Towards Smart Coupling in Multiphysics Simulation

R&D Defense

Deepan Chakravarthi Padmanabhan
February 17th, 2020

Supervisors
Prof. Dr. Paul G. Plöger
Ing. -Inf. Pascal Bayrasy
M. Sc. Ahmad Delforouzi

Hochschule Bonn-Rhein-Sieg
University of Applied Sciences

Fraunhofer SCAI

Table of Contents

Introduction

Introduction

- Multiphysics simulation: The process of studying the interaction between the multiple physical domains in a simulated environment [1].
- Fluid-Structure Interaction (FSI) is a multiphysics simulation.
- Application domains: Automobile [2], robotics [3], and biomedical [4].

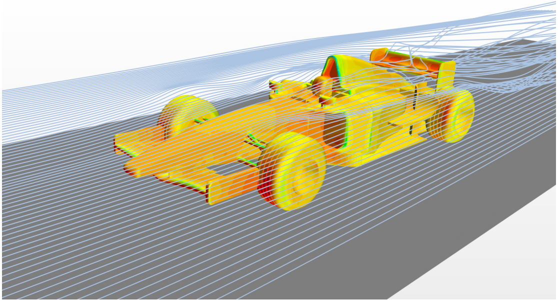


Figure 1: Aerodynamics simulation study of a F1 car [2].

Introduction

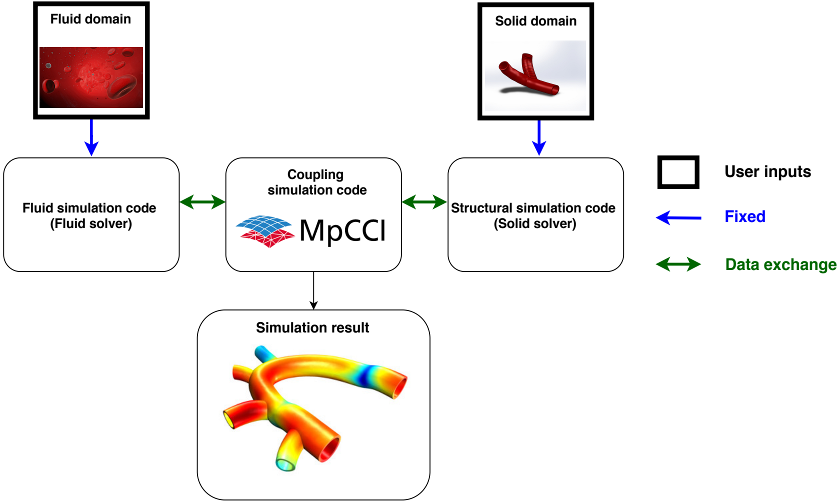


Figure 2: Multiphysics simulation with a coupling tool. MpCCI - Mesh-based parallel Code Coupling Interface. Images adapted from [4] [5] [6].

Introduction

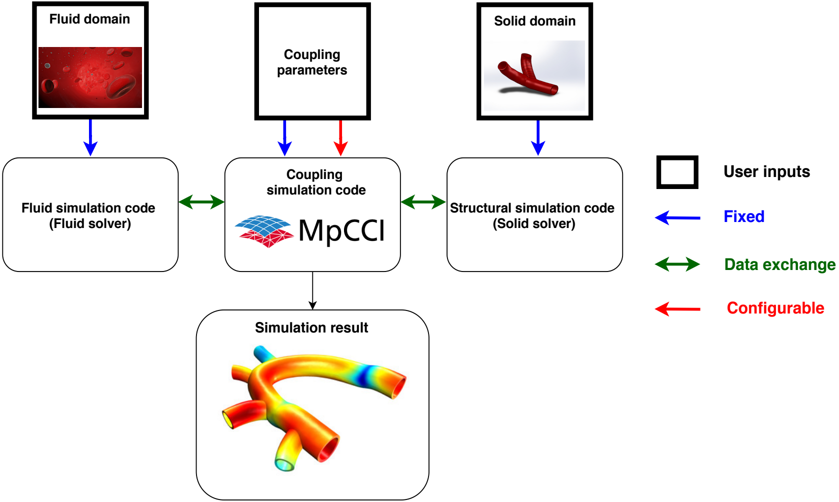


Figure 2: Multiphysics simulation with a coupling tool. MpCCI - Mesh-based parallel Code Coupling Interface. Images adapted from [4] [5] [6].

Problem statement

- Manually fine-tuning coupling tool parameters is tedious and time-consuming.
  - Numerous configurable parameters (approximately 15 parameters).
  - Different types of configurable parameters.
  - Parameters exhibit forbidden and conditional dependencies.

Motivation

- Co-simulation setting up procedure.
  - No assistance to set coupling parameters for FSI simulations.
  - One simulation to study aerodynamics of aircraft $\Longrightarrow$ upto weeks [7] [8].
  - Repeated testing with numerous designs demands a robust coupling tool.

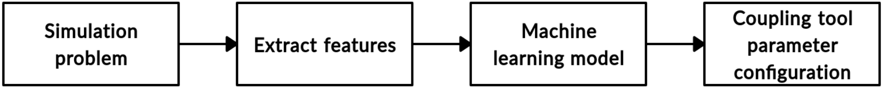
Motivation

- Co-simulation setting up procedure.
  - No assistance to set coupling parameters for FSI simulations.
  - One simulation to study aerodynamics of aircraft $\Longrightarrow$ upto weeks [7] [8].
  - Repeated testing with numerous designs demands a robust coupling tool.
- Coupling tool users.
  - Cambridge University
  - CERN
  - Universität Bonn
  - Airbus
  - Daimler
  - Volkswagen

Proposed strategy

Proposed solution

- Automatically predict coupling tool parameter configurations.
- Configurations with relatively lesser runtime than default configuration.



Proposed solution

- Automatically predict coupling tool parameter configurations.
- Configurations with relatively lesser runtime than default configuration.

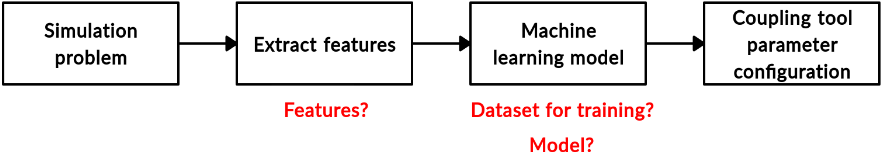
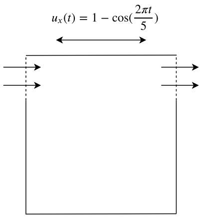


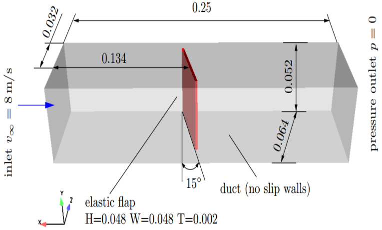
Figure 3: Proposed solution with challenges. Red markings indicate the challenges.

Simulation problem and features

- FSI simulation problems: 3D driven cavity and elastic flap.



(a) 2D cross section of 3D driven cavity.



(b) Elastic flap (Dimension in meter).

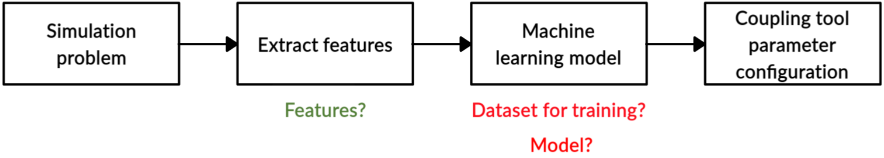
Figure 4: Geometry of the simulation problems [2].

- Features: Fluid density, fluid viscosity, solid elasticity, Poisson's ratio and solid-fluid density ratio (based on [9] [10]).

- Each simulation instance of a problem has different feature values.

Proposed solution

- Automatically predict coupling tool parameter configurations.
- Configurations with relatively lesser runtime than default configuration.



Hyperparameters of the coupling tool

Parameter, P	Parameter type	Domain of the parameter, D
Coupling scheme	Categorical	Implicit-Transient:Implicit-Transient, Explicit-Transient:Explicit-Transient
Initial exchange	Categorical	exchange:exchange, receive:exchange, exchange:receive
Relaxation-0	Categorical	True, False
Relaxation-1	Categorical	True, False
Relaxation method	Categorical	Quasi-Newton, Fixed, Aitken, Ramping
Anderson mix	Categorical	Least squares, Standard, Inverse
Number of levels	Integer	[0, 16]
Omega	Float	[1e-07, 2.0]
Ramp-0	Float	[0.0, 2.0]
Ramp-d	Float	[0.001,1.0]
Relaxation factor	Float	[0.0,2.0]

Table 1: Hyperparameters of the tool. The coloring represents the existence of the parameter, P only if the respective color is selected in domain, D for all the preceding parameter.

Dataset generation

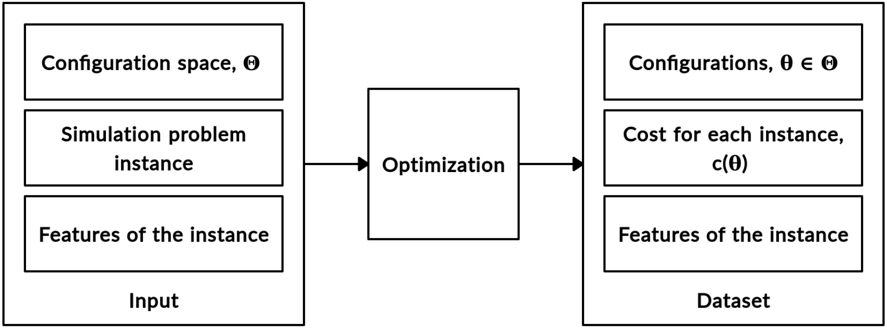


Figure 5: Dataset generation overview. **Cost of each instance is the simulation runtime (in seconds)** for the respective parameter configuration.

Optimization: Sequential Model-based Algorithm Configuration (SMAC)

- Random Forest (RF) surrogate[1] and acquisition function[2].
- Expected Improvement (EI) - how promising is a configuration depending on the model constructed?

[1] surrogate - substitute. A model mimicking the coupling tool.
[2] To select next promising point for evaluation on the target algorithm.

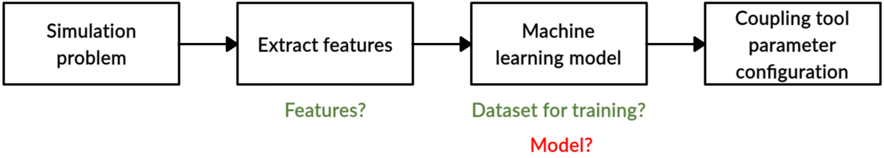
Optimization: Sequential Model-based Algorithm Configuration (SMAC)

- Random Forest (RF) surrogate[1] and acquisition function[2].
- Expected Improvement (EI) - how promising is a configuration depending on the model constructed?
- **Why SMAC?**
  - RF better surrogate model on data with categorical parameters [11] [12].
  - State of the art optimization algorithm [13].

[1] surrogate - substitute. A model mimicking the coupling tool.
[2] To select next promising point for evaluation on the target algorithm.

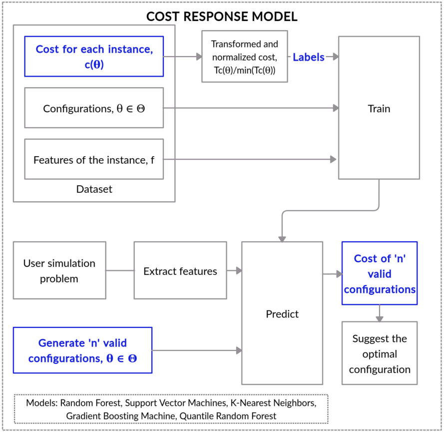
Proposed solution

- Automatically predict coupling tool parameter configurations.
- Configurations with relatively lesser runtime than default configuration.



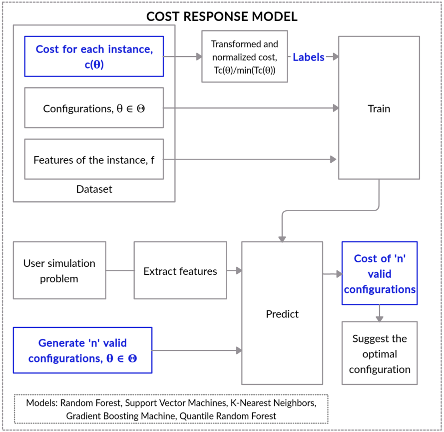
Training and prediction

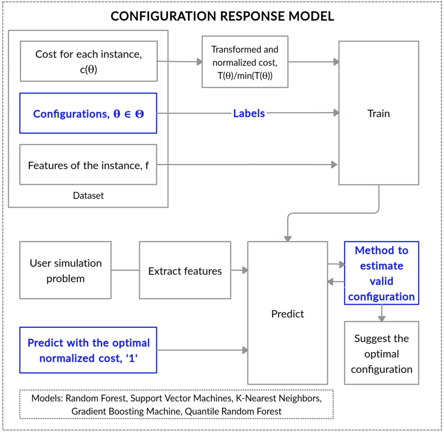
- Trained two types of models (blue indicate the differences).
  - Cost response model
  - Configuration response model



Training and prediction

- Trained two types of models (blue indicate the differences).
  - Cost response model
  - Configuration response model





Experiments and results

SMAC vs Default

- Does SMAC configurations perform better than the default configurations?
- Mean percentage decrease in runtime: 24.92%

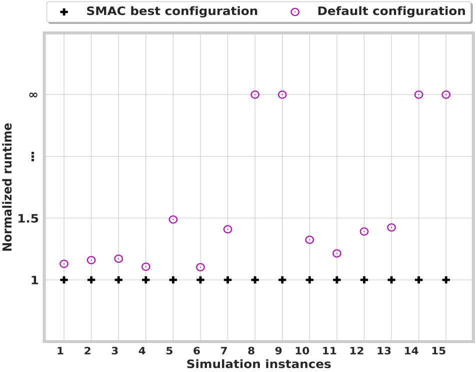


Figure 6: The runtime for each instance is normalized with the SMAC best configuration runtime. ∞ denote cost of crashed simulations.

Model selection across cost transformations

- 5-fold Cross Validation (CV) with Root Mean Square Error (RMSE).
- **Similar trend is observed in configuration response model.**

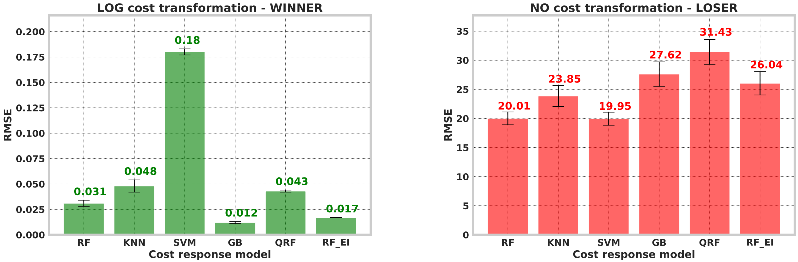


Figure 7: **Lower RMSE is better.** Models - Random forest (RF), K- Nearest Neighbors (KNN), Support Vector Machine (SVM), Gradient Boosting (GB), Quantile Random Forest (QRF), Random Forest predicting EI (RF_EI).

Evaluation on simulation instances

- No benchmarks to compare the parameter configuration performance.
- SMAC best configuration runtime is used to compare the models.
- Normalized Runtime (NR) of a configuration from model 'X'.

$$\text{NR} = \frac{\text{Runtime of the configuration from 'X'}}{\text{Runtime of the best configuration from SMAC}} \tag{1}$$

- Average Normalized Runtime (ANR) of configurations from model 'X'.

$$\text{ANR} = \frac{\sum_{i=1}^{N} \text{NR of a configuration from 'X' on instance i}}{N} \tag{2}$$

Evaluation on simulation instances

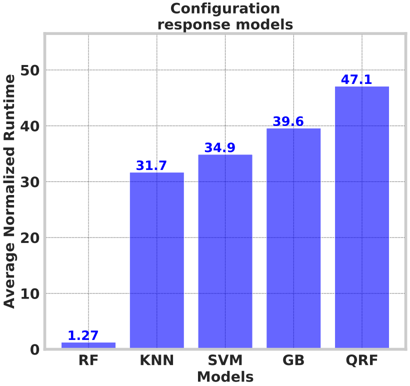


Figure 8: **Lower value is the better model**. The parameter configuration predicted by the top three models with less ANR is suggested by the smart coupling tool.

Model performance on a crashed simulation instance

- Positive impact of the smart coupling tool suggested configurations.

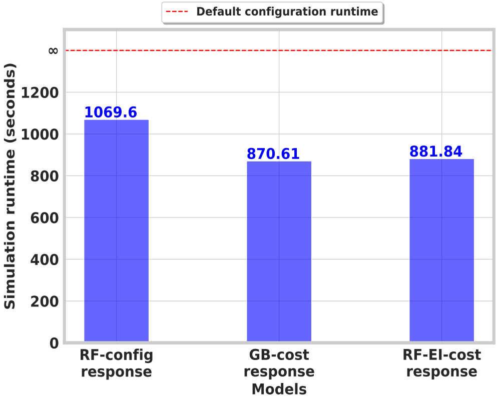


Figure 9: Default configuration results in crashed simulation. The parameter configurations suggested by these models result in a successful simulation.

Conclusion

Contributions

- Methodology to predict parameter configurations of a coupling tool for accelerated simulation.
- A preliminary multiphysics simulation dataset for machine learning.
- Feature reader: To extract features of the simulation instance.
- Smart coupling: To suggest parameter configurations to the user given a FSI simulation instance.

Future work

- Generate dataset v2.0 and evaluate on additional FSI & non-FSI problems.
- Formalize the feature selection method across problems.
- Investigate deep learning based models to suggest parameter configurations.
- Compare performance of SMAC with other optimization algorithms.
- Explainable artificial intelligence (XAI).

References I

[1] Bernhard Gatzhammer, Miriam Mehl, and Tobias Neckel. “A coupling environment for partitioned multiphysics simulations applied to fluid-structure interaction scenarios”. In: Procedia Computer Science 1.1 (2010). ICCS 2010, pp. 681–689.

[2] Fraunhofer Institute for Algorithms and Scientific Computing SCAI. MpCCI 4.5.2-1 Documentation. Accessed on: 2019-05-22. [Online]. Fraunhofer Institute for Algorithms and Scientific Computing SCAI Schloss Birlinghoven, 53754 Sankt Augustin, Germany, 2018. URL: https://www.mpcci.de/content/dam/scai/mpcci/documents/MpCCIdoc-452-20180504.pdf.

[3] Multi-Physics in Robotics Lab. Robots. Accessed on: 2019-01-03. [Online]. URL: https://mpirl.com/portfolio/robotics-2/.

[4] COMSOL Multiphysics. COMSOL Multiphysics Tutorial - Fluid-Structure Interaction in a Network of Blood Vessels. Accessed on: 2019-11-05. [Online]. 2019. URL: https://www.comsol.com/model/fluid-structure-interaction-in-a-network-of-blood-vessels-660.

[5] Dr Zara Kassam. CRISPR used to improve red blood cell transfusion compatibility. Accessed on: 2019-11-05. [Online]. 2018. URL: https://www.drugtargetreview.com/news/31427/crispr-blood-transfusion-compatibility/.

[6] Thomas Philip Cahill. Rough method of creating blood vessel. Accessed on: 2019-11-05. [Online]. 2013. URL: https://grabcad.com/library/blood-vessel-1.

[7] Thomas Ludescher Dr. Marc Ratzel. “Streamlining Aerodynamic CFD Analyses”. In: NAFEMS World Congress (2013).

[8] Donald R. Jones, Matthias Schonlau, and William J. Welch. “Efficient Global Optimization of Expensive Black-Box Functions”. In: Journal of Global Optimization 13.4 (1998), pp. 455–492.

[9] Qun Zhang and Song Cen. “13 - Multiphysics modeling for biomechanical problems”. In: Multiphysics Modeling. Elsevier and Tsinghua University Press Computational Mechanics Series. Oxford: Academic Press, 2016, pp. 363–373.

[10] O.C. Zienkiewicz, R.L. Taylor, and P. Nithiarasu. “Fluid–Structure Interaction”. In: The Finite Element Method for Fluid Dynamics. Seventh. Oxford: Butterworth-Heinemann, 2014, pp. 423–449.

[11] Frank Hutter. “Automated Configuration of Algorithms for Solving Hard Computational Problems”. PhD thesis. Vancouver, Canada: University of British Columbia, Department of Computer Science, 2009.

References II

[12] Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. "Sequential Model-Based Optimization for General Algorithm Configuration". In: Learning and Intelligent Optimization. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 507–523.

[13] Frank Hutter et al. "The configurable SAT solver challenge (CSSC)". In: Artificial Intelligence 243 (2017), pp. 1–25.

[14] Jonathan Gratch and Gerald DeJong. "COMPOSER: A Probabilistic Solution to the Utility Problem in Speed-Up Learning". In: Proceedings Tenth National Conference on Artificial Intelligence (1992), pp. 235–240.

[15] Steven Minton. "An Analytic Learning System for Specializing Heuristics". In: Proceedings of the 13th International Joint Conference on Artifical Intelligence - Volume 2. IJCAI'93. Chambery, France, 1993, pp. 922–928.

[16] Mauro Birattari et al. "A Racing Algorithm for Configuring Metaheuristics". In: Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation. GECCO'02. New York City, New York, 2002, pp. 11–18.

[17] Prasanna Balaprakash, Mauro Birattari, and Thomas Stützle. "Improvement Strategies for the F-Race Algorithm: Sampling Design and Iterative Refinement". In: Hybrid Metaheuristics. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 108–122.

[18] Frank Hutter et al. "ParamILS: An Automatic Algorithm Configuration Framework". In: Journal of Artificial Intelligence Research 36.1 (2009), pp. 267–306.

[19] Carlos Ansótegui, Meinolf Sellmann, and Kevin Tierney. "A Gender-Based Genetic Algorithm for the Automatic Configuration of Algorithms". In: Principles and Practice of Constraint Programming. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 142–157.

[20] Holger H. Hoos. "Automated Algorithm Configuration and Parameter Tuning". In: Autonomous Search. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 37–71.

[21] Hans-Joachim Bungartz et al. "preCICE – A fully parallel library for multi-physics surface coupling". In: Computers & Fluids 141 (2016), pp. 250–258.

[22] Bernhard Gatzhammer. "A partitioned approach for Fluid-Structure Interaction on cartesian grids.". MA thesis. Technische Universität München, Munich, Germany.: Technische Universität München, 2008.

References III

[23] James Bergstra et al. "Algorithms for Hyper-parameter Optimization". In: Proceedings of the 24th International Conference on Neural Information Processing Systems. NIPS'11. Granada, Spain, 2011, pp. 2546–2554.

[24] O. Lima and R. Ventura. "A case study on automatic parameter optimization of a mobile robot localization algorithm". In: 2017 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC). 2017, pp. 43–48.

[25] Peter I. Frazier. "A Tutorial on Bayesian Optimization". In: arXiv e-prints, arXiv:1807.02811 (2018), arXiv:1807.02811. arXiv: 1807.02811 [stat.ML].

[26] Lin Xu et al. "SATzilla: Portfolio-based Algorithm Selection for SAT". In: Journal of Artificial Intelligence Research 32 (2008), pp. 565–606.

[27] Leyton-Brown K Hutter F Hoos H. "Sequential model-based optimization for general algorithm configuration (extended version)". In: Technical Report. TR-2010-10. University of British Columbia, Department of Computer Science.: University of British Columbia, 2010.

[28] Colm McAlinden, Jyoti Khadka, and Konrad Pesudovs. "Precision (repeatability and reproducibility) studies and sample-size calculation". In: Journal of Cataract & Refractive Surgery 41.12 (2015), pp. 2598–2604.

[29] J. W. Bartlett and C. Frost. "Reliability, repeatability and reproducibility: analysis of measurement errors in continuous variables". In: Ultrasound in Obstetrics & Gynecology 31.4 (2008), pp. 466–475.

[30] Klaudius Scheufele. "Robust Quasi-Newton Methods for Partitioned Fluid-Structure Simulations". PhD thesis. Institute of Parallel and Distributed Systems, University of Stuttgart, 2015.

[31] Joris Degroote et al. "Coupling techniques for partitioned fluid-structure interaction simulations with black-box solvers". In: 10th MpCCI User Forum. Fraunhofer Institute SCAI. 2009, pp. 82–91.

[32] Martin Krasser. Bayesian Optimization. GitHub. Accessed on: 2019-11-12. [Online]. 2018. URL: https://github.com/krasserm/bayesian-machine-learning.

Thank you for your time!

Extras

SMAC Good-Bad-Ugly configurations

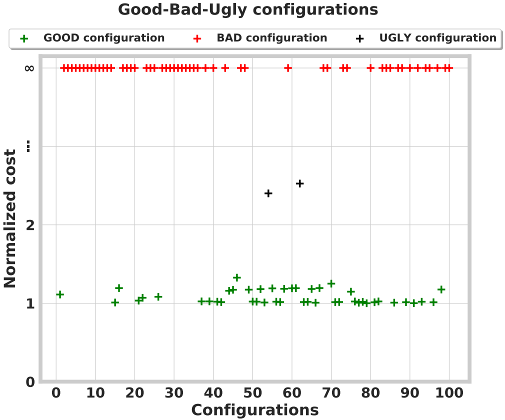


Figure 10: An illustration of good-bad-ugly configurations for single instance optimization over 100 iteration.

- Difficulty to decide whether the configuration is required for training.
- GOOD - YES, I want to use it of course.
- BAD - NO, I definitely do not want you for my training (taking a easy decision of removing them).
- UGLY - - , I do not know what to do with these configurations at the moment.

SMAC Good-Bad-Ugly configurations

Configurations	% of total configurations
The Good (Optimal and sub-optimal)	49.2
The Bad (Crashed)	47.8
The Ugly (Below sub-optimal)	3.0

Table 2: Percentage of good-bad-ugly configurations from SMAC for per-instance optimization (average over 16 instances). The ugly configurations are considered to be the configurations with normalized cost value above the upper inner fence ($Q3 + 1.5 \times IQR$) of the successful simulations distribution.

Related work of AAC

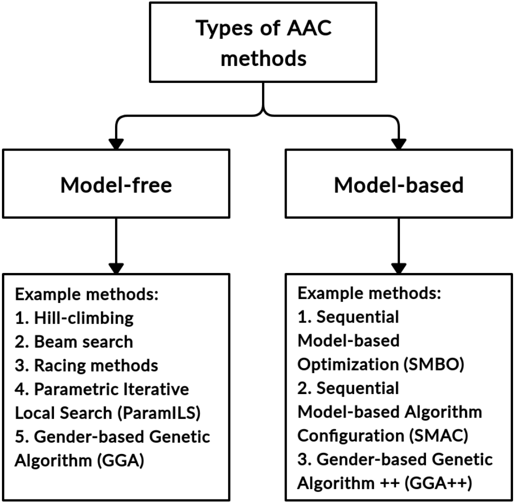


Figure 11: AAC methods.

Related work of AAC

- Traditional methods: Hill-climbing [14], Beam search [15], Racing methods [16] [17].

- Current state-of-the-art: Parameteric-Iterative Local Search (ParamILS) [18], Gender-based Genetic Algorithm (GGA) [19], Sequential Model-based Algorithm Configuration (SMAC) [12].

- Deficits:
  - Traditional methods can handle only lesser number of parameters, approximately five [14] [15]. In addition, the methods are restricted to only numerical and categorical parameters [20].

  - ParamILS is applicable only for configuring algorithms with categorical parameters. It explicitly requires a discretization procedure [18].

  - GGA is computationally expensive involving a fitness function evaluation and supports only per-instance optimization. In addition, fails for complex conditional parameters [13].

Related work of Coupling tool

- preCICE simulation tool, ADVENTURE_Coupler - Competitor coupling tool [21].

- No means of automatically estimating the optimal parameter values of the coupling tools [22].

Related work/Application of SMAC

- Mixed integer programming [12].
- Boolean satisfiability problem (SAT)[3] [12].
- Hyperparameter optimization [23].
- Robot localization [24].

[3]SAT is the problem of determining if there exists an interpretation that satisfies a given Boolean formula

Why SMAC?

- Advantages of SMAC [12] [11]:
  - Model-based AAC method.
  - RF surrogate model performs better with categorical parameters [11] [12].
  - Configures numerical, categorical and conditional parameters [12].
  - SMAC supports forbidden configurations [12].
  - Surrogate model aids in performing fewer direct evaluations of the target algorithm [25].

- Comparative study between ParamILS, GGA and SMAC for configuring Boolean satisfiability problem (SAT) solvers[4] illustrate SMAC is better in algorithm configuration with respect to solution quality and runtime [13].

[4] SAT is the problem of determining if there exists an interpretation that satisfies a given Boolean formula

SMAC steps

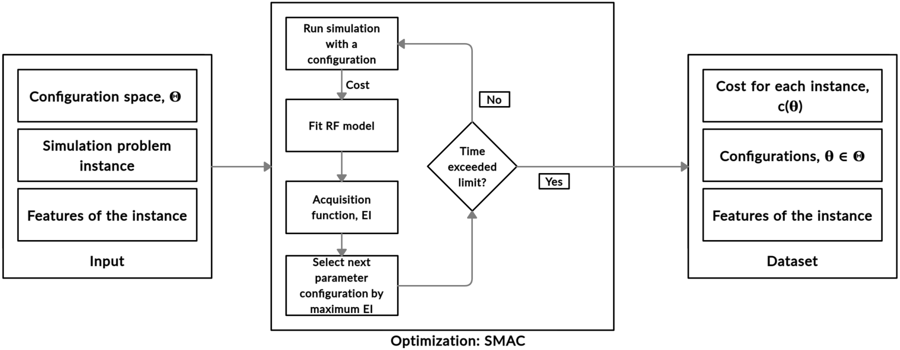


Figure 12: A simplified illustration of SMAC.

Why log-transformation better than normal?

- Pre-processed cost metric - transformed using the shown transformations and normalized to fix the minimum/ best cost metric at 1.

Transformation	Pre-processed cost metric	
	Skewness	Standard deviation
LOG	0.2152	0.6718
LOG-SCALED	-1.036	7.417
INVERSE	-1.532	2501
NONE	**6.331**	**3304**

Table 3: Statistical measures of the pre-processed cost metrics using different transformations. The green and red highlights indicate the first and last ranked statistical values, respectively, across different transformations.

Lessons learned

- Sort out patterns in failed simulations for dataset generation using ML approaches.
- Data normalization.
- Test cases for the input batch provider.
- Test cases for simple implementations and then proceed for higher dimensional regression. To avoid the black box issue.

How did you choose the training models?

- RF, GB performs well on categorical data.
- SATZILLA [26] one of the related works of SMAC utilize KNN and SVM.
- In future have to include different models.
- SATZILLA paper. This paper uses these models to predict the runtime to select the best solver to solve a particular problem instance in SAT solvers.
- SOLVER uses different algorithms to solve different problems.
- SOLVER parameters are pre-configured by AAC algorithms like SMAC, ParamILS.
- So dataset to train: solver name- instance - runtime.
- This is parallel to our domain of application- Algorithm selection.

How can you tell the parameter you find are optimal?

- We cannot practically guarantee optimal parameters in SMAC. SMAC convergence to optimal theorem is provided in [27]. However, we should do intensification with infinte time.
- We have said SMAC suggests a best configuration repeatably with a particular cost value.
- But, the best configurations from SMAC are considered optimal/best for a particular instance for evaluating the predicted configurations effectiveness.
- At the end, the deviation from the best is given as a suboptimal configuration.
- Convergence doesnt mean optimal solution is obtained.

Repeatability of SMAC

- **Test:** Is SMAC repeatable in finding the best configuration of a simulation instance?
- **Oneliner:** Find repeatability co-efficient of SMAC by optimizing a simulation instance using SMAC for 3 trials.
- **Motive:** Important to know the consistency of SMAC to rely on the configurations provided by SMAC. It gives a measure of uncertainity associated with a particular configuration.

Repeatability of SMAC

Instance	Best cost from SMAC (seconds)			Average cost (seconds)	Standard deviation (seconds)
	Trial 1	Trial 2	Trial 3		
1	399.84	397.82	396.41	398.02	1.72
2	545.76	520.97	520.60	529.11	14.42
3	403.16	393.68	416.12	404.32	11.26
4	465.79	452.49	467.15	461.81	8.09
5	2019.46	2026.78	2011.49	2019.24	7.64
6	724.22	726.98	744.65	731.95	11.08

Table 4: Average cost and standard deviation for the simulation instances involved in repeatability test of SMAC. Each trial involves 100 iterations of SMAC

Repeatability coefficient

- Repeatability coefficient: The maximum deviation with a probability of 95% between two successive measurements of the same subject under same measurement conditions and using the same procedure [28] [29].

- $S_r = 1.96 \times \sigma$, where σ is the mean standard deviation of the simulation instance costs (within group standard deviation).

Metric	**Value**
Mean variation in run-time within simulation instances (MS_w)	97.46 seconds
Repeatability coefficient (S_r)	19.34 seconds

Table 5: Repeatability of SMAC

Behavior of SMAC with respect to iteration count

- **Test:** How many iterations is required for SMAC to find better configurations?
- **Oneliner:** Run SMAC with different iteration count and estimate the best cost identified by SMAC.
- **Motive:** Aids in identifying the number of times the target algorithm has to be evaluated by SMAC.

Behavior of SMAC with respect to iteration count

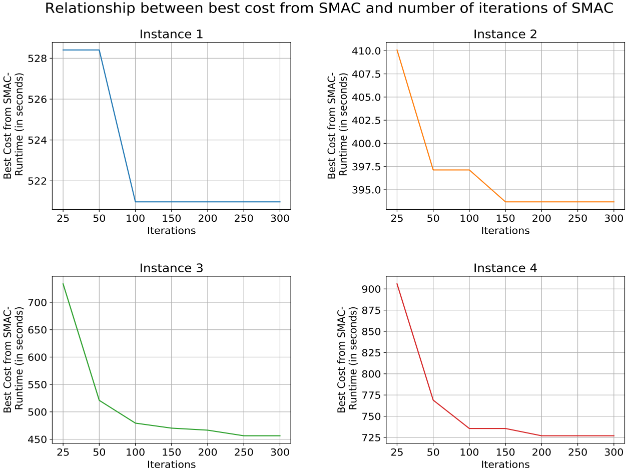
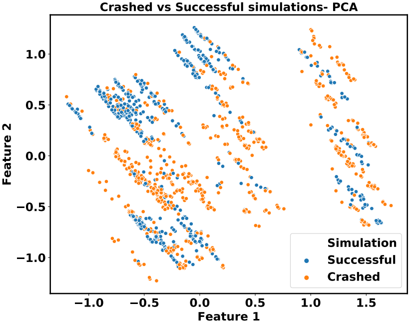


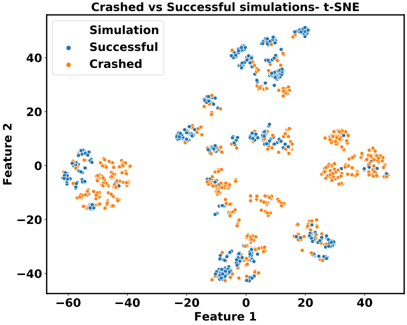
Figure 13: SMAC best cost versus number of iterations. The ability to find good configurations increases with iteration count.

Explainable AI idea

- Feature importance of each Out-Of-Bag (OOB) data using RMSE.
- Learn which feature leads to successful and crashed.
- Intermediate studies: SVM-RBF classifies with an accuracy of 95% (approx).



(a) PCA.

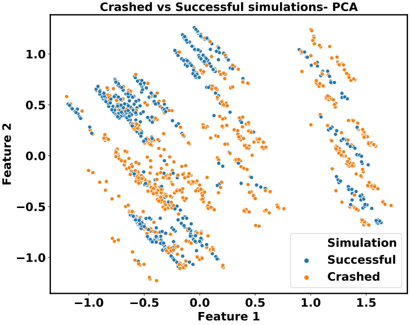


(b) t-SNE.

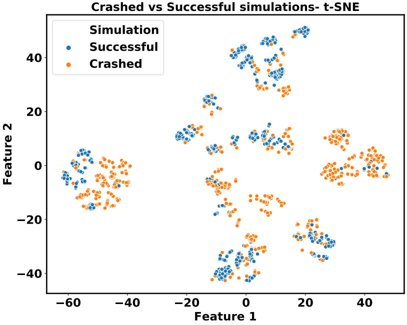
Figure 14: Successful vs crashed simulations. Includes optimization and random generation.

Dataset generation difficulty

- Configuration fixed, features varied (eitherwise).
- Possible reasons: Feature values bounds, or bad default configurations.
- Features are highly overlapping. Manually finding feature and configuration values that lead to successful simulation for generating dataset is difficult.



(a) PCA.



(b) t-SNE.

Figure 15: Successful vs crashed simulations. Includes optimization and random generation.

Significance of hyperparameters

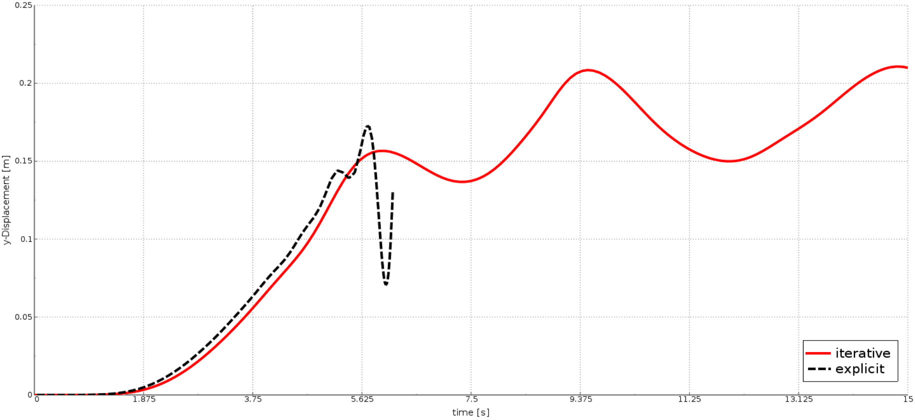


Figure 16: Impact of 'Coupling scheme' hyperparameter in driven cavity- Iterative transient coupling versus explicit coupling for 3D driven cavity [2]. The simulation will fail on configuring the coupling scheme with explicit coupling.

Significance of hyperparameters

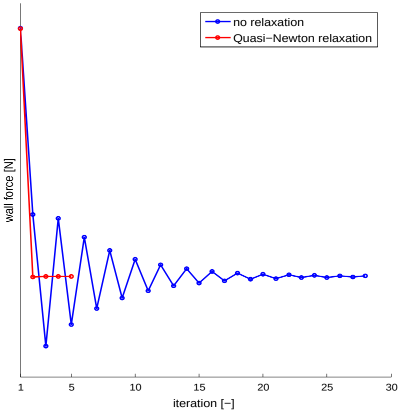


Figure 17: Effectiveness of Quasi-Newton relaxation method for 3D-driven cavity [2].

Default values for the hyperparameters

- In FSI, Quasi-Newton (QN) is the most efficient and robust relaxation method for partitioned coupling [30] [31].
- The following default parameter values are used in MpCCI for QN based problems.

Parameter, P	**Default value**
Coupling scheme	Implicit-Transient:Implicit-Transient
Initial exchange	receive:exchange
Relaxation_0	False
Relaxation_1	True
Relaxation method	Quasi-Newton
Andersonmix type	Inverse
Number of levels	1
Omega	0.1
Ramp_0	0.1
Ramp_d	0.1
Relaxation factor	0.1

Table 6: Default configuration majorly used in MpCCI.

Bayesian Optimization (BO)

- BO with Gaussian Process surrogate model [32].

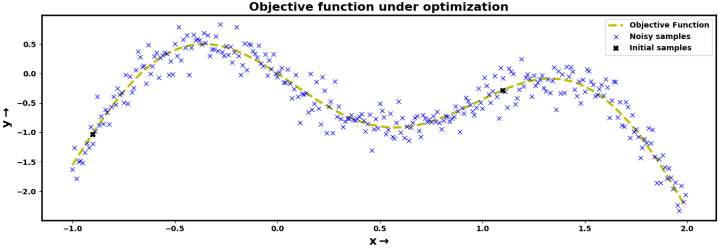
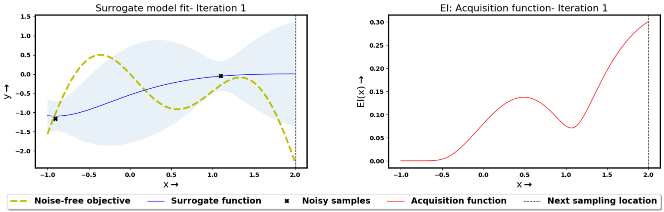
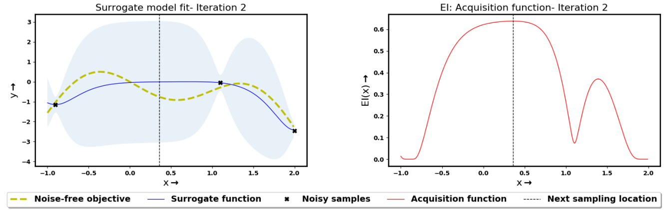


Figure 18: An example objective function to illustrate the working of Bayesian optimization.



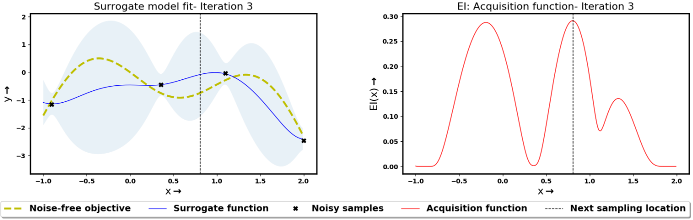
(a) Evaluate the query points (EVALUATE) →Fit surrogate to initial evaluation (FIT)→ Maximize EI to find the next query point (MAXIMIZE EI).



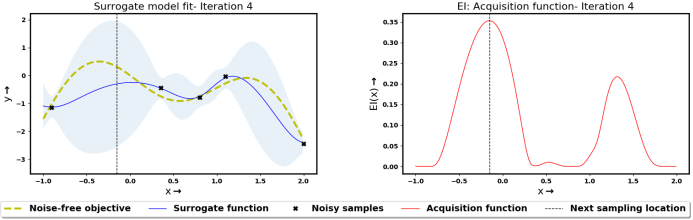
(b) EVALUATE → FIT → MAXIMIZE EI.

Figure 19: Illustration of BO- Iterations 1 and 2.

BO

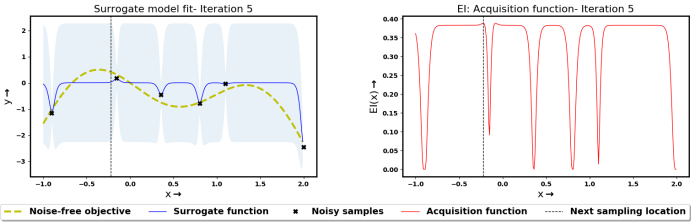


(a) EVALUATE → FIT → MAXIMIZE EI.

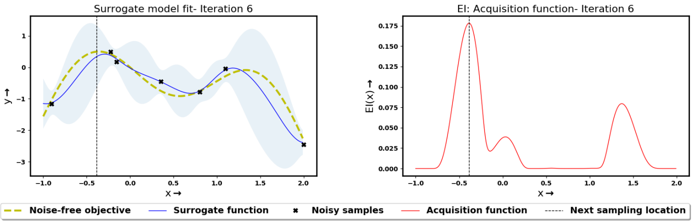


(b) EVALUATE → FIT → MAXIMIZE EI.

Figure 20: Illustration of BO- Iterations 3 and 4.



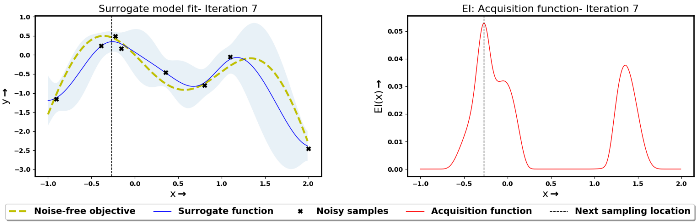
(a) EVALUATE → FIT → MAXIMIZE EI.



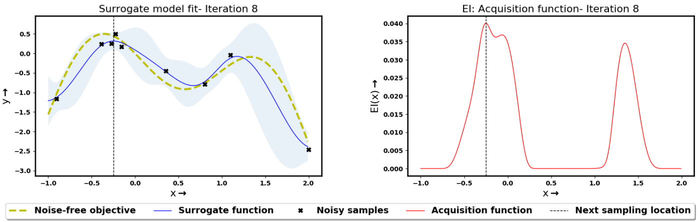
(b) EVALUATE → FIT → MAXIMIZE EI.

Figure 21: Illustration of BO- Iterations 5 and 6.

BO



(a) EVALUATE → FIT → MAXIMIZE EI.



(b) EVALUATE → FIT → MAXIMIZE EI. BO terminates returning the best sample point for 8 iterations.

Figure 22: Illustration of BO- Iterations 7 and 8.

Proposed strategy: Overview

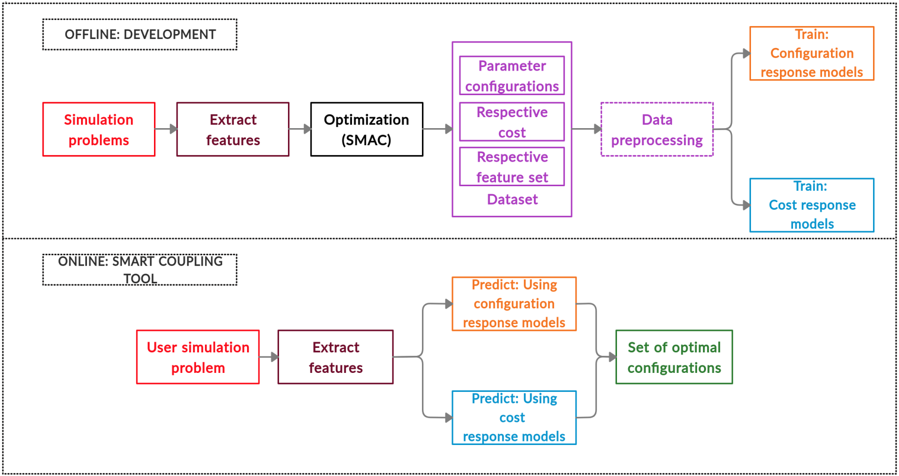


Figure 23: Overview of the proposed strategy to predict optimal parameter configuration. The color resemblances between the online and offline parts of the pipeline illustrate the usage of the same machine learning model, similar feature extraction method and similar input simulation problem.