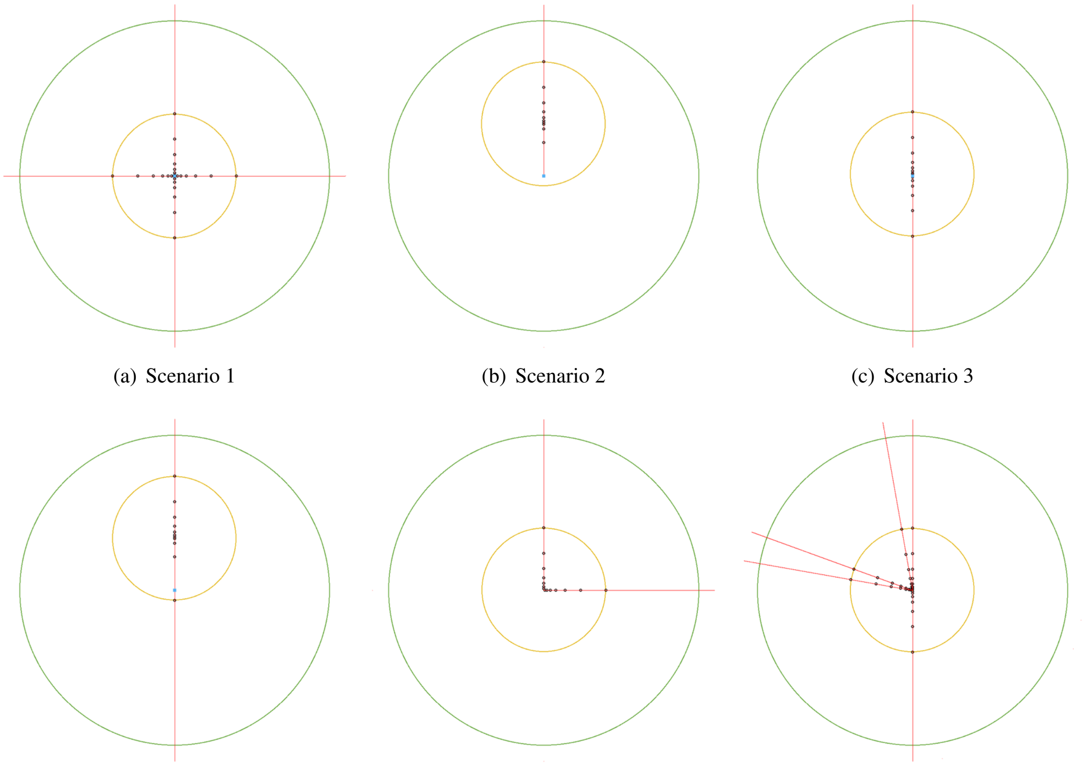

Figure 4. Optimised effector locations. The sensor range is indicated by the green ring, the effector range by the yellow ring, and the threat paths by the red lines. The intercept points are depicted by the black points.

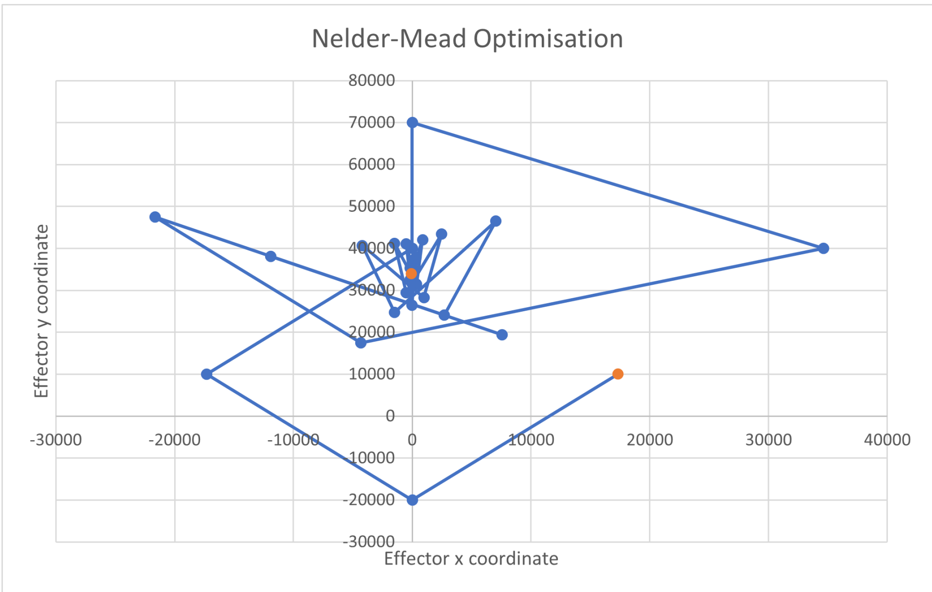

Figure 5. Scenario 2 Nelder-Mead optimisation path

The number of simulations required to find an optimal solution will be important if the time taken to do a high-fidelity simulation will be significant. The Nelder-Mead method takes between 112 and 141 simulations to find an optimal solution.

6 Conclusion and Further Work

We have developed an objective function that can be used to optimise the locations of sensors and missile launchers around a high-value asset in a missile defence system. We assume that we do not know the path that an incoming threat missile will use to approach the target, but we do know the probability of every possible approach path. The objective function value to be minimised is the probability that an incoming threat missile will be missed by every successive intercept attempt.

We have also developed a simple simulation that can calculate the objective score for any specified arrangement of threat approach paths, sensors and missile launchers.

Our objective function is a continuous function of sensor and missile launcher positions so that it can be optimised using popular optimisation tools. We have demonstrated the optimisation using the Nelder-Mead method from the Python SciPy library.

Our work so far has considered simple 2-dimensional models. In reality, the range at which an incoming threat missile can be detected depends on its altitude—'sea-skimming' threat missiles cannot be detected until they are relatively close to the sensor. We could model this without resorting to a full 3-dimensional model by having different detection ranges for different approach paths.

Ultimately, our optimisation approach could be used with DSTG's high-fidelity simulators to help optimise the performance of missile defence systems.

Acknowledgements

This work was conducted as a part of the University of South Australia's Mathematics Clinic for 2021. The Mathematics Clinic is a year-long sponsored research project undertaken by final-year undergraduates studying mathematics, and offers in-depth research experience in real world mathematics where the project is posed from industry and supported by academic advisors. This project is supported by the Commonwealth of Australia as represented by the Defence Science and Technology Group of the Department of Defence.

We thank our DSTG Liaisons Phillip Mellen and Patrick van Bodegom for assisting us in understanding missile defence systems, and for providing assistance and advice throughout the investigation process. We also thank Peter Pudney from UniSA for advising us throughout this project and also Jorge Aarao from UniSA for his assistance.

References

SciPy (2021), 'scipy.optimize.minimize', `https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html`.