Distributed Network Evolution Strategy for Autonomous Robot Swarm

Anonymous Authors[1]

Abstract

In this paper, we aim to build a robust networked multi-robot system that can adaptively evolve the network topology against network dynamics in a distributed manner. We consider network coding deployed multi-robot system where source nodes are connected to terminal nodes with the help of robot swarms as relaying nodes. We show that mixing operations in network coding can induce packet anonymity that allows the inter-connections in a network to be decoupled. This enables each robot to consider complex network inter-connections as a robot-environment interaction such that the Markov decision process (MDP) can be employed at each robot. The optimal policy that can be obtained by solving the MDP provides the robot with optimal amount of changes in transmission range given network dynamics (e.g., the number of robots included in transmission range and channel condition). Hence, the overall network can be adaptively and optimally evolved by responding to the network dynamics. The proposed strategy is used to maximize long-term utility, which is achieved by considering both current network conditions and future network dynamics.

1. Introduction

Rapid development of electronics, wireless communication technologies, and control systems makes it possible to build a multi-robot system or a robot swarm. Each robot can be equipped with sensors, actuators, processors, and wireless communication modules such that it is able to generate, process, transmit and receive information among robots. The multi-robot system has wide range of applications from surveillance(**?**) to unmanned autonomous systems under dynamic environment such as disaster (**?**) or military (**?**) sites.

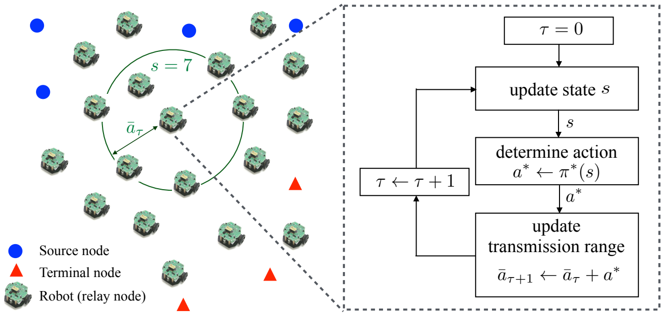
The networked multi-robot systems have unique advantages and challenges. Compared to conventional sensor networks or emerging unmanned aerial vehicle (UAV) networks, the network entities (i.e., the robots) have higher computational capability. Hence, they are able to process information such as encoding and decoding, and to perform real-time decision-making processes. However, the multi-robot applications suffer from inevitable dynamics such as high mobility and unstable channel condition, which becomes critical challenge to build a robust network.

In this paper, we aim to build an autonomous self-evolving robot network, where robots play the role of relaying nodes which collaborate to accomplish information delivery task in the presence of network dynamics. An illustrative example is presented in Fig. **??**(a). By deploying network coding (**?**) at each robot, which combines multiple packets and builds a single *mixed* packets, the complicated network inter-connections among robot swarm can be simplified to the robot-environment interaction such that the MDP framework can be adopted as an optimal solution for decision-making. We design the MDP framework, and enable the robot to determine its optimal wireless transmission range by balancing between energy consumption and overall network throughput, and by responding to network dynamics such as channel condition and robot distribution induced by mobility of robots. For example, in case of sparse robot distribution, a robot decides to increase its transmission range in order to sustain connectivity of overall network. Similarly, if the link failure rate of wireless channel decreases, the robot may shrink its transmission range in order to minimize its transmission power.

The proposed networking system consists of two phases: initialization and evolution. In the initialization phase, the optimal policy for each robot is found and initial network connection is determined. In the evolution phase, each robot follows the optimal policy, and adaptively and optimally changes its transmission range by responding to network dynamics, which is presented in Fig. **??**(b).

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.



(a) An illustrative example of networked multi-robot system at time τ

(b) An overview of network evolution process at the robot with policy π^*

Figure 1. Illustrations of a networked multi-robot system and the proposed network evolution strategy.

2. Network Coding Deployed Multi-robot System

2.1. Network Architecture for Robot Swarm

We model a networked multi-robot system as a directed graph $\mathcal{G}_\tau$, comprising a set $\mathcal{V}(\mathcal{G}_\tau)$ of wireless nodes together with a set $\mathcal{E}(\mathcal{G}_\tau)$ of directed links at time τ. Let $v_i \in \mathcal{V}(\mathcal{G}_\tau)$ be the i-th element in $\mathcal{V}(\mathcal{G}_\tau)$, and there are three types of nodes in the network: source node, robot and terminal node, where the robot can be considered as a relay node. Let $\mathbf{H}$ be an index set of source nodes and its element is denoted as $h \in \mathbf{H}$. An index set of terminals for a source node v_h is denoted as $\mathbf{T}_h$, and the data that v_h generates at time τ are denoted as $x_{h,\tau}$. Because of multipoint-to-multipoint wireless connection characteristics of multi-robot system (**?**), the system has multiple source nodes, and each source node has an independent set of terminal nodes. Specifically, a source node v_h for $h \in \mathbf{H}$ aims to deliver its data $x_{h,\tau}$ to multiple terminal nodes $v_t, \forall t \in \mathbf{T}_h$ so that $\sum_{h\in\mathbf{H}} |\mathbf{T}_h|$ flows are simultaneously considered, where $|\cdot|$ denotes the size of a set. The total index set of all terminals in $\mathcal{G}_\tau$ is denoted as $\mathbf{T} = \bigcup_{h\in\mathbf{H}} \mathbf{T}_h$, and the number of source nodes and terminal nodes are denoted by N_H and N_T, respectively.

In cases where the source nodes are not able to directly transmit data to the terminal nodes, the robot swarm becomes relay nodes. Let v_i for $i \in \mathbf{V}$ be a robot where $\mathbf{V}$ denotes an index set of robots, and N_V be the number of robots. Then, the total number of nodes in the network can be represented by $|\mathcal{V}(\mathcal{G}_\tau)| = N_H + N_V + N_T$.

We use a stochastic geometry model to capture the distribution of robot swarm to model the characteristics of mobility; the number of robots in a bounded region follows an independent homogeneous Poisson point process (PPP) with robot density λ, which is the expected number of Poisson points (**?**). Moreover, each robot can adjust its transmission power that determines potential range of data delivery of the robot, which referred to as *transmission range*. Let $\bar{\delta}_{i,\tau}$ and $\delta_{i,\tau}(v_j)$ be the radius of the transmission range of v_i and the Euclidean distance from node v_i to node v_j at time τ, respectively. Then, v_j is located in the transmission range of v_i if $\delta_{i,\tau}(v_j) \leq \bar{\delta}_{i,\tau}$ and v_j can receive the data from v_i. We assume that links between nodes may be disconnected with probability β, which is referred to as the link failure rate. Hence, the probability that v_j in the transmission range of v_i can successfully receive the data from v_i is given by $1-\beta$.

The deployment of robot swarm naturally leads to a multi-hop ad hoc network, and thus, the network throughput highly depends on the selection of paths constructed by robots. Therefore, node v_i needs to choose a node v_j, which can relay the data to terminal nodes $\mathbf{T}_h$ better than other neighbor robots in terms of node values. The node value of v_j from the perspective of v_i is evaluated by the *node value function* $\Gamma_{i,\tau}(v_j, \mathbf{T}_h)$, expressed as

$$\Gamma_{i,\tau}(v_j, \mathbf{T}_h) = f(\delta_{i,\tau}(v_j), \mathbf{\Delta}_\tau(v_j, \mathbf{T}_h)) \tag{1}$$

where $\mathbf{\Delta}_\tau(v_j, \mathbf{T}_h) = \{\Delta_\tau(v_j, v_t) | \forall t \in \mathbf{T}_h\}$ and $\Delta_\tau(v_j, v_t)$ denotes the number of hops between v_j and v_t at time τ. The function $f(\delta_{i,\tau}(v_j), \mathbf{\Delta}_\tau(v_j, \mathbf{T}_h)) : (\mathbb{R}, \mathbb{W}^{|\mathbf{T}_h|\times 1}) \rightarrow \mathbb{R}$ is a decreasing function of $\delta_{i,\tau}(v_j)$ and $\mathbf{\Delta}_\tau(v_j, \mathbf{T}_h)$, where $\mathbb{R}$ denotes the field of real numbers and $\mathbb{W} = \{0, \mathbb{Z}^+\}$ denotes the whole numbers which includes zero and the positive integers $\mathbb{Z}^+$. By defining the node value as a decreasing function of the distance and the number of hops from v_j to v_t, $\Gamma_{i,\tau}(v_j, \mathbf{T}_h)$ increases as v_j is located closer to v_i, and v_j is connected to $v_t, \forall t \in \mathbf{T}_h$ with a smaller number of hops. Therefore, v_i consumes lower transmission power with a smaller transmission range and reduces delay for data delivery for appropriate selection of v_j.

2.2. Network Coding Based Encoding Process

A source node v_h for $h \in \mathbf{H}$ generates a set of data $x_{h,\tau} = \left\{x_{h,\tau}^{(1)}, \ldots, x_{h,\tau}^{(L)}\right\}$ at time τ and it broadcasts $x_{h,\tau}$ with the transmission power $\bar{a}_h = g(\bar{\delta}_h)$, where the function $g : \mathbb{R} \rightarrow \mathbb{R}$ is determined based on a path loss model of wireless channels[1]. If a robot v_i is located in the transmission range of v_h at time τ, v_i receives $x_{h,\tau}$, and puts $x_{h,\tau}$ into its buffer $\mathcal{L}_i$, i.e., $x_{h,\tau} \in \mathcal{L}_i$ wherein data are sorted by time stamp τ with the oldest time stamp at the head of the queue (**?**)[2].

The robot v_i performs network coding operations by com-

[1] We assume that the radius of transmission range $\bar{\delta}_h$ of v_h is stationary (i.e., time independent), so that the subscription τ is omitted.

[2] Note that the packet has a limited life span (e.g., time to live (TTL) in an internet packet) such that the packets with an expired time stamp can be discarded. For simplicity, we assume that the output capacity of a node is a single packet size such that a node transmits a single packet per unit time (**?**), and a node can receive multiple individual packets by applying multipacket reception techniques (**?**).

bining packets with the same time stamp in $\mathcal{L}_i$ and generates encoded data $y_{i,\tau+1}$ at time $\tau+1$ expressed as

$$y_{i,\tau+1} = \sum_{h=1}^{N_H} \bigoplus (C_{hi,\tau+1} \otimes x_{h,\tau}) \tag{2}$$

where $C_{hi,\tau+1}$ denotes the global coding coefficient of v_i for source data $x_{h,\tau}$. The network coding operations are performed in the Galois field (GF) and the operators $\oplus$ and $\otimes$ denote the addition and multiplication in GF, respectively. When the encoding process is performed in (**??**), the source data with the same time stamp are combined together, and a packet $p_{i,\tau+1}$ is constructed as $p_{i,\tau+1} = [\tau, C_{1i,\tau+1}, \ldots, C_{N_H i,\tau+1}, y_{i,\tau+1}]$ which has the time stamp of the combined source data τ, the global coding coefficient $C_{hi,\tau+1}, \forall h \in \mathbf{H}$ as the header, and the encoded data $y_{i,\tau+1}$ as a payload. An index set of terminals for $p_{i,\tau+1}$ denoted by $\mathbf{T}_{p_{i,\tau+1}}$ can be expressed as

$$\mathbf{T}_{p_{i,\tau+1}} = \cup_{h \in \{h | C_{hi,\tau+1} \neq 0, h \in \mathbf{H}\}} \mathbf{T}_h. \tag{3}$$

This is because $p_{i,\tau+1}$ needs to be delivered to all terminals of combined source data, i.e., for all $v_t \in \mathbf{T}_h$ and $h \in \{h | C_{hi,\tau+1} \neq 0, \forall h \in \mathbf{H}\}$.

If the robot v_i receives the encoded packets $y_{h,\tau+\alpha}$ at time $\tau+\alpha$, it recombines the received data and generates the encoded data $y_{i,\tau+\alpha+1}$ at time $\tau+\alpha+1$, i.e.,

$$\begin{aligned}
y_{i,\tau+\alpha+1} &= \sum_{y_{j,\tau+\alpha} \in \mathcal{L}_i} \bigoplus (c_{ji} \otimes y_{j,\tau+\alpha}) \qquad (4)\\
&= \sum_{y_{j,\tau+\alpha} \in \mathcal{L}_i} \bigoplus \left(c_{ji} \otimes \left(\sum_{h=1}^{N_H} \bigoplus (C_{hj,\tau+\alpha} \otimes x_{h,\tau}) \right) \right)\\
&= \sum_{h=1}^{N_H} \bigoplus \left(\sum_{y_{j,\tau+\alpha} \in \mathcal{L}_i} \bigoplus (c_{ji} \otimes C_{hj,\tau+\alpha}) \right) \otimes x_{h,\tau}\\
&= \sum_{h=1}^{N_H} \bigoplus (C_{hi,\tau+\alpha+1} \otimes x_{h,\tau})
\end{aligned}$$

where $\alpha > 0$, and c_{ji}[3] denotes the local coding coefficient for data from v_j to v_i. In this paper, network coding is implemented based on RLNC (**?**) so that c_{ji} is uniformly and randomly chosen from GF with a size of 2^M (GF(2^M)), i.e., $c_{ij} \in$ GF(2^M). However, the proposed strategy is not limited to RLNC, and deterministic code designs can be considered as well.

If a terminal node receives the same number of packets as the source nodes, the source data can be reconstructed as presented in Appendix A of (**?**).

[3]The time stamp for c_{ji} is omitted because the local coding coefficient is used only for one time slot.

2.3. Impact of Network Coding on Robot Swarm

The robot swarm constructs a large-scale multi-hop wireless network, where the process of recombining incoming packets in (**??**) is performed significantly many times, and eventually allows the recombined packet to include all source data. Therefore, the terminal set of all packets in the network asymptotically converges to $\mathbf{T}$ by making all packets identical. This is defined as *packet anonymity* of network coding, which is expressed in Proposition **??**.

Proposition 1 (Packet Anonymity)**.** *Network coding can asymptotically make both the information and terminal of each packet identical.*

We next consider the impact of the packet anonymity on the node value. Let Φ be the network coding function in (**??**). Then, the node value function $\Gamma_{i,\tau}(v_j, \mathbf{T}_h)$ in (**??**) that is transformed by the network coding function Φ can be expressed as

$$\begin{aligned}
\Phi(\Gamma_{i,\tau}(v_j, \mathbf{T}_h)) &= \Phi\left(f(\delta_{i,\tau}(v_j), \boldsymbol{\Delta}_\tau(v_j, \mathbf{T}_h)) \right) &(5)\\
&= f(\delta_{i,\tau}(v_j), \boldsymbol{\Delta}_\tau(v_j, \mathbf{T})) &(6)\\
&= \Gamma_{i,\tau}(v_j, \mathbf{T}). &(7)
\end{aligned}$$

The equality between (**??**) and (**??**) is based on packet anonymity. $\mathbf{T}$ in (**??**) is constant, so that the node value is only a function of v_j, which concludes (**??**). Therefore, we conclude that if the network coding function Φ is employed in robots, multi-hop connections to terminals (i.e., $\mathbf{T}_h$) need not to be considered for v_i. Rather, only the links directly associated with it should be considered as in a one-hop connection (i.e., v_j), leading to *network decoupling* described in Proposition **??**.

Proposition 2 (Network Decoupling)**.** *Network coding can decouple one-hop connections from the network.*

Network decoupling can lead to the design of decentralized solutions by only considering robot-environment interactions at each robot, which are well-captured by the MDP framework introduced in Section **??**.

3. MDP-based Framework for Decision-making Process

In this section, we propose an MDP-based framework for network evolution, where robots $v_i, \forall i \in \mathbf{V}$ in network are considered as autonomous decision making agents to find the optimal strategy.

For an agent v_i (i.e., a robot), an MDP is a tuple $\langle \mathbf{S}, \mathbf{A}, P(s'|s,a), U(s,a,s'), \rho \rangle$, where $\mathbf{S}$ is the state space, $\mathbf{A}$ is the action space, and $P(s'|s,a) : \mathbf{S} \times \mathbf{A} \times \mathbf{S} \rightarrow [0,1]$ is the state transition probability that action $a \in \mathbf{A}$ in state $s \in \mathbf{S}$ leads to the next state $s' \in \mathbf{S}_i$, which is a real number between 0 and 1. $U(s,a,s') : \mathbf{S} \times \mathbf{A} \times \mathbf{S} \rightarrow \mathbb{R}$ is a utility obtained after transition to state s' from state s with action a, and $\rho \in [0,1]$ is the discount factor. The details are explained as follows.

3.0.1. State Space $\mathbf{S}$

A state $s \in \mathbf{S}$ represents the expected number of effective nodes in the transmission range of the agent. Since the agent can always be the effective node, at most $(N_V + N_T)$ nodes can be located in the transmission range, so that $1 \le s \le \lceil (N_V + N_T)/(1-\beta) \rceil$ in the channel with a β link failure rate.

Note that the definition of the state allows the network to be robust against network dynamics since the agent can adaptively change its transmission ranges by considering mobility of robots and channel condition. For example, if the network is static with robot density λ, simply determining $\bar{\delta}_{i,\tau}$ can provide a solution for network topology, which directly determines the number of successfully received nodes, i.e., s, based on robot density λ. If the network is dynamic, however, s at time τ cannot be directly determined by $\bar{\delta}_{i,\tau}$ since λ may not be a true value in the transmission range of the agent because of the mobility in robot system and the link failure of the channel. Hence, we design network topology based on s rather than $\bar{\delta}_{i,\tau}$, which allows the agent to adaptively change $\bar{\delta}_{i,\tau}$ against network dynamics, leading to a robust network.

3.0.2. Action Space $\mathbf{A}$

An action $a \in \mathbf{A}$ represents the increases in the transmission range as compared to a previous transmission range. Hence, the action at time τ becomes $a = \bar{a}_\tau - \bar{a}_{\tau-1}$, where $\bar{a}_\tau$ and $\bar{a}_{\tau-1}$ denote the transmission ranges at time τ and $\tau-1$, respectively. If $a > 0$, the agent increases the transmission range (i.e., $\bar{a}_\tau > \bar{a}_{\tau-1}$). Similarly, if $a < 0$, the agent decreases the transmission range (i.e., $\bar{a}_\tau < \bar{a}_{\tau-1}$). The agent may keep the same transmission range by taking action $a = 0$.

3.0.3. State Transition Probability $P(s'|s,a)$

A state transition probability represents the probability that a node in state s moves to state s' if action a is taken. Thus, $P(s'|s,a)$ means the probability that s' effective nodes will be included in the transmission range of the agent in the next time stamp by taking action a from current s effective nodes. Since the number of robots in a bounded region follows an independent homogeneous PPP with robot density λ, the state transition probability can be described as

$$P(s'|s,a) = \begin{cases} \frac{(\lambda a)^{\xi'-\xi} \cdot e^{-\lambda a}}{(\xi'-\xi)!} & a > 0 \\ 1 & a = 0 \\ \binom{\xi}{\xi'}(1 - \frac{|a|}{\bar{a}_\tau})^{\xi'} (\frac{|a|}{\bar{a}_\tau})^{\xi-\xi'} & a < 0 \end{cases} \quad (8)$$

where $\xi \triangleq \lceil \frac{s}{1-\beta} \rceil$ and $\xi' \triangleq \lceil \frac{s'}{1-\beta} \rceil$. This implies that the state transition probability is the probability that $(\xi' - \xi)$ nodes are included in the transmission range a for $a > 0$. If $a < 0$, the probability that ξ' nodes are in $\bar{a}_{\tau+1}$ given ξ nodes in $\bar{a}_\tau$ is the probability that $(\xi - \xi')$ nodes are included in $|a|$.

3.0.4. Utility Function $U(s,a,s')$

We define the utility function of the agent v_i as a quasi-linear function that consists of a reward and a cost, i.e.,

$$U(s,a,s') = u + \omega \cdot R(s,s') - (1-\omega) \cdot a \quad (9)$$

where $R(s,s')$ is the reward function that represents immediate throughput improvement given the state transition from s to s' at the cost of taking action a, which increases the transmission range. The cost intrinsically includes transmit power consumption at the agent as well as the penalty for causing wireless inter-node interference. The weight ω ($0 \le \omega \le 1$) can be used to balance the reward and the cost. For example, if $\omega = 1$, the cost associated with taking action a can be ignored, but only the throughput improvement is considered. Since a utility is generally non-negative, a constant u is introduced in (**??**) in order to satisfy $U(s,a,s') \ge 0$. The reward function $R(s,s')$ is defined as $R(s,s') = \gamma(s') - \gamma(s)$ where $\gamma(s)$ denotes network throughput when the agent is in state s, which is a concave increasing function.

3.0.5. Discount Factor ρ

The discount factor $\rho \in [0,1]$ represents the degree of utility reduction over time, so that it determines the cumulative long-term utility. The discount factor can be determined based on the consistency of the network condition (e.g., (**?**)). For example, if the network condition is static, a large value of ρ can be used by imposing a high weight on the predicted future utilities whereas a lower value of ρ needs to be used in more dynamically changing network conditions.

In the next section, we show how the proposed MDP framework enables each robot to make its own optimal decisions, and leads to network evolution strategy.

4. Distributed Network Evolution Strategy

The solution to an MDP is the optimal policy that maps the optimal actions performed in a particular state. Specifically, the policy is a function $\pi : \mathbf{S} \to \mathbf{A}$ which returns an action for a state, i.e, $\pi(s) = a$. The policy π is optimal if it can maximize the state-value function $V(s)$ such that we stochastically find the optimal policy π^* and corresponding optimal action a^* for a state s as below.

$$a^* = \pi^*(s)$$

Algorithm 1 Algorithm for ϵ-Optimal Policy

Require: state space $\mathbf{S}$, action space $\mathbf{A}$, utility function $U(s,a,s')$, weight ω, discount factor ρ, state transition probability $P(s'|s,a)$, optimality level ϵ

1: **Initialize:** $V_0(s) \leftarrow 0, \forall s \in \mathbf{S}$, $\tau \leftarrow 0$

2: **while** $V_\tau(s) - V_{\tau-1}(s) > \frac{1-\rho}{2\rho}\epsilon$ for any $s \in \mathbf{S}$ **do**

3:    **for** $\forall s \in \mathbf{S}$ **do**

4:       update $V_{\tau+1}(s) \leftarrow \max_{a\in\mathbf{A}} \left(\sum_{s'\in\mathbf{S}} P(s'|s,a) \times (U(s,a,s') + \rho V_\tau(s'))\right)$

5:    $\tau \leftarrow \tau + 1$

6: choose ϵ-optimal policy $\pi^{\epsilon^*}(s) \leftarrow \arg\max_a V_\tau(s), \forall s \in \mathbf{S}$

$$= \arg\max_{a\in\mathbf{A}} \sum_{s'\in\mathbf{S}} P(s'|s,a)\left(U(s,a,s') + \rho V(s')\right) \tag{10}$$

In (**??**), the state s sequentially moves into s', and the immediate utility $U(s,a,s')$ and the discounted state-value of successive states $\rho V(s')$ are included.

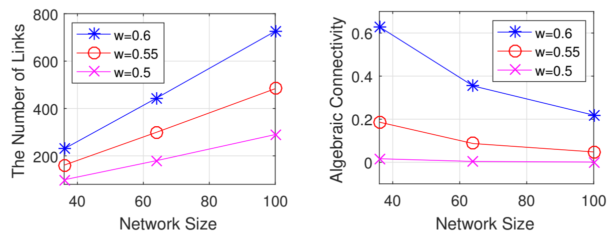
In order to propose a practical algorithm with low computational complexity, we employ a near-optimal policy. We define ϵ-optimal policy π^*_ϵ that satisfies $||V^{\pi^*_\epsilon}(s) - V^{\pi^*}(s)||_\infty \leq \epsilon$ meaning that the error between state-values derived by π^*_ϵ and π^* is bounded by the optimality level ϵ. In Algorithm **??**, the proposed algorithm is provided to find the ϵ-optimal policy.

In Algorithm **??**, the stopping criteria to find the ϵ-optimal policy π^*_ϵ is set as $||V_\tau(s) - V_{\tau-1}(s)||_\infty \leq \frac{1-\rho}{2\rho}\epsilon$ in iteration. It can be easily shown that the optimal policy π^* can always be achieved by setting $\epsilon = 0$, and the convergence speed of Algorithm **??** significantly depends on the discount factor ρ.

With the optimal policy, the robot now can adaptively change its transmission range against network dynamics, which leads to a robust network. It is worth noting that the complexity to find the optimal policy at each robot does not change, even if the total number of robots in network increases. Hence, as the number of robots increases, the total complexity to find the optimal policies of all robots in the network increases linearly. This is because the proposed MDP framework of each robot is not affected by individual network members, instead it is only affected by robot density λ in network.

The initialization phase of the proposed system can be expedited by choosing the initial state of each agent that leads to the initial network. Detailed descriptions on network initialization can be found in Appendix B of (**?**). In adaptation phase, each robot updates its transmission range based on the optimal policy by responding to the network dynamics.

In the next section, we provide simulation results of proposed network evolution strategy.



(a) The number of links   (b) Algebraic connectivity

Figure 2. The impact of ω on network connectivity ($\beta = 0, \rho = 0.5$)

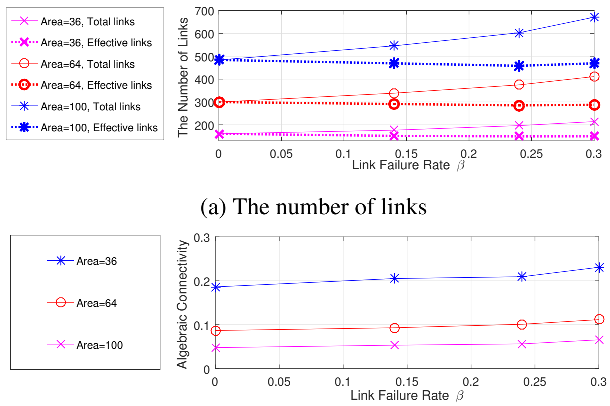
5. Simulation Results

In this simulation, we consider a networked multi-robot system with two source nodes, two terminal nodes and multiple robots.

5.1. Numerical Results

We study the resulting network in terms of two connectivity measures: the number of constructed links and algebraic connectivity (**?**). The number of links constructed in the network reflects the extrinsic connectivity and can be quantified by counting the number of links in the network. In contrast, the algebraic connectivity is the measure of intrinsic connectivity, i.e., how well the overall network is constructed. In this simulation, the network size is defined as considered network region, and the average density of robot distribution in the network region is set as $4/5$. The numerical results are averaged with $1,000$ independent experiments.

Fig. **??** shows the impact of weight ω in utility function (**??**) on network connectivity. Since ω is the weight of reward in the utility function, it is expected that the resulting networks are formed such that the rewards (or the cost) are given more weight than the cost (or the rewards) if ω is high (or low). In the simulations, we assume that there is no link failure in the channels (i.e., $\beta = 0$) and the discount factor is $\rho = 0.5$. Fig. **??** shows that the number of links is proportional to both network size and ω. A robot with high ω may increase the transmission range such that a larger number of links can be covered, leading to throughput gain over power consumption. This is also confirmed in Fig. **??**, which shows high algebraic connectivity with high ω. However, Fig. **??** shows that the algebraic connectivity decreases as network size increases. This is because the proposed strategy does not consider to retain the same algebraic connectivity. Hence, if the same algebraic connectivity is required, a higher ω should be considered in a larger network.



(a) The number of links

(b) Algebraic connectivity

Figure 5. Network connectivity over link failure rate ($0 \leq \beta \leq 0.3, \omega = 0.55, \rho = 0.5$)

Table 1. Numerical Results of Wi-Fi Direct Application

Strategy	System Goodput $[Mbps]$	Successful Connectivity Ratio $[\%]$	Power Consumption $[dBm]$
Proposed	324.60	75.63	89.07
Myopic	276.51	70.28	80.17
Traskov	226.59	38.19	103.14
TCLE	317.36	45.68	76.22

The network connectivity as a function of the link failure rate β ($0 \leq \beta \leq 0.3$) is shown in Fig. **??**. Fig. **??** shows that the proposed strategy enables nodes to make more links as β increases. This enables the networks to maintain approximately the same number of effective nodes. Moreover, it is confirmed from Fig. **??** that the algebraic connectivity increases as β increases. This is because the proposed approach increases the degree of connectivity of the network to overcome unstable channel conditions. Therefore, we conclude that the proposed strategy is successful at adaptively changing network topology by explicitly considering the link failure rates of the channels.

5.2. Performance Comparison in Wi-Fi Direct Application

In this section, we consider an illustrative application with Wi-Fi Direct with IEEE 802.11ac standard MCS-9 (**?**). The considered network region is 60×60 [m^2], and average density of robot distribution is 8×10^{-3}[robots/m^2]. For packet delivery, the RLNC is used in the GF(2^8).

The performance of the proposed strategy is evaluated based on the system goodput (**?**), which is defined as the sum of data rates successfully delivered to terminal nodes, expressed as $\sum_{h=1}^{N_H} \sum_{v_t \in \tilde{\mathbf{T}}_h} \frac{L}{\bar{\tau}(x_h, v_t)}$ where $\tilde{\mathbf{T}}_h \subseteq \mathbf{T}_h$ denotes a set of successfully delivered terminal nodes of x_h, L represents the size data set x_h, and $\bar{\tau}(x_h, v_t)$ denotes the travel time for data set x_h to arrive to a terminal node $v_t \in \tilde{\mathbf{T}}_h$. Moreover, the transmit power is measured by a path loss model, expressed as $P_{TX} = P_{RX} \cdot \left(\frac{4\pi}{\lambda} \cdot d\right)^{\alpha} = \eta \cdot d^{\alpha}$ where P_{TX}, P_{RX}, λ, and d denote transmit power, receive power, wave length, and the distance between transmitter and receiver, respectively.

In this simulation, we simultaneously consider three types of network dynamics: changes in members (i.e., robots) of the network, link failure rates, and location of robots. To produce realistic dynamic network settings, the location of robots is changed in every time stamp, and the network member and the link failure rates β ($0 \leq \beta \leq 0.3$) are updated every 5 time stamps. The simulation parameters are set to $\omega = 0.53$, $u = 0.2$, $\epsilon = 0.01$, $|\mathbf{S}| = 18$ and $|\mathbf{A}| = 7$.

We compare the performance of the proposed strategy with the following three existing network formation strategies.

1. *Myopic*: A myopic strategy is a special case of the proposed strategy with the setting of $\rho = 0$ in (**??**), which focuses on maximizing the immediate utility only.

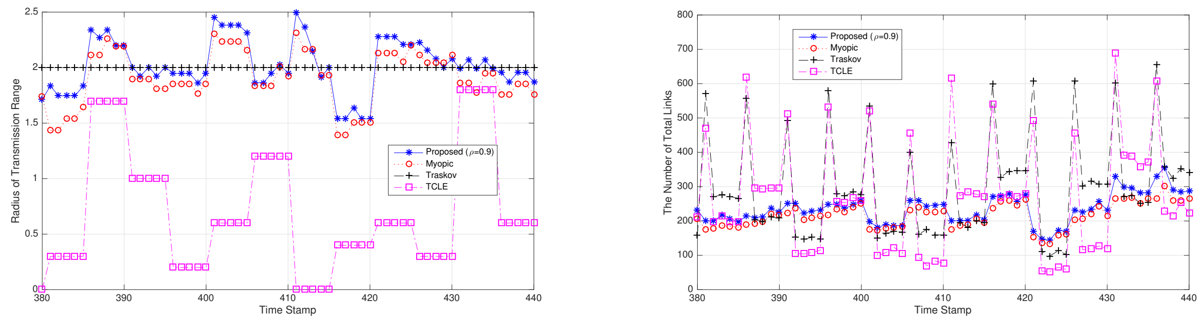
2. *Traskov*(**?**): A well-known centralized network formation strategy for network coding deployed networks. Traskov can provide a static network topology for a given node distribution by exploiting network coding opportunities.

3. *TCLE*(**?**): A state-of-the-art distributed strategy for network formation based on a non-cooperative game. In this strategy, a node chooses its transmission power by balancing the target algebraic connectivity against transmission energy dissipation.

The average numerical results from $1,000$ time stamps are summarized in Table **??**, and illustrative results in the time stamp range of $[380, 440]$ are shown in Fig. **??**.

As shown in Table **??**, the proposed strategy provides the highest system goodput as well as high successful connectivity ratio. The proposed strategy always outperforms the myopic strategy because the policy of the myopic strategy focuses only on the immediate utility, while the proposed strategy considers future utilities. For example, time stamps in $[430, 440]$ of Fig. **??** show that the proposed strategy more proactively responds to network dynamics than the myopic strategy by changing larger number of network links. Moreover, it is confirmed that the myopic strategy tends to result in smaller transmission ranges (shown in Fig. **??**), which leads to a lower total number of links in the network (shown in Fig. **??**).

While the second highest system goodput is achieved by the TCLE, it also shows the second lowest successful connectivity ratio in Table **??**. This implies that the TCLE can make successful connections between a source and a terminal based on significantly short paths. However, the TCLE



(a) Radius of transmission range of a robot

(b) The number of links in the resulting network

Figure 4. The evolution of network in the presence of network dynamics.

is not an appropriate solution for applications that count on successful delivery over throughput. Rather, it is the most energy-efficient strategy (Table **??**) as highlighted in (**?**), and it determines small transmission ranges as shown in Fig. **??**.

Traskov shows the lowest performance in terms of system goodput and successful connectivity ratio while it requires the highest power consumption. As shown in Fig. **??**, Traskov does not change the transmission range once it is determined in the beginning of the simulation such that the network fails to overcome network dynamics.

6. Conclusions

In this paper, we focus on a distributed network evolution strategy that can build a robust network in multi-robot system under the presence of network dynamics. We show that network coding induces packet anonymity and network decoupling such that the MDP framework can be employed at each robot. The robot determines an optimal policy by solving the MDP, and the policy allows the robot to determine optimal transmission range that maximizes the long-term cumulative utilities. Simulation results confirm that the resulting network of the proposed strategy can adaptively change by responding to network dynamics such as unstable channel condition and node mobility.

References

Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. *IEEE Std. 802.11-2013*, 2013.

Ahlswede, R., Cai, N., Li, S.-Y. R., and Yeung, R. W. Network information flow. *IEEE Trans. Inf. Theory*, 46(4): 1204–1216, Jul. 2000.

Andrews, J. G., Ganti, R. K., Haenggi, M., Jindal, N., and Weber, S. A primer on spatial modeling and analysis in wireless networks. *IEEE Commun. Mag.*, 48(11):156–163, November 2010.

Capitán, Jesús, Merino, Luis, and Ollero, Aníbal. Cooperative decision-making under uncertainties for multi-target surveillance with multiples UAVs. *Journal of Intelligent & Robotic Systems*, 84(1-4):371–386, 2016.

Chou, P. A. and Wu, Yunnan. Network coding for the internet and wireless networks. *IEEE Signal Process. Mag.*, 24(5):77–85, Sep. 2007. doi: 10.1109/MSP.2007.904818.

Cloud, J., Zeger, L. M., and Médard, M. Mac centered cooperation - synergistic design of network coding, multi-packet reception, and improved fairness to increase network throughput. *IEEE J. on Sel. Areas in Commun.*, 30 (2):341–349, Feb. 2012.

Gregory, Jason, Fink, Jonathan, Stump, Ethan, Twigg, Jeffrey, Rogers, John, Baran, David, Fung, Nicholas, and Young, Stuart. Application of multi-robot systems to disaster-relief scenarios with limited communication. In *Field and Service Robotics*, pp. 639–653. Springer, 2016.

Ho, T., Koetter, R., Médard, M., Karger, D.R., and Effros, M. The benefits of coding over routing in a randomized setting. In *IEEE Int. Symposium on Inf. Theory*, Cambridge, MA, USA, Jun/Jul. 2003.

Katti, S., Rahul, H., Wenjun, H., Katabi, D., Médard, M., and Crowcroft, J. Xors in the air: practical wireless network coding. *IEEE/ACM Trans. Netw.*, 16(3):497–510, Jun. 2008.

Kwon, Minhae and Park, Hyunggon. Supplemental material: Distributed network evolution strategy for autonomous robot swarm. URL `https://1drv.ms/b/s!AmmlwSalh5bsgTv42l9I2ctHH-qP`.

Miao, Guowang, Zander, Jens, Sung, Ki Won, and Slimane, Slimane Ben. *Fundamentals of Mobile Data Networks*. Cambridge University Press, 2016.

Park, Hyunggon and van der Schaar, Mihaela. A framework for foresighted resource reciprocation in P2P networks. *IEEE Trans. Multimedia*, 11(1):101–116, 2009.

Tanaka, Y. and Hara, S. Initial robots formation for clustering and information networking in a robot swarm. In *International Symposium on Wireless Personal Multimedia Communications*, pp. 1–5, Oct 2011.

Traskov, D., Ratnakar, N., Lun, D. S., Koetter, R., and Médard, M. Network coding for multiple unicasts: An approach based on linear optimization. In *IEEE Int. Symp. Inf. Theory*, pp. 1758–1762, Jul. 2006.

Witkowski, Ulf, Habbal, El, Mostafa, Mohamed Ahmed, Herbrechtsmeier, Stefan, Tanoto, Andry, Penders, Jacques, Alboul, Lyuba, and Gazi, Veysel. Ad-hoc network communication infrastructure for multi-robot systems in disaster scenarios. In *IARP/EURON Workshop on Robotics for Risky Interventions and Environmental Surveillance*, 2008.

Xu, Mengmeng, Yang, Qinghai, and Kwak, Kyung Sup. Distributed topology control with lifetime extension based on non-cooperative game for wireless sensor networks. *IEEE Sensors Journal*, 16(9):3332–3342, 2016.