ICLR 2024 Workshop on AI4DifferentialEquations In Science

Optimizing Computationally Intensive Simulations Using a Biologically-Inspired Acquisition Function and a Fourier Neural Operator Surrogate

John Lins[1]; Wei Liu[2], PhD

[1]University of California, Santa Barbara; [2]Lawrence Livermore National Laboratory

SUMMARY

Achieving optimal simulation results often requires adjusting several key control parameters by comparing many realizations of the intensive simulations. Manually tweaking multiple control parameters can be tedious and inefficient. To alleviate such obstacles in simulation studies, we found that differential evolution combined with Neural Network surrogates could effectively optimize simulations. The differential evolution samples the multi-dimensional optimization space following a multivariate Gaussian distribution. The samples collected may be used to train intelligent surrogate models such as a Fourier Neural Operator (FNO). Once a surrogate is constructed, it could further accelerate the sampling of the optimization search space. This methodology effectively optimized a hypersonic simulation modeled by systems of partial differential equations; it may also be extended to simulation optimization in other disciplines.

MOTIVATION

The current state of high-fidelity multi-physics simulation is marked by its computational intensity. Tweaking multiple control parameters often poses a significant challenge, making it difficult to converge quickly.

Multiple control parameters can influence each other, and the multi-dimensional parameter space can be complex to search.

This study investigates a scheme that

- leverages both exploration and exploitation in the search space,
- systematically optimizes simulation results,
- efficiently automates the process and returns users with optimized control parameters.

METHOD

Differential Evolution with Gaussian Mutation (DE-GM) is a genetic optimization algorithm that optimizes non-convex loss landscapes. The evolution starts with a random distribution over the search space, mutates the population following a multivariate Gaussian distribution, and selects the better population as a result of crossover and loss evaluation. This method is not the same as CMA-DE or DES because we never perform a difference of vectors. A difference of vectors may cause expensive simulation samples to be flung far away from the rest of the population.

Initialization

Mutation

$$\vec{m_i} := \vec{v_1} + \alpha_m \cdot (\vec{v_2} - \vec{v_3}) \quad \text{where } \vec{v_1} \neq \vec{v_2} \neq \vec{v_3} \text{ and } \vec{v_1}, \vec{v_2}, \vec{v_3} \in \mathcal{P}$$

M: Mutant vectors from multivariate Gaussian
P: Current population
T: Trial

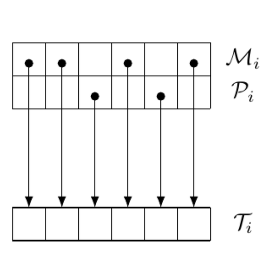
Covariance

$$X \sim \mathcal{N}(\vec{\mu}_{\mathcal{P}}, \vec{\sigma}_{\mathcal{P}})$$

$$\vec{m_i} := X$$

Crossover

$$\mathcal{T}_{i,j} = \begin{cases} \mathcal{M}_{i,j} & rand(0,1) \geq \alpha_c \\ \mathcal{P}_{i,j} & otherwise \end{cases}$$



Selection

$$\mathcal{N}_i = \begin{cases} \mathcal{T}_i & loss(\mathcal{T}_i) < loss(\mathcal{P}_i) \\ \mathcal{P}_i & otherwise \end{cases}$$

Next population

Bounds — Each parameter's search space is bounded

Convergence — Average loss diminishing below tolerance

Test on Ackley function with many local minima and one global minimum

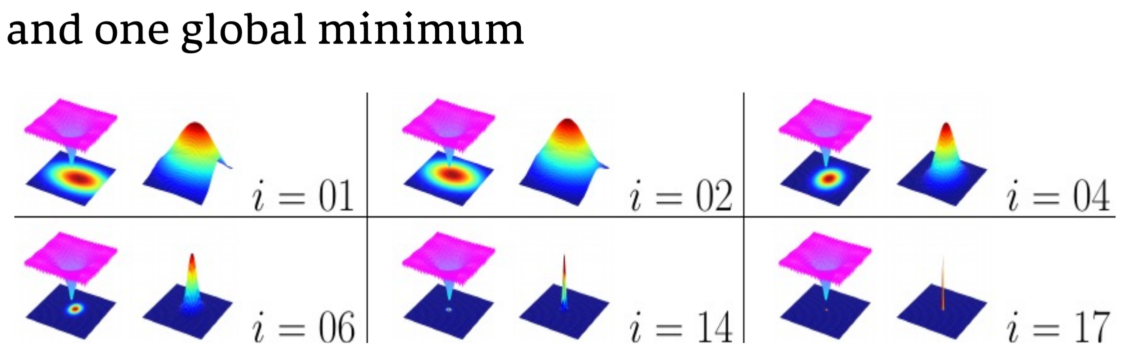


Fig. 1. Convergence of sampling guided by multi-variate Gaussian distribution. Note: 'i' means iteration; the left figure in each cell is the heat map, the right figure is the population Gaussian.

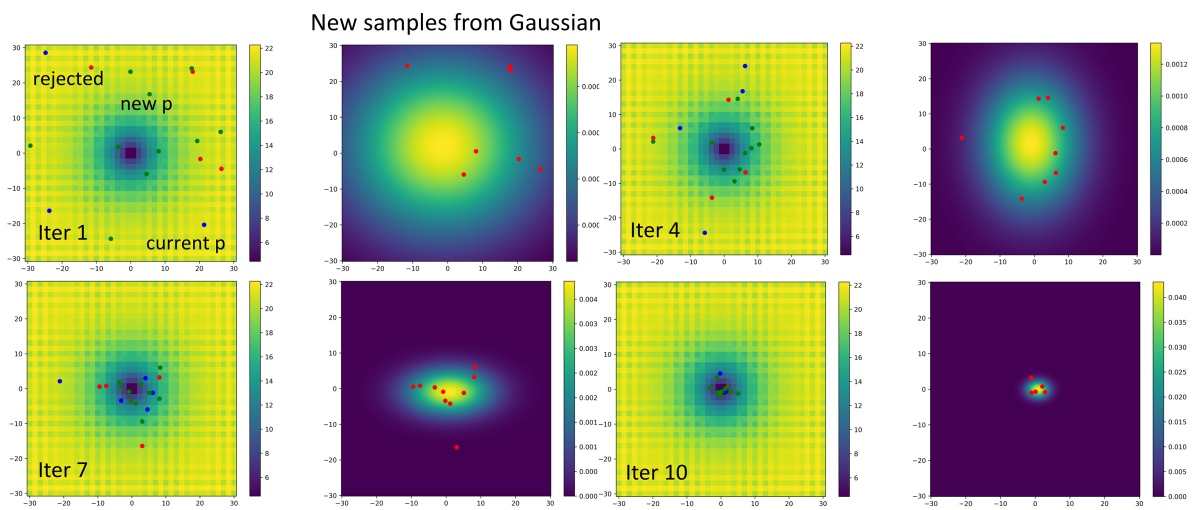
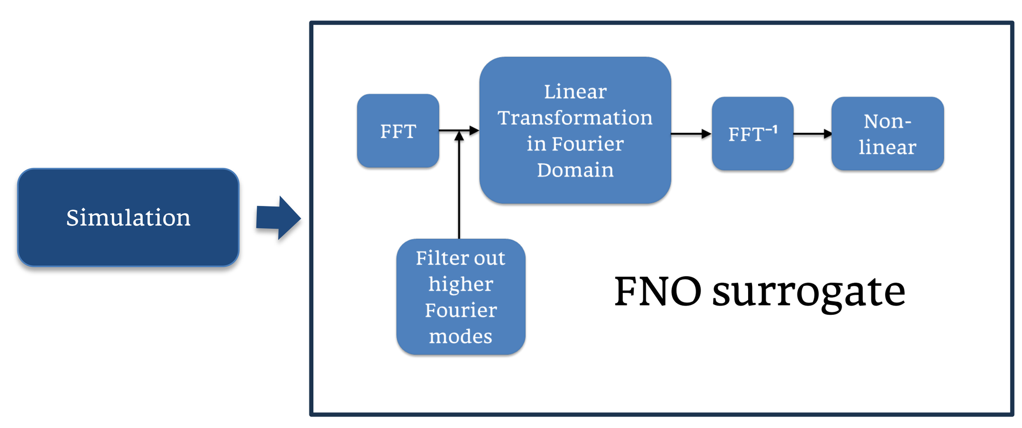


Fig. 2. Evolution of population (samples) through iterations. New samples suggested from the Gaussian distribution participate in cross-over and selection to form new population.

The population became less scattered with more iterations, and the search converged to the global minimum. Multivariate Gaussian distribution accelerates the exploitation of the loss landscape by constraining mutation vectors from being flung far away from the rest of the population.

Simulation NN-surrogate, trained by samples selected by DE-GM, can accelerate the sampling of the loss landscape. The surrogates don't have to be highly accurate pointwise but enough to tell the gradient descent direction. FNO naturally approximates global continuous functions as solutions to differential equations and is computationally efficient in the Fourier domain.

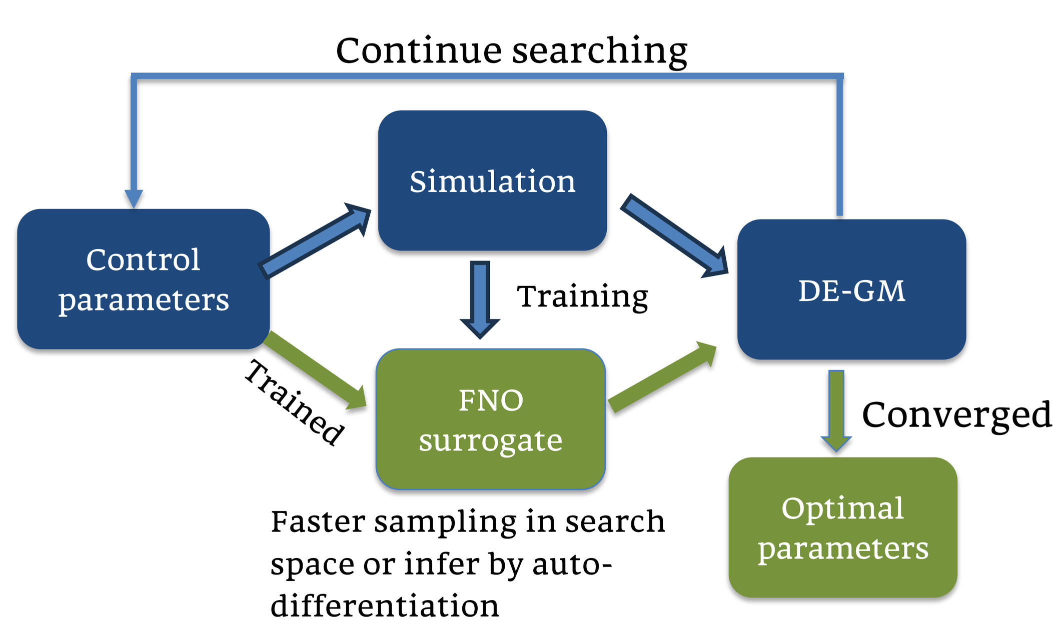


Non-linear activation, Low-pass, FFT

$$v_{t+1}(x) := \sigma(Wv_t(x) + \mathcal{F}^{-1}(\zeta[R_\theta(\mathcal{F}v_t(x))]))$$

Trainable parameters

DE-NN Search and Optimization Workflow



SIMULATION TEST

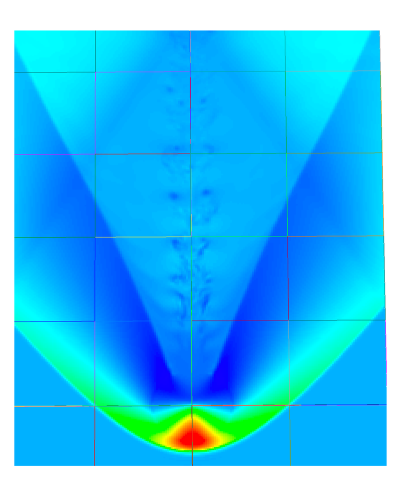


Fig. 3. Hypersonic simulation by MARBL(eul)

Parameter	Description	Default
c_m	Artificial shear viscosity coefficient	0.0001
c_b	Artificial bulk-viscosity dilatation coefficient	0.07
c_k	Artificial thermal conductivity coefficient	0.001
c_l	Ceiling on artificial bulk viscosity	1.0
c_c	Ceiling on artificial thermal conductivity	4.0
c_f	Ceiling on artificial diffusivity	4.0

Table 1. Control parameters of artificial viscosity in hydrodynamic simulation

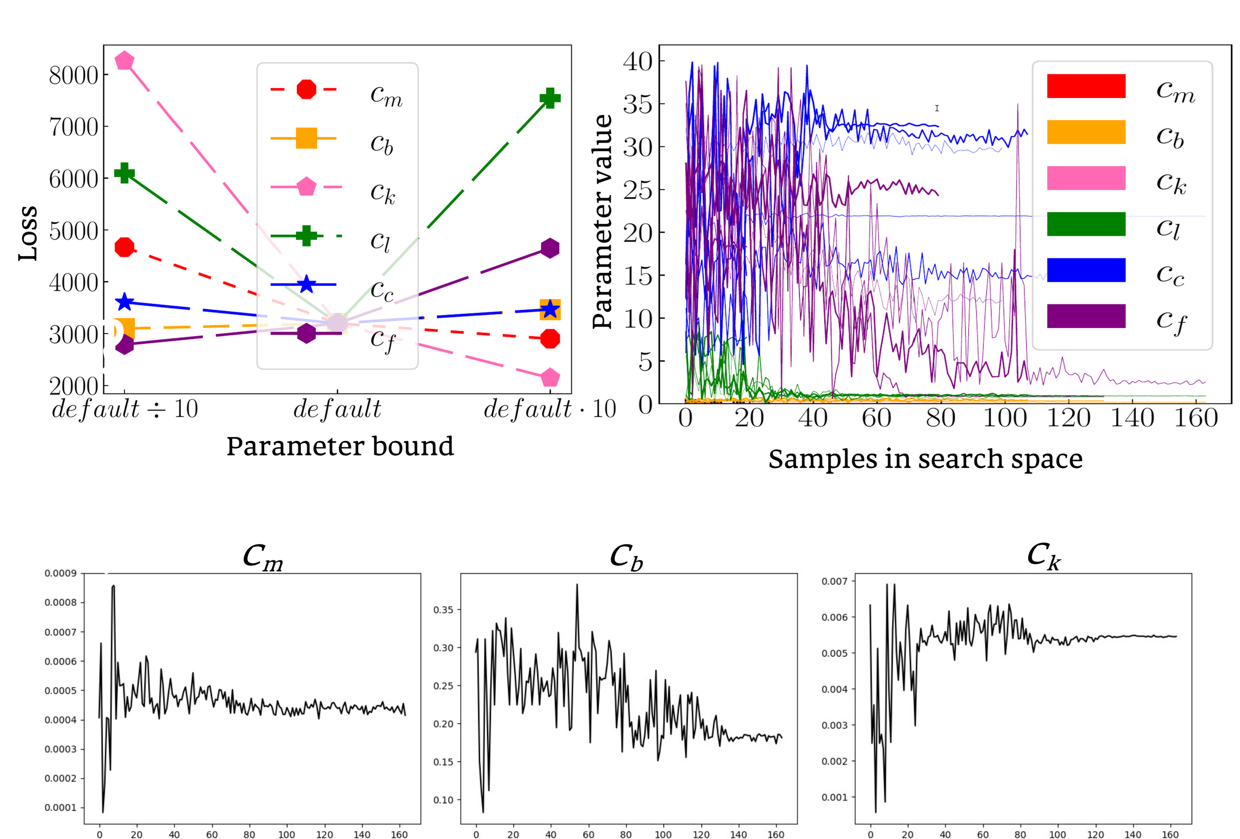


Fig. 4. Zoomed-in convergence of specific control parameters to optimize stability (the loss is a metric of numerical stability such as time step size)

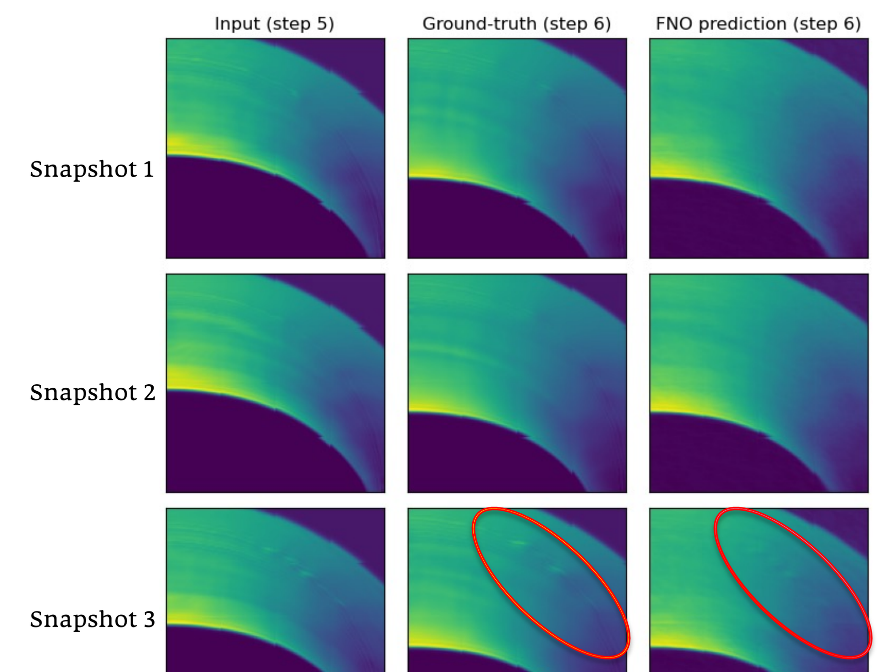


Fig. 5. FNO surrogate gets trained and predicts the next simulation result from sparse training data

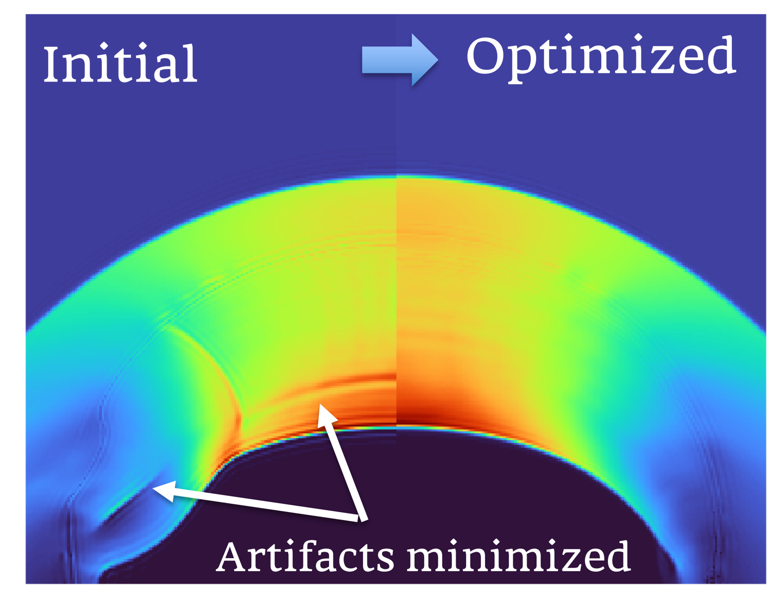


Fig. 6. Hypersonic simulation optimized after searching six control parameters driven by a stability criterion

CONCLUSION

This study proposes and evaluates a workflow integrating multivariate Gaussian-based differential evolution and neural network surrogates to accelerate the search for optimal control parameters for computationally intensive simulations.

The covariance matrix-based differential evolution through a multivariate Gaussian distribution effectively optimizes six key control parameters in a hypersonic simulation with fairly quick convergence and turnaround.

The NN-based surrogate training on hydrodynamic simulation is promising because the output captures the characteristic numerical oscillation with limited training data. It could be used to tell the direction of gradient descent and accelerate sampling of the search space.

The approach can be extended to other kinds of simulations. Users can also construct multi-objectives in the optimization search.

ACKNOWLEDGEMENT

We thank Dr. Britton Olson for his assistance with the MARBL(eul) software, Dr. Andrew Cook and Calvin Young for assistance with the hypersonic simulation, and Dr. Qingkai Kong for his assistance with the Fourier neural operator. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC.

REFERENCES

[1] Andrew W Cook. Artificial fluid properties for large-eddy simulation of compressible turbulent mixing. Physics of fluids, 19(5), 2007.
[2] Andrew W Cook. Enthalpy diffusion in multicomponent flows. Physics of Fluids, 21(5), 2009.
[3] Swagatam Das and Ponnuthurai Nagaratnam Suganthan. Differential evolution: A survey of the state-of-theart. IEEE transactions on evolutionary computation, 15(1):4–31, 2010.
[4] David Gottlieb and Chi-Wang Shu. On the gibbs phenomenon and its resolution. SIAM review, 39(4): 644–668, 1997.
[5] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. Neural networks, 2(5):359–366, 1989.
[6] Christian Igel, Nikolaus Hansen, and Stefan Roth. Covariance matrix adaptation for multi-objective optimization. Evolutionary computation, 15(1):1–28, 2007.
[7] Nikola B. Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew M. Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. CoRR, abs/2108.08481, 2021.
[8] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:2010.08895, 2020.