ISSN 2007-9737

DRL-DC-DQN: A Deep Reinforcement Learning Approach to Decentralized Community Detection

El Yazid Gueddoudj[1,2,*], Abdelouahab Attia[3,4], Zahid Akhtar[5]

[1] University of Batna 1,
Institute of Architecture and Urbanism,
Algeria

[2] University of Tlemcen,
LRIT Laboratory,
Algeria

[3] University Mohamed El Bachir El Ibrahimi of Bordj Bou Arreridj,
Department of Computer Science,
Algeria

[4] University Mohamed El Bachir El Ibrahim of Bordj Bou Arreridj,
Laboratory Materials and Electronic System,
Algeria

[5] State University of New York Polytechnic Institute,
Department of Electrical and Computer Engineering,
USA

elyazid.gueddoudj@univ-batna.dz, abdelouahab.attia@univ-bba.dz, zahid.eltc@gmail.com

Abstract. Decentralized community detection remains a fundamental challenge in network analysis, requiring methods that operate without central coordination using only local information. This paper introduces DRL-DC-DQN (Decentralized Reinforcement Learning for Community Detection with Deep Q-Network), a novel multi-agent reinforcement learning framework where each network node functions as an autonomous agent. Agents decide whether to remain in their current community or transition to a neighboring community based on local observations and neighbor memberships, guided by a neural network approximating the Q-function. Learning is driven by a locally defined reward function that evaluates community cohesion, incentivizing agents to form structurally uniform communities. The deep reinforcement learning paradigm inherently addresses scalability concerns while improving decision quality in large-scale, complex networks. Experimental evaluations on real-world datasets demonstrate that DRL-DC-DQN achieves superior efficiency and robustness compared to state-of-the-art decentralized approaches, establishing its effectiveness for distributed community detection tasks.

Keywords: Decentralized community detection, deep reinforcement learning, multi-agent systems, deep q- network.

1 Introduction

The proliferation of online social networks has generated massive amounts of linked data, in which people, organizations, and automated agents perpetually interact with each other [1]. As a consequence, understanding the structural organization of such networks has become a main research challenge especially via the investigation of community detection [2]. In this work we aim to discover groups of nodes exhibiting strong internal connectivity while maintaining relatively sparse connections with the rest of the network, thereby revealing the inherent community structure. By revealing these communities, we gain a deeper understanding of group behaviors, information flow, rumor

doi: 10.13053/CyS-30-2-6296

propagation, and network-level decision-making. Community detection algorithms including modularity optimization and label propagation have exhibited good performance on static, moderate-scale networks [3,4] and [5]. However, modern networks are much more intricate, traditional methods are challenged in this scenario, as many rely on hand-crafted rules, are only somewhat robust, and do not generalize over time. In the recent state-of-the-art in machine learning based on the general framework of reinforcement learning has been recently shown to provide efficient solutions for graph-based community detection [6] and [7]. Unlike standard heuristic algorithms, they train flexible policies that capture intricate patterns, adapt to temporal networks, and enhance community detection with increasing amount of experience. However, there are still problems with current methods, such as difficulties scaling to very large graphs, imprecise reward mechanisms, and decreased robustness when noise or missing observations are present. These problems continue to motivate further study. The results of our studies on reference datasets show that our suggested model can manage dynamic structures, generalize to heterogeneous networks, and consistently outperform current solutions. In this paper, we propose a novel community detection framework based on graph structure information and reinforcement learning to overcome these difficulties. The framework continuously modifies cluster assignment choices by considering the local connectedness of data items as well as their global structural characteristics. In contrast to traditional methods, our model creates more cohesive and modular community structures with improved detection qualities by learning optimal clustering actions through a reward-based process. This work's primary contributions are:

- A fully decentralized deep reinforcement learning in which every node learns community assignments independently with DQN.
- A locally computable reward function based on the cohesion within communities to achieve the emergence of global structure without any centralized coordination.

The rest of the paper is structured: In Section 2, relevant research on community discovery in dynamic, centralized, and decentralized social networks is presented. The suggested framework is explained in depth in Section 3. The performance evaluation trials and their findings are detailed in Section 4. Finally, we end with a summary and recommendations for more study.

2 Related Work

Global algorithms that presume a centralized view of the entire network, such as modularity maximization, spectral methods, stochastic block models, and matrix factorization approaches, were the main focus of early community detection research [8]. Although these techniques work well on static graphs, thorough surveys [9,10] demonstrate that they are challenging to implement in highly dispersed environments due to their dependency on complete topological knowledge and frequently significant computational cost. A substantial amount of research has examined local community identification, in which each node infers its surrounding community from scant neighborhood information [11], in order to get over scalability [12] and visibility limitations. These approaches, which function with incomplete graph information and are especially well-suited for decentralized or partially viewable social networks, include label propagation schemes and personalized PageRank-based strategies [13]. Parallel to this, distributed implementations of global criteria, such as modularity-based multi-scale approaches [14] have been developed for large data platforms and distributed memory systems; nevertheless, they still usually depend on a central controller or coordinated computation across partitions. The authors of the paper [15] introduce a metric for examining community structure that they name local modularity R. The idea is straightforward: a community is gradually grown by incorporating neighbors that greatly enhance local modularity, beginning with an initial node. The algorithm looks at each possible neighbor at each stage and confirms how much its inclusion would maximize this modularity. In their research, the authors of [16] proposed a novel algorithm based on threephase local expansion technique dubbed LCDPC. Through a methodical analysis of potential relational structures inside complex networks, this approach seeks to find local communities. A novel local community detection technique called LCD-MC was put forth by the authors of the paper [17]. It effectively detects overlapping communities and does away with reliance on a single source node. It outperforms other approaches on both artificial and real-world networks in tests. To overcome scalability and visibility constraints [18], a large body of work, including studies such as [19] and [20], focuses on local community detection, where each node infers its surrounding community from limited neighborhood information. These methods include personalized PageRank-based techniques and label-propagation schemes that operate with partial graph knowledge and are particularly suitable for decentralized or partially observable social networks.

Parallel to this, distributed implementations of global criteria, as [21], have been proposed for big data platforms and distributed-memory systems. However, they still usually depend on a central controller or coordinated computation across partitions. A number of studies specifically focus on completely decentralized environments, in which nodes must coordinate solely through local message exchanges and no single organization is responsible for maintaining the entire graph. In order to maintain correct partitions under significant topological dynamics and limited resources, current dynamic decentralized algorithms, such those proposed by [22], update communities online as nodes migrate and links alter for socially aware and mobile networks. As highlighted by [23], related research on dynamic social community detection also focuses on monitoring community evolution events and using community structures for tasks such as malware containment and opportunistic routing in online and mobile social networks.

Simultaneously, communities are inferred from interaction or transaction graphs under peer-to-peer or consensus constraints in blockchain and decentralized social networks, as studied by [24]. Community detection techniques that preserve user data while creating meaningful communities have been inspired by privacy concerns in decentralized social systems [25] and [26]. By introducing a local differential privacy technique at the node level for decentralized social networks, [27] achieves community structures with formal privacy guarantees by perturbing local information prior to aggregation. More generally privacy preserving and distributed spectral clustering approaches have been studied for the multi-party or federated graph data setting [28] for community detection. The trade-offs within the factors of classification, privacy guarantee and communication overhead are typically considered, and mechanisms like randomized response and bias correction are used to achieve the balance.

At least this what some statistical guarantees tell us, but also these methods are typically tailored towards certain special scenarios and their applicability may be limited when dealing with highly dynamic or complex structured networks. Meanwhile, the study on Q-learning and Deep Q-learning still progresses with many methodological aspects and challenges is the result of the continuous growing. Furthermore, in networks with chaotic or unclear local connection patterns, its deterministic dependence on similarity metrics may weaken robustness. In the proposed framework bases its decisions on learning, which is different from the conventional, similarity based heuristics.

Deep reinforcement learning-trained agents are able to react to decisions and learn to acquire increasingly complex representations of their local and global network contexts. Because of its architecture, the framework can manage a wider range of heterogeneous network structures and maintain its resilience when local neighborhood data is used as an inaccurate indicator of community boundaries. A previous decentralized reinforcement learning scheme was proposed in [30], in which each node behaves as an independent agent attempting to enhance a global quality function, e.g., modularity, via local communications. This work shows that reinforcement learning agents can self-organize into significant community structures without centralized coordination.

However, the reliance on conventional reinforcement learning limits scalability and expressiveness when confronted with large state spaces or highly diverse local patterns. Taken

Table 1. Comparative study of techniques for detecting related communities

Ref	Method Category	Learning / Optimization Paradigm	Centralized vs. Decentralized	Core Objective/ Reward	Network Setting
[15]	Local expansion	Heuristic / greedy	Decentralized	Local conductance	Static
[16]	Potential exploration	Heuristic	Decentralized	Potential community growth	Static
[17]	Clique-based local	Heuristic	Decentralized	Clique expansion	Static
[19]	Localized detection	Algorithmic	Decentralized	Edge density	Large-scale
[20]	Parallel algorithms	Distributed optimization	Centralized	Modularity	Large-scale
[21]	Big-data based	Distributed processing (Spark)	Centralized	Modularity / density	Dynamic
[23]	Dynamic modularity	Optimization	Centralized	Modularity tracking	Dynamic
[27]	Privacy-aware	Optimization	Decentralized	Privacy-preserving modularity	Static
[28]	MARL-based	Q-learning	Decentralized	Neighborhood cohesion	Static
[29]	RL-based (general)	Q-learning	Centralized	Task-specific reward	Non-network
[30]	Information-theoretic	Optimization	Centralized	Minimum description length	High
[31]	Spectral clustering	Optimization	Centralized	Eigenvalue minimization	Low–Medium
[38]	Contrastive learning	Self-supervised DL	Centralized	Representation similarity	Multi-layer
Ours	MARL-based	Deep Q-learning	Fully decentralized	Local cohesion + stability penalty	Static / large-scale

together, while the method in[16] emphasizes deterministic local exploration and the approach in [30] establishes the feasibility of decentralized reinforcement learning, the proposed framework can be viewed as a natural progression that leverages deep reinforcement learning to support adaptive, scalable, and context-aware community detection.

As such, it provides a complementary perspective that addresses several limitations observed in earlier heuristic and reinforcement learning based approaches.

For the sake of conceptualizing our framework within the literature, we provide the comparison of some typical community detection algorithms (Table 1).

The comparison focuses on the essential methodical elements, such as the learning paradigm, the level of decentralization, the observability of environment states and the formulation of rewards.

The above overview clearly reveals the limitations of previous methods and also explains why our design in this paper does.

As can be seen in Table 1, most existing methods are based on centralized optimization or heuristic local expansion, which commonly needs the global information of the whole network.

Latest learning-based methods either stay centralized or do not explicitly consider decentralized decision-making with partial observability.

By contrast, the proposed DRL-DC-DQN architecture integrates deep reinforcement learning with the fully decentralized execution, allowing the scalable community detection using pure local information with the competitive performance maintenance.

3 Proposed Framework

This section presents the methodological design of the proposed framework, which addresses decentralized community detection using the multi-agent deep reinforcement learning paradigm. In the proposed design, each node of the network is modeled as an autonomous agent operating exclusively with local information. Learning is performed by a deep neural network (DQN), allowing each agent to update its community membership based on sequential and localized decisions, rather than global coordination.

3.1 Overview

This section describe the details of the proposed framework. Figure 1 presents an overview of the proposed framework architecture and illustrates how its main components interact during the community detection process. The framework adopts a fully decentralized perspective, in which each node in the network is treated as an autonomous agent that relies solely on information available within its immediate neighborhood. In this framework, agents observe local connectivity patterns, such as the current community affiliations of adjacent nodes, and then evaluate the set of possible actions, such as maintaining their current assignment or transitioning to a neighboring community.

The DQN, which is a function approximator that outputs a Q value for every possible action of the agent given the state, is used to select an action among a set of possible actions. Decisions are taken autonomously at a local level and evolve through iterative processes with the environment of each decision maker without reliance on central coordination or supervision. Learning in this model is localized and is based on a reward signal that represents the quality of the structure of the community in which the agent itself is embedded.

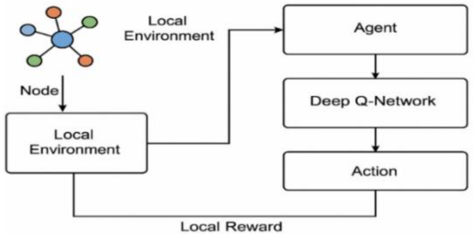

Fig. 1. The proposed DRL-DC-DQN Framework

Such feedback allows each agent to continuously update its policy, leading to the emergence of cohesive community structure in an iterative fashion. As such, the architecture consists of three main parts: (i) the local environment, which specifies what each agent can observe, (ii) the DQN-based decision module, which evaluates the actions, and (iii) an action-reward protocol that enables the policy evolution. With learning and decision-making being done at the agent level, the proposed architecture does not require centralized control and can run on large-scale or dynamic networks. Such decentralization design enables the framework to handle complex and dynamic graph structure and produce stable and adaptive community discovery results.

3.2 Local Environment

In the proposed framework, each agent builds its state representation using only information at its local environment scale. This information mainly consists of the current community assignments of adjacent nodes and a few structural measures that describe local cohesion and connectivity patterns. By restricting observations to the data of the accessible neighbors, the framework runs without the need of the global network information. This local observation scheme enables decentralized execution and at the same time it is applicable to networks of varying size, density, and structural characteristics. Therefore, agents can take actions based on partial, but useful information, extracted from their own neighborhood.

3.3 Agent and Deep Q-Network

To support decision-making in potentially high-dimensional state spaces, each agent is equipped with a DQN that serves as a function approximator for the underlying action value function. The network takes as input a feature representation derived from the agent's local

Algorithm 1. DQN-DC (Deep Reinforcement Learning for Decentralized Community Detection)

```
Input:
- Graph G, neighbors N(i), initial community of node i
- Q-network parameters $\theta$, target network $\theta^-$
- Replay buffer R, batch size B, learning rate $\alpha$
- Discount factor $\gamma$, exploration rate $\varepsilon$
- Max episodes E, max steps per episode T
Initialize $\theta$ randomly
$\theta^- \leftarrow \theta$
$R \leftarrow \emptyset$
for episode = 1 to E do
    for t = 1 to T do
        s ← ObserveState(i)  // Local feature vector
        a ← SelectAction(s, $\theta$, $\varepsilon$)  // $\varepsilon$-greedy policy
        Apply action a (update community of node i)
        s′ ← ObserveState(i)
        r ← Reward(i, a)  // Cohesion-based reward
        Store (s, a, r, s′) in R
        if |R| ≥ B then
            Sample minibatch from R
            for each (sj, aj, rj, s′j) in batch do
                $y_j \leftarrow r_j + \gamma \max_{a'} Q(s'_j, a'; \theta^-)$
                Update $\theta$ using gradient of loss $L = (Q(s_j, a_j; \theta) - y_j)^2$
            end for
        end if
        if t mod K = 0 then $\theta^- \leftarrow \theta$  // Target network update
        $\varepsilon \leftarrow \max(\varepsilon\_min, \varepsilon \cdot \varepsilon\_decay)$
    end for
end for
```

Fig. 2 Training Protocol and Agent Synchronization

state, which may include neighboring community assignments, local connectivity indicators, and other relevant structural descriptors. Based on this input, the DQN outputs estimated Q values for the set of admissible actions available to the agent. Such estimates are updated iteratively as the agent learns, thus the agent can adapt its behavior to changing network structures. This formulation enables agents to adapt to changes in network topology as they learn their decision policies.

3.4 Action Selection

Action selection is controlled by an ε-greedy policy [33, 34], so that a well-defined trade-off between exploratory and exploitative behaviors can be obtained.

Agents explore actions with relevance to the environment under some probability to obtain additional feedback from the environment, and select the action with the highest estimated value under the other condition.

The action space is deliberately represented in a concise but expressive form and includes two possible actions:

(i) stay in the current community assignment, or

(ii) move to a community related to one of the agent's neighbors. This allows agents to locally adapt to community structure and at the same time is computationally efficient.

3.5 Reward Function

The reward is specified locally and is based on some structural feature of the community structure obtained after an agent's action.

Specifically, the reward is a measure of how internally cohesive the community assigned to the agent is, so that configurations with stronger intra-community ties receive higher reward values. Such a local feedback signals agents towards community assignments that are structurally similar in to what it finds in its surrounding neighbors.

Furthermore, since the framework makes full use of local reward signals, it does not depend on

Table 2. Characteristics of Real-World Networks Used in the Experimental Evaluation

Dataset	Type	Nodes	Edges	Reference Communities	Description
Zachary's Karate Club	Social network	34	78	2	Social ties among members of a university karate club.
Dolphins	Animal social network	62	159	2	Interaction network of dolphins in Doubtful Sound.
Football	Sports / affiliation network	115	613	12	College football teams grouped by conferences.
Polbooks	Information / co-purchase	105	441	3	Political books co-purchased on Amazon.
Email-Eu-core	Communication network	1,005	25,571	42	Internal email communications of a European institution.
HEP-Th	Scientific collaboration	8,638	24,827	None	Co-authorship network of high-energy physics researchers.

global optimization objectives or centralized supervision. Instead, broader community organization arises progressively from the accumulation of distributed, agent-level decisions. The reward formulation therefore promotes transitions toward community structures characterized by higher internal consistency, while discouraging assignments that weaken local cohesion.

3.6 Learning Process

The learning process is the protocolized reinforcement learning cycle for each associated node, in which each agent independently observes its own local state, selects an action, receives an associated reward, and updates its action-value estimates.

Agents run decentralized and only have access to local information. Interactions are conceptually synchronous across agents, while community updates are carried out one at a time, so as to preserve the consistency of network states. Policy improvement is done on-line as new local interactions are accrued.

To ensure a more stable training process and address the non-stationarity of independent multi-agent learning, the framework uses experience replay and a target network, which serve to reduce correlation between updates and stabilize the estimation of Q-values.

Together they provide smoother learning dynamics and empirically stable volution.

3.7 Decentralized Operation

The entire design of the proposed system is decentralized. Each agent functions independently and relies just on the information it obtains from its local neighborhood. We do not rely on the existence of a centralized controller, a common global model, or complete network topology information. The fact that agents exchange information is implicit and unfolds through the underlying graph structure and the community assignments. This design decision promotes scalability and makes the framework suitable for large-scale, dynamic or distributed network settings, and also decreases reliance on global communication or coordination schemes.

3.8 Proposed Algorithm

The whole training procedure is structured and organized in (Algorithm 1), which applies Decentralized Deep Q-Learning in a multi-agent setting. In this setting, each node in the network is considered as an agent that tries to locally enhance its community assignment given the

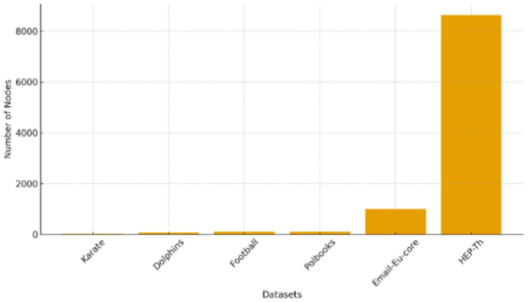

Fig. 6. Distribution of sizes of networks in datasets

Table 3. Characteristics of Datasets Used for Community Detection Analysis

Dataset	Nodes	Edges	Avg Degree	Degree Distribution	Initial Modularity
Karate	34	78	4.59	Light-tailed	0.37
Dolphins	62	159	5.13	Long-tailed	0.52
Football	115	613	10.65	Homogeneous	0.60
Polbooks	105	441	8.40	Mildly heavy-tailed	0.52
Email-Eu-core	1005	25,571	50.9	Skewed	0.48
HEP-Th	8638	24,827	5.75	Heavy-tailed	0.72

Table 4. Comparison of objectives, architectural design principles, and performance trade-offs: Ours vs. [42] and [43]

Criterion	**DRL-DC-DQN**	[42]	[43]
Objective	Learn best local decisions (Q-values)	Maximize modularity	Minimize map equation
Principle	Multi-agent DQN RL	Greedy hierarchical	Information theory, random walk
Strengths	Adaptive, decentralized, robust	Fast, scalable	Detects small communities
Limits	Training cost	Resolution limit	Slower, noise-sensitive

states and reward signals it observes locally. At initialization, each agent maintains a Q-network along with a corresponding target network. During each learning episode, the agent observes its local state, selects an action according to an ε-greedy policy, and applies the selected action to update its community affiliation. A reward reflecting the resulting local community cohesion is then received. The observed transition is stored in a replay buffer, and the Q-network parameters are updated through mini-batch gradient-based optimization. To enhance training stability, the target network is periodically synchronized with the Q-network.

3.9 Training Protocol and Agent Synchronization

The DRL-DC-DQN algorithm proposed in this paper is an independent multi-agent reinforcement learning method, in which each node-agent can get access to its own local observations rather than other node's information in the absence of centralized coordination.

Conceptually synchronous agents engage with the environment: observe local states, select actions at each time step, and community updates are sequentially enforced to maintain consistency.

Policy updates are done separately from local experience replay. Stationarity violation in the environment, which is natural to independent learner, is addressed by target networks, experience replay, and regulated exploration decay. Additionally, the rewards are only a function of local neighborhood cohesion, making inter-agent coupling very weak.

While there are few theoretical guarantees, numerical results show stable convergence for all tested networks.

As shown in Figure 2, each agent observes its local state and selects an action apply independently in sequence and it receives a local reward and updates its Q-network.

The process is the same at every time step and in every episode, so no matter what, the updates are consistent, and learning is stable without any centralized coordination.

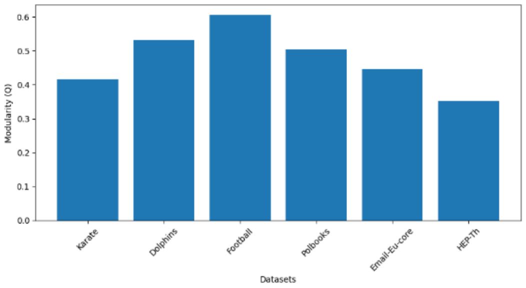

Fig. 3(a). Modularity of the proposed framework across Datasets

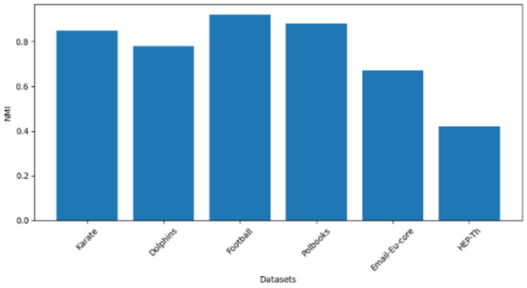

Fig. 3(b). NMI Performance of the proposed framework across Datasets

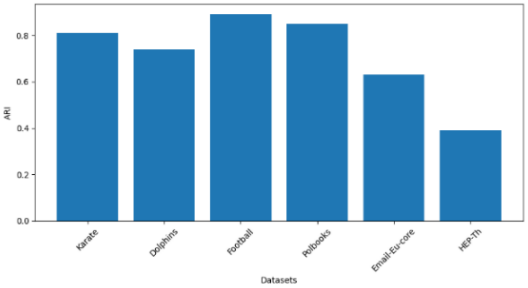

Fig. 3(c). Performance comparison of the proposed framework in ARI over Datasets

4 Experimentations and Results

In this section, we present the experiments and evaluation of the proposed Framework on several widely used real-world networks. Our objectives are to evaluate (i) its potential for evolving a coherent, stable community organization solely through local interactions, and (ii) the quality of the communities identified with respect to classical and learning-based baselines.

3.1 Dataset Characteristics

For all experiments and to observe the behavior of our framework, we took six benchmark networks with different topological structures. To enable assessment in different structural dynamics, the chosen datasets vary with respect to the complexity of the underlying network represented by the network size, the average node degree, the density, and the underlying community structure.

A summary of the datasets employed is shown in Table 2. These data are drawn from a number of different application areas and vary dramatically in scale and density. Size of Baseline Real-world Datasets.

In Fig 6, the relative size of real-world datasets by node is depicted, which shows that our studied baseline covers a wide spectrum of network scales including smaller social interaction graphs and larger scientific collaboration networks.

We present in Table 3 the full technical description of the datasets, including average degree, degree distribution and initial modularity.

3.2 Evaluation Metrics

In order to evaluate the quality of the detected communities, various performance measures were used, including Modularity (Q), It tests if the number of ties within communities is higher than what would be expected by chance according to a random null model [35]. NMI is an information-theoretic based measure which evaluates the similarity between the detected partition and the ground-truth communities. NMI is normalized in the interval [0,1] which allows for the comparison of experiment on different data sets [36] and the ARI that measures the clustering quality by comparing pairwise agreements between the obtained and true partitions and it takes into account the chance [37]. In addition, IDC Local Cohesion metric reports on how cohesive each detected community is internally. Collectively, these measures allow evaluation at both global and local levels, providing a comprehensive basis for performance analysis.

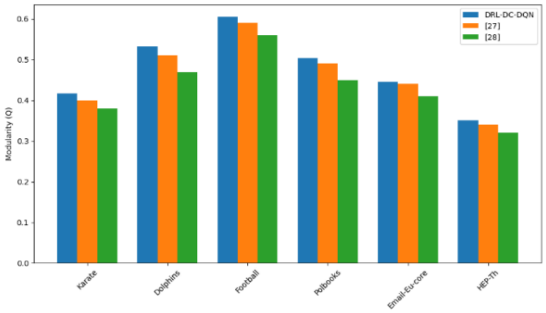

Fig. 4(a). Comparative Modularity Performance

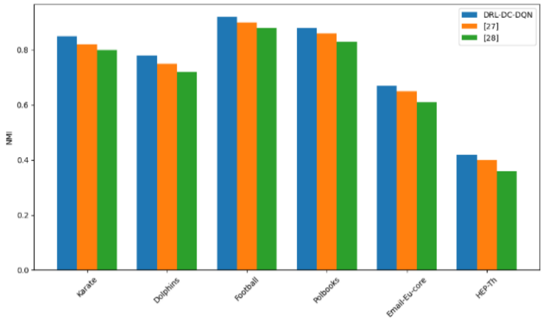

Fig. 4(b). Comparative NMI Performance

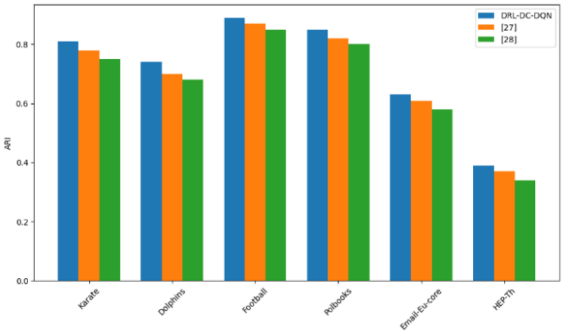

Fig. 4(c). Comparative ARI Performance

3.3 Baseline Methods

The proposed framework was compared with a representative algorithms to guarantee the fairness and comprehensiveness of results. They included tabular Q-learning-based aroad decentralized methods [38], the Label Propagation Algorithm [39], Louvain modularity optimization [40], as well as other multi-agent decentralized community detection methods [41]. This pairing gives a sense of how DQN-DC measures up against both classical and modern techniques.

3.4 Experimental Protocol

In our experiences, each agent operates under a fully decentralized manner using local information from neighbors and without any global controller. Learning takes place across episodes and experience replay is used to decorrelate between observations and to stabilize the training process. We employ a target network to have more stable Q-value updates and an ε-greedy policy to select actions to balance the trade-off between exploration and exploitation. We run each experiment several times to guarantee the reliability of the results, and we report the performance in both the mean as well as the standard deviation across runs.

3.4.1 Experiment 1

The proposed framework was evaluated against baselines on the six datasets with the following three major evaluation metrics: Modularity, NMI and ARI. The results presented in Figures 3(a-c) demonstrate that the framework can find a high-quality community for each network. The NMI values suggest that the detected communities have a very good alignment with ground-truth labels, which demonstrates a good recovery of underlying structures. Similarly, ARI values reveal the quality and consistency of the obtained partitions with respect to multiple executions. These two measurements allow for a complete evaluation of the performance of the framework, and also allow it to be compared to established community detection methods.

3.4.2 Experiment 2

a) Comparison to traditional methods

In order to demonstrate the power of the proposed framework for multi-agent cooperative task navigation, in our setting we compare against two approaches known to perform well: a hierarchical approach based on modularity optimization [42] and an informational algorithm relying on random walk dynamics [43]. Table 4 compares the goals, key ideas, benefits, and

Table 5. Performance metrics of community detection Methods across various Datasets

Datasets	Method	Modularity	NMI	ARI	Runtimes	Convergence (episodes)	Avg_switches (per_node)
Karate	DRL-DC-DQN	0,417	0,85	0,81	12	200	1,2
Karate	[16]	0,395	0,8	0,76	0,5	0	0,3
Karate	[30]	0,39	0,75	0,7	30	500	3
Dolphins	DRL-DC-DQN	0,532	0,78	0,74	20	250	1,5
Dolphins	[16]	0,51	0,75	0,7	1,2	0	0,6
Dolphins	[30]	0,5	0,72	0,68	60	500	4
Football	DRL-DC-DQN	0,605	0,92	0,89	40	300	2
Football	[16]	0,58	0,85	0,82	3	0	1,1
Football	[30]	0,57	0,82	0,79	120	800	5
Polbooks	DRL-DC-DQN	0,504	0,88	0,85	30	280	1,8
Polbooks	[16]	0,49	0,84	0,8	2,2	0	1
Polbooks	[30]	0,47	0,8	0,77	90	700	4,5
Email-Eu-core	DRL-DC-DQN	0,446	0,67	0,63	600	500	3,5
Email-Eu-core	[16]	0,42	0,6	0,55	300	0	2,8
Email-Eu-core	[30]	0,4	0,59	0,58	1200	2000	8
HEP-Th	DRL-DC-DQN	0,351	0,42	0,39	3600	800	5
HEP-Th	[16]	0,33	0,35	0,32	2500	0	4
HEP-Th	[30]	0,3	0,34	0,31	7200	5000	10

drawbacks of the proposed method with existing approaches, so the reader can derive a structured understanding of what the proposed method brings to the table.

The results illustrated in Figures 4(a-c) present a comparison of the three community detection methods (DRL-DC-DQN, [42], and [43]) on the six datasets. The first plot shows that DRL-DC-DQN obtains larger modularity values than the comparing methods on most networks, with the most significant difference for Football dataset. The second graph is the saturated NMI index, in which DRL-DC-DQN demonstrates relatively better agreement with the reference communities, especially for Karate and Football networks. The third graph confirms these observations with respect to the ARI. All these results confirm that the our framework is capable of producing more accurate and structurally consistent community attributions than traditional methods.

b) Comparison with other recent methods

In this experiment, a full assessment of the proposed framework was performed to study its behavior. The results were also compared with two existing algorithms, [16] and [30], in terms of modularity, NMI, the convergence, and the scalability on several real networks. A comparison of the performance of DRL-DC-DQN, [16] and [30] on six datasets is shown in Table 5. For each dataset and method, the following measures are reported: (i) modularity; (ii) NMI; (iii) ARI; (iv) runtime (in seconds), which reflects computational efficiency; (v) the number of convergence episodes, which represents the number of episodes required by the learning methods; and (vi) the average number of changes per node, which captures the stability of community assignments. Overall, our DRL-DC-DQN model (Figure 5(a, b and c)) exhibits consistently higher modularity and NMI values, while maintaining a number of changes per node comparable to that of [30] and a reasonable runtime compared to [16].

5 Conclusion

In this paper, we introduce a new framework, a decentralized deep reinforcement learning based method for community detection in complex networks. It uses a Deep Q-Network based multi-agent learning scheme to achieve a fully decentralized solution where a node atomically makes its community decisions in terms of local interactions with other nodes and the community labels of its neighbor nodes. Experimental results on several benchmark datasets Dolphins, Karate, Polbooks, Football, Email-EuCore and HEP-Th show that the framework is effective under multiple metrics. DRL-DC-DQN also consistently attains higher modularity, NMI and ARI, which means better structural stability and consistency with true community structures.

The proposed algorithm also converges more rapidly with low average node-switching rates, which means stability during community assignments. In contrast to the static or classical decentralized approaches, DRL-DC-DQN achieves a good tradeoff between computation time and scale, and exhibits marked improvements in large scale complex networks, where the traditional methods failed. Such outcomes demonstrate that the decentralized architecture with embedded deep Q-learning not only provides better adaptiveness for the agents and faster convergence, but also leads to higher-quality community detection. On the whole, the research work yields contribution both from the theoretical perspective and practical prospect by proposing a scalable framework that can run in complex and dynamic networks. Future work includes the run-time optimization for large-scale graphs, integrating graph neural networks to enhance state representations, and generalizing the framework to dynamic and heterogeneous network scenarios.

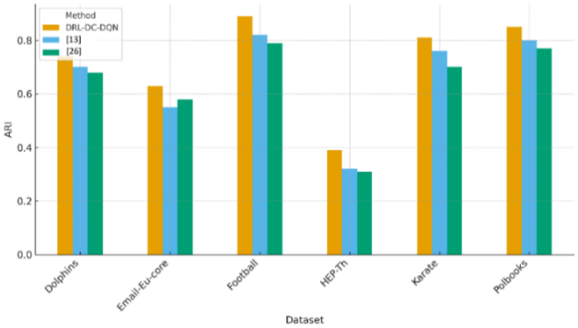

Fig. 5(a) Comparative ARI Scores across Methods and Datasets

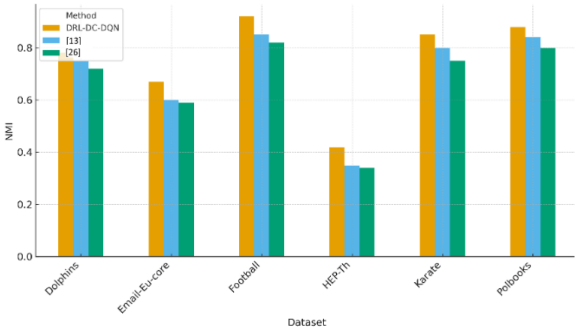

Fig. 5(b) Comparative NMI Scores across Methods

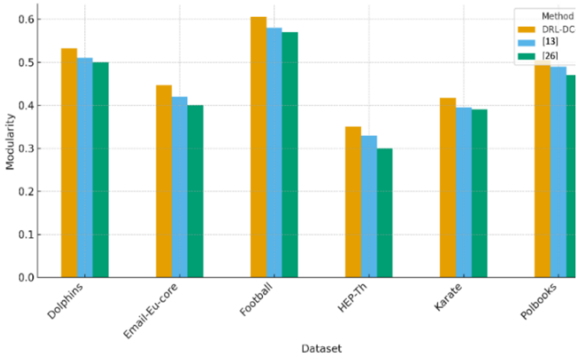

Fig. 5(c) Comparative Modularity Scores across Methods and Datasets

References

1. **Makris, C., Pispirigos, G., Rizos, I.O. (2020).** A Distributed Bagging Ensemble Methodology for Community Prediction in Social Networks. Information, Vol. 11, No. 4, pp. 199. doi: 10.3390/info11040199.
2. **Kwon, S., Cha, M., Jung, K., Chen, W., Wang, Y. (2013).** Prominent Features of Rumor Propagation in Online Social Media. In 2013 IEEE 13th International Conference on Data Mining, pp. 1103–1108. doi: 10.1109/ICDM.2013.61.
3. **Aref, S., Mostajabdaveh, M. (2024).** Analyzing Modularity Maximization in Approximation, Heuristic, and Graph Neural Network Algorithms for Community Detection. Journal of Computational Science, Vol. 78, pp. 102283. doi: 10.1016/j.jocs.2024.102283.
4. **Chaudhary, N., Thakur, H.K., Dwivedi, R. (2022).** An Ensemble Model to Optimize Modularity in Dynamic Bipartite Networks. International Journal of System Assurance Engineering and Management, Vol. 13, No. 5, pp. 2248–2260. doi: 10.1007/s13198-022-01633-1.
5. **Chen, Y., Li, D., Ye, M. (2022).** A Multi-Label Propagation Algorithm for Community Detection Based on Average Mutual Information. Wireless Communications and Mobile Computing, Vol. 2022, No. 1, pp. 2749091.
6. **El-Moussaoui, M., Hanine, M., Kartit, A., Villar, M.G., Garay, H., De La Torre, I. (2025).** A Systematic Review of Deep Learning Methods for Community Detection in Social Networks. Frontiers in Artificial Intelligence, Vol. 8, pp. 1572645.
7. **Yang, S., Wang, J. (2025).** Community Detection Based on Graph Neural Network. In 2025 37th Chinese Control and Decision Conference (CCDC), pp. 5791–5795.
8. **Diboune, A., Slimani, H., Nacer, H., Beghdad Bey, K. (2024).** A Comprehensive Survey on Community Detection Methods and Applications in Complex Information Networks. Social Network Analysis and Mining, Vol. 14, No. 1, pp. 93.
9. **Ning, S., Li, J., Lu, Y. (2023).** A Comprehensive Review of Community Detection in Graphs. arXiv preprint arXiv:2309.11798.
10. **Harenberg, S. (2014).** Community Detection in Large-Scale Networks: A Survey and Empirical Evaluation. Wiley Interdisciplinary Reviews: Computational Statistics, Vol. 6, No. 6, pp. 426–439.
11. **He, C., Fei, X., Cheng, Q., Li, H., Hu, Z., Tang, Y. (2021).** A Survey of Community Detection in Complex Networks Using Nonnegative Matrix Factorization. IEEE Transactions on Computational Social Systems, Vol. 9, No. 2, pp. 440–457.
12. **Gueddoudj, E.Y., Chikh, A., Attia, A. (2023).** Os-ETL: A High-Efficiency, Open-Scala Solution for Integrating Heterogeneous Data in Large-Scale Data Warehousing. Ingénierie des Systèmes d'Information, Vol. 28, No. 3.
13. **Al-Eiadeh, M.R., Abdallah, M. (2025).** PR-DRA: PageRank-Based Defense Resource Allocation Methods for Securing Interdependent Systems Modeled by Attack Graphs. International Journal of Information Security, Vol. 24, No. 1, pp. 1–37.
14. **Ashourvan, A., Telesford, Q.K., Verstynen, T., Vettel, J.M., Bassett, D.S. (2019).** Multi-Scale Detection of Hierarchical Community Architecture in Structural and Functional Brain Networks. PLOS ONE, Vol. 14, No. 5, pp. e0215520.
15. **Clauset, A. (2005).** Finding Local Community Structure in Networks. Physical Review E, Vol. 72, No. 2, pp. 026132.
16. **Wang, S., Yang, J., Ding, X., Zhang, J., Zhao, M. (2023).** A Local Community Detection Algorithm Based on Potential Community Exploration. Frontiers in Physics, Vol. 11, pp. 1114296.
17. **Fanrong, M., Mu, Z., Yong, Z., Ranran, Z. (2014).** Local Community Detection in Complex Networks Based on Maximum Cliques Extension. Mathematical Problems in Engineering, Vol. 2014, No. 1, pp. 653670.
18. **Baltsou, G., Christopoulos, K., Tsichlas, K. (2022).** Local Community Detection: A

Survey. IEEE Access, Vol. 10, pp. 110701–110726.

19. **Hu, P., Lau, W.C. (2012).** Localized Algorithm of Community Detection on Large-Scale Decentralized Social Networks. arXiv preprint arXiv:1212.6323.
20. **Gawande, N., Ghosh, S., Halappanavar, M., Tumeo, A., Kalyanaraman, A. (2022).** Towards Scaling Community Detection on Distributed-Memory Heterogeneous Systems. Parallel Computing, Vol. 111, pp. 102898.
21. **Piyasinghe, P.R., Chang, J.M. (2018).** Community Detection in Dynamic Networks with Spark. International Journal of Data Science, Vol. 3, No. 3, pp. 236–254.
22. **Xiong, Z. (2024).** Simple-Based Dynamic Decentralized Community Detection Algorithm in Socially Aware Networks. Heliyon, Vol. 10, No. 5.
23. **Mazza, M., Cola, G., Tesconi, M. (2023).** Modularity-Based Approach for Tracking Communities in Dynamic Social Networks. Knowledge-Based Systems, Vol. 281, pp. 111067.
24. **Nguyen, N.P., Dinh, T.N., Shen, Y., Thai, M.T. (2014).** Dynamic Social Community Detection and Its Applications. PLOS ONE, Vol. 9, No. 4, pp. e91431.
25. **Seif, M., Nguyen, D., Vullikanti, A., Tandon, R. (2022).** Differentially Private Community Detection for Stochastic Block Models. arXiv preprint arXiv:2202.00636.
26. **Guo, X., Li, X., Chang, X., Ma, S. (2023).** Privacy-Preserving Community Detection for Locally Distributed Multiple Networks. arXiv preprint arXiv:2306.15709.
27. **Fu, N., Ni, W., Hou, L., Zhang, D., Zhang, R. (2024).** Community Detection in Decentralized Social Networks with Local Differential Privacy. Information Sciences, Vol. 661, pp. 120164.
28. **Sadjadi, F., Torra, V., Jamshidi, M. (2024).** Preprocessed Spectral Clustering with Higher Connectivity for Robustness in Real-World Applications. International Journal of Computational Intelligence Systems, Vol. 17, No. 1, pp. 86.
29. **Rai, A.K., Sharma, M., Saravanan, V., Shalini, N., Kumar, M., Singh, P. (2023).** Personalized Healthcare Recommendations with Q-Learning Reinforcement Learning. In 2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI), pp. 1–6.
30. **Paim, E.C., Bazzan, A.L.C., Chira, C. (2020).** Detecting Communities in Networks: A Decentralized Approach Based on Multiagent Reinforcement Learning. In 2020 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 2225–2232.
31. **Nooribakhsh, M., Fernández-Diego, M., González-Ladrón-De-Guevara, M., Mollamotalebi, M. (2024).** Community Detection in Social Networks Using Machine Learning: A Systematic Mapping Study. Knowledge and Information Systems, Vol. 66, No. 12, pp. 7205–7259.
32. **Saeed, H.S., Abbood, A.D. (2025).** A Comprehensive Review of Dynamic Community Detection: Taxonomy, Challenges, and Future Directions. Computers, Materials & Continua, Vol. 85, No. 3.
33. **dos Santos Mignon, A., da Rocha, R.L. de A. (2017).** An Adaptive Implementation of ε-Greedy in Reinforcement Learning. Procedia Computer Science, Vol. 109, pp. 1146–1151.
34. **Tokic, M. (2010).** Adaptive ε-Greedy Exploration in Reinforcement Learning Based on Value Differences. In Annual Conference on Artificial Intelligence, pp. 203–210.
35. **Newman, M.E.J. (2006).** Modularity and Community Structure in Networks. Proceedings of the National Academy of Sciences, Vol. 103, No. 23, pp. 8577–8582.
36. **Danon, L., Diaz-Guilera, A., Duch, J., Arenas, A. (2005).** Comparing Community Structure Identification. Journal of Statistical Mechanics: Theory and Experiment, Vol. 2005, No. 09, pp. P09008.
37. **Vinh, N.X., Epps, J., Bailey, J. (2009).** Information Theoretic Measures for Clusterings Comparison: Is a Correction for Chance Necessary? In Proceedings of the

26th Annual International Conference on Machine Learning, pp. 1073–1080.

38. **Wang, X., Wang, X. (2021).** Reinforcement Learning-Based Multihop Relaying: A Decentralized Q-Learning Approach. Entropy, Vol. 23, No. 10, pp. 1310.
39. **Garza, S.E., Schaeffer, S.E. (2019).** Community Detection with the Label Propagation Algorithm: A Survey. Physica A: Statistical Mechanics and Its Applications, Vol. 534, pp. 122058.
40. **Gach, O., Hao, J.-K. (2013).** Improving the Louvain Algorithm for Community Detection with Modularity Maximization. In International Conference on Artificial Evolution, pp. 145–156.
41. **Yang, B., Huang, J., Liu, D., Liu, J. (2009).** A Multi-Agent Based Decentralized Algorithm for Social Network Community Mining. In 2009 International Conference on Advances in Social Network Analysis and Mining, pp. 78–82.
42. **Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E. (2008).** Fast Unfolding of Communities in Large Networks. Journal of Statistical Mechanics: Theory and Experiment, Vol. 2008, No. 10, pp. P10008.
43. **Rosvall, M., Bergstrom, C.T. (2008).** Maps of Random Walks on Complex Networks Reveal Community Structure. Proceedings of the National Academy of Sciences, Vol. 105, No. 4, pp. 1118–1123.

Article received on 27/03/2026; accepted on 03/05/2026.
**Corresponding authors is El Yazid Gueddoudj.*