Fast Online Learning of Antijamming and Jamming Strategies

Y. Gwon, S. Dastangoo, C. Fossa, H. T. Kung

December 9, 2015

Presented at the 58th IEEE Global Communications Conference, San Diego, CA

LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

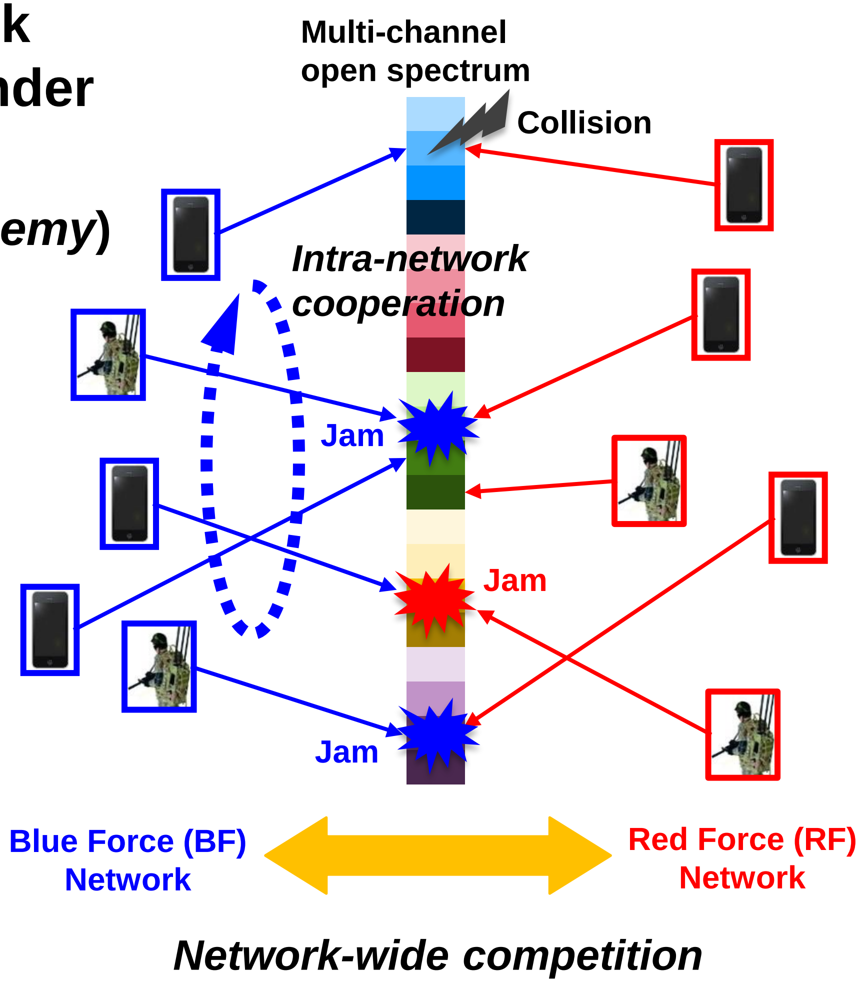
This work is sponsored by the Department of Defense under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the author and are not necessarily endorsed by the United States Government.

Outline

- **Introduction**
- **Background: Competing Cognitive Radio Network**
- **Problem**
- **Model**
- **Solution approaches**
- **Evaluation**
- **Conclusion**

Introduction

- **Competing Cognitive Radio Network (CCRN) models mobile networks under competition**
  - **Blue Force (*ally*) vs. Red Force (*enemy*)**
  - **Dynamic, open spectrum resource**
  - **Nodes are cognitive radios**
    - ➢ **Comm nodes and jammers**
  - **Opportunistic data access**
  - **Strategic jamming attacks**



Background: Competing Cognitive Radio Network

- **Formulation 1: Stochastic MAB**
  - **$<A_B, A_R, R>$**
    - **Blue-force (B) & Red-force (R) action sets:** **$a_B = \{a_{BC}, a_{BJ}\} \in A_B$, $a_R = \{a_{RC}, a_{RJ}\} \in A_R$**
    - **Reward: $R \sim \mathrm{PD}(r|a_B,a_R)$**
  - **Regret $\Gamma = \max_{a \in AB} \sum_T r(a) - \sum_T r(a_B{}^t)$**
    - **Optimal regret bound in $O(\log T)$ [Lai&Robbins'85]**

- **Formulation 2: Markov Game**
  - **$<A_B, A_R, \mathrm{S}, R, \mathrm{T}>$**
    - **Stateful model with states S and probabilistic transition function T**
  - **Strategy π: $S \longrightarrow \mathrm{PD}(A)$ is probability distribution over action space**
    - **Optimal strategy $\pi^* = \arg\max_\pi \mathrm{E}[\sum \gamma R(s,a_B,a_R)]$ can be computed by Q-learning via linear programming**

New Problem Formulation

- **Assume intelligent adversary**
  - **Hostile Red-force can learn as efficiently as Blue-force**
  - **Also, applies cognitive sensing to compute strategies**

- **Consequences**
  - **Well-behaved stochastic channel reward invalid ⇒ *time-varying* channel rewards**
    - **More difficult to predict or model**
  - **Nonstationarity in Red-force actions**
    - **Random, arbitrary changepoint ⇒ introduces dynamic changes**

Revised Regret Model

- **Stochastic MAB problems model regret Γ using reward function r(a)**
  - $\Gamma = \max_{a \in AB} \sum_T r(a) - \sum_T r(a_B^t)$

- **Using loss function l(a), we revise Γ**
  - **Revised regret Λ with loss function *l*(.)**

    $\Lambda = \sum l\,(a_B^t) - \min_{a \in AB} \sum l\,(a)$

- **Loss version is equivalent to reward version Γ**
  - **But provides *adversarial view* as if:**

    "Red-force alters potential loss for Blue-force over time, revealing only $l^t(a_B^t)$ at time t"

New Optimization Goals

- **Find best Blue-force action that minimizes Λ over time**

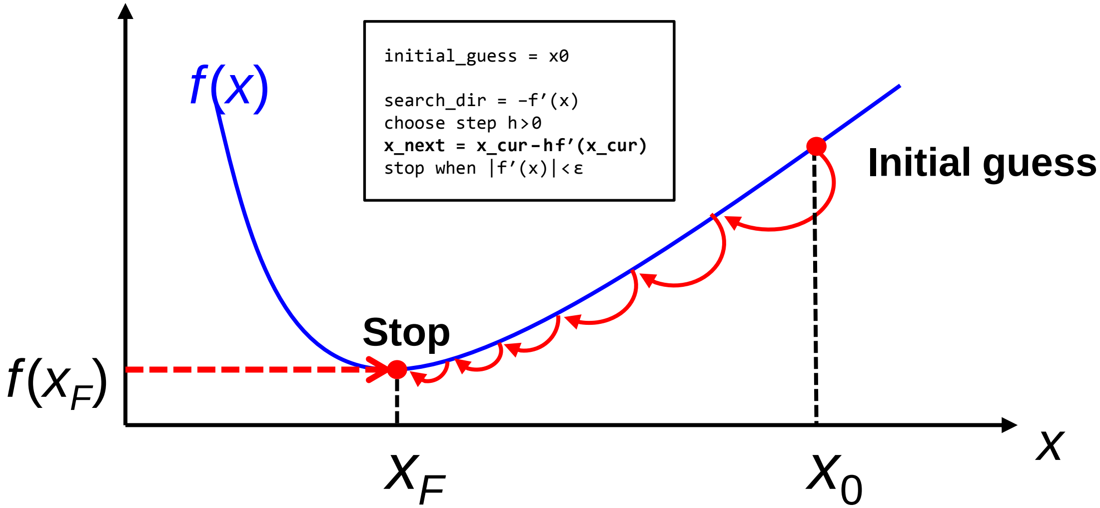
  $a^* = \arg\min_a \sum l^t(a_B{}^t) - \min_{a \in AB} \sum l^t(a)$

- **It's critical to estimate $l^t(.)$ accurately for new optimization**
  - **$l(.)$ evolves over t, and intelligent adversary makes it difficult to estimate**

Our Approach: Online Convex Optimization

- If $l^t(.) \in$ convex set, optimal regret bound can be achieved by online convex programming [Zinkevich'03]
  - Underlying idea is gradient descent/ascent

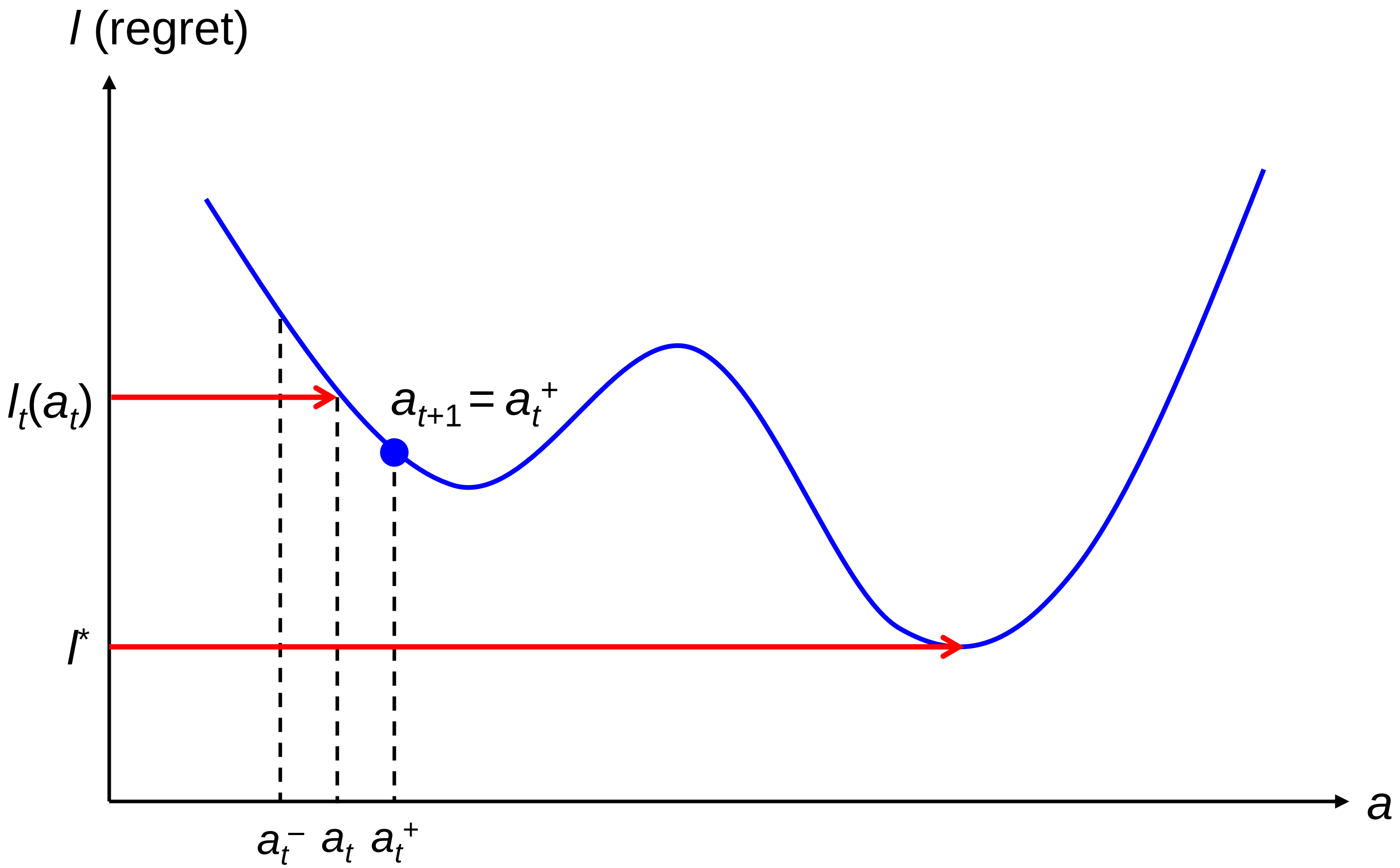
- What is gradient descent?
  - Find minima of loss by tracing estimated gradient (slope) of loss



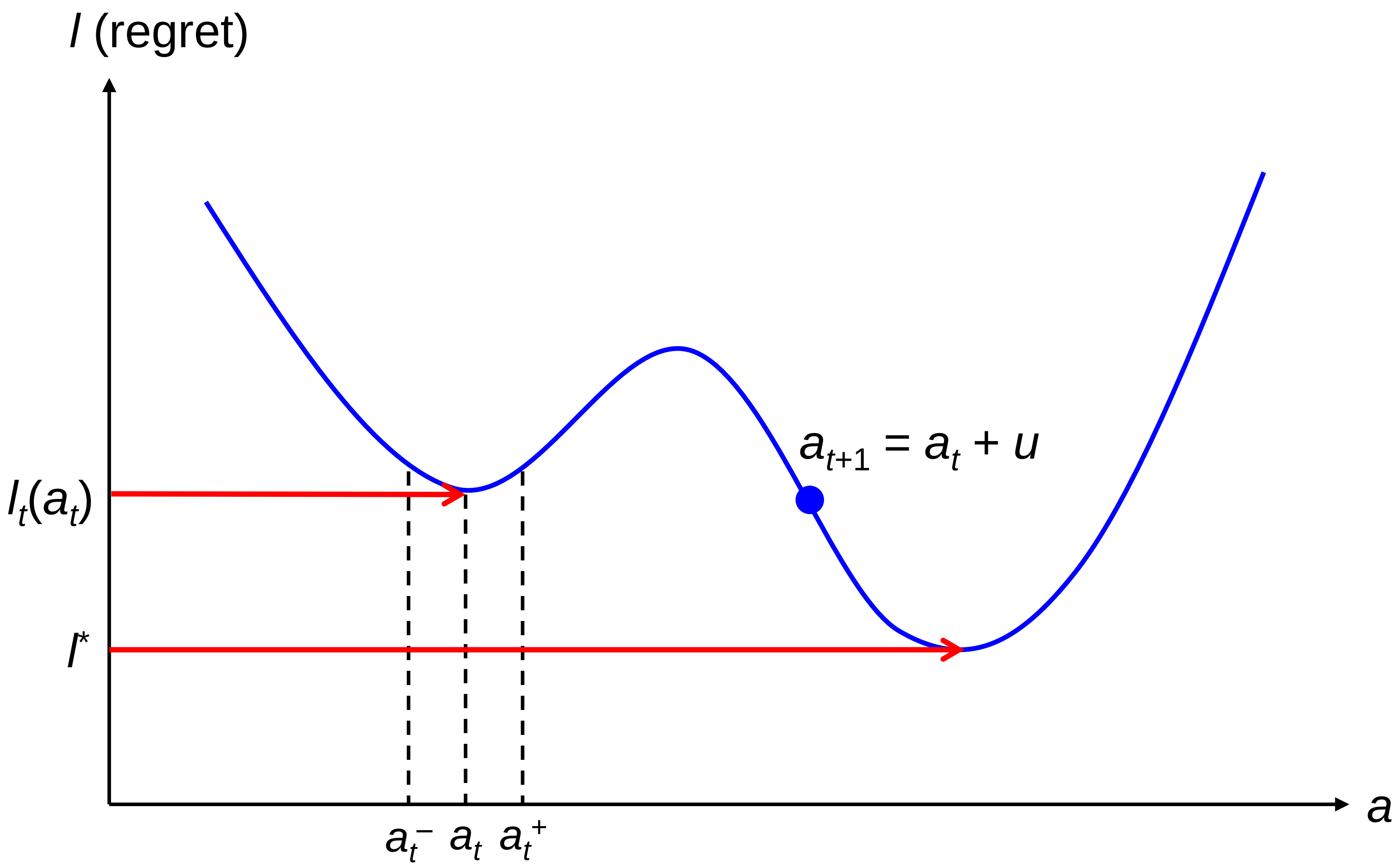
Our New Algorithm: Fast Online Learning

- **Sketch of key ideas**
  - **Estimate expected loss function for next time**
  - **Take gradient that leads to minimum loss iteratively**
  - **Test if reached minimum is global or local**
  - **When stuck at inefficiency (undesirable local min), use escape mechanism to get out**
  - **Go back and repeat until convergence**

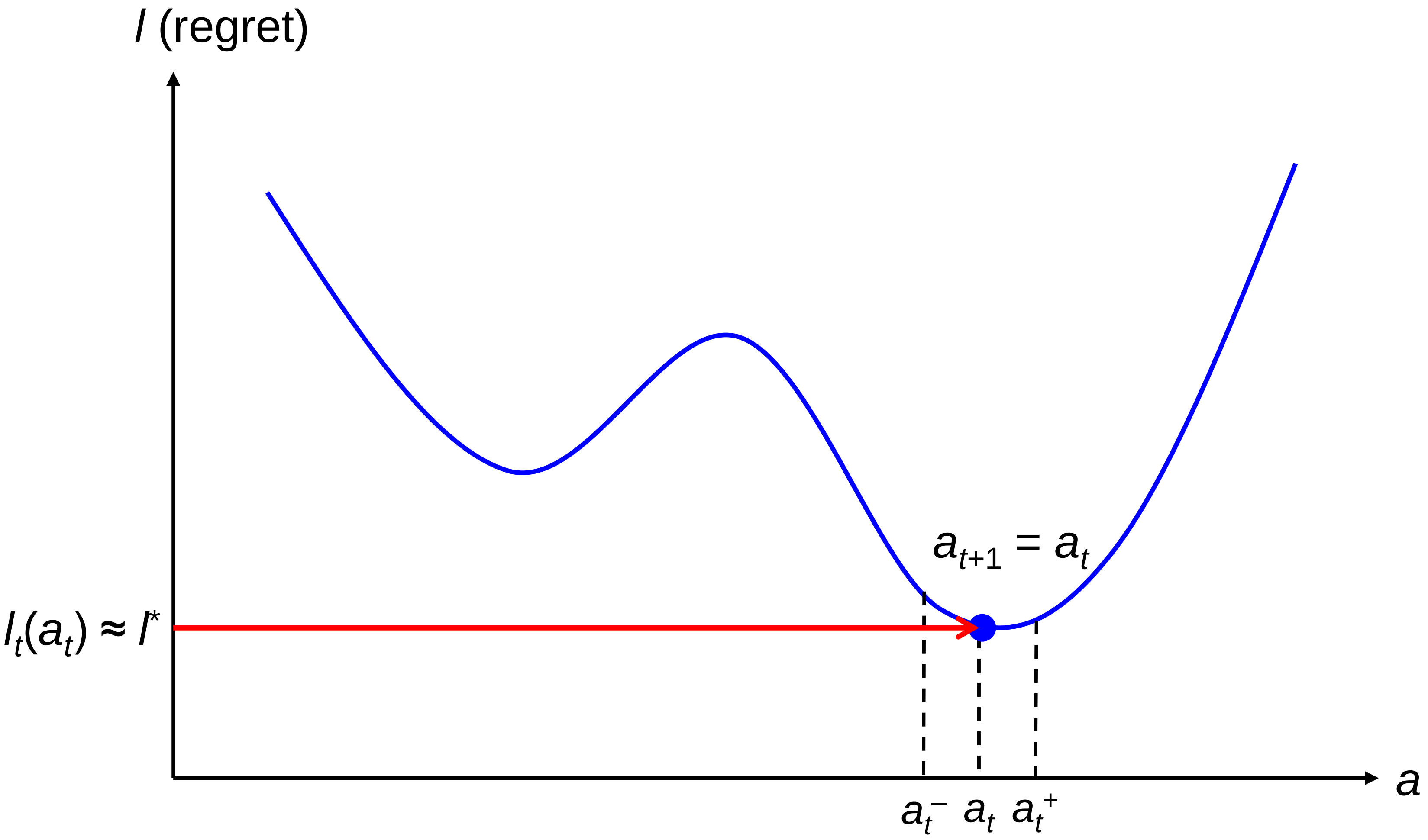
New Algorithm Explained (1)



New Algorithm Explained (2)



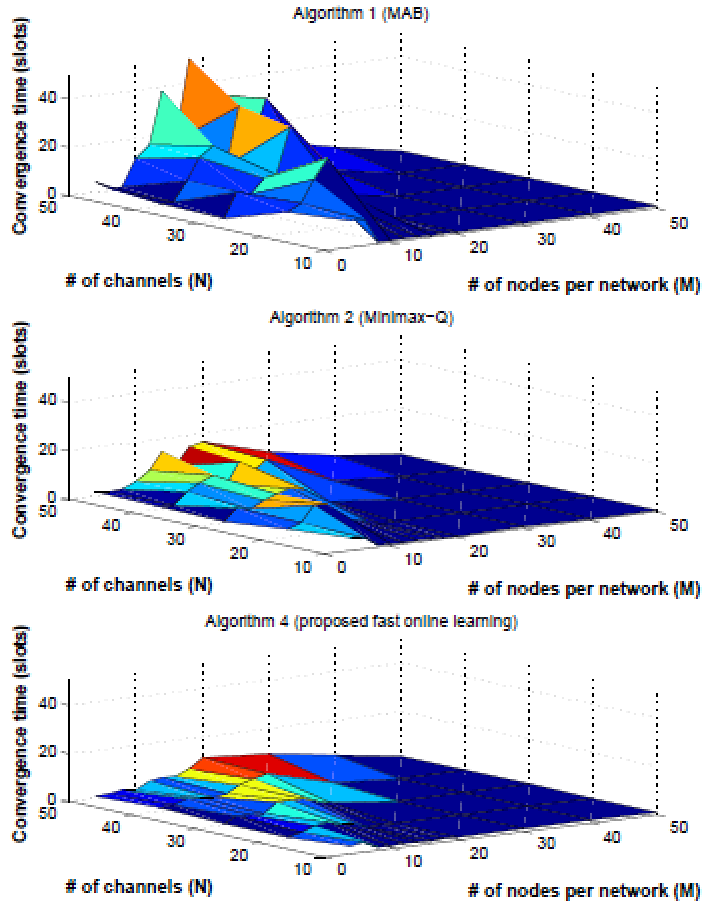
New Algorithm Explained (3)



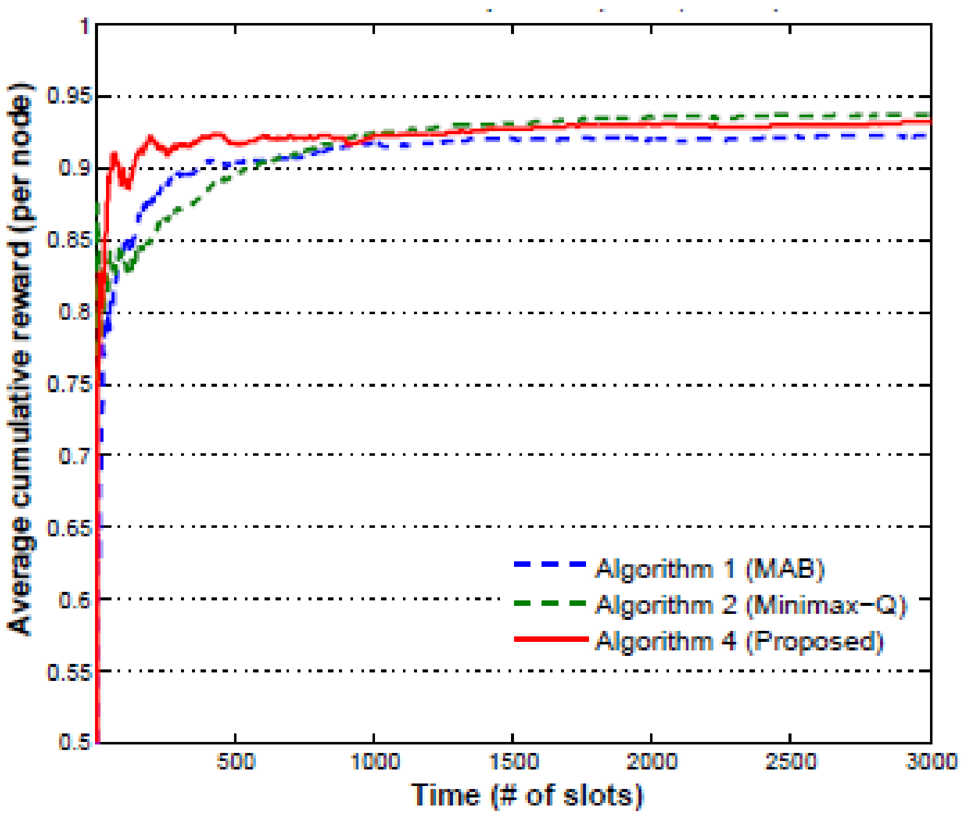
Evaluation

- Wrote custom simulator in MATLAB
  - Simulated spectrum with N = 10, 20, 30, 40, 50 channels
  - Varied number of nodes M = 10 to 50
    - Number of jammers in M total nodes varied 2 to 10
  - Simulation duration = 5,000 time slots
- Algorithms evaluated
  1. MAB (Blue-force) vs. random changepoint (Red-force)
  2. Minimax-Q (Blue-force) vs. random changepoint (Red-force)
  3. Proposed online (Blue-force) vs. random changepoint (Red-force)
- All algorithmic matchups in centralized control

Results: Convergence Time



Results: Average Reward Performance (N = 40, M = 20)



New algorithm finds optimal strategy much more rapidly than MAB and Q-learning based algorithms

Summary

- **Extended Competing Cognitive Radio Network (CCRN) to harder class of problems under nonstochastic assumptions**
  - **Random changepoints for enemy channel access & jamming strategies, time-varying channel reward**
- **Proposed new algorithm based on online convex programming**
  - **Simpler than MAB and Q-learning**
  - **Achieved much better convergence property**
  - **Finds optimal strategy faster**
- **Future work**
  - **Better channel activity prediction can help estimate more accurate loss function**

Support Materials

Proposed Algorithm

Algorithm 4 (CCRN online gradient descent learning)

choose a^1 randomly
while $t \geq 1$
  execute a^t and observe r^t
  compute $\hat{l}^t(a^t)$
  if $|l^* - \hat{l}^t(a^t)| < \epsilon$
    $a^{t+1} := a^t$
    continue
  end
  $a_-^t := a^t - \delta_-$ such that $\|a^t\|_0 = \left\|a_-^t\right\|_0$
  $a_+^t := a^t + \delta_+$ such that $\|a^t\|_0 = \left\|a_+^t\right\|_0$
  $\nabla \hat{l}^t := \min\{\hat{l}^t(a_-^t), \hat{l}^t(a_+^t)\}$
  if $\nabla \hat{l}^t < \hat{l}^t(a^t)$
    $a^{t+1} := \arg\min_{x \in \{a_-^t, a_+^t\}} \hat{l}^t(x)$
  else
    $a^{t+1} := a^t - w + u$
  end
end

Channel Activity Matrix, Outcome, Reward, State (1/2)

- **Example: there are two comm nodes and two jammers for each BF and RF network**
  - **BF uses channel 10 for control, RF channel 1**

- **At time *t*, actions are the following**
  - $A_B^t = \{a_{B,comm} = [7\ 3],\ a_{B,jam} = [1\ 5]\}$
    - $a_{B,comm} = [7\ 3]$ **means BF comm node 1 transmit at channel 7, and comm node at 2 channel 3**
  - $A_R^t = \{a_{R,comm} = [3\ 5],\ a_{B,jam} = [10\ 9]\}$

- **How to figure out channel outcomes, compute rewards, and determine state?**
  - **Channel Activity Matrix**

Channel Activity Matrix, Outcome, Reward, State (2/2)

CH	Blue Force		Red Force		Outcome	Reward	
	Comm	Jammer	Comm	Jammer		BF	RF
1	–	Jam	–	–	BF jamming success	+1	0
3	Tx	–	Tx	–	BF & RF comms collide	0	0
5	–	Jam	Tx	–	BF jamming success	+1	0
7	Tx	–	–	–	BF comm Tx success	+1	0
9	–	–	–	Jam	RF jamming fail	0	0
10	–	–	–	Jam	RF jamming success	0	+1