A graph Patrol Problem with a random attacker and an observable patroller

Thomas Lowbridge
School of Mathematical Sciences
University of Nottingham

January 22, 2018

Contents

1 Introduction to a random attacker patroller game with observation

The model has a graph, $Q = (N, E)$, with a set of nodes labeled 1 to n, $N = \{1, ..., n\}$, and a set of edges linking these nodes. The adjacency matrix $a = (a_{i,j})_{i,j \in N}$, has $a_{i,j} = 1$ if i and j are adjacent and $a_{i,j} = 0$ if they are not adjacent. By definition we will use $a_{i,i} = 1 \quad \forall i \in N$.

An attacker has some attack time for node i, called X_i and chooses to attack node i with some probability, p_i. The attackers arrive according to some Poisson process with rate Λ, so by Poisson thinning they arrive at node i according to a Poisson process with rate $\lambda_i = \Lambda p_i$.

The patroller, uses some walk (with possible waiting) to patrol the graph.We assume that a patrollers walk is able to capture all attacks that have already begun, but not completed. But unlike the 'normal' setting the past unit time, the attackers do not start their attacks and instead will wait for the patroller to leave. Each missed attack at node i inccures a cost of c_i to the patroller.

We can formulate the state space, as the delineation of separate nodes. $\Omega = \{(\boldsymbol{s}, \boldsymbol{o}) = \quad | \quad s_i = 1, 2, ..., o_i = 0, 1, 2, ... \quad \forall i \in N\}$. Where $\boldsymbol{s} = (s_1, ..., s_n)$ has each s_i represent the number of time periods since the last visit for that node i and $\boldsymbol{o} = (o_1, ..., o_n)$ has each o_i represent the number of attackers present in the last time period when the node i was last visited (i.e The number of attackers known to be beginning their attack s_i time ago at node i).

The s_i increment by 1 if the node is not visited upon each action, or if the node is visited reset to $s_i = 1$. The o_i do not change for nodes not visited, when a node is visited, the 0_i 'reset' according to the Poisson distribution $Po(\lambda_i \times 1) = Po(\lambda)$. Due to $s_i = 1$ if and only if the patroller is currently at this node, we will use $l(\boldsymbol{s}) = \arg\min_{i \in N} s_i$ to represent the current node.

As the future of the process is independent of its past, the process can be formulated as a Markov decision process(MDP), where at the end of the period, the patroller chooses which adjacent node to visit. Thus the action space is $\mathcal{A} = \{j \,|\, a_{l(\boldsymbol{s}),j} = 1\}$, with a deterministic, stationary policy, $\pi : \Omega \to \mathcal{A}$.

The transitions of the MDP aren't entirely deterministic, $\boldsymbol{s}$ is purely deterministic, but $\boldsymbol{o}$ is partially probabilistic. In state $(\boldsymbol{s}, \boldsymbol{o})$ with the decision to visit node $i \in \mathcal{A}$, then the state will transition to $(\widetilde{\boldsymbol{s}}, \widetilde{\boldsymbol{o}})$ where $\widetilde{s}_j = s_j + 1$ if $j \neq i$ and $\widetilde{s}_j = 1$ if $j = i$ and $\widetilde{o}_j = o_j$ if $j \neq i$ and $o_j \sim Po(\lambda)$ if $j = i$.

To write down the cost function, which is dependent on the state $(\boldsymbol{s}, \boldsymbol{o})$ and the action to visit node i chosen, we will look at the expected cost of incurred at all nodes and sum these costs for the next time period.

$$C_j(\boldsymbol{s},\boldsymbol{o},i) = \begin{cases} c_j\lambda_j \int_0^{s_j} P(t-1 < X_j \leq t)dt + o_j P(0 < X_j \leq s_j) \text{ for } i \neq j \\ c_j\lambda_j \int_0^{s_j-1} P(t-1 < X_j \leq t)dt + o_j P(0 < X_j \leq s_j) \text{ for } i = j \end{cases}$$
$$= \begin{cases} c_j\lambda_j \int_{s_j-1}^{s_j} P(X_j \leq t)dt + o_j P(X_j \leq s_j) \text{ for } i \neq j \\ c_j\lambda_j \int_{s_j-2}^{s_j-1} P(X_j \leq t)dt + o_j P(X_j \leq s_j) \text{ for } i = j \end{cases} \tag{1}$$

With $C(\boldsymbol{s},\boldsymbol{0},i) = \sum_{j=1}^{n} C_j(\boldsymbol{s},\boldsymbol{o},i)$ being the cost function for the MDP.

We will now make the assumptions that X_j is bounded by B_j and that instead of using $Po(\lambda)$ for the observation transition and placing a bound on this Poisson distribution, named b_j, so we are now drawing from a truncated Poisson distribtion, henceforth called $TPo(\lambda, b_j)$. Then we can immediately say that the $o_j \leq b_j$ state is finite and the state s_j has the same cost function for $s_j \geq B_j + 2$ and hence we will restrict our space to this. So our modified transition is $\widetilde{s_j} = \min(s_j + 1, B_j + 2)$ if $j \neq i$ and $\widetilde{s_j} = 1$ if $i = j$. $\widetilde{o_j} = o_j$ if $i \neq j$ and $o_j \sim TPo(\lambda, b_j)$ if $i = j$.

Further reduction is possible as if $X_j \leq B_j$ then any observations o_j which started s_j time units ago is bound to have finished if $s_j \geq B_j + 1$. So our new state space is further reduced to having only $(\lfloor B_j \rfloor + 1, 0)$ when $s_j = \lfloor B_j \rfloor + 1$.

So $\Omega = \{(\boldsymbol{s},\boldsymbol{0}) | s_i = 1, 2, .., \lfloor B_i \rfloor + 1, o_i = 1, ..., b_i \,\forall i \in N\} \cup \{(\lfloor B_j \rfloor + 2, 0)\}$.

With further modified transitions that if $s_j = \lfloor B_j \rfloor + 1$ then $\widetilde{o_j} = 0$ for $i \neq j$.

Now our state space and action space are finite we need only consider deterministic, stationary policies. Applying such a policy generates a sequence of states under a given policy π, namely $\{\psi_\pi^k(\boldsymbol{s}_0, \boldsymbol{o}_0), k = 0, 1, 2, ...\}$. However we are not guaranteed to every have a regenerating process when the same node is visited due to the unpredictable nature of $o_i \sim TPo(\lambda, b_i)$. Unless $b_i = 0 \,\forall i \in N$ then we have removed the probabilistic nature of o_i's transition. We will not focus on the special case of $b_i = 0 \,\forall i \in N$ but it is shown how to develop a index for the single node problem in Appendix A.

Appendices

A Observations are always zero

On a single node we are limited to the state space of $\Omega = \{(1, 0), ..., (\lfloor B \rfloor + 2, 0)\}$, then on this state space we can implement a policy which returns every k time units, this will gives an average long run cost of

$$f(k) = \frac{c\lambda \int_0^{k-1} P(X \leq t)dt + \omega}{k} \tag{2}$$

So to find out when the patroller would be indifferent from choosing to return every k or every $k+1$, solve $f(k+1) - f(k) = 0$ giving

$$\frac{1}{k(k+1)}(c\lambda(k\int_{k-1}^{k} P(X \leq t)dt - \int_0^{k-1} P(X \leq t)dt) - \omega) = 0$$

Prompting an index of

$$W(k) = c\lambda(k\int_{k-1}^{k} P(X \leq t)dt - \int_0^{k-1} P(X \leq t)dt)$$

We note that $W(0) = 0$ and for $k \geq B+1$ $W(k) = c\lambda(k - int_0^{k+1} P(X \leq t)dt = c\lambda(1 + \int_0^{k-1} P(X > t)dt) = c\lambda(1 + E[X])$. We will now show that:

- $W(k)$ is non-decreasing
- The optimal policy when $\omega \in [W(k-1), W(k)]$ is to visit every k time units
- If $w \geq c\lambda(1 + E[X])$ then it is optimal to never visit

Proof. First

$$\begin{aligned}
W(k+1) - W(k) = c\lambda((k+1)\int_k^{k+1} P(X \leq t)dt - \int_0^k P(X \leq t)dt \\
- (k\int_{k-1}^{k} P(X \leq t)dt - \int_0^{k-1} P(X \leq t)dt)) \\
= c\lambda((k+1)\int_k^{k+1} P(X \leq t)dt - k\int_{k-1}^{k} P(X \leq t)dt - \int_k^{k-1} P(X \leq t)dt) \\
= c\lambda(k+1)(\int_k^{k+1} P(X \leq t)dt - \int_{k-1}^{k} P(X \leq t)dt) \geq 0
\end{aligned}$$

As $P(X \leq t)$ is non-decreasing.

Second if $\omega \in []$

$\square$