Multi-Objective Dynamic Dispatch Optimisation using Multi-Agent Reinforcement Learning*

(Extended Abstract)

Patrick Mannion
National University of Ireland Galway
p.mannion3@nuigalway.ie

Karl Mason
National University of Ireland Galway
k.mason2@nuigalway.ie

Sam Devlin
University of York
UK
sam.devlin@york.ac.uk

Jim Duggan
National University of Ireland Galway
jim.duggan@nuigalway.ie

Enda Howley
National University of Ireland Galway
ehowley@nuigalway.ie

ABSTRACT

In this paper, we examine the application of Multi-Agent Reinforcement Learning (MARL) to a Dynamic Economic Emissions Dispatch problem. This is a multi-objective problem domain, where the conflicting objectives of fuel cost and emissions must be minimised. We evaluate the performance of several different MARL credit assignment structures in this domain, and our experimental results show that MARL can produce comparable solutions to those computed by Genetic Algorithms and Particle Swarm Optimisation.

Keywords

Multi-objective; Reinforcement Learning; Reward Shaping; Difference Rewards; Multi-Agent Systems; Smart Grid

1. INTRODUCTION

In a Multi-Agent System (MAS), multiple autonomous agents act independently in a common environment. The majority of MAS research focuses on optimising systems with respect to a single objective, despite the fact that many real world problems are inherently multi-objective in nature. This is the issue addressed by multi-objective optimisation (MOO) approaches: the requirement to make a trade-off between competing objectives. MOO approaches typically seek to approximate the true Pareto front of a problem, i.e. the set of solutions that are incomparable, where each solution is not dominated by any of the others on every objective.

Multi-Agent Reinforcement Learning (MARL) has proven to be successful in developing suitable joint policies in numerous complex single-objective problems, but research into its application to multi-objective problems is still at a very early stage. MARL problems may be formalised using the Stochastic Game (SG) framework. A SG is defined as a tuple $< S, A_{1...n}, T, R_{1...n} >$, where n is the number of agents, S is the set of states, A_i is the set of actions for agent i (and A is the joint action set), T is the transition function, and R_i is the reward function for agent i. In MARL, agents learn to maximise the return from R, and thus the design of R directly affects the joint policies learned. R may be augmented by an additional shaping reward F, in order to provide additional feedback to the agents. Reward shaping has been investigated as a method to improve learning speed and/or the final joint policy learned in MARL problems [3]. Potential-Based Reward Shaping (PBRS) is a form of reward shaping which has been proven not to alter the set of Nash equilibria of a MAS [2]. In PBRS the shaping term is of the form $F(s, s') = \gamma\Phi(s') - \Phi(s)$, where $\Phi(s)$ is a potential function which represents preferences for agents to reach certain system states.

Two typical MARL reward functions exist: **local rewards** (L_i) based on the utility of the part of a system that agent i can observe directly, and **global rewards** (G) based on the utility of the entire system. A **difference reward** (D_i) is a shaped reward signal that aims to quantify each agent's individual contribution to the system performance, with $R = G$ and $F = -G(z_{-i})$. The global utility for a theoretical system without the contribution of agent i is represented by the counterfactual $G(z_{-i})$. Here z is a general term that may represent states or state-action pairs.

Counterfactual as Potential (CaP) [3], is an automated method of generating multi-agent potential functions using the same knowledge represented by D. CaP automatically assigns potentials to states using counterfactuals, so that $\Phi(s) = G(z_{-i})$. Here $R = G$, and F is calculated as normal in PBRS. CaP preserves the guarantee of consistent Nash equilibria when using PBRS, while incorporating knowledge based on D in an automated manner.

2. DYNAMIC ECONOMIC EMISSIONS DISPATCH (DEED)

In the DEED problem a number of electricity generators must be scheduled to meet a specified customer demand over a period of time, while minimising the conflicting objectives of fuel cost and emissions. Basu [1] analysed DEED as a multi-dimensional optimisation problem, with each dimension in the problem space representing the power output of

*A full version of this paper is available at `endahowley.ie`

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

a generator at a given time. Approaches such as Genetic Algorithms (GA) [1] and Particle Swarm Optimisation (PSO) [4] have previously been applied to DEED.

In our MARL approach, each agent $i \in \{2, ..., N\}$ controls the power output of a generator $n \in N$ at each hour $m \in M$, and the first generator is a slack generator. The local cost $f_c^L(n,m)$ and emissions $f_e^L(n,m)$ terms for generator n over hour m are calculated as:

$$f_c^L(n,m) = a_n + b_n P_{nm} + c_n (P_{nm})^2 + |d_n sin\{e_n(P_n^{min} - P_{nm})\}| \quad (1)$$

$$f_e^L(n,m) = E(\alpha_n + \beta_n P_{nm} + \gamma_n (P_{nm})^2 + \eta \exp \delta P_{nm}) \quad (2)$$

where a_n, b_n, c_n, d_n and e_n are the cost coefficients for each generator, α_n, β_n, γ_n, η_n and δ_n are the emission coefficients for each generator, P_{nm} is the power output from generator n at time m, P_n^{min} is the minimum permissible power output of generator n, and $E = 10$ is the emissions scaling factor.

The global cost and emissions for hour m may then be calculated as the summation of $f_c^L(n,m)$ and $f_e^L(n,m)$ respectively over the $N = 10$ generators in the system. To discourage violations of system operating limits, a global penalty function f_p^G is used, which returns the total number of constraint violations multiplied by 10^6. The counterfactual versions of the global cost, emissions and violations terms for agent i are calculated by assuming that the agent chose the same action as in the previous timestep.

We combine the reward signals L_o, G_o, D_o and CaP_o for each objective $o \in O$ into single reward signals, using two different scalarisation techniques: linear scalarisation (+) and hypervolume scalarisation (λ). The objective weights used for the linear scalarisation are: $w_c = 0.45$, $w_e = 0.55$, and $w_p = 1.0$. The agents receive one of these scalarised reward signals while learning: $L(+)$, $L(\lambda)$, $G(+)$, $G(\lambda)$, $D(+)$, $D(\lambda)$, $CaP(+)$ or $CaP(\lambda)$. Note that all reward signals given to agents are negative, as this is a minimisation problem. We apply multiple individual Q-learners with ϵ-greedy exploration to DEED, learning using the reward functions above for $20,000$ episodes. Each episode comprises $M = 24$ hours. The learning parameters for all agents are: $\alpha = 0.10$, $\gamma = 0.75$, $\epsilon = 0.05$. All values for cost coefficients, emission coefficients, ramp limits, generator capacity limits, power demands and transmission line loss coefficients can be found in the work of Basu [1].

3. RESULTS AND DISCUSSION

The average results over 50 statistical runs for the MARL approaches tested are presented in Table 1. All claims of statistical significance are supported by two-tailed t-tests assuming unequal variances, with $p = 0.05$ selected as the threshold for significance. As expected, L performs poorly here, encouraging agents to greedily minimise their own fuel cost and emissions without considering the utility of the system as a whole. D converges to a stable policy most quickly with both scalarisations, while both variants of G learn good policies, but at a slower rate than D. CaP initially learns more quickly than G for both scalarisations; increased learning speed is a typical characteristic of PBRS. However, the final joint policies learned by CaP are not as good as those learned by G or D.

No statistical difference was found between the final performance of the scalarisation approaches for $G(+)$ and $G(\lambda)$, or for $CaP(+)$ and $CaP(\lambda)$. The differences in the means between $D(+)$ and $D(\lambda)$ were statistically insignificant for the cost objective, but were significant for the emissions objective ($p = 1.19 \times 10^{-8}$). The differences in the mean final performance of $D(+)$ and $G(+)$ were found to be significant for both the cost objective ($p = 5.01 \times 10^{-22}$), and the emissions objective ($p = 3.20 \times 10^{-10}$). Overall, $D(+)$ offered the best performance of all the MARL approaches tested.

GA results reported by Basu [1] and PSO results reported by Mason [4] are included in Table 1 for comparison purposes. Our experiments show that MARL produces results that are comparable to those produced by GA and PSO based approaches, although not quite as good. For example, Basu's NSGA-II [1] has 4.2% lower costs, and 6.8% lower emissions than $D(+)$ on average in this problem. However, MAS is arguably a more interesting paradigm to use when studying these types of optimisation problems, due to the ability to modify simulation parameters while learning online, and the possibility of modelling system disturbances (e.g. generator failure). MAS are inherently suited to distributed control and optimisation problems like DEED, and we intend to investigate further applications of MAS and MARL to these types of problems in the future.

Table 1: DEED Average solutions

	Cost ($ $\times 10^6$)	Emissions (lb $\times 10^5$)
$L(+)$	4.1127	28.8266
$L(\lambda)$	4.1149	17.6606
$CaP(+)$	2.8777	7.4774
$CaP(\lambda)$	2.8919	9.6431
$G(+)$	2.7647	3.9098
$G(\lambda)$	2.7607	3.9788
$D(+)$	2.6641	3.3255
$D(\lambda)$	2.6748	3.8980
NSGA-II [1]	2.5226	3.0994
PSO-AWL [4]	2.5463	2.9455

Acknowledgments

Patrick Mannion is funded by the Irish Research Council.

REFERENCES

[1] M. Basu. Dynamic economic emission dispatch using nondominated sorting genetic algorithm-ii. *International Journal of Electrical Power & Energy Systems*, 30(2):140–149, 2008.

[2] S. Devlin and D. Kudenko. Theoretical considerations of potential-based reward shaping for multi-agent systems. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 225–232, 2011.

[3] S. Devlin, L. Yliniemi, D. Kudenko, and K. Tumer. Potential-based difference rewards for multiagent reinforcement learning. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 165–172, 2014.

[4] K. Mason. Avoidance techniques & neighbourhood topologies in particle swarm optimisation. Master's thesis, National University of Ireland Galway, 2015.