Torpedoes on Target: EAs on Track

Nicole Patterson, Luigi Barone and Cara MacNish
School of Computer Science & Software Engineering
The University of Western Australia
M002, 35 Stirling Highway, Crawley, Western Australia, 6009
pattern@tpg.com.au, {luigi, cara}@csse.uwa.edu.au

Abstract
Designing tracks for torpedoes is a challenging task. Deployers are required to generate tracks that incorporate multiple constraints in a high pressure environment. This article presents an application of Evolutionary Algorithms (EAs) to torpedo track planning. The algorithm seeks a good track for a torpedo attacking a moving target in an environment containing a range of natural and man-made features. Three key scenarios are presented which show that the EA is producing promising results for single torpedo, single target engagements. Tracks produced show that the EA makes intelligent decisions about using the engagement environment in producing a solution.

Keywords: path-planning, evolutionary algorithms, torpedo tracks, naval engagements

1 Introduction
Planning tracks or paths for a torpedo to follow en-route to a target is a challenging task for deployers in a stressful environment. Combat situations typically require a track to be found quickly and accurately as information about the situation becomes available. Also tracks for the torpedo are generated using torpedo course instructions rather than in more intuitive Cartesian coordinates, adding an extra layer of difficulty to an already arduous task.

Evolutionary algorithms (EAs) can be used to find good solutions for complex problems. They work well in situations where the optimal solution either can not be found or is difficult to find analytically. EAs are modelled on genetic evolution where there is constant pressure on species to improve. As the torpedo environment is a complex domain, EAs should be a useful tool in finding optimal tracks. EAs have already been demonstrated as being useful in path planning for missiles by Creaser [2] and Hughes [4] both in single engagements and swarms. While the dynamics of torpedoes are similar to missiles, there are significant differences between the two domains. These differences include: detection capabilities, propulsion and the respective environmental operating conditions. In addition, most of the work on missile guidance has been on developing a lower level guidance system. Torpedo track-planning occurs at a higher level, taking advantage of the existing control system to implement the low level commands necessary. This work focusses on the high level planning of torpedo tracks.

This article considers single torpedo, single target engagements where the tracks are generated offline or pre-launch of the torpedo. Section 2 describes how a torpedo track can be modelled and Section 3 introduces EAs and describes how we have applied them to torpedo track planning. Section 4 describes some typical single torpedo-single target engagement scenarios and presents the solutions produced by the EA. Section 5 outlines the conclusions of this work and discusses future developments of the system.

2 Torpedo Modelling
A torpedo has a number of modes of operation from its launch to arriving at the target location. These modes, passive, active and search, relate to the type of sonar employed by the torpedo at that time. Torpedoes typically have both passive and active sonar onboard. When using the quiet passive sonar (passive mode), the torpedo is less aware of its environment but has a lower rate of detection by enemy craft. Active sonar is much noisier and is used to find and lock onto a target (active mode). Search mode is used to locate a target when the torpedo can not find it at the specified location. Typically, a torpedo is given instructions pre-launch as to how to travel (in passive mode) to a point before it is to switch to active mode for a final engagement. Search mode is not considered in this article for reasons described later.

Torpedoes use rudders to alter their orientations and propellers driven by a motor as propulsion. Control of these rudders and propulsion is onboard the torpedo so that the deployer can provide high level instructions for the torpedo control system to implement.
Torpedo tracks are usually controlled using bearing, depth and speed instructions. A *leg* is a set of instructions for the desired depth, bearing and speed of the torpedo. A duration is specified for the leg and includes both the time taken to change course and the time spent on the new course. Thus a leg can be described by:

\[
Leg = \{\text{duration, bearing, depth, speed}\}.
\]

A torpedo track is made up of a number of these legs and here is denoted as an *instruction set* for the track:

\[
Instruction\ Set = [Leg_1, Leg_2, \ldots, Leg_n].
\]

Instruction sets can be converted into Cartesian space by assuming that the torpedo changes its leg variables smoothly to the desired values within given linear and angular acceleration bounds. Six coordinates are used to describe the position and orientation of the torpedo at intervals of time $dt$. We define the track in these Cartesian coordinates.

\[
track = \begin{bmatrix}
  x_1, y_1, z_1, \alpha_1, \beta_1, \gamma_1 \n
  x_2, y_2, z_2, \alpha_2, \beta_2, \gamma_2 \n
  \vdots \n
  x_n, y_n, z_n, \alpha_n, \beta_n, \gamma_n
\end{bmatrix}^T.
\]

The torpedo has its own onboard control system that can account for externalities. Thus the torpedo can follow the instructions given without the deployer needing to consider currents or changes in water conditions.

## 3 Evolutionary Algorithms

### 3.1 Evolutionary Algorithms as an Optimisation Tool

Evolutionary Algorithms (EAs) are a good optimisation tool because they are flexible and robust [1]. They are modelled on the biological evolution process observed in nature. A population of solutions is constructed and subjected to evolutionary operations (reproduction, mutation). Environmental selection of the best solution, based on Darwin’s theory of survival of the fittest, is applied to this population and the species evolves its way towards an optimum.

The basic algorithm can be implemented as follows:

1. Initialise population of candidate solutions.
2. For a number of generations:
   1. Apply reproduction operators to candidates to produce offspring.
   2. Evaluate the fitness of candidate solutions (offspring and parents).
   3. Select a number of candidates to continue to the next generation.
3. Repeat until either the number of allowed iterations has been reached or a solution with a better fitness than some threshold has been found.

The *genotype* of a candidate is the genetic makeup of the candidate whilst the *phenotype* is the actual behaviour of the candidate in the environment [3]. Reproduction operators are typically crossover and mutation [3, 1] and are applied to the genotype of the candidate. Crossover is analogous to sexual reproduction in genetics where offspring contain a part of each parent’s genetic material. Mutation operators probabilistically change some genes in a solution.

Solutions are evaluated and ranked according to the fitness of the phenotype.

### 3.2 Applying The EA structure to the Torpedo Optimisation Problem

The smallest definable genetic building block in this problem is the instructions for a leg of the torpedo’s track. We define this as a *gene*. The genotype or chromosome used as a candidate solution is the set of instructions given by a deployer to the torpedo. The phenotype or behaviour of the candidate is the track produced by this instruction set. These representations are shown in Figure 1.

#### 3.2.1 Initial Population

The starting population is initialised randomly with the bearing, depth and speed variables chosen uniformly within the allowed ranges. The duration for each leg is chosen from a normal distribution with the mean and standard deviation designed to balance the duration of each leg in the path.

#### 3.2.2 Reproduction

The two reproduction operators employed in this EA are mutation and two parent crossover. Crossover is performed by choosing a cut point on each of the two chromosome parents using a uniform random variable and swapping the two parts at this cut to form new offspring solutions. The first parent is always selected from the top half of the population (sorted by fitness score) while the second parent can be any other member of the current population. Mutation is applied after the recombination process using normal distributions about the current values. The standard deviations of these distributions are reduced exponentially with the number of iterations, trading exploration for convergence.
predictable paths and so if the torpedo travels in their
of a large amount of noise. They travel along highly
fixed percentage. Merchant ships tend to be producers
plane that reduces noise travelling through it by some
torpedo. We model a thermal inversion as a horizontal
level should decrease as the square of the distance from
the target (distance and orientation compared to the target's
predicted position) and the speed at which the torpedo
finishes. Koopman [5] provides a good understanding
of these factors as applied to naval engagements.

The actual path taken to reach the endpoint is rated
on estimated detection rate by the target and how long
the target is given to react. Detection rates are not
a definitive measure but an estimate of how easily a
target should be able to detect the torpedo. We use a
quantitative estimate based on the torpedo noise gener-
ated, which in turn depends on the speed the torpedo
is travelling and any manoeuvring (change of direc-
tion or speed) that the torpedo undergoes. The noise
level should decrease as the square of the distance from
the torpedo. Further environmental factors such as a
thermal inversion layer or noise sources such as mer-
chant ships are added into the detection measure as
they may reduce the overall level of detection of the
torpedo. We model a thermal inversion as a horizontal
plane that reduces noise travelling through it by some
fixed percentage. Merchant ships tend to be producers
of a large amount of noise. They travel along highly
predictable paths and so if the torpedo travels in their
wake, it is less likely to be detected by potential targets.
A detection measure can estimated from these factors
as follows:

\[
detection \text{ measure} = \text{noise}(\text{speed}, \text{manoeuvres}) \cdot \text{mask}(\text{distance}, \text{environment}).
\]

Apart from measuring how easily detectable a partic-
ular track is, the fitness function also considers when
along the path the torpedo is considered to be easily de-
tected. Early detection gives the target a greater chance
of evading or launching countermeasures so is consid-
ered less effective than later detection. Also, the longer
the torpedo spends in the water, the more chance it has
of being detected so the total time of the engagement is
included.

One of the most important elements of the track is that
the torpedo can not travel too close to any object or
it may collide with that object. This is particularly
relevant to neutral and friendly forces in the area, so
a large penalty is imposed if the torpedo does come too
close to any object it is not targeting. The issue of false
targeting arises again when the torpedo becomes active
and acquires its target. If another non-target entity is
considered to be closer or more easily acquired by the
 torpedo, the track is penalised severely.

These different objectives are combined in a weighted
sum. The weights can be changed and are used to
assign priorities to different objectives; some engage-
ments place reaching the target quickly above the wish
to remain hidden and in other cases secrecy is given top
priority.

The total fitness is illustrated in Equation (1), repre-
sented as a minimisation problem.

\[
\text{fitness}(\text{track}) = w_1 \cdot \text{collision risk} + w_2 \cdot \text{miss distance} + w_3 \cdot \text{duration} + w_4 \cdot \text{final speed offset} + w_5 \cdot \text{orientation} + w_6 \cdot \text{detection measure} + w_7 \cdot \text{time from detection}
\]
In Equation (1), the objectives are listed in the order of priority assigned. The element **collision risk** adds a severe penalty if the torpedo track goes too close to any non-target craft in the engagement. The **miss distance** refers to the distance the torpedo is from the target at the end of the track, the **orientation** refers to the difference between the direction the torpedo points and the direction the target is relative to the torpedo at the end of the track and the **final speed offset** is an indicator of if the torpedo’s speed is sufficiently high. The **path detection measure** assesses the noise levels received by enemy ships, the **time from detection** estimates how long the enemy has to react from detecting the torpedo and the **duration** refers to the time taken from launch until the torpedo goes active.

### 4 Engagement Scenarios

In these experiments we consider only engagements of single-torpedo single-target scenarios in which knowledge of the environment and target is known pre-launch. The torpedo’s environment can include other craft in the area such as neutral craft, for example merchant ships, and friendly forces. The target is assumed to move in a single direction with constant velocity. The three scenarios presented illustrate paths evolved in increasingly complex environments. The EA parameters are shown in Table 1.

#### 4.1 Scenario 1: Selection Based on Distance

The torpedo is launched at a depth of 100m below the surface facing due North. The target is a ship at a bearing of 045 (45°) from this start location and is heading due East. There is a thermal inversion layer at a depth of 150m.

Qualitatively, we expect the best track to be one of two options:

1. Ignore the inversion layer and aim directly for the earliest possible intercept point with the target and increase speed to a maximum. This is the expected behaviour when the target is relatively close.

2. Move below the thermal inversion layer (thus reducing the chance of early detection) but intercept the target later. This type of behaviour is expected when the target is much further away.

The tracks generated by the EA exhibit these behaviours depending on the distance of the target. When the target is 11.3km away as shown in Figure 2, the EA directs the torpedo to use the thermal inversion layer to mask its noise. When the distance is much less, as shown in Figure 3 where the target is 1.4km away, the EA directs the torpedo to ignore the inversion layer and aim directly for the target. Both of these situations agree with the expected behaviour and show that the EA makes a correct assessment on whether or not to use the thermal inversion layer.

#### 4.2 Scenario 2: Using a Merchant Vessel

In this scenario, the target is launched as before, heading due North at a depth of 100m. The target is at a bearing of 045 (45°), 11.3km away heading due East at 5m/s. There is a single thermal inversion layer 150m below the surface shown as a dark plane. The target’s initial location and direction is indicated with an arrow. The EA directs the torpedo to use the inversion layer to hide its motion.

In this engagement, the torpedo can mask its noise by running underneath the merchant ship for some part of its journey however if it gets too close it may lock on to the wrong target or collide with the ship.

Figure 4 shows a track evolved when the merchant ship is initially 200m South of the torpedo. The EA guides the torpedo to use the ship to mask the noise as it is beneficial to do so in this case. The ship is only used for part of the torpedo track. To use the ship as a mask,
Figure 3: (target is close) The torpedo is launched at 15m/s, at a depth of 100m facing due North. The target is a ship at a bearing of 045, 1.4km away heading due East at 5m/s. There is a single thermal inversion layer at 150m below the surface shown as a dark plane. The target’s initial location and direction is indicated with an arrow. The EA directs the torpedo to ignore the inversion layer and take a direct route. The torpedo must be at the correct depth below the ship. As the torpedo moves it changes its course however there is no gain in changing its course until the depth is correct. The torpedo breaks away from the ship to hit the target at the end of its track.

If the merchant ship starts 1km further South (1.2km South of the torpedo) but otherwise is unchanged in speed and bearing, the resulting path is vastly different. The ship is now too far away. The cost of modifying the path to use the ship outweighs the noise reduction gains. The EA now generates a path in which the torpedo uses the ship to mask its noise only when there is a net benefit to do so.

4.3 Scenario 3: Complex Case

This engagement includes two friendly submarines, two neutral ships, a thermal inversion layer and a moving target. Such a scenario would be complex for unassisted deployer as there are a significant number of moving craft in the area, including friendly forces.

Table 1: EA parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mutation Rate</td>
<td>10%</td>
</tr>
<tr>
<td>Mutation StDev</td>
<td>0.125 * Range, scaled down by 10% per 500 gen</td>
</tr>
<tr>
<td>Population Size</td>
<td>50</td>
</tr>
<tr>
<td>Maximum number of generations</td>
<td>5000</td>
</tr>
<tr>
<td>Initialisation of Variables</td>
<td>uniform 0-359°</td>
</tr>
<tr>
<td>Bearing</td>
<td>uniform 5-500m</td>
</tr>
<tr>
<td>Depth</td>
<td>uniform 1-22m/s</td>
</tr>
<tr>
<td>Speed</td>
<td></td>
</tr>
<tr>
<td>Number of legs</td>
<td>1-8</td>
</tr>
</tbody>
</table>

Figure 4: (merchant ship is close) The torpedo is launched at 15m/s, at a depth of 100m facing due North. The target is a ship at 045 at 2.2km heading N030E at 5m/s as indicated by an arrow. A ship is 200m South of the torpedo heading parallel to the target but at a greater speed. The ship’s path is indicated by a dotted line. The torpedo is guided along a track that uses the ship to mask its own noise and so takes a slightly less direct route.
Figure 5 shows the set up of this scenario and the torpedo track proposed by the EA. The evolved track avoids the first submarine by going above the submarine’s expected path, then uses the paths of the two merchant ships to partially mask its approach to the target. The torpedo manages to hit the target, avoid all other craft in the region and make use of noise sources to mask its own presence. This kind of track would be difficult to set manually as there are five moving objects other than the torpedo to consider and a large cost for failure.

5 Conclusion

There is a problem generating suitable tracks for a torpedo to follow in an engagement situation because of the pressure the deployer is under and the coordinate system used in developing the tracks. Deployers also may have difficulty optimising and using all the features of the environment in the scenario and there is a high cost for failure. Evolutionary Algorithms (EAs) were applied to this problem because of their ability to find good solutions when the problem is complex and not able to be solved analytically. In this work, we have shown that EAs can be successfully applied to generating tracks for torpedoes in the single torpedo single target scenarios. In these scenarios, the EA directs the torpedo to modify its behaviour to use the environment if appropriate and useful. Feedback from engineers working in the application domain suggests that the tracks generated are consistent with those a deployer would currently develop.

It is expected that EAs will also be useful when applied to multiple torpedo and/or multiple target engagements. These scenarios require tracks to be developed for torpedos to work together in acquiring a target rather than just optimising for their own track. This problem will be addressed in our future work.

6 Acknowledgements

The authors would like to acknowledge Raytheon Australia for their assistance and technical information for this project. The first author is partially supported by a studentship from Raytheon Australia.

References


