Inferring Structure and Parameters of Dynamic Systems using Particle Swarm Optimization

Muhammad Usman
Department of Computing Science, University of Aberdeen
Aberdeen, UK
m.usman.17@abdn.ac.uk

Abubakr Awad
Department of Computing Science, University of Aberdeen
Aberdeen, UK
abubakr.awad@abdn.ac.uk

Wei Pang
Department of Computing Science, University of Aberdeen
Aberdeen, UK
pang.wei@abdn.ac.uk

George M. Coghill
Department of Computing Science, University of Aberdeen
Aberdeen, UK
g.coghill@abdn.ac.uk

ABSTRACT

Inferring models of dynamic systems from their time series data is a challenging task for optimization algorithms due to its potentially expensive computational cost and underlying large search space. In this study, we aim to infer both the structure and parameters of a dynamic system model simultaneously by Particle Swarm Optimization (PSO), enhanced by effective stratified sampling strategies. More specifically, we apply Latin Hyper Cube Sampling (LHS) with PSO. This leads to two novel swarm-inspired algorithms, LHS-PSO which can be used efficiently to learn the structure and parameters of simple and complex dynamic system models. We used a complex biological cancer model called Kinetochores, for assessing the performance of PSO and LHS-PSO. The experimental results demonstrate that LHS-PSO can find promising solutions with corresponding structure and parameters, and it outperforms PSO during our experiments.

CCS CONCEPTS

• **Mathematical optimization → Continuous optimization**; *Bio-inspired optimization*;

KEYWORDS

Dynamic Systems, Particle Swarm Optimization, Genetic Algorithm, Latin Hypercube Sampling, Learning Structure and Parameter

ACM Reference Format:
Muhammad Usman, Abubakr Awad, Wei Pang, and George M. Coghill. 2019. Inferring Structure and Parameters of Dynamic Systems using Particle Swarm Optimization. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3319619.3322010

GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6748-6/19/07…$15.00
https://doi.org/10.1145/3319619.3322010

1 INTRODUCTION

Real-time complex dynamic systems are commonly modelled by Ordinary Differential Equations (ODEs). An *ODE* model captures the dynamic behaviour of a system evolving over time. An ODE model has a model structure and a set of parameters associated with its structure. The dynamic behaviour of an ODE model is determined by both of its structure and parameter values.

Modelling the dynamic behaviour of a dynamic system from its time series data requires us to identify the parameters and structure of an ODE through simulation and compare the simulation results with the target data. Moreover, time series data is used to assess the performance of the target model in order to conclude the values of parameters and structure of an ODE model. The general form of an ODE is presented in Equation.1, where X_i is the state variable, n is the number of observable components and f_i is the relationship among variables.

$$\frac{dX_i}{dt} = f_i(X_1, X_2, ..., X_n)(i = 1, 2, ..., n) \tag{1}$$

Particle Swarm Optimization (PSO) is a well known meta-heuristic algorithm which provides a sufficiently good solution to an optimization problem, commonly with incomplete or imperfect information. Highlighting the broader scope and the complexity in dynamic system, it put forward us to search for an efficient sampling technique to be used with PSO. We explore the use of stratified sampling approach called Latin Hypercube Sampling (LHS) , as it has have been proven to be effective approach in many problems.

2 RELATED WORK

There have been several studies for identifying ODE models. In [7], the authors demonstrated the use of an evolutionary method for recognizing a gene regulatory network from time series data. *Yang et al*. [7] used Multi Expression Programming (MEP) and Particle Swarm Optimization (PSO) to solve structural optimization problems. They inferred the right-hand side of ODE by MEP and optimized its parameters by PSO so as to obtain the superlative structure of ODE from a small population. To reduce its complexity they partitioned the search space in order to achieve better performance and aimed to deploy the approach on real large-scale biochemical network problems.

Darania et al. [1] established a substantial work to approximate the solutions of differential equation using swarm intelligence. *Tan*

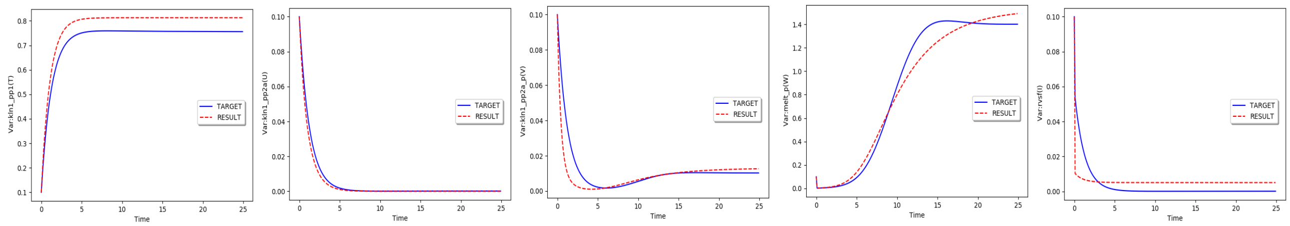

Figure 1: Time course data of variable `kln1_pp1(T)`,`kln1_pp1_2a(U)`, `kln1_pp2a_p(V)`, `melt_p(W)` and `rvsf(I)` respectively

et al. [4] presented a shared method to a series of solutions of non-linear differential equations. *Lee et al.* [3] estimated the solutions of non- linear differential equations using the Method of Bilaterally Bounded (MBB). *Mateescu et al.*[6] suggested a modest formulation for solving Cauchy problem using Genetic Algorithm (GA). *Mastorakis et al.* [5] presented various new ideas on inferring an ODE as an optimization problem. They used finite element models to advance the fitness functions.

Eberhart et al. [2] optimized the parabolic function using PSO. They kept the population size of PSO 20 in order to have a quick output because the system performance is sensitive to the number of iterations and population.

3 METHODOLOGY

In this paper, PSO and LHS-PSO are analyzed to infer both the structure and parameter of an ODE at the same time. In order to achieve our goal, we are taking a real biological cancer tumour model called *kinetochore* as a testing model, which consists of five ODEs as described in Equations.2~ 6. These equations express the rate of change of components d/dt.

$$\frac{dT}{dt} = 0.9 * I * 1 * 10.24 - T * 0.0012 \tag{2}$$

$$\frac{dU}{dt} = V * T * 0.0154 - U * 1 * 0.81 \tag{3}$$

$$\frac{dV}{dt} = W * 0.1 * 0.12 + U * 1 * 0.81 - V * 1.64 - V * T * 0.015 \tag{4}$$

$$\frac{dW}{dt} = V*1.64+0.7*1*0.47-W*0.1*0.12-W*U*1591.5*1 \tag{5}$$

$$\begin{aligned}\frac{dI}{dt} &= T * 0.0012 + 0.8 * U * 1410.79 * 1 - 0.9 * I * 1 * 10.24 \\ &\quad -I * 1 * 1934.77 * 1\end{aligned} \tag{6}$$

3.1 The LHS-PSO Algorithm

A widely used sampling technique is LHS, which stratifies the input probability distributions. It ensures that the samples are distributed across the entire search space evenly. LHS-PSO improves the original PSO algorithm by assigning the initial position to each particle rather than initializing the particle position randomly. Our proposed LHS-PSO algorithm is shown in Table.4.

Our goal of using this algorithm is to reform a better multivariate space by eliminating redundant realizations from the initial pool and to make it easy for PSO to search for better structure and parameter in the search space.

Table 1: LHS-PSO Algorithm

LHS-PSO Algorithm
1) Initialize population.
1.1) The population is initialized randomly.
1.2) The population is initialized with LHS.
2) Calculate the fitness value of each particle.
3) Population evolution.
3.1) Update velocity and position of each particle.
5) Evaluate termination condition
5.1) Yes, stop iteration
5.2) No, go to Step (2)

4 EXPERIMENTAL RESULTS

We applied LHS-PSO to infer the structure and parameters of dynamic system ODEs and test the effectiveness of these algorithms. The corresponding target and resultant ODE models for LHS-PSO are shown in Figure.1, which proves that LHS-PSO algorithm have a better ability to search for parameters and structure of the system of ODEs by using known time course data.

ACKNOWLEDGMENTS

Muhammad Usman and Abubakr Awad are supported by Elphinstone PhD Scholarship (University of Aberdeen).

REFERENCES

[1] M Babaei. 2013. A general approach to approximate solutions of nonlinear differential equations using particle swarm optimization. *Applied Soft Computing* 13, 7 (2013), 3354–3365.
[2] Russell C Eberhart and Yuhui Shi. 2001. Tracking and optimizing dynamic systems with particle swarms. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, Vol. 1. IEEE, 94–100.
[3] Zong-Yi Lee. 2006. Method of bilaterally bounded to solution Blasius equation using particle swarm optimization. *Appl. Math. Comput.* 179, 2 (2006), 779–786.
[4] S Liao and Y Tan. 2007. A general approach to obtain series solutions of nonlinear differential equations. *Studies in Applied Mathematics* 119, 4 (2007), 297–354.
[5] Nikos E Mastorakis. 2006. Unstable ordinary differential equations: solution via genetic algorithms and the method of Nelder-Mead. *WSEAS Transactions on Mathematics* 5, 12 (2006), 1276.
[6] George Daniel Mateescu et al. 2006. On the application of genetic algorithms to differential equations. *Romanian Journal of Economic Forecasting* 7, 2 (2006).
[7] Bin Yang, Yuehui Chen, and Qingfang Meng. 2009. Inference of differential equation models by multi expression programming for gene regulatory networks. In *International Conference on Intelligent Computing*. Springer, 974–983.