# OPTIMAL ATTACK AND DEFENCE OF LARGE SCALE NETWORKS USING MEAN FIELD THEORY

*Jorge del Val, Santiago Zazo, Sergio Valcarcel Macua, Javier Zazo, Juan Parras*

Universidad Politécnica de Madrid

## ABSTRACT

We address the issue of large scale network security. It is known that traditional game theory becomes intractable when considering a large number of players, which is a realistic situation in today's networks where a centralized administration is not available. We propose a new model, based on mean field theory, that allows us to obtain optimal decentralised defence policy for any node in the network and optimal attack policy for an attacker. In this way we settle a promising framework for the development of a mean field game theory of large scale network security. We also present a case study with experimental results.

***Index Terms—*** Dynamic programming, game theory, mean field, network, optimal control, security

## 1. INTRODUCTION

Network security has been a prominent research field in the last years, as networks continue to gain importance in today's society. Useful frameworks such as game theory have been applied successfully to this issue [1, 2, 3], by considering a game with just two players: a single attacker and a single defender. Nevertheless, such model becomes limited for large networks. A more realistic model is to consider every node in the network as a player. However, the problem becomes intractable as the network scales in size.

We propose a mean field approach where every node is a player, but they do not have to take into account the state of every other individual node, just some aggregate state called *mass function*. This is similar to the situation where a predator wants to catch a shoal of fish. In such case, each fish may not take into account every other fish in the shoal to choose it's action, but just the aggregate of the whole shoal. This consideration can be used to simplify a game formed by a large number of players and transform it into a game of two agents, the fish and the shoal [4, 5]. In our case the whole network behaves in a way as a shoal of fish, where the hacker is the predator. The mean field approach has been successfully applied to fields such as energy or interference management [6, 7], but to our knowledge very little research has been done linking dynamic mean field theory to network security. Also, most of the current theoretical research is based on continuous frameworks, despite the potential of discrete mean field games for multiple applications. Reference [8] proposes a model based on mean field games for mobile ad-hoc networks. We provide a different approach to the security problem in large scale networks, where the incentives are based on network topology rather than information assets and energy. In addition, as we will see in Section 3, we look for solutions in a *stationary* state. This allows us to relax the assumptions regarding network observability, which is not realistic in large scale networks.

Our main theoretical contributions are as follows: First, we introduce a new model for large scale network security. Secondly, we apply a mean field approximation to overcome the intractability for a high number of nodes. Next, we show how the network can find the optimal response for a given attack in the mean field framework, as well as how the attacker can find an optimal response for a given defence policy. Lastly, we solve the problem numerically for an example case.

In section 2 we define the system model, i.e., the components, incentives and dynamics of both the attacker and defender. In section 3 we propose a mean field approximation of the system. In section 4 we calculate the best response functions in the stationary regime with dynamic programming methods. In section 5, an example case is studied and solved.

## 2. SYSTEM MODEL

We consider a large network represented with a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{E}$ is the set of edges and $\mathcal{V}$ is the set of nodes with large $|\mathcal{V}| = N$. Each node $i \in \mathcal{V}$ has a degree $k^i$, which is the number of neighbour nodes to which it is connected. The network has a degree distribution $G(k)$, which is the proportion of nodes with degree $k$ in the network.

We define the importance of each node to the network as an increasing function of its connectivity degree $I(k)$, since hacking nodes with more connections generally do more harm to the network. Each node has a discrete security state level $x^i \in \mathcal{X} = \{0, ..., N_x\}$ that evolves stochastically but can be controlled with maintenance.

We represent the attacker with superscript 0 and the defender with superscript $i$, where $i = 1, ..., N$.

### 2.1. Attacker

At every time step $t$ the attacker chooses the action pair $(k_t^0, a_t^0) \in \mathcal{K} \times A^0$, where $\mathcal{K}$ is the set of degrees of the network, and $A^0 = \{A_1, ..., A_L\}$ is the set of available attacks to the attacker (such as hack ftp server, brute force attack, etc).

This work was supported in part by the Spanish Ministry of Science and Innovation under the grant TEC2013-46011-C3-1-R (UnderWorld), the COMONSENS Network of Excellence TEC2015-69648-REDC and by an FPU doctoral grant to the fourth author.

978-0-9928-6265-7/16/$31.00 ©2016 IEEE

When the attacker has chosen degree $k_t^0$ and action $a_t^0$, some node $i$ is drawn randomly from the nodes of the network with that degree. The failure of the attack at time $t$ is modelled by a Bernoulli random variable $f_t \sim Ber(p(\cdot|a_t^0, x_t^i))$, where $x_t^i$ is the security level of the attacked node at time $t$ and where $p$ is the probability that $f_t = 1$. Note that the distribution of $f_t$ depends on both the action of the attacker and the security level of the defender, modelling the compromise between attack strength and defence level. The success of the attack at time $t$ is denoted by $s_t = 1 - f_t$. Since the node is chosen randomly by the attacker, the security level is also a random variable $X$. We define $m_t(x, k)$ as the *mass function* on time $t$, representing the proportion of nodes of degree $k$ and security level $x$ in the network. By defining also $m_t(x|k)$ as the proportion of nodes in the level $x$ of those with degree $k$, we easily obtain $m_t(x, k) = m_t(x|k)G(k)$. Now we introduce the probability of $f_t$ taking a specific value given the actions of the attacker and the network configuration:

$$p(f_t|a_t^0, k_t^0; m_t) = \sum_{x\in\mathcal{X}} p(f_t|a_t^0, x)p(X = x|k_t^0) = \\ = \sum_{x\in\mathcal{X}} p(f_t|a_t^0, x)m_t(x|k_t^0) \quad (1)$$

The attacker should be discouraged to make strong attacks on important nodes repeatedly, since if it fails then the attacker gains notoriety so that it is more likely to be caught. Therefore, we induce caution in the attacker. Define the state of the attacker as the notoriety level denoted by $q$. When an attack fails, the attacker gains $n(a_t^0, k_t^0)$ units of notoriety. Here, $n(a_t^0, k_t^0)$ is an increasing function of $k_t^0$, since failing an attack in important nodes results in more notoriety. The dynamics for $q_t$ are

$$q_{t+1} = (1 - \alpha)q_t + n(a_t^0, k_t^0)f_t \quad (2)$$

The $(1-\alpha)q_t$ term is a forgetting factor, which makes the dynamics stable and determine the time of recovery if the attack fails.

The attacker aims to minimize its notoriety and the cost $c(a_t)$ associated to its actions. It also aims to maximize the number of successful attacks, which are weighted by the importance of the hacked objective $I(k)$. Finally, we include an *entropy penalization* term $\epsilon$, meaning that it is willing to sacrifice some of its reward in order to become more unpredictable. To do this, the attacker plays on each state $q_t$ a *mixed strategy*, i.e., probability distribution over the space of actions $\pi^0(a_t^0, k_t^0|q_t)$, which we refer as the *attack policy*. Note that $a_t^0$ and $k_t^0$ are now random variables distributed according to $\pi^0$. Thus, we propose the following cost function:

$$J^0(q, \pi^0, m_t) = \mathbb{E}\left\{\sum_{t=0}^{\infty} \beta^t L^0(q_t, a_t^0, k_t^0, \pi^0, s_t)|q_0 = q\right\} \quad (3)$$

where

$$L^0(q_t, a_t^0, k_t^0, \pi^0, s_t) = w_1 q_t^2 - w_2 I(k_t^0)s_t + w_3 c(a_t^0) \quad (4)$$
$$+ \epsilon \log(\pi^0(a_t^0, k_t^0|q_t)) \quad (5)$$

is the *running cost* and the expectation is taken over $s_t \sim Ber(1 - p(\cdot|a_t^0, k_t^0; m_t))$ and over $(a_t^0, k_t^0) \sim \pi^0(a_t^0, k_t^0|q_t)$. Here, $w_1$, $w_2$, and $w_3$ are weighting parameters.

### 2.2. Defender

Each node in the network has the ability to maintain its security level at discrete steps. This level is subject to stochastic deterioration (e.g. new vulnerabilities are found). Every node $i$ can choose an action $a^i$ from $\mathcal{A} = \{0, 1\}$, where 1 means "update" and 0 "not update". He can also *randomize* its actions according to some set of probabilities[1], extending the space of actions to $\mathcal{A}^r = \{p_1, ..., p_D\}$, where each action means that it can randomly update the system with probability $p_d$ or do nothing with probability $1 - p_d$. The nodes' objective is to find a *defence policy* $\mu^i : \mathcal{X} \times \mathcal{K} \to \mathcal{A}^r$ that maps current state to optimal action. We assume all nodes have identical incentives and dynamics. Thus, they solve the same problem and the optimal policy will be the same for all of them, i.e., $\mu^i(x, k) = \mu(x, k) \quad \forall i \in \mathcal{V}$.

The dynamics for the network nodes are modelled with a transition probability matrix $P_k^\mu$ for each degree $k$, with entries $ij$ given by:

$$P_{ij,k}^\mu = P(X_{t+1} = j|X_t = i; \mu(i, k), k) \quad (6)$$

Note that the transition probabilities depend on the policy $\mu$. Each row of the matrix corresponds to the transition probabilities from the state $i$ while choosing the action $\mu(i, k)$, so that $P_k^\mu$ is a row stochastic matrix. The transition probabilities are known for the case of "update", denoted by $\mu(i, k) = 1$, and "not update", denoted by $\mu(i, k) = 0$. If a defender of degree $k$ chooses some probability of updating the system $\mu(i, k) = p_d \in \mathcal{A}^r$, the transition probabilities can be obtained with the law of total probability:

$$P_{ij,k}^\mu = p_d P(X_{t+1} = j|X_t = i; \mu(i, k) = 1, k) \\ + (1 - p_d)P(X_{t+1} = j|X_t = i; \mu(i, k) = 0, k) \quad (7)$$

The defender aims to minimize a cost term associated to the updating of the system, denoted by $ca_t^i$ where $c$ is some cost parameter. It will also aim to minimize the number successful attacks to the network, weighted by the importance of each attacked node. This last term couples the optimization process for all nodes in the network, settling a collaborative framework.

We define the random variable $s_t^i$ to model the event that the attacker draws node $i$ as its objective and the attack is successful. We propose the following cost function for the defender

$$J(x^i, x^{-i}, \mu, \pi^0) = \mathbb{E}\left\{\sum_{t=1}^{\infty} \beta^t L(a_t^i, s_t^i, s_t^{-i})|x_0^i = x^i, \forall i\right\} \quad (8)$$

where

$$L(a_t^i, s_t^i, s_t^{-i}) = ca_t^i + l\sum_{i=1}^{N} I(k^i)s_t^i \quad (9)$$

and the expectation is taken over $s_t^i \sim Ber(1 - p(\cdot|a_t^0, x_t^i))$, $(a_t^0, k^i) \sim \pi^0(a^0, k^i|q)$, and $a_t^i \sim Ber(\cdot|\mu(x, k))$. Here $l$

[1]The discretization of the probabilities is made for computational tractability, and the continuous version will be addressed in future work

is a weighting parameter, $x^{-i}$ represents the state of all the players other than $i$ and $s_t^i$ models the event that the hacker attacks node $i$ successfully.

## 3. MEAN FIELD APPROXIMATION

In this section we derive the mean field approximation of the cost in equation (8) to the form $J(\mu, m, \pi^0)$, where $m$ is the initial mass function of the network.

We look for solutions defined on *stationary* states, i.e., states where the system does not evolve with time. For a given policy, those states are reached asymptotically as $t \to \infty$ independently of the initial conditions, since the system is ergodic due to the Markov-like evolution of the mass and the mean field limit, as discussed in section 3.2. This allows any player to predict the long-term mass function for a given policy without the need to measure the actual state of the network.

### 3.1. Defender approximation

To introduce a mean field approximation for the nodes in the network (i.e., defenders), we take the expected value of the aggregative term in (9). We also define the random variable $I_t^0$ as the node of degree $k_t^0$ drawn by the attacker on time $t$:

$$\sum_{i=1}^{N} I(k^i)\mathbb{E}\{s_t^i\} =$$
$$\sum_{i=1}^{N} I(k^i) \sum_{a^0 \in \mathcal{A}^0} p(G|a^0, x_t^i) p(I_t^0 = i|k^i)\pi^0(a^0, k^i|q_t) \quad (10)$$

Noting that $p(I_t^0|k_i) = \frac{1}{N}\frac{1}{G(k_i)}$, we can simplify equation (10) into:

$$\sum_{a^0 \in \mathcal{A}^0} \frac{1}{N} \sum_{i=1}^{N} I(k^i) p(G|a^0, x^i) \frac{\pi^0(a^0, k^i|q)}{G(k^i)} \quad (11)$$

Now, using the definition of mass function we can write equation (11) in terms of it as:

$$g(m, \pi^0|q) = \sum_{i=1}^{N} I(k^i)\mathbb{E}\{s_t^i\} =$$
$$\sum_{a^0 \in \mathcal{A}^0} \sum_{x \in \mathcal{X}} \sum_{k \in \mathcal{K}} I(k) p(G|a^0, x)\pi^0(a^0, k|q)\frac{m(x,k)}{G(k)} \quad (12)$$

However, the defender does not now exactly in which state the attacker is to evaluate $g(m, \pi^0|q)$. Recalling the fact that we look for stationary solutions, we can obtain a stationary probability density function $f(q)$ from equation (2). One way to approximate it is to discretize $q$ in the points $q_i$ for $i = 1, ..., N_q$, and then obtain an approximation of the transition probabilities given the policy $\pi^0$ from the point $q_i$ to the point $q_j$. Finally, we obtain the stationary distribution $f(q_i)$ of the underlying Markov chain and define

$$g(m, \pi^0) \approx \sum_{i=1}^{N_q} f(q_i) g(m, \pi^0|q_i) \quad (13)$$

Since the resulting cost function of the defender is the same for every node, we can drop the superscript and consider the cost for a generic node.

### 3.2. Mass evolution

As we have discussed, the evolution of the state of each player is stochastic, and consequently the evolution of the mass can not be accurately predicted in a general case. However, as $N \to \infty$ it becomes *deterministic* due to the law of large numbers, so that the system as a whole is predictable. This is the so called mean field limit.

Now we approximate the evolution of the mass function, which depends on the defence policy of the network, using the transition probabilities:

$$m(x|k)_{t+1} = \sum_j P(X_{t+1} = x|X_t = j, \mu, k) m(j|k)_t \quad (14)$$

which can be expressed more compactly in vector form for every security level state:

$$m_{t+1}^k = P_k^{\mu T} m_t^k \quad (15)$$

where $m_t^k \in \mathbb{R}^{N_x}$ is the conditional mass function vector for the degree $k$ in the time $t$.

Because we look for stationary solutions, the conditional mass function for each $k$ is the right eigenvector of $P_k^{\mu T}$ associated to the simple eigenvalue 1. Since $P_k^{\mu}$ is a row stochastic matrix, by Perron-Frobenious theory the first eigenvector is unique and can be obtained by iteration of (15) until convergence. Since the stationary mass only depends on the defenders' policy $\mu$, we define the mapping from the space of policies to the space of mass functions as $m_\mu$, so that its $k_{th}$ column is $[m_\mu]_k = m_\mu^k = \lim_{t\to\infty} m_t^k$.

### 3.3. Outline and final cost

With this mean field approximation, the cumulative cost of a generic defender becomes:

$$J(x, \mu, \pi^0) = \mathbb{E}\left\{\sum_{t=1}^{\infty} \beta^t L(a_t, m_\mu, \pi^0)|x_0 = x\right\} \quad (16)$$

where

$$L(a_t, m_\mu, \pi^0) = ca_t + lg(m_\mu, \pi^0) \quad (17)$$

and the expectation is taken over $a_t^i \sim Ber(\cdot|\mu(x,k))$.

We remark that now the cost in (16) not depend on all the network nodes, but just on the mass function and the defender policy. Thus, we have transformed the original problem to a tractable one whose complexity does not increase with the scale of the network.

## 4. BEST RESPONSE FUNCTIONS

### 4.1. Optimal defense

The optimal defense of the network given a known attack policy $\pi^0(a^0, k^0|q)$ is the solution to the following problem

$$BR(x, k, \pi^0) = \arg\min_{\mu(x,k)} \{J(x, \mu, \pi^0)\} \quad (18)$$

which can obtained by solving the Bellman equation:

$$V(x,k,\pi^0) = \min_{\mu(x,k)} \left\{ \mathbb{E}\{L(a, m_\mu, \pi^0) + \beta V(x',k)\} \right\} \tag{19}$$

where $V(x,k)$ is the *value function*. The expectation is taken over the next state $x'$, given that the actual state is $x$. In order to solve (19), we use a fixed point iteration known as *value iteration* [9].

For the sake of clarity, we drop the dependence of $k$, since the problem for each $k$ can be solved independently. We denote the iteration as $z$ and define $[\mu(x), \mu^z]$ as the vector $(\mu^z(x_1), ..., \mu(x), ..., \mu^z(x_{N_x}))$. Taking the expectation in (19) the fixed point iteration becomes:

$$V^{z+1}(x) = \min_{\mu(x)} \left\{ c\mu(x) + g(m_{[\mu(x),\mu^z]}, \pi^0) + \beta \sum_j P_{x,j}^{\mu(x)} V^z(j) \right\} \tag{20}$$

A detailed convergence analysis of value iteration can be found on [10]. However, the convergence behaviour including terms depending on $[\mu(x), \mu^z]$ is beyond the scope of this paper.

### 4.2. Optimal attack

Given a defence policy, and thus a network mass function, the problem of the attacker is:

$$BR^0(q,\mu) = \arg\min_{\pi^0} \{J^0(q, \pi^0, m_\mu)\} \tag{21}$$

This problem can be solved in a similar manner to the defender, solving the Bellman equation for the attacker:

$$V(q) = \min_{\pi^0} \left\{ \mathbb{E}\{Q(q, a^0, k^0, V) + \epsilon \log(\pi^0(a^0, k^0))\} \right\} \tag{22}$$

where the expectation is taken over $(a^0, k^0) \sim \pi^0$. Here, the $Q$ factor is defined in the following manner:

$$Q(q, a^0, k^0, V) = \mathbb{E}\left\{ w_1 q^2 - w_2 I(k^0) s_t + w_3 c(a^0) + V(q')|q, a^0, k^0 \right\} \tag{23}$$

where the expectation is taken over $s$ and $q'$. Note that $Q$ does not depend on $\pi^0$, since the probabilities of $q'$ and $s$ given $q$, $a^0$ and $k^0$ are independent of the attacker's policy. Now, we define the fixed point iteration

$$V^{z+1}(q) = \min_{\pi^0} \left\{ \sum_{a^0,k^0} \pi^0(a^0,k^0|q) \left( Q(q, a^0, k^0, V^z) + \epsilon \log(\pi^0(a^0,k^0|q)) \right) \right\} \tag{24}$$

It can be shown from the first order optimality conditions that a unique minimizer of the right side of (24) is

$$\pi^0(a^0,k^0|q) = \frac{e^{\frac{-Q(q,a^0,k^0,V^z)}{\epsilon}}}{\sum_{a,k} e^{\frac{-Q(q,a^0,k^0,V^z)}{\epsilon}}} \tag{25}$$

where the summation goes over $a^0 \in \mathcal{A}^0$ and $k^0 \in \mathcal{K}$. Then we just have to evaluate $Q$ for all $(a^0, k^0)$, given $V^z$ and $q$, to perform the iteration.

However, in contrast with the defender, the state $q$ is a continuous variable. For numerical reasons, this requires some sort of approximation in the value function to be evaluated at each iteration. We address this using a truncated Chevyshev basis and projecting the value function in that subspace. This is

$$V^z(q) \approx \sum_{m=1}^{n} \alpha_m \psi_m(q) \tag{26}$$

where $\psi_m(q)$ is the Chevyshev polynomial of order $m$ defined on the range of the approximation. Then on each iteration we evaluate the Bellman equation on a set of $n$ points over the space of states. Finally, we obtain the coefficients of the next iteration in the MSE sense [2].

## 5. CASE STUDY

Consider a network with $|\mathcal{K}| = 5$ and $G(k) = \frac{\exp(-0.2k)}{Z}$, where $Z$ is a normalization constant. Here, $N_x = 5$, $\mathcal{A}^0 = \{1, 2, ..., 6\}$ and $c(a^0) = a^0$.

We define exponentially decaying transition probabilities for "not update" actions, and similar ones for "update" actions, with its peak on $x+1$, if $x$ was the actual state. This way we model the decaying of the security level if the system is not updated, and its increment in the other case. Also, we set $c = 1$ for the action of updating the system and a weight of $l = 3$ units to the mass term in the cost function of the defender. We also allow the defender to randomize actions with probabilities $\mathcal{A}^r = \{0, 0.25, 0.5, 0.75, 1\}$.

We use $\beta = 0.6$ in both cases and a forgetting factor of $\alpha = 0.7$ for the attacker. In addition, we use a factor of $\epsilon = 20$ units of penalization for entropy. We define the increment in notoriety for unsuccessful attacks as $n(a,k) = 0.1ak$. We also define the probability that the variable $f_t$ (defined on section 2.1) takes value 1 as

$$p(f_t|a_t^0, x_t^i) = \frac{1}{1 + \exp(a_t^0 - 2x_t^i + 1)}. \tag{27}$$

Then, the failure probability of an attack with action $a^0$ on a node of security level $x$ is of sigmoid form, increasing rapidly with the difference between $a^0$ and $2x$.

We also define the importance of a node to the network as the square of the connectivity degree $I(k) = k^2$.

[2] The detailed procedure will be addressed in a future version

### 5.1. Defending the network

We assume that a hacker which attacks all degrees with equal probability and random attack actions, i.e. $\pi^0(a,k|q) = \frac{1}{N_a N_k}$. The resulting defence policy is shown on Figure 1, and the resulting conditional mass function is shown on Figure 2. We observe that the most important nodes defend

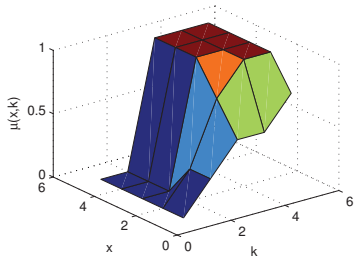


**Fig. 1**. Optimal defence policy

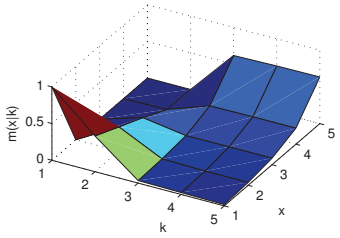


**Fig. 2**. Conditional mass function

themselves with more intensity in spite of the cost. This makes the mass more concentrated in higher security levels for the most important nodes.

### 5.2. Attacking the network

We assume that the mass function is the resulting one from Section 5.1, and that it is known to the attacker. We show the results for $q = 2$ in Figure 3 as an example. For a given net-

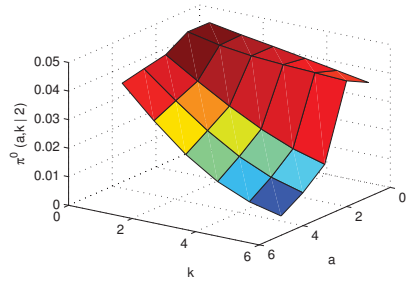


**Fig. 3**. Optimal attack policy for q=2

work configuration, the attacker prefers to attack with more intensity the lower degrees while lowering the intensity of its actions for nodes with higher degree. It also prefers to attack lower degrees rather than high ones. This is logical in this scenario, since high degrees are very well defended.

## 6. CONCLUSIONS AND FUTURE WORK

We have proposed a new theoretical model for the security problem in large scale networks using a mean field approximation. We have also shown how to obtain best response attack and defence policies in tractable and scalable way.

This work also settles the basis for mean field reinforcement learning in large scale networks for security domains. We think that is a subject of great interest, and will also be the object of future research.

Future work also includes addressing the computation of Nash equilibrium points of the game with continuous randomized policies, and the formalization of numerical methods for discrete mean field games.

## 7. REFERENCES

[1] S. Shiva D. Dasgupta V. Shandilya Q. Wu S. Roy, C. Ellis, "A survey of game theory as applied to network security," *Proceedings of the 43rd Hawaii International Conference on System Sciences*, 2010.

[2] M. Hossein, Z. Quanyan, T. Alpcan, T. Basar, and J.P. Hubaux, "Game theory meets network security and privacy," in *EPFL Technical Report*, 2010.

[3] N. Poolsappasit, R. Dewri, and I. Ray, "Dynamic security risk management using bayesian attack graphs," *Dependable and Secure Computing, IEEE Transactions on*, vol. 9, no. 1, pp. 61–74, Jan 2012.

[4] O. Guéant, J. M. Lasry, and P. L. Lions, "Mean field games and applications," in *Paris-Princeton lectures on mathematical finance 2010*, pp. 205–266. Springer, 2011.

[5] J. M. Lasry and P. L. Lions, "Mean field games," *Japanese Journal of Mathematics*, vol. 2, no. 1, pp. 229–260, 2007.

[6] F. Mériaux, V. Varma, and S. Lasaulce, "Mean Field Energy Games in Wireless Networks," *ArXiv e-prints*, Jan. 2013.

[7] H. Tembine, P. Vilanova, M. Assaad, and M. Debbah, "Mean field stochastic games for sinr-based medium access control," in *Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools*, 2011.

[8] Y. Wang, F. R. Yu, H. Tang, and M. Huang, "A mean field game theoretic approach for security enhancements in mobile ad hoc networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1616–1627, March 2014.

[9] R. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, NJ, USA, 1 edition, 1957.

[10] Dimitri P. Bertsekas, *Dynamic Programming and Optimal Control*, Athena Scientific, 2nd edition, 2000.