Online optimization approaches for fugitive interception

Irene S. van Droffelaar, Jan H. Kwakkel, Alexander Verbraeck

The fugitive interception problem asks, "what is the optimal set of routes for a fleet of police units to traverse in order to maximize the probability of intercepting a fleeing fugitive?". Models can be used to support the decision, as long as the optimal solution is calculated in near-real-time. Static optimization methods can be fast, but do not integrate new incoming information. If the police unit routes are optimized without taking into account the possibility of new information, the routes are, in actuality, not optimal.

A thought experiment quickly illustrates the optimality gap. Consider a network, where a policy unit and a fugitive start on opposite ends. Their respective nodes are connected by two paths of equal length. Assuming the fugitive wants to move away from its current location, there are two possible routes for the fugitive: through node 1 or node 5. The target nodes of the fugitive are nodes 4 and 8, and the fugitive is considered 'escaped' when they reach either. If the police does not have additional information on the preferred route of the fugitive, the probability of intercepting the fugitive is 50%.

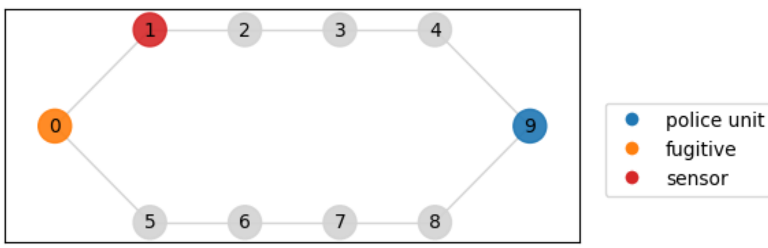

Figure 1. A 9-node test network

There are two possible approaches to utilize the new information: re-optimizing for the new situation when it occurs, or using decision triggers. The first approach, iterative optimization, recalculates the optimal route of the police unit when the fugitive is detected. Initially, the police unit takes the top or bottom path with equal probability. At t=1, the fugitive is detected, and the optimal path of the police unit is recalculated. If the computation time is 0, the police unit pivots and intercepts the fugitive on the edge on node 4. If the computation time is longer than 3 time steps, the fugitive escapes.

Alternatively, we can find tipping points (conditions under which a current approach ceases to meet its objectives), which are used to dynamically adapt the optimal sequence of actions. On the same network, depicted in Figure 1, the police unit should wait and go to node 4 or 8, depending on whether or not the sensor detects the fugitive. Another strategy that yields 100% probability of interception is: the police unit is sent to node 8, and pivots to node 4 if the sensor detects the fugitive. A tipping point approach has a higher initial computation cost, as multiple scenarios and adaptive strategies have to be evaluated. However, adaptation upon detection is instant.

In short, the iterative optimization approach and the tipping point approach consider incoming information in a different way. We compare the approaches, considering various graphs, number of units to be positioned, and number of sensors. The respective probabilities of interception are evaluated through a simulation of the computation time and optimal solution, over a wide range of initial positions.