Bandit networks

Charles Auguste & Louis Trezzini

January 24, 2019

Goals of the project

Review the article “Stochastic bandits on a social network: Collaborative learning with local information sharing” (Kolla, Jagannathan, and Gopalan)

- Understand the proposed framework and algorithms, along with their theoretical guarantees
- Implement them in C++ to speed up and parallelize the computations
- Reproduce the experimental results
- Pinpoint the limitations of the model and try to improve it

Multi-agent stochastic multi-armed bandit (MAB) problem

- Undirected graph $G = (V, E)$ with $|V| = m$ users
- All users are playing the same MAB problem with K arms
- A user v can observe the actions and the respective rewards of itself and its one-hop neighbors up to round t, before deciding the action for round $t + 1$
- The objective is regret minimization

Upper-Confidence-Bound-Network (UCB-Network) policy

Algorithm 1 Upper-Confidence-Bound-Network (UCB-Network)

Each user in G follows UCB-user policy

UCB-user policy for a user v:

Initialization: For $1 \leq t \leq K$

- play arm t

Loop: For $K \leq t \leq n$

- $a^v(t+1) = \underset{j}{\text{argmax}}\ \hat{\mu}_{m_j^v(t)} + \sqrt{\frac{2 \ln t}{m_j^v(t)}}$

- $m_i^v(t)$: number of times arm i has been chosen by node v and its one-hop neighbors up to round t
- $\hat{\mu}_{m_i^v(t)}$: average reward for playing arm i obtained by node v and its one-hop neighbors up to round t

Follow Your Leader (FYL) policy

Algorithm 2 Follow Your Leader (FYL) policy

Input: Graph G, a dominating set D and a dominating set partition

Leader - **Each node in** D**:**

Follows the UCB-user policy by using the samples of itself and its neighbors

Follower - **Each node in** $V \setminus D$**:**

In round $t = 1$:

- Chooses an action randomly from $\mathcal{K}$

In round $t > 1$:

- Chooses the action taken by the leader in its component, in the previous round $(t - 1)$

Issue with the FYL policy

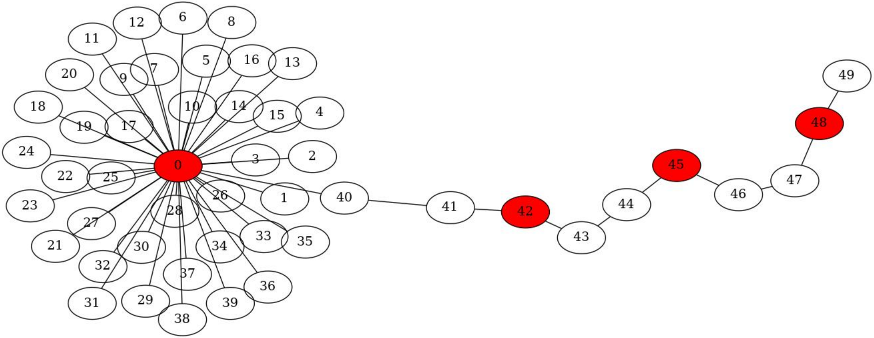

Figure 1: Star-chain graph, with optimal dominating set in red

Nodes 41-49 are missing on a lot of information !

Follow Best Informed (FBI) policy

- FYL policy is myopic
- In addition to their previous action, nodes can output the number of samples (information) they used to compute it
- Nodes can follow their best informed neighbor and use UCB-policy if they are better informed
- Actually, the structure of a graph fully determines the behavior of the nodes (but not their precise actions obviously)

Example of the usefulness of the FBI policy

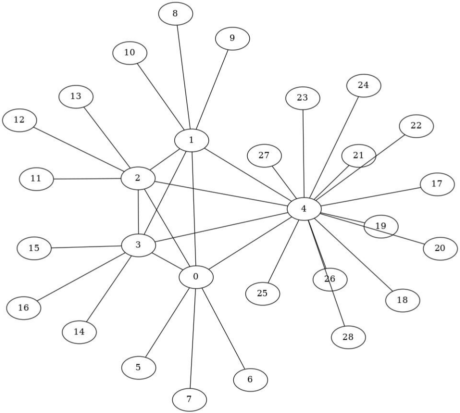

Figure 2: Fully connected stars graph

After first iteration, node 4 has the most information. It can pass it to nodes 0-3, who will then pass it to their children.

Results for a fully connected stars graph

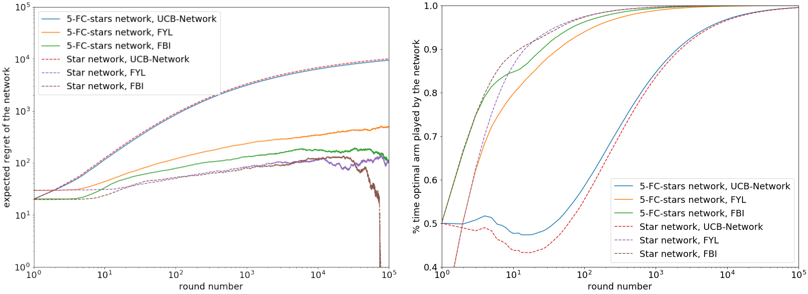

Figure 3: Performance comparison of UCB-Network, FYL, and FBI policies on a 100-nodes star network and on the 100-nodes 5-FC-stars network: 2 arms, Bernoulli rewards with means 0.5 and 0.7 (1000 sample paths).

Results for a star-chain graph

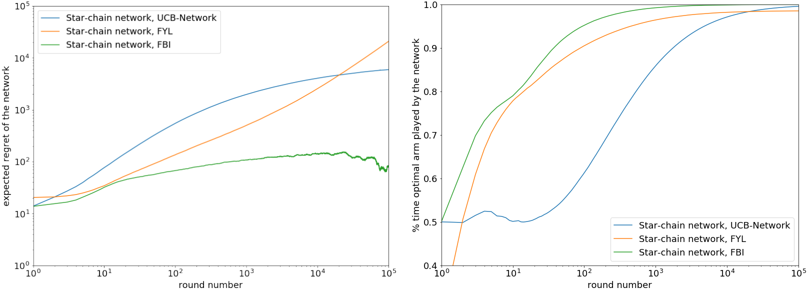

Figure 4: Performance comparison of FYL and FBI policies on the pathological graph structure (star graph with 70 nodes, among which a 20-nodes long chain): 2 arms, Bernoulli rewards with means 0.5 and 0.7 (1000 sample paths).

A deeper look at the FBI policy

- **Downside:** If one node has more information than the rest, every node is going to follow it (at a delayed rate) $\Rightarrow$ Strong correlation in the nodes actions
- **Further improvements:** it may be smart to randomly follow one with a probability depending on its amount of information. But then the behavior of the nodes is not determined by the structure of the graph...

Conclusion

- UCB-user policy is very useful to study a bandit problem on a graph...
- ... But the naive UCB-network policy is not very efficient!
- For simple policies, interesting bounds can be computed
- More general types of graphs provide hints to build more robust strategies

Any Questions ?

References

Ravi Kumar Kolla, Krishna P. Jagannathan, and Aditya Gopalan. "Stochastic bandits on a social network: Collaborative learning with local information sharing". In: *CoRR* abs/1602.08886 (2016). arXiv: 1602.08886. URL: http://arxiv.org/abs/1602.08886.