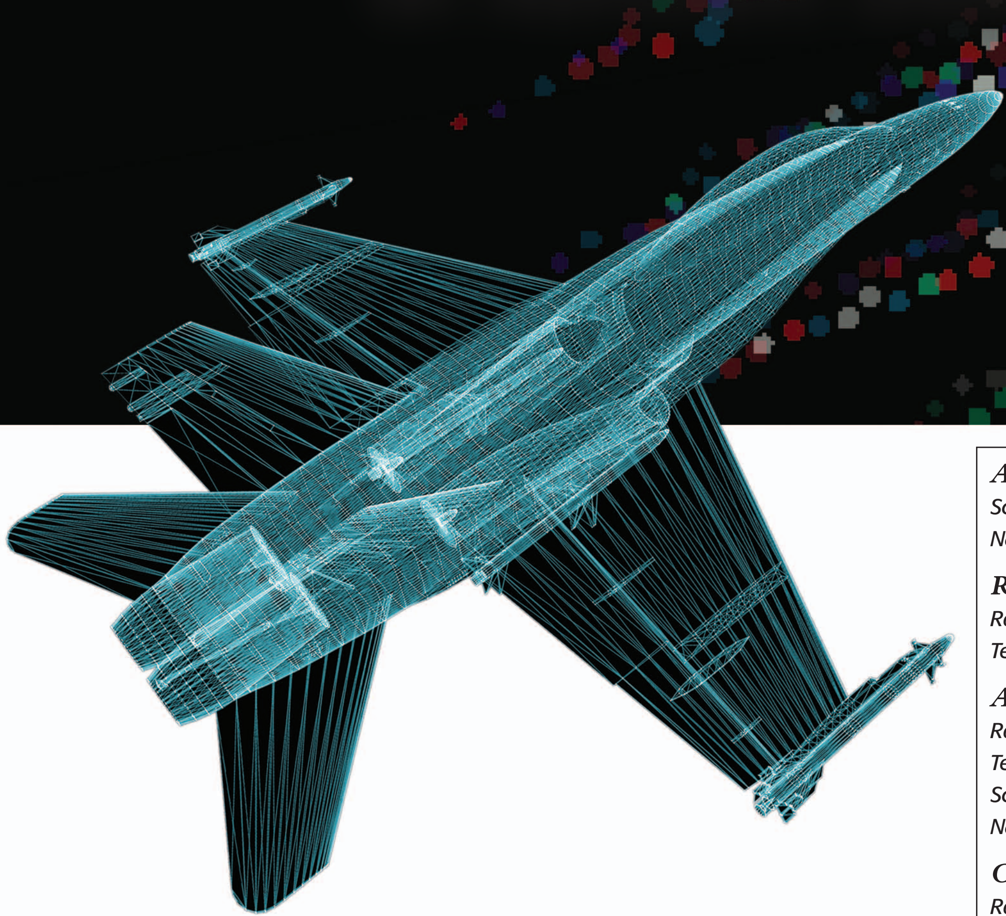
# Knowledge Transfer Through Machine Learning in Aircraft Design




***Alan Tan Wei Min***
*School of Computer Science and Engineering, Nanyang Technological University, Singapore*

***Ramon Sagarna***
*Rolls-Royce@NTU Corporate Lab, Nanyang Technological University, Singapore*

***Abhishek Gupta and Yew-Soon Ong***
*Rolls-Royce@NTU Corporate Lab, Nanyang Technological University, Singapore*
*School of Computer Science and Engineering, Nanyang Technological University, Singapore*

***Chi Keong Goh***
*Rolls-Royce@NTU Corporate Lab, Nanyang Technological University, Singapore*



***Abstract***—The modern aircraft has evolved to become an important part of our society. Its design is multidisciplinary in nature and is characterized by complex analyses of mutually interdependent disciplines and large search spaces. Machine learning has, historically, played a significant role in aircraft design, primarily by approximating expensive physics-based numerical simulations. In this work, we summarize the current role of machine learning in this application domain, and highlight the opportunity of incorporating recent advances in the field to further its impact. Specifically, regression models (or surrogate models) that represent a major portion of the current efforts are generally built from scratch assuming a zero prior knowledge state, only relying on data from the ongoing target problem of interest. However, due to the incremental nature of design processes, there likely exists relevant knowledge from various related sources that can potentially be leveraged. As such, we present three relatively advanced machine learning technologies that facilitate automatic knowledge transfer in order to improve design performance. Subsequently, we demonstrate the efficacy of one of these technologies, i.e. transfer learning, on two use cases of aircraft engine design yielding noteworthy results. Our aim is to unveil this new application as a well-suited arena for the salient features of knowledge transfer in machine learning to come to the fore, thereby encouraging future research efforts.



*Digital Object Identifier 10.1109/MCI.2017.2742781*
*Date of publication: 12 October 2017*

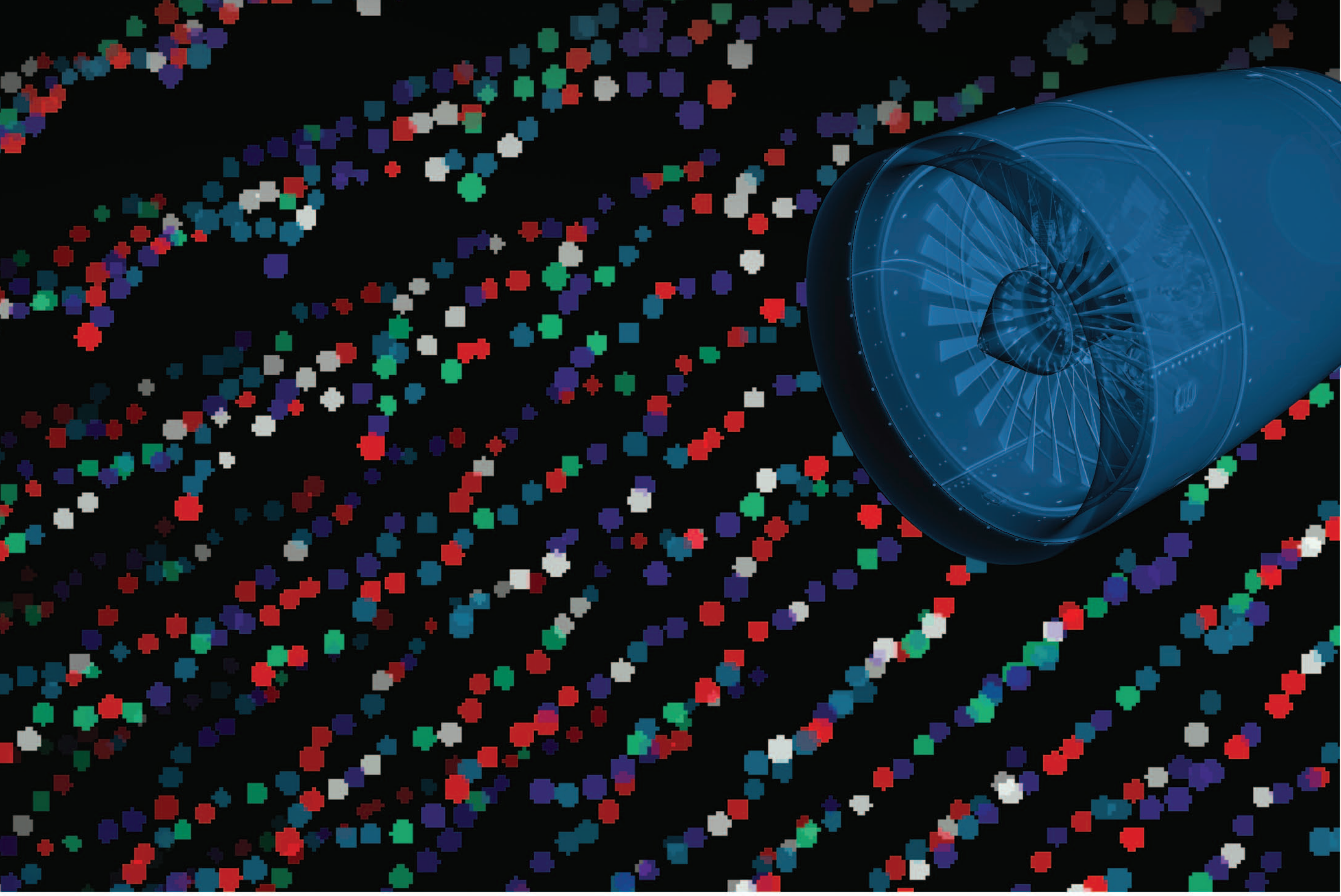


Corresponding Author: Yew-Soon Ong (Email: asysong@ntu.edu.sg)





1556-603X/17©2017IEEE





## I. Introduction

The modern-day aircraft is an integral part of today's society. It provides access to fast, efficient long distance travel and has facilitated the growth of many important areas such as international politics, economics, and medicine. Its design is an extremely complex task that involves the interactions of a variety of mutually interdependent systems. In addition to strict safety standards and performance requirements, they must often be tailored to a particular application or mission [1]. Despite these challenges, great strides have been made in the design effort since the first successful wood and fabric based construction by the Wright brothers in 1903. In particular, Aviation received a major boost during World War II with the success of the Rolls-Royce Supermarine Spitfire in the Battle of Britain victory of 1940 [2]. Since then, considerable improvements have been accomplished in all technical areas. Fundamental properties such as lift, drag, weight and flight performances are now well understood. Computational methods and tools have been steadily improved and their accuracy verified by substantial empirical data. However, despite these noteworthy advancements, aircraft design remains extremely laborious and time-consuming, taking as much as 6 years from the initial conception to the first product delivery [3].

The design of an aircraft is a multidisciplinary effort and is characterized by complex analyses with large search spaces involving many disciplines. Ignoring the disciplinary boundaries, a multidisciplinary design optimization (MDO) problem can be framed as a nonlinear programming task with the goal of optimizing the objective(s) subject to a set of constraints. However, the disciplinary analyses that model the behavior of the system components are mutually interdependent, i.e. one must account for components interacting with one another through coupling variables or scenarios wherein the outputs of one analysis (corresponding to a particular component) are the inputs of another. Further, the objectives and constraints themselves are evaluated from the analyses outputs of multiple disciplines. The key challenge, therefore, is to effectively and efficiently manage the couplings of the system. It should be noted that this interdependence is sometimes suppressed in favor of single-discipline optimizations occurring sequentially or in parallel. Nevertheless, it is naturally regarded that a proper consideration of these interdependencies will result in a more accurate approximation of the actual system behavior [4].

While simple design efforts may be adequately completed by a single master designer, the complexity and level of detail required for the modern-day aircraft necessitate a more thorough treatment. Generally, aircraft design requires the involvement of groups of specialists and sound mathematical formulations (or architectures) to manage the disciplinary interdependencies. In addition to numerous applications of MDO on a wide variety of systems (e.g. aircraft [5], buildings [6], and wind turbines [7]), the development of MDO architectures is an active field of research and a number of surveys have been published over the last two decades [8]–[11]. Many include discussions on monolithic and distributed MDO architectures, where the former refers to formulations that cast the MDO task as a single optimization problem, while the latter refers to formulations that decompose the overall optimization problem into smaller subproblems. Fairly recently, Martins and Lambe [4] presented a comprehensive survey of all existing architectures including a comparison of the various features, merits, and performances.

In all variants of MDO architectures, a critical component is the engineering analyses. In the past, these tasks were performed manually using (often oversimplified) analytical theory and/or labor intensive physical experimentation [12]. With the advent of high-performance computing and the advancements of computational engineering methods, modern analyses heavily utilize physics-based numerical simulations, where mathematical models of physical systems are solved for a discretized domain. However, the significant improvements to accuracy, particularly for large-scale complex systems, come at a price. Computational fluid dynamics (CFD) simulations, for example, can take anywhere from several minutes to several weeks per simulation [13], depending on the complexity of the system and the desired level of accuracy. As each MDO task may require hundreds or thousands of such simulations, there are strong financial motivations to find ways of speeding up the analysis procedures.

Machine learning, a cornerstone of artificial and computational intelligence defined loosely as a means to build approximations from data, has played a key role in reducing the impact of the aforementioned challenge. The existing methods in the literature have been successfully applied in a host of practical domains spanning natural language processing [14], computer vision [15], [16], recommender systems [17], etc. In aircraft design, machine learning is predominantly used for approximating the expensive physics-based simulations using supervised regression models, or more commonly, surrogate models. Its application has been widespread throughout various stages of aircraft design. However, despite the promising outcomes achieved thus far, it is deemed that the full capabilities of machine learning are yet to be fully unveiled and exploited in this domain.

In the present paper, we summarize the efforts by the aerospace community to apply machine learning in aircraft design, and highlight the new opportunities presented by recent advances in the field to further its impact. Specifically, it is noted that in current practices surrogate models are typically built from scratch assuming zero prior knowledge, only relying on data sampled from the ongoing target problem of interest. However, it is contended that any practically useful intelligent system in an industrial setting will be faced with a large number of problems over a lifetime, with the problems likely sharing domain-specific overlaps. With this in mind, we describe three relatively advanced machine learning technologies that are especially developed to enable automatic knowledge transfer as a means of improving upon existing *tabula rasa* efforts. Our aim is to unveil meta-machine learning as a promising approach to enhance the efficiency of aircraft design and facilitate the realization of a more *agile* design process [18], [19]. Specifically, improving design procedures and methodologies so as to allow for rapid adaptation to change, lower development costs, and shorter time-to-market. The remainder of the paper is organized as follows: Section II provides an overview of the general role of machine learning in aircraft design; Section III sheds light on several potential directions for enhancing the future impact of machine learning through the incorporation of automatic knowledge transfer; Section IV presents an illustrative experimental case study; Section V concludes the paper.

## II. The General Role of Machine Learning in Aircraft Design

In engineering, machine learning predominantly appears in the form of supervised regression models, more commonly known as surrogate modeling, in engineering (including aircraft design) [20], [21]. Multiple surveys have been published regarding this particular application [22]–[24], detailing factors such as design of experiments, and cross-validation procedures, in addition to the specificities of the machine learning algorithms themselves. The range of regression model types that have been used in the past is quite large. These include polynomial regression (the 2nd order variant in particular) [25]–[27], neural networks [28]–[30], Gaussian process (or kriging) [31]–[33], co-kriging [34], support vector regression [35], [36] and radial basis functions [30].

Next, we briefly discuss the use of machine learning in two areas of aerospace applications that have drawn considerable research interest: (1) data-driven surrogate models of physical phenomena, and (2) machine-learning complemented physics simulations. The list of works described herein is not comprehensive but is rather a representative subset meant to provide a general impression of the concentration of efforts in this domain. Notably, a common facet of existing works is their rather myopic view of an ongoing target problem at hand; often overlooking the possibility of automatically extracting and transferring knowledge across different problems (that may well share underlying commonalities) to enhance the outcome of machine learning. This observation illuminates a key deficiency of current practices that is holding back the efficacy of machine learning in aircraft design, thereby motivating the future research directions that are sketched in Section III.

### A. Data-Driven Surrogate Models of Physical Phenomena

In aircraft design, regression models are widely used to approximate expensive physics-based numerical simulations [37]. For example, Huang et al. [38] presented the design optimization of an aeroengine turbine disc under thermal and mechanical loads. To reduce computational cost, the authors constructed a kriging regression model which approximates the computational structural mechanics analysis of the engine component. In [39], Papila et al. presented the optimization of the shape of a two-stage supersonic turbine involving $O(10)$ design variables. In their optimization framework, the authors utilized a combination of radial-basis neural networks and polynomial regression in order to approximate the expensive CFD numerical simulations. In [40], Sellar et al. proposed an MDO framework that utilized neural networks to approximate non-discipline level information (i.e. outputs from other subsystems) in order to improve overall system optimization performance.

Another typical application of regression models is to facilitate the use of variable-fidelity physics simulations. To elaborate, high-fidelity simulation models are more accurate but at the cost of being very computationally expensive, while on the other hand, low-fidelity models are cheaper but less accurate. In [41], a kriging response surface was utilized to predict the difference between a high-fidelity CFD simulation and a low-fidelity empirical tool. The kriging model was then used in conjunction with the empirical tool in an optimization process and was found to be more accurate than simple response surfaces built on the CFD data directly. In [42], Jaeggi et al. utilized a sparse pseudo-input Gaussian process regression model to predict the divergence between low and high-fidelity CFD simulations, which then served as a proxy for *simulation risk* to be optimized along with other performance measures in a multi-objective optimization algorithm. In this way, the model can be used to guide engineers in deploying CFD simulations of appropriate fidelity in order to make better use of computational resources and develop designs that are minimally affected by simulation error. More recently, Minisci and Vasile [43] applied a neural network-based approximation model in place of full CFD simulations to calculate the forces acting on an unmanned space vehicle. Interestingly, the database for training the network was updated by a multi-fidelity approach, such that low fidelity models were used for global sampling while high-fidelity ones were used for the local refinements.

### B. Machine Learning Complemented Physics Simulation

The governing equations of CFD, known as the Navier-Stokes equations, are a set of highly complex partial differential equations (PDEs) that do not have an analytical solution and must be solved numerically. However, a direct numerical simulation of these equations can be extremely expensive, especially in the case of turbulent flows ubiquitous in aerospace applications. As such, a common simplification is to solve the Reynolds-Averaged Navier-Stokes (RANS) equations instead. The turbulence modeling closure terms, a critical component of the RANS approach, is derived mostly based on ad-hoc heuristics that can produce erroneous results under certain unseen flow regimes. In [44], the basic idea proposed was that in order to enhance the consistency and accuracy of RANS, generalizable machine learning algorithms could be used to build a representation of turbulence modeling closure terms. To demonstrate the feasibility of this idea, the authors first trained a neural network using data from a set of RANS-based CFD runs with the Spalart-Allamaras (SA) turbulence model. They then ran a new set of RANS-based CFD runs using the neural network as the turbulence model instead, and showed that the original results were successfully reproduced. In [45], [46], a methodology was presented to identify a correction factor that could account for deficiencies in simplified physics-based models (e.g. RANS equations) in comparison to more precise higher-fidelity direct numerical simulations [47]. Using inverse modeling, the authors first identified these correction factors for multiple variations of a problem and then built a Gaussian process regression model to learn the correction factor as a function of problem-specific features. The learned model was thereafter injected into the simplified physics simulations to produce more accurate results.

The numerical solution of PDEs that govern all physical phenomena rely on specialized mathematical techniques, of which the finite element method and the finite volume method are among the most popular. These methods have been widely used in a variety of simulation settings from structural mechanics to fluid mechanics. The first step of a finite element or finite volume analysis is to create an appropriate discretization of the physical domain (to form what is often referred to as a *mesh*). Regions of high interest require a dense mesh in order to precisely capture all relevant physics, while other regions can be made coarser in order to reduce computational overhead [48]. Improper design of the mesh can drastically affect the accuracy of the simulation. Therefore, the mesh (or discretization) must often be adaptively refined during the simulation in order to adapt the precision of the numerical computations—a process which can be expensive. In [48], [49], the idea of building an expert system that is able to automatically determine the resolution of the mesh to be used for the analysis (by learning from a database of geometries of similar type) was proposed. The application of self-organizing neural networks for mesh generation has also been considered in [50], [51]. In these works, the authors assumed that the mesh density for a given geometry is provided (e.g. by an expert system as discussed above) and the self-organizing neural network is used to automatically generate the mesh while ensuring that user-specified constraints are met (e.g. triangular v.s. quadrilateral elements, well-proportioned elements, appropriate interior angles, etc.).

## III. Future Directions for Machine Learning in Aircraft Design

Despite the advancements achieved through the application of machine learning in aircraft design, there remain certain shortcomings in current methods that open up new directions for future research. Of particular interest to this paper is the observation that most machine learning algorithms require tens, hundreds, or even thousands of training instances to build sufficiently good models capable of making effective predictions. However, in aerospace applications, obtaining even a single data point from a CFD simulation or physical experiment can be very costly, leading to the threat of problem-specific *data-scarcity* [52]. As a result, the existing uses of machine learning continue to suffer from the so-called *cold start* problem where considerable resources are spent in generating the initial database for a given new problem—which can take several hours, days, or even weeks depending on the cost of a single simulation/laboratory experiment [4].

With the above in mind, it is contended that in many industrial settings, especially in various aspects of aircraft design, there likely exists an abundance of relevant data that has been accumulated from related design exercises over several decades of gradual progress. Indeed it is the ability to exploit existing knowledge that often distinguishes a domain expert from an amateur (with zero prior knowledge). Therefore, to push the envelope of machine learning applications in aircraft design—particularly with regard to overcoming the cold start problem which can be a severe bottleneck for tight computational budgets in the face of increasing demands in modern engineering systems—there emerges immense scope for developing algorithms that can automatically extract and transfer knowledge from related design exercises to improve the efficiency of future design efforts. As an illustrative example, recent studies (albeit in a different field of application) have shown that due to the use of Gaussian process regression models facilitating cross-domain knowledge exchange [53], the impetus provided to the optimization search reduces the time taken to find high-quality solutions by as much as 40% compared to traditional methods without transfer [54]. Clearly, such time/cost savings will be of much value in aerospace applications as well, thereby serving as the main motivation behind our subsequent discussions.

The notion of knowledge transfer has, however, been largely unexplored by the members of the aerospace community, with few notable exceptions. In [55], Han and Görtz proposed a hierarchical kriging model that approximates a high-fidelity function by using a sampled lower-fidelity function for its model trend. Valenzuela et al. [56] proposed a multi-fidelity Gaussian process model that enhances predictions of new concepts using observations from a previous concept with similar trends. These works mark the beginnings of initiatives intending to exploit knowledge transfer within the domain of aircraft design. Doubtless, the members of the aerospace community can further benefit from the large body of research and mathematical rigor provided by machine learning. Accordingly, in this section, we highlight three relatively modern technologies that can help make significant strides in aircraft design by facilitating automatic knowledge transfer across problems and domains in surrogate modeling. The need to consider advanced techniques, in addition to more traditional incremental learning strategies (where problem-specific data appears over time [57]), shall be clarified through numerical case studies in Section IV. In particular, it will be shown that the direct (naïve) reuse of past knowledge may at times turn out to be worse than no transfer at all if the knowledge is not properly adapted—this is because the features, data distribution, and/or characteristics of target functions (such as the mean and scale parameters) need not precisely match across distinct problems/domains [58].

### *A. Transfer Learning*

The development of complex designs draws heavily from prior knowledge, as *tabula rasa*-style design efforts would be extremely difficult (or impossible) to conduct. This reliance is especially significant in aircraft design, where the enormity of the task necessitates simplifying assumptions made based on technical knowledge and past experiences. This fact, coupled with the incremental nature of design and the preferred decomposition approach of MDO, indicates that there is a great deal of knowledge from past (and in-parallel) projects that can be exploited in order to improve the quality of designs. For instance, an engineering team designing a turbine for an aircraft engine would use, as reference, past designs that have been successful and modify them accordingly to suit the current application. Relevant transferable knowledge may also originate from different parts of the aircraft that may fill a similar role. The fan blades from a low-pressure compressor and a high-pressure compressor of a gas turbine engine, for example, may have similar design characteristics despite the differences in the environments in which they operate.

However, this notion of knowledge transfer in optimization has largely been unexplored with only a few related works published fairly recently [59]–[64]. While prior knowledge incorporation can occur through appropriate problem formulations, the optimization algorithm itself is conducted from scratch i.e. without reusing knowledge (e.g. data or regression models) gathered from previous optimization efforts. As such, a possible research direction is the application of transfer learning as a means to exploit knowledge embedded in sources of information, such as data or regression models, obtained from related but distinct design efforts.

As illustrated in Figure 1, the objective of transfer learning is to improve the learning of the predictive function of a target task using knowledge from a

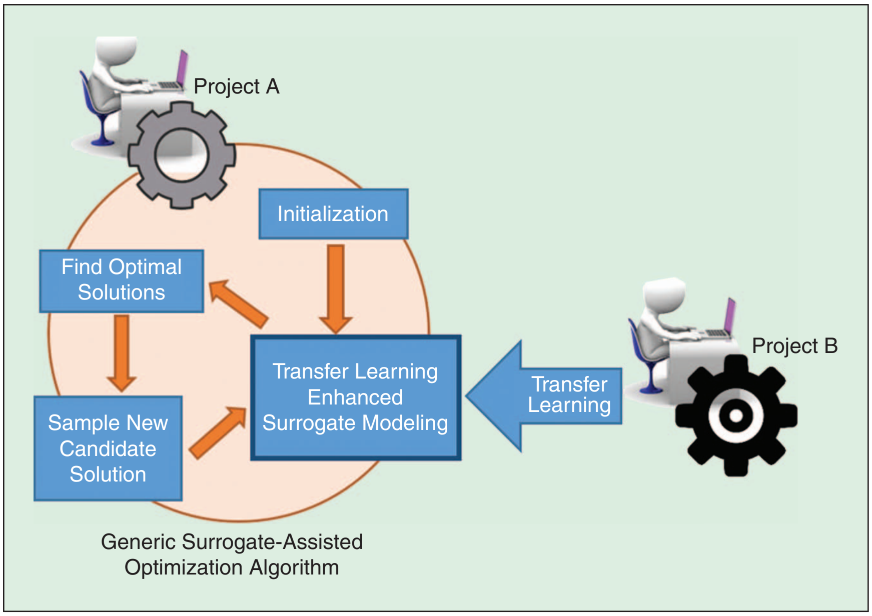



**FIGURE 1** Illustration of transfer learning applied to a generic optimization algorithm. In the design effort of Project A, knowledge from a related (but distinct) Project B is used to enhance optimization performance.

source task, where either the domains or the tasks differ [65]. One suitable use case would be an optimization task of an aircraft engine in which source information would originate from a design effort to optimize a different engine configuration comprising several recurring parameters and outputs. Another potential use case would be to investigate the addition of new design parameters or an altogether different set of design parameters relative to a prior (source) task.

### B. Multi-Task Learning

In real-world aircraft designs, it is not uncommon for the total number of design parameters throughout the entire system to number in the thousands. Consider, for example, a generic turbofan engine used by most commercial airliners. Its functionality depends on a host of interdependent systems such as the compressors, turbine, and combustion chamber—each of which is itself reliant on thousands of design variables. Due to effects such as the curse of dimensionality [66] coupled with the high computational cost of precision simulations, a monolithic optimization of the entire design space would be prohibitively expensive.

As such, a common practice is to analyze only a small subset of these design parameters (which constitutes a subproblem). Consequently, there are often different variations of these subproblems that are conditioned on the specific choice of all remaining design parameters. A typical example in aircraft design of such a situation would be the optimization of an aircraft wing. In each variation, the design team may aim to optimize performance parameters (e.g. lift, drag, and weight) as a function of the geometry of the wing [67], conditioned on certain fixed variables (e.g. materials, and flight conditions) subsequently varied as part of distinct studies. To elaborate, one variation might study a wing utilizing composite material A while another, composite material B; each with its own unique set of material properties.

The domain of multi-task learning is uniquely suited for these types of problems. In contrast to transfer learning which leverages on *past* experiences, multi-task learning assumes that there are multiple related tasks that can be *simultaneously* solved in order to facilitate knowledge transfer [53], [54], [68]. In the context of multi-task regression, the model aims to learn a mapping as follows: $f: X \to \mathbb{R}^N$ where $X$ represents a common set of input parameters and the $N$ outputs are considered to belong to different tasks [53].

By learning related tasks together using a shared representation, the generalization performance is contended to improve by exploiting the shared knowledge in the training signals of the related task(s). As illustrated in Figure 2, from an aircraft design perspective, a single multi-task regression model built from data originating from various related design efforts can be used to boost multiple optimization algorithms at the same time—in the spirit of multi-task optimization [54], [69]–[71]. Approaching the problem in this manner facilitates knowledge transfer and reduces the overall computational cost of the design effort as the same regions of the parameter space need not be searched repeatedly [54], thereby resulting in the need for fewer expensive numerical (finite element or finite volume) simulations. It is worth noting here that, although the dimensionality for each of the subproblems is expected to be significantly lower than for the entire design space, it could be the case that the common set of input parameters defines a large space for the multi-task regression. In such scenarios, while the arguments in favor of savings in simulations would still hold, the process of multitasking does not rule out the application of classic approaches, such as space reduction techniques [72] (the beginning of Section IV elaborates more on this point).

### C. Multi-View Learning

Due to the computationally expensive nature of disciplinary analyses, a common strategy is to approximate the numerical simulations using models of varying fidelity. In certain situations, physical phenomena or structures of interest can be analyzed using either a full 3-D view or even a restricted 2-D/1-D view, depending on the types of statistics the engineer is looking for. For example, while analyzing plate-like structures (which are common in aircraft), one may consider a 3-D discretization if through-thickness effects are of interest, or may emphasize on the 2-D discretization if only the in-plane effects are of paramount importance [73]. Thus, data that provides an understanding of the mechanics from different perspectives is made available in situations where computational budget is a prime limitation.

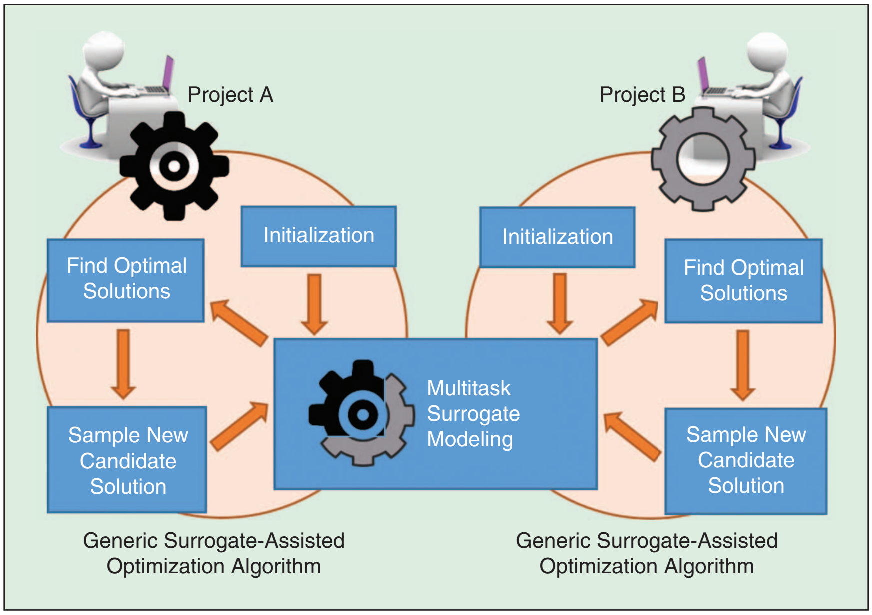



**FIGURE 2** Illustration of multi-task learning applied to generic optimization algorithms for two related projects. Both Projects A and B are assisted by a multi-task surrogate model constructed from a combination of data from both projects.

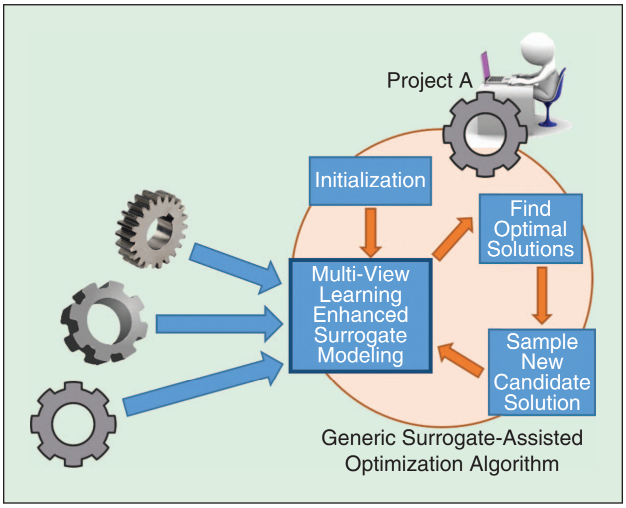



**FIGURE 3** Illustration of multi-view learning applied to a generic optimization algorithm. Data from different perspectives of the system under study are used to build a multi-view learning enhanced surrogate model in order to obtain a more robust and accurate approximation of reality.

The domain of multi-view learning is uniquely suited to analyzing multifaceted data-streams of the aforementioned type. This fairly new paradigm jointly learns a function for each perspective in order to exploit the redundant views of the same domain and, consequently, improves prediction performance [74]. This form of learning is characterized by heterogeneous features that can be partitioned into groups (i.e. views). Typical examples of views, in the commonly cited context of person identification, include fingerprints, photographs, and personal identification documents—each with its own set of features. While there have been substantial efforts put into advancing the domain of multi-view learning (e.g. regression [75] and clustering [76]), the application of this particular approach in aircraft design has, thus far, been unexplored.

Multiple viewpoints of a particular component or subsystem generally occur in the form of numerical simulations that emphasize on a specific facet of that component or subsystem (as described previously in the context of plate-like structures). This is often achieved by skewing the fidelity of the simulation in favor of phenomena of interest. For example, in the design optimization effort of a compressor, engineers could model the object with all its 3-dimensional intricacies, detailing the complex shape of each blade, the effects of gravity, and many other carefully defined minutiae. As these simulations would be extremely expensive, engineers may often restrict the scope of the model by emphasizing only those physical phenomena that are temporarily of the most interest. As illustrated in Figure 3, instead of treating each view independently of the others, multi-view learning could be exploited to provide a much more robust, comprehensive, and consequently accurate approximation of the overall reality.

## IV. Case Study

In order to demonstrate the potential of the machine learning paradigms described above, we present next an illustrative case study of simulation data reuse through transfer learning (Section III-A) in the context of aircraft engine design.

A major task in the early phase of the development of a new engine is a preliminary assessment of new design concepts. More precisely, overall engine attributes, such as weight, thrust, or specific fuel consumption, are estimated through simulation for the given design performance parameters of a different kind; e.g. pressures, temperatures, velocities, etc. This space is typically explored with the aim of resolving a good starting design point which will be further refined upon at the component-level analyses that are often based on high-fidelity physics-based numerical simulations. The adoption of surrogate modeling to assist engineers in this search comes as no surprise if we consider the sparseness of the design space that is caused by the (typically) high dimensionality and the significant cost of numerical simulations [77]. Such a broad search space for the task at hand poses some issues to the learning of surrogate models. With no additional information but the simulation data, an insufficient amount of the latter may lead to a poor fit of the response function. Moreover, we are keen on enhancing the quality of the fit so as to improve the accuracy of the starting design points, which calls for even more simulations. On the other hand, we simultaneously seek to reduce the computational cost of the response function approximation process so that the design space can be more rigorously navigated. This brings about the need for enhanced approaches capable of learning models that make the most effective use of available data, i.e., providing practically admissible prediction performance given a small number of data instances.

A widespread solution direction to deal with these issues relies on space reduction techniques [72]. One broad family of approaches adaptively refines the region of the design space to sample from. However, this implies that a progressively increasing number of "physical" simulations have to be executed at each round. From a complementary point of view, a well-known group of methods aims to identify and remove unimportant input variables, e.g. through sensitivity analysis. It is contended that such a strategy may nevertheless be ineffective in some high-dimensional scenarios since the number of remaining variables continues to be too high.

In contrast to the above, the alternative solution direction advocated in this paper is to reuse the large amounts of simulation data generated from distinct (but possibly related) engine design exercises in the past or progressing concurrently. If the machine learning algorithm is able to effectively exploit the underlying synergy between designs, then an additional pool of simulations would become technically accessible *for free*. Before proceeding any further, we highlight that data transfer and space reduction can be complementary approaches. Indeed, their proper combination has already been shown to be advantageous in machine learning research [78], [79]; alluding to the possibility of further improvements in surrogate modeling for engine design.

In this case study, we devise two use cases that represent scenarios design engineers commonly face in practice and which readily accommodate a data transfer approach.

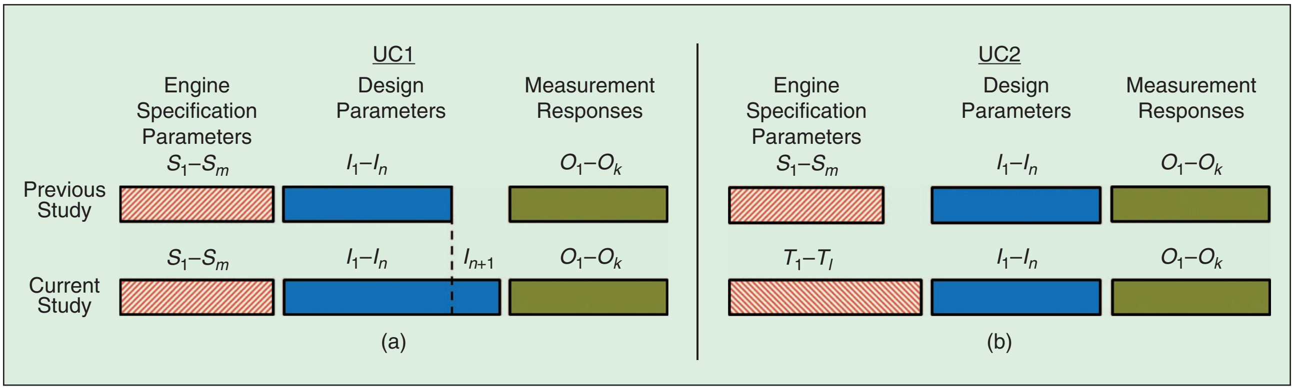



**FIGURE 4** Diagrams representing use cases UC1 (left) and UC2 (right).

**TABLE 1 List of the input and output parameters for the previous and current study datasets for UC1 and UC2. For similarly labeled parameters, the subsystem where it affects is indicated inside the parenthesis.**

| INPUT PARAMETERS | | | OUTPUT PARAMETERS |
|---|---|---|---|
| **UC1 CURRENT STUDY** | **UC1 PREVIOUS STUDY** | **UC2 PREVIOUS AND CURRENT STUDIES** | **UC1 & UC2 PREVIOUS AND CURRENT STUDIES** |
| ALTITUDE-FT | ALTITUDE-FT | ALTITUDE-FT | O1: NET THRUST |
| FAN BYPASS RATIO | FAN BYPASS RATIO | – | O2: FNET/W |
| PRESSURE RATIO (COMPRESSOR) | PRESSURE RATIO (COMPRESSOR) | PRESSURE RATIO (COMPRESSOR) | O3: FUEL FLOW |
| PRESSURE RATIO (BURNER) | PRESSURE RATIO (BURNER) | PRESSURE RATIO (BURNER) | O4: TSFC |
| EFFICIENCY (TURBINE) | EFFICIENCY (TURBINE) | EFFICIENCY (TURBINE) | O5: CORE AIRFLOW |
| A8/A2 (NOZZLE) | – | A8/A2 (NOZZLE) | O6: WEIGHT |

**Use Case 1 (UC1):** In this case, two consecutive studies are conducted, both based on the same engine specifications and both looking at the same responses. Importantly, the study addresses the impact of including an additional design parameter (such as the compressor pressure ratio) to the initial study. A diagram representation of UC1 may be found on the left side of Figure 4.

**Use Case 2 (UC2):** This use case depicts a scenario where data for a different engine specification to the current one of interest has been captured previously. Some underlying relationship between both engines is indeed recognized (e.g. similar turbojet engines distinguished mainly by the presence or absence of an afterburner). Notably, the current and previous studies consider the same set of design parameters and output responses. The right side of Figure 4 shows a schematic representation of UC2.

In the remainder of this section, we showcase an empirical assessment of the possible improvements to surrogate models that are well within our reach by simply incorporating transfer learning ideas into the aforementioned use cases.

### *A. Benchmark Data Generation*

For the evaluation of the different methods, benchmark data was generated for each of the use cases by utilizing EngineSim (version v1.8a) [80], a popular open-source simulator developed by NASA Glenn Research Center. For UC2, real-world data was also provided by Rolls-Royce.

EngineSim is a simple, middle-scale simulator of the aircraft propulsion system whose outputs are macroscopic performance measurements, in tune with the preliminary nature of the tasks considered in this demonstration. Additionally, the simulator possesses some desirable properties for benchmarking:[1] it is problem specific, scalable and time economical.

In each use case, 300 engine configurations are simulated for the source (previous study) dataset, and 500 configurations are sampled for the target (current study of interest) dataset. Each dataset is sampled using a latin hypercube design of experiment (DOE) procedure and rescaled to a mean of zero and a standard deviation of one. For UC1, both the current and previous study datasets are generated from the CF6 Turbofan model. More generally, the baseline parameters that specify this engine are identical for both source and target studies. However, the previous study is carried out with 5 input parameters, while the current one has 6 parameters. Table 1 shows the set of inputs for both datasets. The table also presents the 6 outputs of interest that are considered. In UC2, the target dataset is sampled from the CF6 Turbofan model while the source dataset is sampled from a J85 Turbojet model. In UC2, distinct from UC1, the baseline parameters are different. The input parameters are, however, overlapping for both studies; these are shown in Table 1. The output responses considered are the same as in UC1.

[1]For the sake of replicability, all the generated datasets are available for download at https://blogs.ntu.edu.sg/rr-ntucorplab/dacs/dacs-publications/.

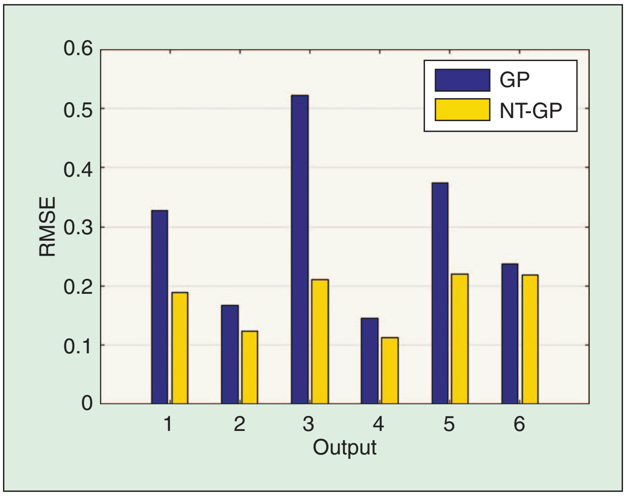



**FIGURE 5** *RMSE* values of GP and NT-GP for each output on UC1.

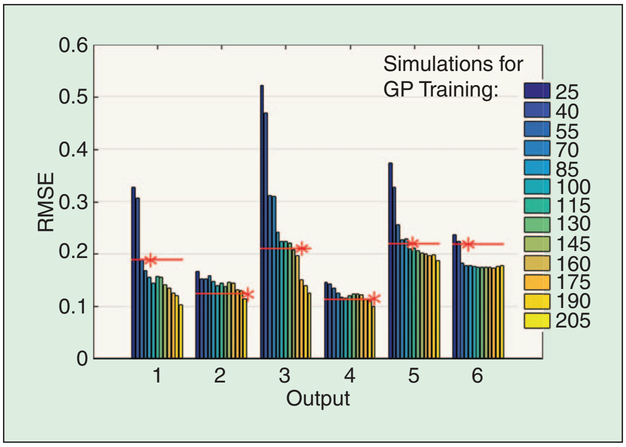



**FIGURE 6** *RMSE* for each output with increasing number of simulations for training a GP model. The red segments show the *RMSE* of NT-GP trained with 20 (target) simulations, and the red stars highlight the points where GP starts outperforming NT-GP.

### *B. Experimental Evaluation*

As discussed in Section 3.1, given a target learning task and knowledge acquired when solving a different source learning task, transfer learning (TL) techniques attempt to improve the learning of the target task using the knowledge from the source. To evaluate the performance enhancement that can be achieved by leveraging ideas from TL, an empirical comparison with a traditional surrogate modeling strategy is carried out. The latter consists of applying Gaussian process regression (GP) [81]—also referred to as kriging—on the target dataset. It is important to note that the TL strategies considered here augment the baseline learning method and, therefore, are used with GP for fairness of comparison. All GP models utilize a standard squared exponential covariance function. The hyperparameters of each method are determined by approximating the maximum model likelihood for the given data. A conjugate gradient approach is adopted for this search, with a maximum of 200 evaluations. The reported results correspond to the model providing the best objective function value over 10 independent runs with random hyperparameter initializations. The generalization error of each method is approximated through a hold-out scheme where 20 instances from the target dataset are sampled for training, and the remaining 480 instances are used for testing. The error is calculated using the root mean squared error (RMSE) measure.

#### 1) Evaluation of Use Case 1

Perhaps one of the simplest TL approaches is to combine source and target data, and then to apply a classical learning method, e.g. GP. To test this Naïve-Transfer GP (NT-GP) in UC1, the 6th parameter is manually added into the source dataset and filled with the mean of the 6th parameter in the target dataset.

Figure 5 shows the RMSE results of two approaches, i.e., the standard GP with no transfer, and the NT-GP. As can be observed, NT-GP outperforms GP in all the outputs, with a dramatic improvement in three. The overall result suggests that in a situation where the input-output functions between the source and target datasets are identical (as a consequence of identical engine specification), even a simple combination of datasets can result in non-trivial improvements to the accuracy of predictions. In general, it is expected that the accuracy of a learning method will improve with an increase in the amount of target training data ([82], ch. 18). Accordingly, the results indicate that when following the traditional GP methodology, a larger volume of physical simulation data is needed in order to approach the performance of NT-GP. This observation is substantiated in Figure 6, where the RMSE of GP trained on a progressively increasing number of simulations is shown. The red star (∗) on the horizontal bar in Figure 6 indicates the data size from which GP begins to perform better than the NT-GP. Taking the RMSE of NT-GP as a baseline provides an estimation of the amount of extra simulation data that would need to be generated with GP. As can be observed, the savings with NT-GP are lower for outputs O1 and O6 (55 and 40 simulations, respectively) than for the rest, which reach up to 190 simulations (O2).

#### 2) Evaluation of Use Case 2

The evaluation for UC2 involves three TL methods: NT-GP, a more sophisticated Adaptive Transfer Gaussian Process (AT-GP) [83], and a Metaheuristic-based Instance Selection for Transfer (MIST) approach.

In the first round, NT-GP is once again compared with GP. Unlike UC1, Figure 7 shows that the results of NT-GP are now, generally, worse than GP. Indeed, *negative transfer* may occur if the naïve combination of those datasets results in adulterated samples from the source distribution that misleads the GP. This is due to the dissimilarities in the response surfaces of the source and target engines. In general, most TL methods build upon the implicit assumption that the source domain is relevant to the target domain in a certain sense [65]. Therefore, there is a need for approaches that are capable of adapting the source knowledge to be reused by the TL method applied for the target problem at hand. This has been acknowledged to be a critical issue in order to avoid negative transfer [84], [65]. Here we evaluate two such adaptive TL approaches.

**AT-GP [83]:** GP provides a means for a non-parametric, flexible modeling and subsequent inference on the set of response functions of a domain of interest. For example, in our case, such a domain is given by the input design space and the engine performance output. The correlations among output values at distinct input locations are captured in the covariance matrix resulting from the evaluation of a covariance function $k(\cdot,\cdot)$ on the set of input pairs. In general, inputs that are close to each other, according to the similarity measure defined by $k$, will induce high correlations in the output space. In a scenario with two domains, where one of them is the target, a direct use of $k$ on all the available inputs may lead the model to endorse deceptive dependencies if the source points are not consistent with the correspondence between inputs similarity and output dependencies in the target domain. This suggests that the similarities between the domains should also be taken into account. AT-GP accounts for the relatedness between the source and target domains through a transfer covariance function explicitly capturing their correlation. More precisely, denoting the source and target domains by $\mathcal{S}$ and $\mathcal{T}$ respectively, and given two inputs $\mathbf{x}, \mathbf{x}'$ from either domain, AT-GP employs the following covariance function:

$$k^*(\mathbf{x},\mathbf{x}') = \begin{cases} \lambda k(\mathbf{x},\mathbf{x}'), & \mathbf{x}\in\mathcal{S}\ \&\ \mathbf{x}'\in\mathcal{T} \quad \text{or} \quad \mathbf{x}\in\mathcal{T}\ \&\ \mathbf{x}'\in\mathcal{S} \\ k(\mathbf{x},\mathbf{x}'), & \text{otherwise} \end{cases}$$

where $k(\cdot,\cdot)$ is any valid covariance function and $\lambda$ is a hyper-parameter measuring the correlation between points from $\mathcal{S}$ and $\mathcal{T}$. Note that, for $k^*$ to represent a valid covariance function, it must be positive semi-definite. This is fulfilled whenever $-1 \leq \lambda \leq 1$ [83].

Through model learning, $\lambda$ is expected to capture the varying transfer strengths between different source-target domain pairs. If a source is highly correlated with the target (either positively or negatively), $|\lambda|$ would approach 1 and the source data points would play an important role in the inference (i.e. output prediction, in our case), mostly if the number of target points is comparatively low. By contrast, for sources completely unrelated to the target, $|\lambda|$ would become 0 and so they would not be taken into account. In this way, the beneficial transfer effect is enhanced and the harmful transfer effect is automatically reduced.

**MIST [52]:** Different to AT-GP, MIST represents a model-independent alternative. In this case, the adaptive strategy consists of automatically selecting the most relevant subset of points from the source dataset. With this aim, we have developed a general approach relying on a search for the optimal subset of source instances that is in tune with the similarity assumption of the TL method at hand. Here the latter is NT-GP. In the scenario of UC2, assuming the DOE sampling method for the source and the target is the same, the underlying marginal distribution of inputs will also be the same. So, the conditional distribution of outputs given the inputs is what

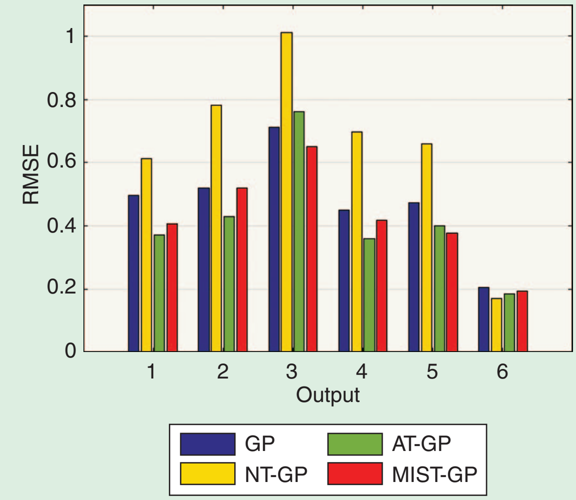



**FIGURE 7** *RMSE* values of GP, NT-GP, AT-GP and MIST-GP, for each output on UC2.

determines whether a sample of source points has been drawn from a source distribution equal to the target distribution or not. By keeping only the source points with equal conditional distribution in source and target we obtain a larger, representative sample of the target domain which we would expect to help NT-GP in the building of the GP model. In this case, therefore, we would be using MIST to search for the sources subset that, together with the target points, forms the most representative sample of the target distribution. Notice that if this subset is empty, then the source data is completely irrelevant and therefore no transfer takes place.

More generally, since the objective is to enhance predictive performance, in order to evaluate a candidate source subset, the TL method at hand is run and its generalization error is estimated. This dependence on the TL method implies that the objective function of the optimization problem may vary significantly. Furthermore, if the predictive behavior of the method is unclear, then the properties of the function may be unknown. Additionally, the size of the search space (i.e. the power set of the source dataset) grows exponentially with the number of source simulations. The MIST approach relies on population-based metaheuristics to address such an ill-defined optimization scenario [85]. These algorithms navigate the search space by iteratively updating a set of candidate solutions (population) until a stopping condition is fulfilled (e.g. a maximum number of iterations). The population update consists of selecting a high-quality subset from the current population, capturing the information about the search space from this subset into a model, and sampling this model to obtain new solutions which may enter the population if the quality is improved. A broad range of search space models has been developed over the years, leading to several families of metaheuristics (e.g. genetic algorithms, differential evolution, etc.).

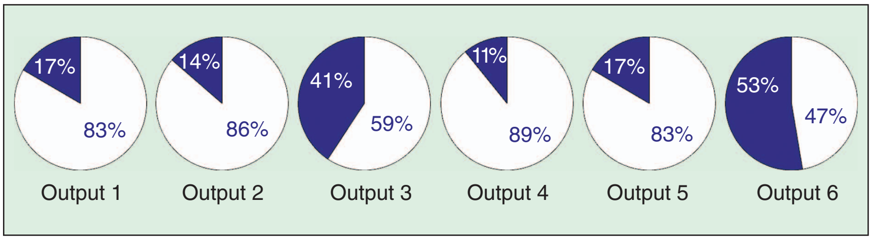



**FIGURE 8** Percentage of source data employed (dark colored) for training an NT-GP model with MIST for each output in UC2.

As Figure 7 shows, the two adaptive TL methods considered here are able to outperform GP and, more noticeably, NT-GP in almost all the cases. AT-GP achieves the largest improvements in three of the outputs, however, it also shows a slightly worse RMSE (compared to standard GP) for output O3. By contrast, MIST seems to offer a more consistent behavior by producing enhancements over GP across all the outputs. Figure 8 presents the averaged proportion of the source dataset selected after the search in MIST for each output. As can be observed, the percentage does not increase further than about 50%, suggesting that a portion of the source data may not be viable for reuse, while there indeed exists a significant subset that is beneficial. These figures serve to better understand the poor predictions of NT-GP. Notice that the only algorithmic difference between MIST and NT-GP comes from the proportion of source data points that are selected for reuse while learning the GP. In the latter method, it is 100% of the data. This implies that simply via an appropriate selection of points we can make dramatic improvements in surrogate modeling, turning the negative transfer into a positive one. Overall, this observation stresses the importance of a suitable prior investigation into *what knowledge to transfer*.

For this case study, we have augmented our preliminary version of MIST [52] by incorporating an estimation of distribution algorithm [86] in which, at each step of the search, a probability distribution of the most promising solutions (source subsets) encountered so far is estimated with a tree-structured Bayesian network. Since NT-GP learns a GP model by maximizing the likelihood, the contribution of one sample point to this measure is affected by the rest of the points considered in the sample. This indicates that the objective function (generalization error) of the optimization problem is defined on interrelated variables (source data points). Instead of considering these variables as independent throughout the search, using a tree model of the search space allows to capture order-1 dependencies from the promising source points and subsequently sample new solutions taking the dependencies into account. The evaluation of a candidate subset consists of estimating the generalization error of NT-GP by means of an internal $k$-fold cross-validation scheme where $k = 3$. The population size in MIST is set to 400, and 100 iterations are performed during the search. Since the population is initialized at random, MIST is repeated 10 times to get an estimate of the prediction error of the approach.

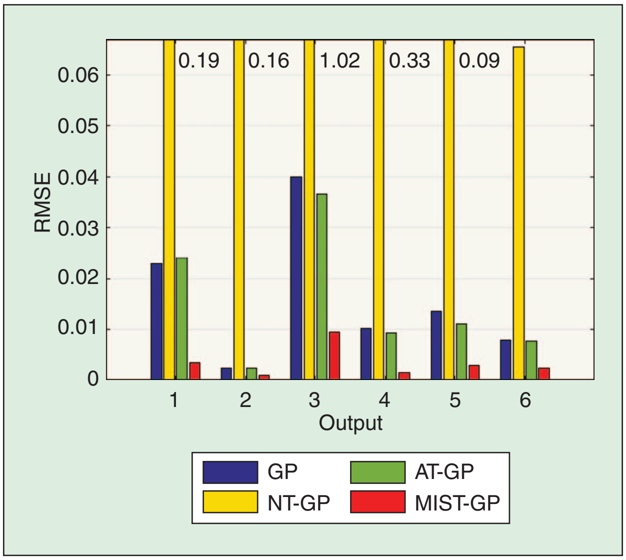



**FIGURE 9** *RMSE* for GP, NT-GP, AT-GP, and MIST-GP for each output in UC2 of the real-world data. The numbers within the figure window represent the actual values of the yellow bars that extend outside the figure.

### 3) Results on Real-World Data

In order to further explore their applicability, the TL methods above were tested on real-world data provided by Rolls-Royce that falls under the categorization of UC2. The pair of datasets was sampled using a latin hypercube DOE procedure from two different engines, where the number of source and target simulations were 274 and 100 respectively. Both datasets have the same 5 input parameters and 6 outputs. In this case, a 10-fold cross-validation procedure was used to approximate the generalization error.

As is shown in Figure 9, the significant difference between the performance of GP and NT-GP is a further indication that a more rigorous and adaptive approach is required. Note that this disparity is much larger than was observed in UC2 on EngineSim data (Figure 7). Once again, the two adaptive methods clearly overcome the deficiencies of NT-GP. This time, however, AT-GP offers a moderate improvement over GP. In contrast, MIST shows a dramatic increase in the accuracy of the predictions for all the outputs. In spite of these results, one clear problem with the MIST approach is that it relies on a wrapper strategy in order to find the (sub)optimal subset of source data. While this is advantageous if the goal is to minimize the prediction error, it also limits the scalability of the method. Therefore, the AT-GP method could still be a more desirable option if efficiency is an issue.

Nonetheless, in light of the promising outcomes of these experiments, the investigation of cost-effective refinements of the MIST approach becomes a potential avenue for additional enhancements. More generally, in the field of TL, various algorithms with different strengths and weaknesses have been developed. Aircraft surrogate modeling could be boosted further by leveraging these techniques to address more advanced scenarios, such as the availability of multiple sources of data, or by designing tailored TL methods upon aircraft design-specific domain knowledge.

## V. Conclusion

The aircraft plays an extremely important role in the modern-day society. It facilitates long distance travel and the growth of critical areas such as medicine, politics, and trade. Considered an effort in MDO, its design is characterized by complex analyses of mutually interdependent disciplines and large search spaces. To reduce the computational cost of modern disciplinary analyses, many have leveraged machine learning, a cornerstone of artificial and computational intelligence.

The design of an aircraft is generally done incrementally, drawing heavily from past experiences and technical knowledge. Consequently, there exists knowledge from various sources such as previously completed (and in-parallel) projects, components of similar functionality, and multiple fidelity simulations, that can be exploited in order to improve design performance. In this work, we summarize the current role of machine learning in aircraft design and identify a limitation of existing approaches. Specifically, the approximations are built only from data sampled for the current problem of interest, generally overlooking vast sources of relevant information that may be present outside the self-contained scope of a particular design task. Subsequently, we discuss three comparatively advanced machine learning technologies (*transfer learning*, *multi-task learning*, and *multi-view learning*) that can help address this issue by facilitating knowledge transfer across design tasks.

Transfer learning utilizes data from source domains in order to augment the approximation of a target domain of interest. In aircraft design, source domains would refer to related previously completed (or in-progress) projects. Multi-task learning, on the other hand, aims to simultaneously learn approximations for multiple variants of a similar domain; sharing information through a unified model representation in order to achieve better performance. It is generally the case that the input spaces between tasks are unaltered. In aircraft MDO, such scenarios can commonly occur when the design optimization of a particular component or subsystem is repeated several times conditioned upon the design of all other components in the overall system. This naturally gives rise to multiple (potentially related) variants of a particular task [87]. Multi-view learning, on the other hand, aims to utilize data from different views or perspectives (generally with differing input spaces) in order to build more accurate approximations. In aircraft design, these varying views could originate from physical simulations that provide an understanding of the system mechanics from different perspectives.

While there exists a fairly large body of research that has been performed thus far in regard to the application of these "advanced" machine learning technologies on a variety of domains, few (or none) have investigated their viability in aircraft design. In this work, we have provided a demonstration of one of these technologies (i.e. transfer learning) in engine design. In general, the transfer learning-enhanced surrogate models have outperformed its traditional counterparts in the two use cases that we have outlined. However, while these results are indeed promising, there remains a great deal of research that remains to be done. It is our hope that this work would help facilitate a merging of the two fields, namely, machine learning and aerospace engineering, to assist in progressing the state-of-the-art in aircraft technologies for the 21st century.

## Acknowledgment

This work was conducted within the Rolls-Royce@NTU Corporate Lab with support from the National Research Foundation (NRF) Singapore under the Corp Lab@University Scheme. The authors would like to thank Rolls-Royce PLC for permission to publish the results.

## References

[1] J. D. Mattingly, W. H. Heiser, and D. T. Pratt, *Aircraft Engine Design*, 2nd ed. Reston, VA: AIAA Education Series, 2002.
[2] S. Garber. (2017, Feb. 6). Part I: The age of propellers, Chapter 5: Design Refinement, 1939-45, examples of World War II aircraft. [Online]. Available: https://history.nasa.gov/SP-468/ch5-3.htm
[3] E. Torenbeek, *Advanced Aircraft Design: Conceptual Design, Technology and Optimization of Subsonic Civil Airplanes*. Hoboken, NJ: Wiley, 2013.
[4] J. R. Martins and A. B. Lambe, "Multidisciplinary design optimization: A survey of architectures," *AIAA J.*, vol. 51, pp. 2049–2075, July 2013.
[5] I. Kroo, S. Altus, R. Braun, P. Gage, and I. Sobieski, "Multidisciplinary optimization methods for aircraft preliminary design," in *Proc. 5th AIAA Symp. Multidisciplinary Analysis and Optimization*, Panama City Beach, FL, Sept. 7–9, 1994, pp. 697–709.
[6] R. Choudhary, A. Malkawi, and P. Papalambros, "Analytic target cascading in simulation-based building design," *Automat. Construct.*, vol. 14, pp. 551–568, Nov. 2005.
[7] P. Fuglsang and H. A. Madsen, "Optimization method for wind turbine rotors," *J. Wind Eng. Ind. Aerodyn.*, vol. 80, pp. 191–206, Mar. 1999.
[8] R. T. Haftka, J. Sobieszczanski-Sobieski, and S. L. Padula, "On options for interdisciplinary analysis and design optimization," *Struct. Optimization*, vol. 4, pp. 65–74, Jan. 1992.
[9] E. J. Cramer, J. E. Dennis, Jr., P. D. Frank, R. M. Lewis, and G. R. Shubin, "Problem formulation for multidisciplinary optimization," *SIAM J. Optimization*, vol. 4, pp. 754–776, July 1994.
[10] R. J. Balling and J. Sobieszczanski-Sobieski, "Optimization of coupled systems: A critical overview of approaches," *AIAA J.*, vol. 34, pp. 6–17, Jan. 1996.
[11] N. M. Alexandrov and M. Y. Hussaini, *Multidisciplinary Design Optimization: State of the Art*, vol. 80. Philadelphia, PA: SIAM, 1997.
[12] R. W. Claus, A. L. Evans, and G. J. Follen, "Multidisciplinary propulsion simulation using NPSS," in *Proc. 4th AIAA Symp. Multidisciplinary Analysis and Optimization*, Cleveland, OH, Sept. 21–23, 1992, pp. 191–196.
[13] G. Medic, G. Kalitzin, D. You, M. Herrmann, F. Ham, E. van der Weide, H. Pitsch, and J. Alonso, "Integrated RANS/LES computations of turbulent flow through a turbofan jet engine annual research briefs," in *Annual Research Briefs*. Stanford, CA: Center for Turbulence Research, Univ. Stanford, 2006, pp. 275–285.
[14] G. G. Chowdhury, "Natural language processing," *Annu. Rev. Inform. Sci. Technol.*, vol. 37, pp. 51–89, Jan. 2003.
[15] N. M. Oliver, B. Rosario, and A. P. Pentland, "A bayesian computer vision system for modeling human interactions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 831–843, Aug. 2000.
[16] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European Conf. Computer Vision*, Graz, Austria, Sept. 7–13, 2006, pp. 430–443.
[17] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, pp. 734–749, Apr. 2005.
[18] R. C. Martin, *Agile Software Development: Principles, Patterns, and Practices*. Upper Saddle River, NJ: Prentice Hall, 2002.
[19] A. Gunasekaran, *Agile Manufacturing: The 21st Century Competitive Strategy*. Amsterdam, The Netherlands: Elsevier, 2001.
[20] D. Lim, Y. Jin, Y. S. Ong, and B. Sendhoff, "Generalizing surrogate-assisted evolutionary computation," *IEEE Trans. Evol. Comput.*, vol. 14, pp. 329–355, Dec. 2010.
[21] M. N. Le, Y. S. Ong, Y. Jin, and B. Sendhoff, "A unified framework for symbiosis of evolutionary mechanisms with application to water clusters potential model design," *IEEE Comput. Intell. Mag.*, vol. 7, pp. 20–35, Jan. 2012.
[22] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. K. Tucker, "Surrogate-based analysis and optimization," *Progr. Aerospace Sci.*, vol. 41, pp. 1–28, Jan. 2005.
[23] T. W. Simpson, V. Toropov, V. Balabanov, and F. A. Viana, "Design and analysis of computer experiments in multidisciplinary design optimization: A review of how far we

have come or not," in *Proc. 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conf.*, Victoria, Canada, Sept. 1–22, 2008, pp. 10–12.
[24] A. I. Forrester and A. J. Keane, "Recent advances in surrogate-based optimization," *Progr. Aerospace Sci.*, vol. 45, pp. 50–79, Jan. 2009.
[25] O. Bandte, "A probabilistic multi-criteria decision making technique for conceptual and preliminary aerospace systems design," Ph.D. dissertation, School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, 2000.
[26] H.-M. Kim and K.-Y. Kim, "Design optimization of rib-roughened channel to enhance turbulent heat transfer," *Int. J. Heat Mass Transfer*, vol. 47, pp. 5159–5168, Nov. 2004.
[27] Y. Yanhui, L. Dong, H. Ziyan, and L. Zijian, "Optimization of preform shapes by RSM and FEM to improve deformation homogeneity in aerospace forgings," *Chinese J. Aeronautics*, vol. 23, pp. 260–267, Apr. 2010.
[28] N. K. Madavan, M. M. Rai, and F. W. Huber, "Neural net-based redesign of transonic turbines for improved unsteady aerodynamic performance," National Aeronautics and Space Administration, Ames Research Center, Los Angeles, CA, Tech. Rep. NASA/TM-1998-208754, Nov. 1998.
[29] M. M. Rai and N. K. Madavan, "Aerodynamic design using neural networks," *AIAA J.*, vol. 38, pp. 173–182, Sept. 2000.
[30] C. J. Miller, A. C. Maser, E. Garcia, and D. N. Mavris, "Invent surrogate modeling and optimization of transient thermal responses," in *Proc. 50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Nashville, TN, Jan. 9–12, 2012, doi: 10.2514/6.2012-1123.
[31] A. I. J. Forrester, A. J. Keane, and N. W. Bressloff, "Design and analysis of "noisy" computer experiments," *AIAA J.*, vol. 44, pp. 2331–2339, Oct. 2006.
[32] W. Huang, Z.-G. Wang, D. B. Ingham, L. Ma, and M. Pourkashanian, "Design exploration for a single expansion ramp nozzle (sern) using data mining," *Acta Astronaut.*, vol. 83, pp. 10–17, Nov. 2013.
[33] W. Huang, "Design exploration of three-dimensional transverse jet in a supersonic crossflow based on data mining and multi-objective design optimization approaches," *Int. J. Hydrogen Energy*, vol. 39, pp. 3914–3925, Mar. 2014.
[34] D. J. J. Toal and A. J. Keane, "Efficient multipoint aerodynamic design optimization via cokriging," *J. Aircraft*, vol. 48, pp. 1685–1695, Sept. 2011.
[35] M. F. Ozcan, B. Kestner, J. C. Tai, and D. N. Mavris, "Pattern classification of a civilian turbofan's state space for real-time surrogate modeling," in *Proc. 49th AIAA/ASME/SAE/ASEE Joint Propulsion Conf.*, Cleveland, OH, July 28–30, 2013, doi: 10.2514/6.2013-4003.
[36] Q. Zeeshan, D. Yunfeng, S. Ghumman, A. F. Rafique, and A. Kamran, "Support vector regression-driven multidisciplinary design optimization of multistage ground based interceptor," in *Proc. AIAA Modeling and Simulation Technologies Conf.*, Chicago, IL, Aug. 10–13, 2009, doi: 10.2514/6.2009-6238.
[37] M. N. Le, Y. S. Ong, S. Menzel, C.-W. Seah, and B. Sendhoff, "Multi co-objective evolutionary optimization: Cross surrogate augmentation for computationally expensive problems," in *Proc. IEEE Congr. Evolutionary Computation*, Brisbane, Australia, June 10–15, 2012, doi: 10.1109/CEC.2012.6252915.
[38] Z. Huang, C. Wang, J. Chen, and H. Tian, "Optimal design of aeroengine turbine disc based on kriging surrogate models," *Computers Structures*, vol. 89, pp. 27–37, Jan. 2011.
[39] N. Papila, W. Shyy, L. Griffin, and D. J. Dorney, "Shape optimization of supersonic turbines using global approximation methods," *J. Propulsion Power*, vol. 18, pp. 509–518, May 2002.
[40] R. Sellar, S. Batill, and J. Renaud, "Response surface-based, concurrent subspace optimization for multidisciplinary system design," in *Proc. 34th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, Jan. 15–18, 1996, doi: 10.2514/6.1996-714.
[41] A. Keane, "Wing optimization using design of experiment, response surface, and data fusion methods," *J. Aircraft*, vol. 40, pp. 741–750, July 2003.
[42] D. M. Jaeggi, G. T. Parks, W. N. Dawes, and P. J. Clarkson, "Robust multi-fidelity aerodynamic design optimization using surrogate models," in *Proc. 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conf.*, Victoria, Canada, Sept. 10–12, 2008, doi: 10.2514/6.2008-6052.
[43] E. Minisci and M. Vasile, "Robust design of a re-entry unmanned space vehicle by multifidelity evolution control," *AIAA J.*, vol. 51, pp. 1284–1295, June 2013.
[44] B. Tracey, K. Duraisamy, and J. J. Alonso, "A machine learning strategy to assist turbulence model development," in *Proc. 53rd AIAA Aerospace Sciences Meeting*, Kissimmee, FL, Jan. 5–9, 2015, doi: 10.2514/6.2015-1287.
[45] K. Duraisamy, Z. J. Zhang, and A. P. Singh, "New approaches in turbulence and transition modeling using data-driven techniques," in *Proc. 53rd AIAA Aerospace Sciences Meeting*, Kissimmee, FL, Jan. 5–9, 2015, doi: 10.2514/6.2015-1284.
[46] E. J. Parish and K. Duraisamy, "A paradigm for data-driven predictive modeling using field inversion and machine learning," *J. Comput. Phys.*, vol. 305, pp. 758–774, Nov. 2016.
[47] P. Moin and K. Mahesh, "Direct numerical simulation: A tool in turbulence research," *Annu. Rev. Fluid Mech.*, vol. 30, pp. 539–578, Jan. 1998.
[48] B. Dolšak, I. Bratko, and A. Jezernik, "Application of machine learning in finite element computation," in *Machine Learning and Data Mining: Methods and Applications*. Hoboken, NJ: Wiley, 1998, ch. 5.
[49] B. Dolšak, "Finite element mesh design expert system," *Knowledge-based Syst.*, vol. 15, pp. 315–322, July 2002.
[50] C.-H. Ahn, S.-S. Lee, H.-J. Lee, and S.-Y. Lee, "A self-organizing neural network approach for automatic mesh generation," *IEEE Trans. Magn.*, vol. 27, pp. 4201–4204, Sept. 1991.
[51] L. Manevitz, M. Yousef, and D. Givoli, "Finite-element mesh generation using self-organizing neural networks," *Microcomput. Civil Eng.*, vol. 12, pp. 233–250, July 1997.
[52] A. Tan Wei Min, R. Sagarna, A. Gupta, and Y. S. Ong, "Coping with data scarcity in aircraft engine design," in *Proc. 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conf. Forum*, Denver, CO, June 5–9, 2017, doi: 10.2514/6.2017-4434.
[53] E. V. Bonilla, K. M. A. Chai, and C. K. Williams, "Multi-task gaussian process prediction," in *Advances in Neural Information Processing Systems 20.* Vancouver, Canada: Curran Associates, 2008, pp. 153–160.
[54] K. Swersky, J. Snoek, and R. P. Adams, "Multi-task bayesian optimization," in *Advances in Neural Information Processing Systems 26.* Lake Tahoe, NV: Curran Associates, 2013, pp. 2004–2012.
[55] Z.-H. Han and S. Görtz, "Hierarchical kriging model for variable-fidelity surrogate modeling," *AIAA J.*, vol. 50, pp. 1885–1896, Sept. 2012.
[56] J. E. Valenzuela del Río and D. Mavris, "Gaussian process surrogate model for levering similar trends across concepts," *AIAA J.*, vol. 53, pp. 1002–1015, Apr. 2015.
[57] A. Bouchachia, "Incremental learning," in *Encyclopedia of Data Warehousing and Mining*, 2nd ed. Hershey, PA: IGI Global, 2009, pp. 1006–1012.
[58] D. Yogatama and G. Mann, "Efficient transfer learning method for automatic hyperparameter tuning," *17th Proc. Int. Conf. Artificial Intelligence and Statistics*, Apr. 22–25, 2014, vol. 33, pp. 1077–1085.
[59] D. Lim, Y. S. Ong, A. Gupta, C. K. Goh, and P. S. Dutta, "Towards a new praxis in optinformatics targeting knowledge re-use in evolutionary computation: simultaneous problem learning and optimization," *Evol. Intell.*, vol. 9, pp. 203–220, Dec. 2016.
[60] L. Feng, Y. S. Ong, S. Jiang, and A. Gupta, "Autoencoding evolutionary search with learning across heterogeneous problems," *IEEE Trans. Evol. Comput.*, Mar. 2017, doi: 10.1109/TEVC.2017.2682274.
[61] L. Feng, Y. S. Ong, M.-H. Lim, and I. W. Tsang, "Memetic search with interdomain learning: A realization between CVRP and CARP," *IEEE Trans. Evol. Comput.*, vol. 19, pp. 644–658, Oct. 2015.
[62] M. Iqbal, W. N. Browne, and M. Zhang, "Reusing building blocks of extracted knowledge to solve complex, large-scale boolean problems," *IEEE Trans. Evol. Comput.*, vol. 18, pp. 465–480, Sept. 2014.
[63] M. W. Hauschild, M. Pelikan, K. Sastry, and D. E. Goldberg, "Using previous models to bias structural learning in the hierarchical boa," *Evol. Comput.*, vol. 20, pp. 135–160, Feb. 2012.
[64] S. Baluja, "Incorporating a priori knowledge in probabilistic-model based optimization," in *Scalable Optimization via Probabilistic Modeling*, M. Pelikan, K. Sastry, and E. CantuPaz, Eds. New York: Springer, 2006, pp. 205–222.
[65] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, pp. 1345–1359, Oct. 2010.
[66] Y. Zhai, Y. S. Ong, and I. W. Tsang, "The emerging "big dimensionality," *IEEE Comput. Intell. Mag.*, vol. 9, pp. 14–26, July 2014.
[67] Y. S. Ong, P. B. Nair, and A. J. Keane, "Evolutionary optimization of computationally expensive problems via surrogate modeling," *AIAA J.*, vol. 41, pp. 687–696, Apr. 2003.
[68] B. Rakitsch, C. Lippert, K. Borgwardt, and O. Stegle, "It is all in the noise: Efficient multi-task gaussian process inference with structured residuals," in *Proc. Advances in Neural Information Processing Systems* Lake Tahoe, NV, 2013, vol. 26, pp. 1466–1474.
[69] A. Gupta, Y. S. Ong, and L. Feng, "Multifactorial evolution: toward evolutionary multitasking," *IEEE Trans. Evol. Comput.*, vol. 20, pp. 343–357, July 2016.
[70] A. Gupta, Y. S. Ong, L. Feng, and K. C. Tan, "Multiobjective multifactorial optimization in evolutionary multitasking," *IEEE Trans. Cybern.*, vol. 47, pp. 1652–1665, May 2016.
[71] Y. S. Ong and A. Gupta, "Evolutionary multitasking: A computer science view of cognitive multitasking," *Cogn. Comput.*, vol. 8, pp. 125–142, Mar. 2016.
[72] S. Shan and G. G. Wang, "Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions," *Struct. Multidisciplinary Optimization*, vol. 41, pp. 219–241, Aug. 2010.
[73] Y. X. Zhang and C. H. Yang, "Recent developments in finite element analysis for laminated composite plates," *Compos. Struct.*, vol. 88, pp. 147–157, Mar. 2009.
[74] S. Sun, "A survey of multi-view machine learning," *Neural Comput. Applicat.*, vol. 23, pp. 2031–2038, Feb. 2013.
[75] S. M. Kakade and D. P. Foster, "Multi-view regression via canonical correlation analysis," in *Proc. 20th Annu. Conf. Learning Theory*, 2007, pp. 82–96.
[76] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan, "Multi-view clustering via canonical correlation analysis," in *Proc. 26th ACM Annu. Int. Conf. Machine Learning*, Montreal, Canada, 2009, pp. 129–136.
[77] Y. Mack, "CFD-based surrogate modeling of liquid rocket engine components via design space refinement and sensitivity assessment," Ph.D. dissertation, Dept. Mechanical and Aerospace Eng., Univ. Florida, Gainsville, FL, 2007.
[78] S. Uguroglu and J. Carbonell, "Feature selection for transfer learning," in *Proc. European Conf. Machine Learning and Knowledge Discovery in Databases*, Athens, Greece, Sept. 5–9, 2011, vol. III, pp. 430–442.
[79] D. Kale and Y. Liu, "Accelerating active learning with transfer learning," in *Proc. IEEE 13th Int. Conf. Data Mining*, Dallas, TX, Dec. 7–10, 2013, pp. 1085–1090.
[80] (2014, Aug. 19). Enginesim v1.8a. NASA Glen Research Center. [Online]. Available: http://www.grc.nasa.gov/WWW/K-12/airplane/ngnsim.html
[81] C. E. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.
[82] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. 3rd ed. London, U.K.: Pearson, 2009.
[83] B. Cao, S. J. Pan, Y. Zhang, D.-Y. Yeung, and, and Q. Yang, "Adaptive transfer learning," in *Proc. 24th AAAI Conf. Artificial Intelligence*, Atlanta, GA, July 11–15, 2010, pp. 407–712.
[84] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications*. Hershey, PA: IGI Global, 2009, pp. 242–264.
[85] T. F. Gonzalez, *Handbook of Approximation Algorithms and Metaheuristics*. Boca Raton, FL: Chapman & Hall, 2007.
[86] P. Larrañaga and J. A. Lozano, *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*. Boston, MA: Kluwer Academic, 2002.
[87] A. Gupta, J. Mańdziuk, and Y. S. Ong, "Evolutionary multitasking in bi-level optimization," *Complex Intell. Syst.*, vol. 1, pp. 83–95, Feb. 2015.