

Co-Design of Embeddable Diagnostics using Reduced-Order Models[★]

Gregory Provan^{*}

^{*} *University College Cork, Cork, Ireland (e-mail: g.provan@cs.ucc.ie).*

Abstract: We develop a system for generating embedded diagnostics from an ODE model that can isolate faults given the memory and processing limitations of the embedded processor. This system trades off diagnosis isolation accuracy for inference time and/or memory in a principled manner. We use a Polynomial Regression approach for tuning the performance of an ensemble of low-fidelity ODE diagnosis models such that we achieve the target of embedded processing limits. We demonstrate our approach on a non-linear tank benchmark system.

© 2017, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: Fault Detection, Supervision and Safety of Technical Process; AI methods for FDI; Design of fault tolerant/reliable systems; Computational methods for FDI

1. INTRODUCTION

With the increasing availability of component libraries (e.g., in MATLAB/Simulink¹ or Modelica²) for systems design, it is important to use these libraries for generating embedded code for applications like control and diagnostics. However, the high-fidelity models and complex simulators used for design are typically incompatible with the memory and processing limitations of embedded processors. As a consequence, we need to transform the high-fidelity simulation models into embedded models that obey memory and processing limitations.

This article proposes an approach for generating an embedded diagnostic system tailored to an embedded processor. We show how to tune an ensemble of low-fidelity (reduced) models ψ_L (called a surrogate model), using data \mathbf{x} and/or outputs from a high-fidelity model ψ_H such that we adhere to the constraints \mathcal{C} of the embedded processor. We learn a function ζ such that $\psi_H(\mathbf{x}) \simeq \zeta(\psi_L(\mathbf{x}))$.

A big literature exists for model-order reduction (e.g., (Schilders et al., 2008)), but this approach does not address guarantees on memory and processing limitations for the generated models. Previous work (Cui et al., 2016) has shown that the performance of systems employing reduced models (also known as meta-modeling) depends on the test problems, the model parameter settings, and the design of experiments. Using an ensemble of meta-models can achieve better performance than using a single meta-model (Cui et al., 2016; Yin et al., 2014). For example, Yin et al. (2014) show that the ensemble meta-modeling method performs better than a single static meta-model in a task comprising multi-objective design optimization using dynamic ensemble metamodeling methods.

Our proposed approach can use any class of model as a reduced representation. Further, we can choose a wide variety of functional representations for ζ to learn, including polynomials (Bianchi et al., 2016), Bayesian (Gaussian Process) models (Bliznyuk et al., 2012), neural networks (Holena, 2009) and fuzzy membership functions (Bardossy et al., 1990). Once the function has been learned, we perform inference only on the computationally cheap reduced model in the embedded processor.

This article introduces several novel issues, of which the following are significant. First, in the co-design process of generating a diagnosis model subject to embedded system constraints \mathcal{C} , we use a design-of-experiments approach to identify the fault scenarios for computing the embedded model. This approach enables us to trade off the diagnostics accuracy for computational constraints in a principled manner. Second, we learn a surrogate model that performs inverse inference (diagnosis) rather than forward inference (simulation), which is typical of surrogate modelling. In other words, rather than optimizing simulation accuracy we optimize a measure of diagnostics accuracy, subject to \mathcal{C} . Third, we introduce a framework that is general and based on principled trade-offs of embedded system properties. Hence, a designer can choose the type of high-fidelity model used to generate the fault scenario data, as well as the surrogate modelling approach (e.g., polynomial regression, Gaussian Process, neural network, etc.).

Our contributions are as follows.

- (1) We introduce a novel method for embedded system co-design, based on tuning an ensemble of low-fidelity (reduced) models ψ_L using data and/or outputs from a high-fidelity model ψ_H , such that we adhere to the constraints \mathcal{C} of the embedded processor.
- (2) We propose a two-stage process for embedded model generation, where the first stage creates a set \mathbf{X} of tuning data, and the second stage generates from \mathbf{X} the embedded model that obeys constraints \mathcal{C} .
- (3) We demonstrate our approach on a hydraulic benchmark system.

[★] The paper has been supported by SFI grants 12/RC/2289 and 13/RC/2094.

¹ <https://uk.mathworks.com/help/physmod/simscape/ug/introducing-the-simscape-block-libraries.html>

² www.modelica.org/libraries

2. RELATED WORK

Our approach builds on work in a variety of areas, including reduced-order models, co-design, and model-based design optimization. Model-order reduction (MOR) consists of a broad set of mathematical methods to generate and evaluate reduced models that, in comparison to a full model, (a) have reduced degrees of freedom but comparable input-output accuracy, and (b) preserve properties of the original system, such as stability, controllability, passivity, etc. For recent surveys of the field, see (Baur et al., 2014; Benner et al., 2015; Schilders et al., 2008). MOR techniques have only been applied to simulation models; however, in our case we have a diagnosis model, for which the preservation of diagnostics properties like fault isolation or diagnosability has not been studied.

A significant amount of work has been done for hardware/software co-design, most of it focused on the design of embedded controllers, e.g., (Staunstrup and Wolf, 2013). There has been very little work done on the design of embedded diagnostics, apart from (Simon et al., 2013). (Simon et al., 2013) describe a range of techniques for co-design, focusing on the time delays and data dropout rates of networked control systems. In contrast, we focus on processor limitations such as memory, stack size, etc.

Our work is related to work in model-based system design using meta-models, e.g., (Van Gigch, 2013). What is unique to our approach is the use of pre-defined reduced-order diagnosis models, the use of most-likely failure scenarios as the basis for generating data for learning the embedded models, and the use of embedded processor constraints.

This work extends prior research on the use of meta-models (or surrogate models), which are designed to mimic the simulation behaviour of a system, but at a considerably reduced computational cost. Meta-models have been designed using many different approaches, including techniques such as Polynomial Response Surface (PRS) (Paduart et al., 2010), Kriging (Kleijnen, 2009), Artificial Neural Networks (ANN), Radial Basis Function (RBF) (Bliznyuk et al., 2012), and Support Vector Regression (SVR) (Kromanis and Kripakaran, 2013). (Frangos et al., 2010; Wang and Shan, 2007) presents a more detailed overview on several meta-modeling techniques. The same mathematical functions can be used as linking functions; see, e.g., (Razavi et al., 2012; Wang and Shan, 2007; Zhao and Xue, 2010) for details of these techniques together with analyses of their advantages and disadvantages.

The fidelity (accuracy) of meta-models governs the computational cost and convergence characteristics of the meta-model-based inference. There is a clear trade-off between high accuracy and low computational expense. The fidelity of the meta-models directly depends on the number of scenarios simulated by the HFM, which are then used to tune the LFM. Generally, the number of scenarios (sample points) is proportional to the fidelity and computational cost (Shan and Wang, 2010).

3. APPROACH

3.1 Preliminaries

We consider a computational model ψ that represents the behaviour of a physical system with an M -dimensional input vector $\mathbf{x}(t) = \{x_1(t), \dots, x_M(t)\}$ and N -dimensional response vector $\mathbf{y}(t) = \{y_1(t), \dots, y_N(t)\}$, i.e., $\mathbf{y}(t) = \psi(\mathbf{x}(t))$.³ $d(x)$ denotes the domain of values for variable x . Since we are interested in nominal and fault behaviours, we partition \mathbf{x} into state variables \mathbf{x}_S and fault variables \mathbf{x}_f , WLOG. This enables us to specify the fault space explicitly, if we assume that each $x \in \mathbf{x}_f$ takes on a finite number of discrete values, e.g., a pipe could have values {OK, leak, partial-blockage}, where OK denotes nominal behaviour and the other values faulty behaviour. The space of faults is the cross-product space of the $x_i \in \mathbf{x}_f$: $\mathfrak{F} = \{\times_i x_i | x_i \in \mathbf{x}_f \wedge d(x_i) \neq OK\}$.

Our objective is to build an embedded diagnostics model $\hat{\psi}$. We aim to design $\hat{\psi}$ so it can isolate the most important subset of the fault space, since it is unlikely that we will be able to capture all fault combinations in a resource-constrained model. If we define \mathfrak{F} as the discrete fault space, we can specify a discrete probability distribution for $\{P(f) : f \in \mathfrak{F}\}$. Given a threshold Π^* of the fault probability mass, our aim is to compute a multiple-fault diagnosis, i.e., a subset $\mathcal{F} \subseteq \mathfrak{F}$, of the fault space such that $P(\mathcal{F}) \geq \Pi^*$.

3.2 Objectives

We sub-divide our task into two sub-tasks:

- (1) Design a set of experiments to guarantee a notion of embedded fault coverage.
- (2) Generate the embedded model to guarantee a given fault isolation accuracy within the constraints of the embedded processor.

Design of Experiments Our first objective is to generate a set of experiments to guarantee embedded fault coverage. We assume that we cannot run experiments to collect data for all of \mathfrak{F} , so we will use ψ to simulate the necessary experimental data. In other words, we use ψ to simulate a response \mathbf{y} at n distinct locations $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$. We call each pair (\mathbf{x}, \mathbf{y}) an experiment γ , and denote the full space of experiments as Γ . We call \mathbf{X} the *design of experiments (DoE)* and \mathbf{Y} the *observation space*.

We assume that we have a prior distribution $P(\psi)$ over the model space Ψ , for $\psi \in \Psi$. Given data D acquired from an experiment, we can compute the *a posteriori* distribution using Bayes' theorem:

$$P(\psi|\Gamma) = \frac{P(\Gamma|\psi)P(\psi)}{P(\Gamma)}, \quad (1)$$

where $P(\Gamma) = \sum_{\psi \in \Psi} P(\Gamma|\psi)P(\psi)$. From this, we can compute the expected information gain of an experiment as follows (Busetto and Lygeros, 2014)

$$I(\psi|\Gamma) = \sum_{\Gamma, \psi} P(\psi, \Gamma) \log_2 \left(\frac{P(\Gamma, \psi)}{P(\Gamma)P(\psi)} \right). \quad (2)$$

³ We will suppress temporal notation for simplicity of exposition.

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات