Local Optima Networks for Reinforcement Learning - A Case Study: Coupled Inverted Pendulum Task

Yuyang Zhou
School of Computer Science
University of Nottingham Ningbo China
Ningbo, China
Yuyang.Zhou@nottingham.edu.cn

Alexander Turner
School of Computer Science
University of Nottingham
Nottingham, United Kingdom
alexander.turner@nottingham.ac.uk

Ferrante Neri
School of Computer Science
and Electronic Engineering
University of Surrey
Guildford, United Kingdom
f.neri@surrey.ac.uk 0000-0002-6100-6532

***Abstract*—Reinforcement Learning (RL) refers to a set of methods where the agent learns directly from interactions without explicitly constructing a model of the environment. In RL, the agent interacts with an environment, takes actions, receives feedback, and learns to make decisions to maximize cumulative rewards over time. The primary goal is to find an optimal policy or value function that guides the agent's decision-making. Although RL can be formulated as an optimisation problem, it is rarely analysed or studied in depth. Conversely, just like any other optimisation task, an understanding of the problem might help detect high-quality policies. This study employs the use of Local Optima Networks (LONs) to analyse the fitness landscape associated with RL and modify the sampling method for the case of the coupled inverted pendulum tasks. Deep Deterministic Policy Gradient serves as a local search algorithm to refine the characterization of the fitness landscape. Experimental results on the two pendulum tasks in part confirm and extend the conclusions of a study on the same problem carried out from a robotics and engineering standpoint. However, the proposed approach uniquely identifies both known and previously unknown local optima solutions. A sensitivity analysis of a key LON parameter, the perturbation strength, offers deeper insights into the fitness landscape. The constructed LON indicates that, for the coupled inverted pendulum task, some basins of attraction are much stronger than others.**

***Index Terms*—Fitness landscape analysis, reinforcement learning, inverted pendulum task, robotics**

I. Introduction

Reinforcement Learning (RL) is a machine learning paradigm where an agent learns to make decisions by interacting with an environment. The agent receives feedback in the form of rewards or penalties based on its actions, and the objective is to learn a policy that maximizes the cumulative reward over time. Due to this model free approach, RL has found extensive application in robotics, contributing to the development of intelligent, adaptive systems [1].

Although RL can be perceived as a trial-and-error black-box approach, it inherently involves solving an optimisation problem. Consequently, we can associate a fitness landscape with any RL problem and conduct an analysis of this fitness landscape, that is Fitness Landscape Analysis (FLA) [2]. FLA comprises a set of techniques aimed at extracting features from a fitness landscape to inform the design of an appropriate solver. These features include, for example, the number of optima [3] and the correlation between pairs of variables [4].

While this approach has been extensively used in the discrete domain [5]–[7], it is gaining attention in the continuous domain as well [8]–[10]. Among the FLA visualisation techniques, Local Optima Networks (LONs) [11] have recently received increasing attention from the evolutionary computation community. A LON provides an intuitive 2D or 3D graphic visualisation of multivariate landscapes. In a LON graph, the nodes represent local optima while edges represent possible transitions between these local optima. Due to their potential to represent highly dimensional landscapes, LONs are currently considered a popular tool to study learning landscapes of neural networks [12] and have even been experimented within a neural architecture space [13].

In the context of RL, particularly Deep RL, FLA involves examining the structure and characteristics of the solution space in which learning algorithms operate. It explores how the performance of a learning algorithm, or agent, is influenced by changes in its parameters or policies. By visualizing and analyzing the fitness landscape, researchers can gain insights into the complexity, smoothness, and potential challenges of the learning process. This analysis aids in understanding how different configurations of the agent's behavior relate to its performance and guides the optimization of reinforcement learning algorithms for more effective and efficient training.

The study in [14] performed a FLA on the training landscape of a neural network for RL applied in the context of the control of the coupled inverted pendulum task. Numerical results indicated a correlation between the ruggedness of the landscape and the performance of multiple operators, thus providing a recommendation for which operator to use under various ruggedness scenarios.

The present study, building on the analysis in [14], introduces an approach based on Local Optima Networks (LONs) to gain insights into the RL problem associated with the coupled inverted pendulum task. Our computational intelligence perspective's findings are compared with those obtained through a classical engineering approach in [15]. This cross-disciplinary comparison reveals that our proposed approach partially confirms the results in [15] for the two pendulum tasks, as both approaches identify the same typical local optima. However, our LON approach indicates the presence of some typical optima that were previously unknown while

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00118

reveals the strength of different basins of attraction.

To address the multi-modal nature of the fitness landscape, we propose a modified resampling strategy, namely parameter-level basin hopping, that directly operates on the parameters of neural networks adopted in RL algorithm, aims to detect unexplored basins of attraction by perturbations of the neural network parameters. We combine this strategy with the Deep Deterministic Policy Gradient (DDPG) [16], a gradient-based method traditionally used in RL. In the proposed algorithm, DDPG acts as a local search to optimize the behaviour of RL agents on the coupled inverted pendulum task. Meanwhile, DDPG is resampled by parameter-level basin hopping to mitigate the issue of converging into typical local optima. The proposed method, by combining a thorough exploration of the search space with an exploitation of promising areas, enables the detection of previously unknown local optima.

The remainder of this paper is organised as follows. Section II introduces the coupled inverted pendulum task and formulates it as a RL problem. Additionally, a brief overview of Local Optima Networks (LONs) is provided. Section III describes the proposed method, distinguishing between the new sampling method called parameter-level basin hopping and the use of DDPG as a local search. Section IV presents the results of this study. Finally, Section V provides the concluding remarks for this study.

II. Background

A. The Coupled Inverted Pendulum Task

The coupled inverted pendulums task is a control task that was designed as a proxy for the dynamics locomotion of multi-legged robots [15]. The purpose of this was to create a generalised algorithm capable of evaluating the performance of control algorithms for the purpose of robotic control. Different control algorithms can be compared, to measure their ability to solve the task and to better understand their properties and functionality. This alleviates the time pressures associated with building bespoke tasks in which to evaluate a particular control algorithm [14].

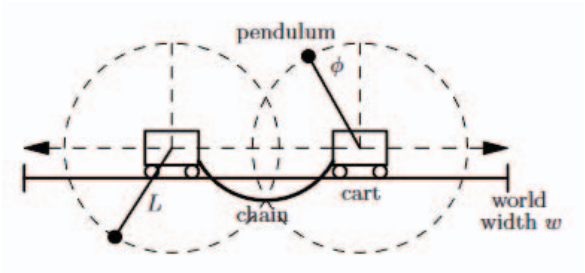

Fig. 1: The coupled inverted pendulums task configured for two carts. The objective is to move the carts in such a way to move the pendulums to the upright position and maintain their balance [15].

The task consists of a 1-dimension track which can be configured to contain between 1 - 5 carts. Each cart has a centrally mounted pendulum at its center hanging below it. In scenarios with multiple carts, they are interconnected, restricting their movement. If this is the case, the task necessitates coordinated movements to prevent overstretching the tether or collisions. The simulation ends, recording the fitness score at that moment, if any cart collides with another or hits the track's boundary. The objective of the task is to move the cart(s) in such a way as to swing the pendulum into the upright position and maintain it there. The complexity of the task escalates with the number of carts, transitioning from simple movements in single-cart scenarios to intricate coordination and spatial navigation in multi-cart setups. The complexity and execution times of tasks vary markedly between single and double pendulums. Therefore, to keep the computational runtime of the experiments within reasonable limits, the focus will be exclusively on the one and two pendulum problems

TABLE I: Parameters of the coupled inverted pendulums task [15].

ID	Sensor Name	System to sensor mapping
S_0	Pendulum Angle 0	$\phi \in$ [0,0.5π] $\rightarrow$ [127, 0], 0 else
S_1	Pendulum Angle 1	$\phi \in$ [1.5π,2π] $\rightarrow$ [0,127], 0 else
S_2	Pendulum Angle 2	$\phi \in$ [0.5π,π] $\rightarrow$ [127, 0], 0 else
S_3	Pendulum Angle 3	$\phi \in$ [π,1.5π] $\rightarrow$ [0,127], 0 else
S_4	Proximity 0	Distance left $\rightarrow$ [0,127]
S_5	Proximity 1	Distance right $\rightarrow$ [0,127]
S_6	Cart Velocity 0	$v \in$ [-2,0] $\rightarrow$ [127,0], 0 else
S_7	Cart Velocity 1	$v \in$ [0,2] $\rightarrow$ [0,127], 0 else
S_8	Angular Velocity 0	$w \in$ [-5π,0] $\rightarrow$ [127,0], 0 else
S_9	Angular Velocity 1	$w \in$ [0,5π] $\rightarrow$ [0,127], 0 else
A_i	Actuators 0	$A_i \in$[0,127], for $i \in$ 0,1
u	Motor Control 0	2(A_0/127 - A_1/127) $\rightarrow$ [0,1]

Control of each cart is independent, with inputs at every time step dictating the speed. At each time step, each cart provides 10 sensor readings that describe its current state which can be seen in Table I. The differential between two consecutive actuator values determines the cart's next movement. The simulation spans a maximum of 3000 time steps. An aggregate fitness function measures the proportion of time steps during which all pendulums remained in their upper equilibrium positions.

B. Continuous Control by Deep Reinforcement Learning

Reinforcement learning is a subsection of machine learning where an agent seeks to maximise a given reward over a number of time steps [17]. Unlike supervised learning, which relies on labeled training data which can be complex to acquire, reinforcement learning operates without the need for both data acquisition and labelling. This field has gained renewed interest following the successful integration of deep neural networks with reinforcement learning, demonstrated in surpassing human-level performance [18]. A notable implementation in this domain is deep Q-learning, utilizing deep neural networks [19]. These networks, characterized by multiple neuron layers between input and output [20], are employed in this study to control each cart in the simulation.

The neural network parameters are encoded as a vector whose elements are weights and biases of the network

$$\left(w^{1,1}, w^{1,2} \ldots, b^{1,1}, b^{1,2}, \ldots\right) \tag{1}$$

with indices indicating the layer and neuron position.

Each network mathematically models a cart, taking ten kinematic parameters $S_0, S_2, \ldots, S_9$ listed in Table I as inputs and outputting two acceleration values for the cart's wheels, A_1 and A_2 respectively.

In scenarios with multiple interconnected carts, the candidate solution $\mathbf{x}$ includes weights and biases for each cart's network 1. In order to assess the quality, i.e., the fitness f of X, is evaluated through simulation, recording the aggregate duration each pendulum(s) remains upright during the task. The fitness function $f(\mathbf{x})$ to be maximised is the fraction of time during which each pendulum is in the upright position. Simulations are continued for a prearranged observation window and interrupted if the distance between two carts exceeds the length of the chain or the carts have collided.

The pseudocode (Algorithm 1) details the calculation of the fitness function $f(\mathbf{x})$ for the coupled inverted pendulums studied in this article and depicted in Figure 1 is shown in Algorithm 1.

Algorithm 1 Fitness function f for the coupled inverted pendulums task using deep neural networks also used in [14]

INPUT The candidate solution $\mathbf{x}$ containing weights and biases of two neural networks (one for each cart)
$k = 1$
while Carts within the simulation are in bounds, proximity of each other and within 3000 step limit ($k \leq 3000$).
do
  for $j = 1 : n_{pend}$ carts **do**
    extract and normalise the sensor values $S_0, S_1, \ldots S_9$ for cart j from the simulation, see Table I
    execute the network and collect the acceleration values for each wheel
    execute one step of the pendulum simulation and record the time $t_{up}^{k,j}$ during which the pendulum is upright
    update the sensor values $S_0, S_1, \ldots S_9$ representing the new state of the cart
  end for
  $k = k + 1$
end while
Calculate $t_{up} = \sum_{k=1}^{3000} \sum_{j=1}^{n_{pend}} t_{up}^{k,j}$ and normalise it $t_{up} = \frac{t_{up}}{\text{total time}}$
OUTPUT The fitness value $f(\mathbf{x}) = t_{up}$

C. Local Optima Networks

Local Optima Network (LON) was proposed in [11] to analyse and visualise the fitness landscape of problems with discrete search space. Subsequently, in [21], LONs were extended to the continuous domain with the Basin-Hopping algorithm [22] as the sampling method. LONs have been recognized as a valuable tool for effectively visualizing and analysing the global structure of fitness landscapes.

A LON is a directed graph with sampled local optima as the nodes and transitions between local optima as the edges. During sampling, Basin-Hopping perturbs an existing local optima and performs local search on the perturbed solution. If the local search leads to a non-deteriorating local optima, a valid transition is established, and a new edge is created from the existing local optima to the improved local optima. A monotonic sequence is a connected sequence of nodes. The start node of the sequence is the first local optima sampled in a run, and the end node is the last local optima sampled in that run, which does not have an outward edge. This implies that the perturbation fails to move to a non-deteriorating local optima.

LONs have been applied to various continuous and discrete domain problems [23] [24]. However, it is worth noting that, to the best of our knowledge, this current work represents the pioneering effort in utilizing LONs to analyze the fitness landscape of high-dimensional problems, surpassing 1000 dimensions (the number of dimensions is give by the number of trainable parameters of the neural network). Furthermore, this study is the first attempt to construct LONs based on the trainable parameters of neural networks. By visualizing and analyzing LONs in such complex and high-dimensional domains, we gain valuable insights into the global structure, connectivity, and optimization challenges of such landscapes.

III. Proposed Method: LON for Reinforcement Learning

A. Parameter-Level Basin-Hopping

Algorithm 2 presents an adaptation of the Basin-Hopping algorithm [25] specifically designed to operate on the trainable parameters of neural networks. It takes several inputs: perturbation strength (p), tolerance (T), a RL method ($F(\theta, \omega, e, h)$), the necessary hyperparameters (h) for the RL method, and the environment parameters (e) for the RL task. The algorithm returns a list of sampled fitness values (f^*) and their corresponding local optima parameters (θ^*, ω^*). The perturbation strength (p) determines the range of random values added to each parameter during perturbation. The tolerance (T) determines when the sampling process should stop. If more than T consecutive runs of the local search fail to find non-deteriorating solutions, the algorithm terminates.

Algorithm 2 initializes random parameters θ and ω for Actor and Critic networks. The RL method is executed to train the Actor and Critic to find local optima θ_l and ω_l with the local maximal fitness f. Next, a perturbation process applies random values within $[-p, p]$ to θ^* and ω^*, followed by another RL method execution. The process repeats until the tolerance criterion is met. Throughout the algorithm, the best fitness f^* and corresponding parameters θ^* and ω^* are updated when non-deteriorating local optima are discovered.

Algorithm 2 Parameter-Level Basin-Hopping

INPUT Perturbation strength p, Tolerance T, Environment parameters e, Hyperparameters h, RL method $F(\theta, \omega, e, h)$
Randomly initialize Actor θ, Critic ω
Set tolerance counter $t = 0$
Observe fitness and local optima parameters $f, \theta_l, \omega_l = F(\theta, \omega, e, h)$
Update best fitness $f^* = f$
Update best parameters $\theta^* = \theta_l$, $\omega^* = \omega_l$
Append best fitness and parameters to list $list \leftarrow (f^*, \theta^*, \omega^*)$
while $t < T$ **do**
  $\theta = \text{Perturbation}(\theta^*, p)$
  $\omega = \text{Perturbation}(\omega^*, p)$
  $f, \theta_l, \omega_l = F(\theta, \omega, e, h)$
  if $f \geq f^*$ **then**
    $f^* = f$
    $\theta^* = \theta_l$, $\omega^* = \omega_l$
    $list \leftarrow (f^*, \theta^*, \omega^*)$
    $t = 0$
  else
    $t = t + 1$
  end if
end while
RETURN $list$

B. Deep Deterministic Policy Gradient

The DDPG algorithm [16] is employed as the local search method to solve the RL task. DDPG is a well-known algorithm designed to address RL problems with continuous states and action. We chose DDPG for local search as it naturally suits the continuous action space of the inverted pendulum problem under consideration. In this setup, the Actor and Critic networks are both 4-layer feedforward neural networks. Each layer in the networks uses sigmoid activation function, including the output layer of the Actor network, which uses the sigmoid function to constrain the action values within the range of $[0, 1]$. On the other hand, the output layer of the Critic network has no activation function, allowing the Q-value to range in $[-\infty, \infty]$. To facilitate learning, an Experience Replay buffer of size 40000 is utilized, from which 2000 state-action-reward-state transitions are sampled at each step. During the sampling process, a Gaussian noise $\epsilon \sim \mathcal{N}(0, 0.1)$ is added to the selected action. The learning rate is set to 0.001, and the discount factor, which determines the importance of future rewards, is set to 0.99. These configurations are specified in the input h of the Basin-Hopping algorithm.

C. Single Action Conversion Trick

As mentioned in Section II-A, each cart has 2 actuators A_1 and A_2 for two opposite directions, left and right. The motor gives an acceleration to left only when $A_1 - A_2 > 0$ and to right when $A_1 - A_2 < 0$. Since only the difference between the actuators is relevant, the Actor is designed to output a single action value a in the range of $[0, 1]$ for each cart. This action value is then interpreted as follows:

$$a \xrightarrow{a<0.5} \{254 \times (0.5 - a), 0\},$$

$$a \xrightarrow{a\geq 0.5} \{0, 254 \times (a - 0.5)\}.$$

In the case of the two pendulum tasks, the sensor values of the two carts are concatenated and used as input to the Actor network. The Actor network then generates two action values as output. The sizes of the layers in the Actor network, including the input layer, are [20, 20, 20, 20, 2]. The input layer of the Critic network receives the concatenated state and action as input, and the network outputs a single Q-value. The sizes of the layers in the Critic network, including the input layer, are [22, 20, 20, 20, 1]. The total number of trainable parameters is 2623.

IV. Experimental Results

The parameter-level Basin-Hopping method, utilizing DDPG as the local search, is employed for the inverted pendulum task on the two carts scenario with 2000 steps of simulation. A total of 100 runs of Basin-Hopping are conducted with two different perturbation strengths, namely $p \in \{\text{1e-3, 1e-4}\}$, and a tolerance value of $T = 5$.

Figure 2 summarizes the performance of local optima sampled by Basin-Hopping. There are some key differences between the two perturbation strengths, with the smaller perturbation (1e-4) resulting in increased performance, especially at above a fitness value of 0.6. This suggests that the smaller perturbations are able to traverse the landscape more optimally than larger perturbations. More specifically, the smaller perturbations allow for the optimum behaviour to be learned, whereby the pendulums are able to be balanced in the upright position, which typically occurs with finesses of 0.72 and above.

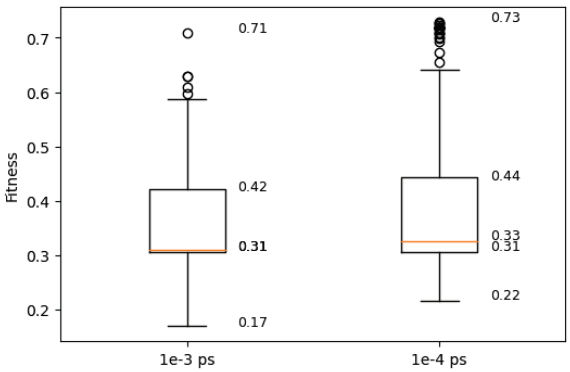

Fig. 2: Local optima fitness boxplot for two pendulums task with different perturbation strength (ps).

The boxplot analysis indicates that the LON results with a perturbation strength of 1e-4 warrant a closer investigation. Figure 3 presents a 3D visualization of the fitness landscape

for the two pendulums task with a perturbation strength of 1e-4, revealing a complex landscape of local optima. Three notable characteristics are captured by the LON visualization.

- Due to the high-dimensional search space, the landscape of local optima appears sparse. Different runs seem not converging to the same local optimum, as nodes are never visited by multiple inward edges.
- The problem exhibits multiple distinct typical local optima as nodes tend to cluster around specific levels of the graph rather than being evenly distributed along the z-axis.
- Some basins of attraction are challenging to escape. The nodes at the bottom of the LON graph have a limited number of edges connecting them to the upper part of the graph, suggesting that most attempts to escape these local optima are unsuccessful. This emphasizes the importance of proper initialization of the neural network parameters, as poorly initialized Actors and Critics may struggle to escape basins of attraction with low fitness.

The remaining of this section further investigates the LON results with various numerical analyses.

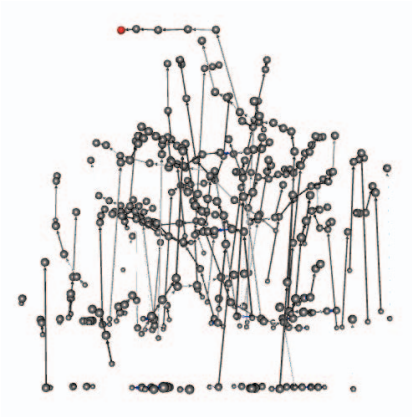

Fig. 3: 3D LON visualization for two pendulums task with perturbation strength 1e-4. Grey nodes are local optima, red node is the local optima with the best fitness, small grey nodes are start of a path of connected nodes. Directed edges represents perturbations from the start node to the end node. The nodes with higher fitnesses are positioned higher.

Figure 4 displays the distribution of basins of attraction, referred to as 'sinks' in [23], at the level of local optima in the parameter space of neural networks for the two pendulums task with a perturbation strength of 1e-4. In the context of the LON, a sink represents the end node of a monotonic sequence where the sampling method fails to discover any non-deteriorating local optima through perturbation. The distribution of fitness values associated with these sinks reveals the presence of five peaks, representing local maxima in terms of frequency of the occurrences of fitnesses. These peaks are observed around fitness values of $0.22, 0.32, 0.46, 0.55$, and 0.72. In [15], the fitness values of $0.32, 0.46, 0.55$, and 0.72 are recognized as typical local optima with distinct behavioral patterns. However, the sink corresponding to a fitness value of 0.22, which is not reported in [15], represents a new behavior pattern. In this scenario, both carts move in the same direction and come to an immediate stop upon approaching the boundary. Consequently, the pendulums remain in the lower position throughout the task, resulting in a low fitness.

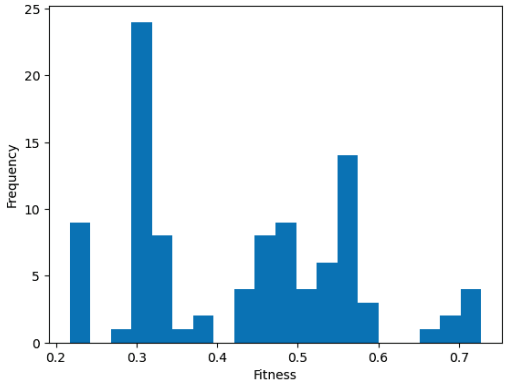

Fig. 4: Fitness histogram of basins of attraction of local optima for two pendulums task with perturbation strength 1e-4.

The distribution of fitnesses of all local optima sampled for two pendulums task with perturbation strength 1e-4 is shown in figure 5. Similarly as in figure 4, the figure reveals the presence of five peaks located around $0.22, 0.32, 0.46, 0.55, 0.72$ fitness values. This observation suggests that the end nodes of a monotonic sequence serve as typical basins of attraction. Moreover, the limited occurrence of local optima between fitness values of 0.6 and 0.7 indicates a surprisingly smooth landscape beyond a fitness of 0.6. Consequently, local search algorithms easily reach a fitness of 0.7 once they escape the basins of attraction surrounding a fitness of 0.6. Furthermore, it is noteworthy that the peaks at 0.22 and 0.32 significantly surpass the other peaks, indicating the remarkable strength of these basins of attraction compared to others. This observation suggests a high probability for agents to become trapped in these particular basins of attraction. Therefore, an efficient solver for the two pendulums coupled inverted pendulum task should possess the capability to effectively escape these basins of attraction.

V. Conclusion

In this paper, we present a method for sampling data to construct Local Optima Networks (LONs) and apply this approach to characterise the training landscape of a neural network used in Reinforcement Learning (RL) within the context of a robotic control problem—the coupled inverted task. The derived solutions are grounded in the physical principles governing dynamic balance and control. The insights gained from the analysis of the fitness landscape, particularly around the basin-hopping strategies and the identification of optimal

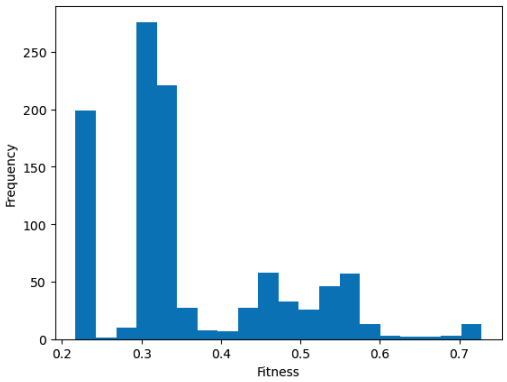

Fig. 5: Fitness histogram of local optima for two pendulums task with perturbation strength 1e-4.

and suboptimal local optima, can potentially inform the development of more adaptive and resilient control algorithms for real world robotics.

The global search in our proposed method is executed through a novel adaptive modification of the Basin-Hopping method. Its purpose is to identify unexplored areas within the search space. Complementing this, the local search is performed by Deep Deterministic Policy Gradient (DDPG), aiming to exploit the basins identified by the global search.

The proposed computational intelligence approach confirms and extends the findings of a classical engineering study. In addition to the previously identified typical basins of attraction, a new narrow but strong suboptimal basin of attraction has now been detected. Knowledge of this additional feature in the landscape will aid engineers and AI practitioners in carefully designing algorithms. More importantly, this study highlights that the problem under consideration is not yet fully understood by the scientific community. Future work will involve exploring alternative basin-hopping techniques and utilising surrogate models to rapidly predict the local performance of DDPG runs. The objective of these efforts will be to expand our understanding of the landscape and potentially identify new unexplored basins of attraction.

Acknowledgements

This work was partially supported by Jiangsu Distinguished Professor Programme.

References

[1] J. Kober, J. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, 2013. [Online]. Available: doi:10.1177/0278364913495721

[2] K. M. Malan and A. P. Engelbrecht, "A survey of techniques for characterising fitness landscapes and some possible ways forward," *Information Sciences*, vol. 241, pp. 148 – 163, 2013.

[3] P. Caamaño, A. Prieto, J. A. Becerra, F. Bellas, and R. J. Duro, "Real-valued multimodal fitness landscape characterization for evolution," in *Neural Information Processing. Theory and Algorithms*, K. W. Wong, B. S. U. Mendis, and A. Bouzerdoum, Eds. Springer, 2010, pp. 567–574.

[4] F. Neri, "Generalised pattern search with restarting fitness landscape analysis," *SN Comput. Sci.*, vol. 3, no. 2, p. 110, 2022.

[5] P. Merz, "Advanced fitness landscape analysis and the performance of memetic algorithms," *Evolutionary Computation*, vol. 12, no. 3, pp. 303–325, 2004.

[6] C. Reeves and J. E. Rowe, *Genetic Algorithms: Principles and Perspectives*. Springer, 2002.

[7] P. Merz and B. Freisleben, "Fitness landscape analysis and memetic algorithms for the quadratic assignment problem," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 4, pp. 337–352, 2000.

[8] K. M. Malan and A. P. Engelbrecht, "Quantifying ruggedness of continuous landscapes using entropy," in *2009 IEEE Congress on Evolutionary Computation*, 2009, pp. 1440–1447.

[9] ——, "A progressive random walk algorithm for sampling continuous fitness landscapes," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, 2014, pp. 2507–2514.

[10] N. D. Jana, J. Sil, and S. Das, "Continuous fitness landscape analysis using a chaos-based random walk algorithm," *Soft Computing*, vol. 22, pp. 921–948, 2018.

[11] G. Ochoa, M. Tomassini, S. Vérel, and C. Darabos, "A study of nk landscapes' basins and local optima networks," in *GECCO*, ser. GECCO08. ACM, Jul. 2008. [Online]. Available: http://dx.doi.org/10.1145/1389095.1389204

[12] N. M. Rodrigues, K. M. Malan, G. Ochoa, L. Vanneschi, and S. Silva, "Fitness landscape analysis of convolutional neural network architectures for image classification," *Information Sciences*, vol. 609, pp. 711–726, 2022.

[13] I. Potgieter, C. W. Cleghorn, and A. S. Bosman, "A local optima network analysis of the feedforward neural architecture space," in *International Joint Conference on Neural Networks (IJCNN)*, 2022, pp. 1–8.

[14] F. Neri and A. Turner, "A fitness landscape analysis approach for reinforcement learning in the control of the coupled inverted pendulum task," in *Applications of Evolutionary Computation*, J. Correia, S. Smith, and R. Qaddoura, Eds. Cham: Springer Nature, 2023, pp. 69–85.

[15] H. Hamann, T. Schmickl, and K. Crailsheim, "Coupled inverted pendulums: a benchmark for evolving decentral controllers in modular robotics," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, 2011, pp. 195–202.

[16] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2016. [Online]. Available: http://arxiv.org/abs/1509.02971

[17] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[18] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[19] C. J. C. H. Watkins and P. Dayan, "Technical note q-learning," *Mach. Learn.*, vol. 8, pp. 279–292, 1992.

[20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[21] J. Adair, G. Ochoa, and K. M. Malan, "Local optima networks for continuous fitness landscapes," in *GECCO*. New York, NY, USA: ACM, 2019, p. 1407–1414. [Online]. Available: https://doi.org/10.1145/3319619.3326852

[22] D. J. Wales and J. P. K. Doye, "Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms," *The Journal of Physical Chemistry A*, vol. 101, no. 28, p. 5111–5116, Jul. 1997. [Online]. Available: http://dx.doi.org/10.1021/jp970984n

[23] C. W. Cleghorn and G. Ochoa, "Understanding parameter spaces using local optima networks: A case study on particle swarm optimization," in *GECCO*. New York, NY, USA: ACM, 2021, p. 1657–1664. [Online]. Available: https://doi.org/10.1145/3449726.3463145

[24] G. Ochoa, S. Verel, F. Daolio, and M. Tomassini, *Local Optima Networks: A New Model of Combinatorial Fitness Landscapes*. Springer, 2014, pp. 233–262.

[25] D. J. Wales and J. P. K. Doye, "Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms," *The Journal of Physical Chemistry A*, vol. 101, no. 28, pp. 5111–5116, 1997.