Game Theory, Robotics and the Pursuit-Evasion Problem

Kris Beevers
Rensselaer Polytechnic Institute
Algorithmic Robotics Laboratory
beevek@cs.rpi.edu

November 6, 2002

What is Game Theory?

- In a nutshell: multiperson decision-making [1]
- Psychologists would say: the theory of social situations [2]
- Two main branches [2]
  - *Cooperative*: formation of coalitions
  - *Non-cooperative*
- *Dynamic* game theory: the order in which decisions are made is important [1]

Non-cooperative Game Theory [1]

- Each person involved pursues his or her own (partly conflicting) interests

- A ‘game’:
  - ‘players’
  - ‘moves’
  - what players know (about moves of other players, the environment, etc.)
  - *payoffs* (both good and bad): depend on the *values* of the player

- Conflict situation: players value possible outcomes differently

Strategies [2]

- *Strategy*: fundamental notion in noncooperative game theory
- "Set of instructions that a player could give to a friend or program" (to play on their behalf)
- *Strategic form*: map from strategies to payoffs

Strategic Form: Prisoners' Dilemma

Player 1	Player 2	*confess*	*not confess*
confess		1,1	9,0
not confess		0,9	5,5

- Total payoff highest when neither confesses (5,5)

- *BUT* reasoning is as follows:
  - if other player doesn't confess, best for me to confess (9 instead of 5)
  - if other player confesses, also best to confess (1 instead of 0)
  - no matter what other player does, it's best for me to confess

- Game theory predicts each player will thus follow their own self-interests and confess

Zero- vs. Nonzero-sum Games [1]

- *Zero-sum* game: sum of cost functions of the players is zero
  - usually two players
  - constant-sum: transform to zero-sum
- *Nonzero-sum* games: sum of cost functions nonconstant
  - cooperation between two or more players may lead to mutual advantage

Nash Equilibrium

- Another example game [3]: 'Chicken' (nonzero-sum)
- Players *Johnny*, *Oscar*: both have option to escalate a brawl or give in
- Payoff matrix for Johnny:

Johnny	Oscar	*escalate*	*yield*
escalate		-10	1
yield		-1	0

  - by giving in, both can maximize their minimal payoff
  - *BUT*: both won't necessarily give in
    * if one guesses the other will give in, he will escalate
    * if both escalate, both are worse off

Nash Equilibrium (cont.)

- x, y: probabilities that J, O (resp.) escalate

- Expected payoff for J: $p_J = -10xy + x - y$
  - so, if O escalates with probability $> \frac{1}{10}$, J should yield
  - if O escalates with probability $< \frac{1}{10}$, J should escalate
  - if both J and O escalate with probability $= \frac{1}{10}$, they are in *Nash equilibrium*
    * neither has anything to gain by deviating from equilibrium

Game Theory: Applications in Robotics

- S. LaValle
  - *A game-theoretic framework for robot motion planning* (PhD thesis) [8]
    - motion planning under uncertainty in sensing and control
    - motion planning under environment uncertainties
    - multiple-robot motion planning (coordination)

- J. Hespanha, M. Prandini, S. Sastry
  - *Probabilistic Pursuit-Evasion Games: A One-Step Nash Approach* [6]

Pursuit-Evasion Games

- Several obvious applications in robotics/distributed robotics [4]
- ‘Degenerate’ cases (inanimate ‘opponents’)
  - obstacle avoidance
  - foraging/search-and-rescue
  - navigation
- Collective behaviors
  - following
  - flocking
  - aggregation
  - dispersion

Pursuit-Evasion and Game Theory [4]

- Problem first posed by Isaacs in 1950s [5]
- Considered extensively in aerial combat context (e.g. missiles)
- Different from previous games
  - continuous unfolding of moves, continuous variation in strategies
- Game theory can handle pursuit-evasion
  - optimal pursuit strategy depends on evasion strategy adopted by other player and vice-versa—just what game theory is good at
  - continuous nature modeled by differential equations
  - approach: pursuers minimize time to capture, evaders maximize time to capture

Probabilistic Pursuit-Evasion Games: A One-Step Nash Approach [6]

- Team of agents pursuing smart evader in non-accurately mapped terrain
- Integrates map-learning and pursuit
  - describes problem as a *partial-information Markov game* (nonzero sum)
- Finds Nash solution to the game
  - shows solution always exists
  - method to compute: reduce to an equivalent zero-sum matrix game

Notation

- n_p pursuers (called player U), single evader (player D)
- Pursuit region: finite collection of cells $\mathcal{X} = \{1, 2, \ldots, n_c\}$
- All events take place at a set of equally-spaced times $\mathcal{T} = \{1, 2, \ldots\}$
- Some cells may contain obstacles; configuration of obstacles not perfectly known
- Positions at time t: $\mathbf{x}_p^i(t)$ (pursuer i), $\mathbf{x}_e(t)$ (evader)
- Obstacle positions (fixed): $\mathbf{x}_o^i(t) = \mathbf{x}_o^i(t+1)\ \forall t \in \mathcal{T}$
- Game state at time t: $\mathbf{s}(t) = (\mathbf{x}_e(t), \mathbf{x}_p(t), \mathbf{x}_o(t)) \in \mathcal{S}$

Transitions

- Every time instant t each player can choose control actions from $\mathcal{U}, \mathcal{D}$, the sets of actions available to U, D resp.

- Next desired positions for pursuers, evader: $\mathbf{u}(t) \in \mathcal{U}, \mathbf{d}(t) \in \mathcal{D}$

- *Transition probability*: probability that next state will be $\mathbf{s}'(t) \in \mathcal{S}$ given $\mathbf{u}(t), \mathbf{d}(t)$
  - e.g., modelling uncertainty that an action will produce the desired outcome

- Set of cells reachable in one time step by an agent at x: $\mathcal{A}(x) \subseteq \mathcal{X}$

- Pursuers, evaders reach chosen adjacent cells with probability ρ_p, ρ_e

Observations

- A set of measurements is available to each player at every t:
  $\mathbf{Y}_t = \{y_0, y_1, \ldots, y_t\}, \mathbf{Z}_t = \{z_0, z_1, \ldots, z_t\}$ (for U, D resp.)

- $\mathcal{Y}, \mathcal{Z}$: *measurement space* for U, D resp. (finite sets); realizations of random variables $\mathbf{y}(t), \mathbf{z}(t)$

- Assume worst-case scenario: D has access to all information available to U
  - $\mathbf{Y}_t \subseteq \mathbf{Z}_t$

- *Game over set*:
  $\mathcal{S}_{over} = \{(x_e, x_p, x_o) \in \mathcal{S} | x_e = x_p^i \text{ for some } i \in \{1, \ldots, n_p\}\}$
  - both players can detect end of game

Stochastic Policies

- μ, δ: stochastic ‘policies’
  - each player selects action for time t according to some probability distribution (a ‘policy’)
- Probability measures vary with policies, so we denote them as $P_{\mu,\delta}$ (e.g. for a probability that depends on μ and δ)

Problem Formulation

- Pursuers/evader choose stochastic actions so as to maximize/minimize (resp.) probability of finishing game at next instant

- Consider:
  - $t \in \mathcal{T}, \mathbf{s}(t) \notin \mathcal{S}_{over}$
  - current measurements available to U, D are $Y \in \mathcal{Y}, Z \in \mathcal{Z}$ respectively
  - player U: select action $\mu(Y)$ to maximize $V_U(Y,t) = P_{\mu,\delta}(\mathbf{T}_{over} = t+1|Y)$
  - player D: select action $\delta(Z)$ to minimize $V_D(Z,t) = P_{\mu,\delta}(\mathbf{T}_{over} = t+1|Z)$

Since each player has a different set of information, the resulting game evolves through a succession of *nonzero-sum static games*

One-Step Nash Equilibrium Solution

- Cost functions:

$$
\begin{aligned}
J_D(p,q,Z) &= \sum_{u,d} p_u q_d(Z) \sum_{s' \in \mathcal{S}_{over}} p(s,s',u,d) \left(P_{\mu_{t-1},\delta_{t-1}}(\mathbf{s}(t) = s | \mathbf{Z}_t = Z)\right) \\
J_U(p,q) &= E_{\mu_{t-1},\delta_{t-1}} \left[J_D(p,q,\mathbf{Z}_t) | \mathbf{Y}_t = Y \right]
\end{aligned}
$$

  - $p(s,s',u,d)$ is a *transition probability function* (e.g. probability given s and actions u, d that next state will be s' at $t+1$)
  - p_u: scalar in distribution p over $\mathcal{U}$ corresponding to action u
  - $q_d(Z)$: similar to p_u, but takes into account Y (pursuer's information) since the evader knows it

- So, pursuers try to maximize estimate of evader's cost based on observations

Nash Equilibrium Solution (cont.)

- J_U, J_D represent cost functions optimized at time t by U and D
- Since each player's incurred cost depends on the other player's choice of moves, what exactly does "optimize a cost" mean?
- Well-known solution: Nash equilibrium
- Natural tendency for the game to be played at Nash equilibrium

Nash Equilibrium Solution (cont.)

- Players choose actions $\mu(Y), \delta(Z)$ equal to p^*, q^* satisfying

$$\begin{aligned} J_U(p^*, q^*) &\geq J_U(p, q^*) \; \forall p \\ J_D(p^*, q^*, Z) &\leq J_D(p^*, q, Z) \; \forall q \end{aligned}$$

- Pair (p^*, q^*) is called a *one-step Nash equilibrium*

Nash Equilibrium Solution (cont.)

- Note: in general, for nonzero-sum games there are multiple Nash equilibria corresponding to different values of costs

- However, we can reduce the pursuit-evasion problem to the determination of a Nash equilibrium for a fictitious zero-sum game with cost J_U
  - Then, it follows that all Nash pairs (p^*, q^*) are *interchangeable* and correspond to the same value for $J_U(p^*, q^*)$
  - We call this the *value of the game*

- Essentially, can do this because if persuer chooses p^*, a *rational evader* is 'forced' to choose q^*

Nash Equilibrium Solution (cont.)

- Pursuers (even though they have less information) can influence the best achievable value for $J_D(p^*, q, Z)$

- Paper shows that finding the Nash equilibrium for a one-step zero-sum game with cost J_U is equivalent to finding 'saddle-point equilibrium' for two-player zero-sum matrix game

- Reduces computation of stochastic policies to a Linear Programming problem

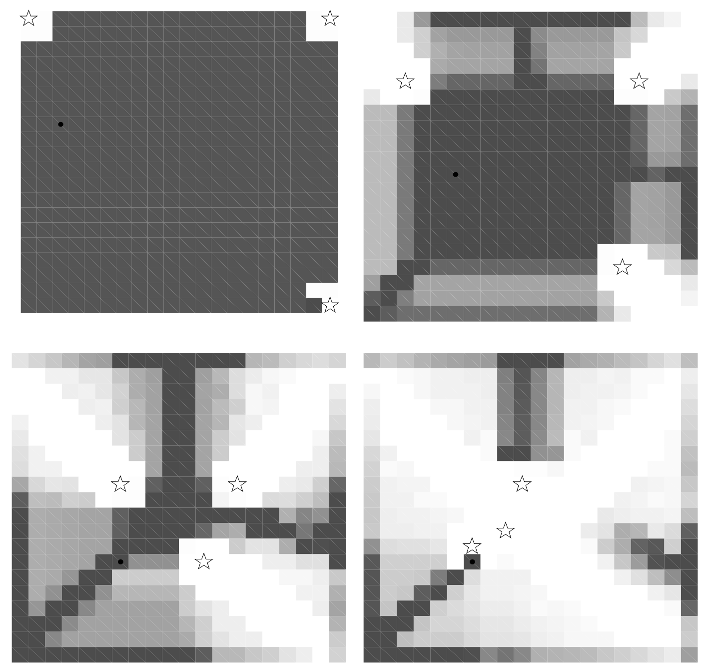
Example (Simulation)

- Pursuers
  - can perfectly determine position x
  - can perfectly sense adjacent cells $\mathcal{A}(x)$ for obstacles
  - senses for evaders
    - perfect sensing for cell pursuer is currently in
    - false positives (f_p) and false negatives (f_n) for $\mathcal{A}(x)$
- Evader
  - can perfectly determine position x
  - can perfectly sense adjacent cells $\mathcal{A}(x)$ for obstacles
  - knows pursuers' locations perfectly (because it has access to their measurement data)

Example (cont.)

- Parameters
  - $n_c = 400$ cells
  - $n_p = 3$ fast pursuers ($\rho_p = 1$) [light stars]
  - slow evader ($\rho_e = 0.5$) [dark circle]
  - $f_p = f_n = 0.01$
- Frames taken every four time steps

Example (cont.)



Some Problems With Game Theory and Robotics Applications

- Computation
  - this paper: $\approx 9n_p \times 9^4$ calculations per time 'instant'
  - LaValle/Hutchinson [7]: coordination problem solved with Nash equilibrium; 2-3 robots, up to an hour of computation

- 'Rationality' assumption
  - who's to say other players aren't *irrational*
  - modern game theory offers (among other approaches) *evolutionary game theory* [3]
  - still can't develop a strategy to deal with 'random' opponents [4]

References

1. T. Basar and G.J. Oldser. *Dynamic Noncooperative Game Theory*. Academic Press, 2nd edition, 1995.

2. D. Levine. *Game Theory*. `http://levine.sscnet.ucla.edu/general/cogsci.htm`.

3. J. Hofbauer and K. Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University press, 1998.

4. G.F. Miller and D.T. Cliff. *Co-evolution of pursuit and evasion I: Biological and game-theoretic foundations*. Technical Report CSRP311, University of Sussex, School of Cognitive and Computing Science, 1994.

5. Isaacs, R. *Games of pursuit*. Technical Report P-257, Rand Corporation, 1951.

6. J. Hespanha, M. Prandini and S. Sastry. *Probabilistic Pursuit-Evasion Games: A One-Step Nash Approach*. Submitted to the 39th Conf. on Decision and Contr., Mar. 2000.

7. S. M. LaValle and S. A. Hutchinson. *Path selection and coordination of multiple robots via Nash equilibria*. In Proc. 1994 IEEE Int'l Conf. Robot. & and Autom., pages 1847-1852, May 1994.

8. S. M. LaValle. *A game-theoretic framework for robot motion planning*. PhD thesis, University of Illinois, Urbana, IL, July 1995.