ANALYSIS OF HYPERPARAMETER TUNING TECHNIQUES:
PROPOSALS, BENEFITS AND FUTURE RESEARCH DIRECTIONS

[1] “Automated Reinforcement Learning (AutoRL): A Survey and Open Problems” by Jack Parker-Holder, Raghu Rajan, Xingyou Song, André Biedenkapp, Yingjie Miao, Theresa Eimer, Baohe Zhang, Vu Nguyen, Roberto Calandra, Aleksandra Faust, Frank Hutter, Marius Lindauer (2022).

[2] “Practical Bayesian Optimization” by Jasper Snoek, Hugo Larochelle, Ryan P. Adams. (2012) - 7598 citations.

[3] "Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization" by Li, J., Jamieson, K., DeSalvo, G., Rostamizadeh, A., and Talwalkar, A. in the Journal of Machine Learning Research (JMLR) in 2018 - 1930 citations.

[4] “A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning” by Brochu, E., Cora, V. M., & de Freitas, N. (2010) - 2529 citations.

[5] “Deep Reinforcement Learning that Matters” by Yan Duan, John Schulman, Xi Chen, Peter L. Bartlett, Ilya Sutskever, Pieter Abbeel (2016) in the Proceedings of the 33rd International Conference on Machine Learning (ICML 2016) - 1718 citations.

[6] “Batch Bayesian Optimization via Local Penalization” by Javier González, Zhenwen Dai, Philipp Hennig, Neil D. Lawrence (2015) - 324 citations.

[7] "Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves." by Tobias Domhan, Jost Tobias Springenberg, and Frank Hutter in Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI), 2016 - 627 citations.

[8] “Fast Bayesian Optimization of Machine Learning Hyperparameters on Large Datasets” by Klein, A., Falkner, S., Bartels, S., Hennig, P., & Hutter, F. (2017) - 563 citations.

[9] “Bayesian optimization with tree-structured dependencies” by Jenatton, R., Archambeau, C., Gonzalez, J., & Seeger, M. (2017) In Proceedings of the 34th International Conference on Machine Learning-Volume 70 (pp. 1655-1664) - 57 citations.

[10] "BOHB : Robust and Efficient Hyperparameter Optimization at Scale " by Stefan Falkner, Aaron Klein and Frank Hutter in 2018 - 838 citations.

[11] “Think Global and Act Local: Bayesian Optimisation over High-Dimensional Categorical and Mixed Search Spaces” by Xingchen Wan, Vu Nguyen, Huong Ha, Binxin Ru, Cong Lu, Michael A. Osborne, ICML 2021.

PAPER	PROPOSAL	BENEFITS OF THE TECHNIQUE	POTENTIAL RESEARCH DIRECTIONS	BRAINSTORMING
[1]	Examines the recent advancements and challenges in the field of Automated Reinforcement Learning. The paper presents an overview of existing techniques and frameworks for automating the design and tuning of reinforcement learning systems, including hyperparameter tuning, algorithm selection, reward shaping, and curriculum learning. Hyperparameter tuning techniques Evaluation methods Environment Design Benchmarks Future Directions	1. Suggest that AutoRL has potential to significantly reduce the amount of time and effort required for hyperparameter tuning in reinforcement learning, as well as to improve the quality of the resulting models. 2. States wide variety of algorithms proposed in AutoRL 3. Methods for Automating Reinforcement Learning (i) Random/Grid Search (ii) Bayesian Optimization (iii) Evolutionary Approaches (iv) Meta-Gradients for Online Tuning (v) Blackbox Online Tuning (vi) Learning Reinforcement Learning algorithms (vii) Environment Design (viii) Hybrid Approaches 4. Benchmarks for algorithm evaluation. 5. The paper provides a comprehensive survey of the existing literature on the topic Hyperparameter Optimization in Reinforcement Learning	1. Random/Grid Search approaches unable to leverage the information regarding promising regions of the hyperparameter search space for making informed decisions. 2. BO-based approaches usually perform static tuning which may not be the most effective for RL. BO approaches with temporal nature of optimization 3. Evolutionary Algorithms :- Computational inefficiency. Disadvantage of PBT methods:- Relative data inefficiency 4. Meta-Gradient methods:- Rely on meta-parameters being initialised well and current meta-gradient methods cannot tune non-differentiable methods. 5. Learning Reinforcement Learning Algorithms 6. Automating better representation of states, actions, rewards - to obtain optimal policy faster	1. Population-based meta-learning 2. Different evaluation metrics 3. Rainbow Algorithm: Combination of several methods to improve stability and performance 4. Fidelities 5. Meta-Gradients on differentiable hyperparameters Differentiable Neural Architecture Search (DARTS) 6. Metrics based on Robust Statistics 7. Dynamic selection of configurations 8. BOIL - modeling training curves 9. Bandits to model degree of exploration, degree of optimism for off-policy methods, amount of diversity to add to population
[2]	Automatic hyperparameter tuning problem within the framework of Bayesian optimization, in which a learning algorithm's generalization performance is modelled as a sample from a Gaussian process. The tractable posterior distribution induced by the GP leads to efficient use of the information gathered by previous experiments, enabling optimal choices about what parameters to try next. BO with Gaussian Process Priors Practical Considerations for Bayesian Optimization of Hyperparameters.	1. Takes into account the variable cost (duration) of learning experiments 2. Leverages the presence of multiple cores for parallel experimentation. 3. States important consideration for practical implementation:- (a) Appropriate choice for covariance function and its hyperparameters (b) The evaluation of the objective function can often take a significant amount of time (c) Optimization algorithms should take advantage of multi-core parallelism 4. Highlights the challenges in hyperparameter optimization. 5. Shows that the effects of the Gaussian process prior and the associated inference procedure can have a large impact on the success or failure of Bayesian optimization. 6. Examines the impact of the kernel itself and whether the default choice of the squared-exponential covariance function is appropriate. 7. Fully Bayesian treatment of expected improvement.	1. Exploration of different covariance functions 2. No clear way to handle categorical variables 3. Incorporating domain knowledge 4. Unable to handle non-stationary functions 5. More advanced acquisition functions 6. The Gaussian Process is too simple to handle complex functions. 7. Thq Acquisition function to choose the next configuration to evaluate on, can be improved. 8. The power of the Gaussian Process to express rich distribution rests solely on the choice of Covariance function. This can be improved.	The measure, “Expected improvement per second” can be used to choose if a configuration is valuable to evaluate. It can also give us insights on early stopping for that configuration. The Fully Bayesian treatment of GP kernel parameters can be extended for oher complex models to approximate the objective function. Improving on the choice of Covariance Functions to work effectively on complex models(with respect to GP model) Two major choices to consider:- 1. Prior over functions to express assumptions about the function being optimized. 2. Choice of acquisition function.

PAPER	PROPOSAL	BENEFITS OF THE TECHNIQUE	POTENTIAL RESEARCH DIRECTIONS	BRAINSTORMING
[3]	Formulates hyperparameter optimization as a pure-exploration non-stochastic infinite-armed bandit problem to speedup random search through adaptive resource allocation and early-stopping.	1. Novel paper on HyperBand Hyperparameter Optimization Algorithm 2. Over order-of-magnitude speedup over other methods 3. Integrates with other "subset selection" algorithms 4. Resource efficiency: Evaluates the configuration further only if the configuration is promising. 5. Scalable to Multi-Fidelity. 6. Easily reproducible 7. Minimal assumptions 8.Efficient exploration of hyperparameter space. 9. Inherently parallel 10. High convergence rate	1. Optimization function computationally expensive 2. Limited in complexity when dealing with complex environments. 3. Integration with other machine learning techniques 4. Better methods for early stopping 5. Inefficient when hyperparameters are correlated 6. Assumes evaluations at different levels of resource allocation as equally informative 7. Lacks guidance/ Unable to leverage the information regarding promising regions for making informed decisions 8. Solely focus on exploration and do not exploit the information	Probabilistic Model to improve the search efficiency by modeling the dependencies between the hyperparameters. Parallelizing the algorithm by forming batches. Adaptive resource allocation based on the confidence of the hyperparameter configuration being successful. Explorative configurations don't get much resources initially + combined with techniques which have a high rate of change of performance in early stages. Multi-fidelity Optimization
[4]	Comprehensive tutorial on the use of Bayesian Optimization to optimize expensive, black-box functions. The hyperparameter values are learned by seeding with a few random samples and maximizing the log-likelihood of the evidence given theta, aided with an informative hyperprior on the hyperparameters. Ideas for Hierarchical Reinforcement Learning.	1. This optimization technique has the nice property that it aims to minimise the number of objective function evaluations. 2. Methodology for selecting prior distribution, acquisition functions and Gaussian Processes. 3. Technique to choose the next best point. 4. Two applications of Bayesian Optimization which includes application to Hierarchical Reinforcement Learning 5. The paper provides a comprehensive overview of the Bayesian optimization literature and summarizes the latest advances and future directions in the field. 6. BO is especially beneficial where the objective function doesn't have a closed form representation. 7. Comparative study of different acquisition functions. 8. Describes the role of Gaussian noise to increase the robustness. 9. Works with non-convex objective functions with multiple local optimums.	1. There are alternative modeling techniques, such as neural networks, TPE, and SVM, that can outperform Gaussian processes in modeling the cost function of BO 2. Conditionally converges to global optimum:- (i) The acquisition function must be continuous and minimizes the expected deviation from global minimum (ii) Objective function must be continuous (iii) Prior is homogeneous (iv) Optimization is independent of the m-th differences. 3. Comparison with other algorithms not presented. 4. Experimental evaluation of the algorithm not presented. 5. The surrogate function, acquisition function can be extended.	Comprehensive analysis on Bayesian Optimization with explanation on the underlying mathematics. We can use BO in our project to Incorporate prior belief about the problem to help direct the sampling, and to trade off exploration and exploitation of the search space. Acquisition functions:- 1. Improvement-based acquisition functions. 2. Exploration-exploitation trade-off. 3. Confidence bound criteria.

PAPER	PROPOSAL	BENEFITS OF THE TECHNIQUE	POTENTIAL RESEARCH DIRECTIONS	BRAINSTORMING
[5]	Investigates challenges posed by reproducibility, proper experimental techniques and reporting procedures. Experiment Analysis 1. Hyperparameters 2. Network Architecture 3. Reward Scaling 4. Random Seed and Trials 5. Environments 6. Codebases “Reporting Evaluation” Metrics 1. Online View vs. Policy Optimization 2. Confidence Bounds 3. Power Analysis 4. Significance	1. First paper to examine reproducibility and good experimental practice in the context of DeepRL. 2. Analysis on the key factors affecting reproducibility. 3. Guidelines to make future results in deep RL more reproducible. 4. Methods to accurately judge improvements offered by novel methods. 5. Emphasis on the importance of benchmarking and comparing different RL algorithms on a suite of standard tasks, to facilitate reproducibility and accelerate progress in the field 6. Demonstrate the benefits of significance testing. 7. Due to the high variance across trials and random seeds of reinforcement learning algorithms, many trials must be run taking into account the different factors that can lead to misleading results.	1. Focuses only on Policy Gradient methods in continuous control. 2. Limited comparison with other approaches: Focuses on DeepRL, does not compare with non-DeepRL methods or the hybrid methods. 3. Limited evaluation in non-standard tasks. 4. Building hyperparameter agonistic algorithms. 5. Analysis done only on four RL algorithms(TRPO, DDPG, PPO, ACKTR) for only three OpenAI gym environments(HalfCheetah, Hopper, Swimmer) 6. Effect of reward scaling on different reinforcement learning algorithms.	We can incorporate the methods mentioned in the paper to account to the different factors which affect the significance testing of our improved algorithm. Using those methods we ensure that the results of our improved algorithm are trustworthy as we have considered the different possibilities of misinterpreting the results obtained. The paper also explains the necessary considerations before reproducing or extending the concepts of a research paper.
[6]	Simultaneously propose batches of hyperparameter values to explore to find the optimal hyperparameter value leveraging parallelization. Proposes a highly effective heuristic, based on an estimate of the function’s *Lipschitz constant* that captures “local repulsion” at negligible overhead. Uses a penalised acquisition function (novel) which penalises the function values of the candidate points according to their local distances to previously evaluated points	1. Improvement of the typical sequential Bayesian Optimization method, allows multiple points to be evaluated simultaneously. 2. Inherently parallel 3. Higher convergence rate 4. Scalable to high-dimensional search spaces. 5. Probabilistic framework to approximately infer the Lipschitz constant 6. This method allows us to place bounds on how far the optimum of f is, from a certain location. 7. Improved Exploration-Exploitation tradeoff - The local penalization technique uses a penalty term into the acquisition function that encourages the algorithm to explore different regions of the input space while also exploiting regions that are likely to yield high function values. 8. Direct penalization of the acquisition function around its most recent maximum. 9. The effect of a local penalizer is to smoothly reduce the value of the acquisition function in the neighbourhood of x.(instead of modelling it as narrowly peaked) 10. Consider the interdependencies between different batches of function evaluations.	1. Computational burden due to the optimization-marginalisation loop 2. The heuristic used to estimate a function's Lipschitz constant can be improved. 3. The objective function may not obey the requirement that it must be a Lipschitz continuous function. 4. Main challenge of this technique is that the Lipschitz constant is unknown. 5. Transfer learning 6. Discovering optimal batches 7. Better models for to incorporate Local Penalization	Find other better techniques similar to this, which also allows us to place bounds on how far the optimum of f is, from a certain location. Proposing batches of hyperparameter configurations to exploit parallelism can be incorporated in our project. The paper also considers dependencies between different batches of function evaluations, which is one of the major drawbacks of parallelizing evaluations. The BOIL algorithm can be used to model the training curves. The method transforms the whole training curve into a numeric score to represent high vs low-performing curves. Instead of producing a single numeric value, we can generate a vector of values to better tell about the characteristics of the training curve.

PAPER	PROPOSAL	BENEFITS OF THE TECHNIQUE	POTENTIAL RESEARCH DIRECTIONS	BRAINSTORMING
[7]	Method for accelerating the hyperparameter optimization process by extrapolating the learning curve of a deep neural network. By extrapolating the learning curve, the method can estimate the performance of the model with different hyperparameter settings, without the need to train the model from scratch. Constructs a probability model that extrapolates the learning curve of a hyperparameter setting to quickly detect poorly performing configurations. Devises a *predictive termination criterion* that can be combined with any hyperparameter optimization method.	1. Novel paper on Extrapolation of learning curves to speed up hyperparameter optimization of DNNs. 2. Agonistic to the hyperparameter optimizer used. (Can be integrated with other hyperparameter optimizers) 3. Transferability - Information gained in analysing. 4. Reduces computational costs 5. Captures a diverse set of learning behaviours of standard learning curves. 6. Finds maximum likelihood estimate for all the parameters. MCMC to sample from the posterior distribution of the model parameters given the earning curves. 7. Improved reproducibility	1. Unable to handle non-stationary data efficiently. 2. Lacks the explicit incorporation of prior knowledge or domain expertise into the hyperparameter optimization process 3. Does not balance exploration-exploitation of hyperparameter configurations. 4. Does not take into account that the model is initially exploring more hence performs worse. 5. Categorical hyperparameters not handled. This method considers all hyperparameters to be continuous. 6. Assumes non-stochastic behaviour of the learning curve. 7. Potential mis-evaluation due to choice of validation environment. 8. Risk of losing hyperparameter configuration who have promising long term performance.	The idea of using a model to extrapolate the learning curve by observing the initial improvements, to perform early stopping if necessary can be highly beneficial to problems with high dimensional search spaces. In the paper, a simple linear regression model was used to extrapolate the learning curve. We can improve this approach by using a more complex model which also considers the dependencies between hyperparameters and dependencies between hyperparameter configurations to predict the long term performance of the current configuration. This can be integrated with the Hyperband approach to use resources efficiently. Alternatively, we can use Bayesian regression to model the uncertainty in the curve and make more accurate predictions.
[8]	Novel approach with a focus on scalability for large datasets. To accelerate hyperparameter optimization, we propose a generative model for the validation error as a function of training set size, which is learned during the optimization process and allows exploration of preliminary configurations on small subsets, by extrapolating to the full dataset.	1. Leverage dataset size as an additional degree of freedom enriching the representation of the optimization problem 2. Pick a small subset of the training data and see how well the model does. 3. FABOLAS models loss and computational cost across dataset size and uses these models to carry out Bayesian optimization with an extra degree of freedom. 4. Uses Marten 5/2 kernel in its Automatic Relevance Determination form contrast to the standard Gaussian kernel, it makes less restrictive smoothness assumptions. 5. Uses the Mahalanobis distance to account for the covariance between variables. 6. This technique has a very high rate of change of performance in early stages.	1. Distributed version of the algorithm 2. If the optimization problem is stationary, the methods may not be effective. 3. No specific way to handle categorical variables in the proposed method 4. Complex kernels and acquisition functions 5. Dynamic surrogate model	Leverages the configuration's performance on small datasets to model the effectiveness of the configuration if trained on the whole dataset. This paper takes into account single fidelity parameter - Training set size This idea can be extended to a Multi-Fidelity model to account for other important fidelities.

PAPER	PROPOSAL	BENEFITS OF THE TECHNIQUE	POTENTIAL RESEARCH DIRECTIONS	BRAINSTORMING
[9]	Describes a new approach to Bayesian Optimization that takes into account the dependencies between hyperparameters. The authors propose a model that represents the joint distribution of hyperparameters using a tree structure, where each node in the tree corresponds to a hyperparameter and the edges represent the conditional dependencies between the hyperparameters. The authors show that this approach can improve the efficiency of Bayesian optimization by reducing the number of function evaluations required to find the optimal hyperparameters	1. Novel surrogate model for Bayesian Optimization which combines independent Gaussian Processes with a linear model that encodes a tree-based dependency structure and can transfer information between overlapping decision sequences. 2. As the constructed surrogate model is tailored to the structure, we can reduce the number of evaluations required. 3. The same structure also speeds up acquisition function decisions. 4. Acquisition function which is able to exploit the tree structure 5. This technique can cover the weakness of the Hyperband Algorithm by reducing the search space by a huge factor. 6. We explore the search space more efficiently and posterior inference scales more favourably with the number of observations than Gaussian-Process based approaches. 7. In Reinforcement Learning many hyperparameters are correlated due to the special reason that they are responsible to make decisions. 7. Tree structured kernel to model the dependencies between hyperparameters. 8. A more efficient exploration of hyperparameter space by exploiting the dependency structure.	1. Dependence on predefined distributions for sampling hyperparameters and partitioning the input space. 2. Overfits. 3. Evaluation of algorithms done only on a small set of benchmark datasets. 4. TPE can easily get stuck in local optima. 5. Multi-level tree not explored (required in cases where dependencies among hyperparameters are high). 6. No methods provided for scaling of algorithms to larger search spaces. 7.Hyperparmaters may not have tree-structured dependency, instead the hyperparameters can have dependency on many other hyperparameters. 8. Requires significant tuning: The choice of kernels is crucial to achieve optimal performance. 9. Exploring the impact of different kernel functions. 10. Integration with deep learning	The idea to extract the dependencies between hyperparameters of RL algorithm can be incorporated into our project, since in complex environments the dependencies are expected to be very high due to the co-working of the hyperparameters to attain highest reward. In the paper, the authors propose tree structured dependencies. We believe the employing of the Bayesian Neural Network to model the dependency structure between the different related hyperparameters can be highly beneficial. This allows us to capture many layers of dependencies effectively which can be very beneficial in complex environments.
[10]	Combine the benefits of both Bayesian Optimization and Bandit-Based Methods	1. Best of both worlds BO: Leveraging prior information to explore promising regions HB: Exploration infinite-armed bandit adaptive resource allocation 2. Robust to noisy objectives 3. Strong anytime performance 4. Fast convergence to optimal configurations 5. Versatile 6. Inherently parallel	1. There are estimators that outperform TPE in certain scenarios 2. Assumes reliable estimation given the limited number of trials 3. Computationally expensive for large hyperparameter spaces 4. Assumes independence of hyperparameters 5. Assumes objective function is static 6. Limited applicability to online learning algorithms	Bayesian Optimization can be used to use the information gained from previous evaluations of different hyperparameter configurations. Hyperband algorithms can be used to introduce multiple fidelities initially then gradually increasing the fidelity based on whether the configuration is expected to better with the more fidelity provided. A model can be constructed to assess the significance of increasing a particular fidelity. This can be used to choose the next configuration and fidelity level.

PAPER	PROPOSAL	BENEFITS OF THE TECHNIQUE	POTENTIAL RESEARCH DIRECTIONS	BRAINSTORMING
[11]	Combines local optimisation with a tailored kernel design, effectively handling high-dimensional categorical and mixed search spaces, whilst retaining sample efficiency. Derives convergence guarantee for the proposed approach.	1. Novel GP-based BO approach using tailored kernels and trust regions 2. Used kernel design to incorporate both categorical and continuous variables. 3. The use of a mixed kernel allows for different types of covariance structures to be included in the model, which can better capture the underlying correlations. 4. Kernel is positive definite which guarantees convergence to global optima 5. Trust region for categorical space defined in terms of Hamming distance 6. GP-UCB principle to restart the method 7. Instead of using multi-bandit for categorical parts and acquisition function for continuous parts, both the categorical and continuous inputs are handled by a single, unified GP. 8. Can be integrated with	1. Acquisition function cannot be optimised via gradient-based techniques. 2. Uses Local search on categorical variables 3. No clear way to deal with highly imbalanced categorical variables. 4. Not parallelized 5. Dynamic search spaces 6. Kernel design can be improved 7. Assumption that the objective function is smooth and continuous. 8.	The methods used in this paper can be incorporated in our paper to handle high-dimensional categorical search. This can be integrated with continuous search space methods to get better exploration on large dimensional spaces. This method can also be integrated with other optimization techniques such as gradient based optimization to improve its efficiency and performance.