EXTREME ENSEMBLE COMPUTATIONS FOR OPTIMIZATION UNDER UNCERTAINTY

John Axerio-Cilies*
Gianluca Iaccarino
Mechanical Engineering Department
Stanford University
488 Escondido Mall, Bldg. 500
Stanford, CA, 94305, USA
Email: axerio@stanford.edu, jops@stanford.edu

Giovanni Petrone
Department of Aerospace Engineering
University of Naples "Federico II"
Via Claudio 21, 80125 Napoli, Italy
Email: gpetrone@stanford.edu

Vijay Sellappan
Enginsoft Americas
2445 Faber Pl., Suite 102
Palo Alto, CA 94303-3347
Email: v.sellappan@enginsoft.com

Abstract. The development of robust design strategies coupled with detailed simulation models requires the introduction of advanced algorithms and computing resource management tools. On the algorithmic side, we explore the use of simplex-based stochastic collocation methods to characterize uncertainties, and multi-objective genetic algorithms to optimize a large-scale, three-dimensional geometry using a very large number (extreme ensemble) of CFD simulations on HPC clusters. The problem of interest is the optimization under uncertainty of a Formula 1 tire brake intake to maximize cooling efficiency and minimize aerodynamic resistance. The uncertainties are introduced in terms of the tire deformation and free stream conditions. A simulations environment – Leland – has been developed to dynamically *schedule, monitor and stir* the calculation ensemble and extract runtime information as well as simulation results and statistics. Leland is equipped with an auto-tuning strategy for optimal load balancing and fault tolerance checks to avoid failures in the ensemble.

Key words: meta-scheduler, fault tolerance, uncertainty quantification, optimization, parallel, high performance computing

1 INTRODUCTION

In the last few years, clusters with 10,000 CPUs have become available, and it is now feasible to design and optimize complex engineering systems using computationally intensive simulations. This development highlights the need to create resource managers that deliver cost-effective utilization with fault tolerance.

The BlueGene/L cluster with 65,536 nodes was designed to have less than one failure every ten days. In fact, this cluster and others like it experience an aver-

age of one processor failure every hour[1]. In light of this, it is necessary to study, develop, and continually improve strategies for efficient completion of large simulations. Theoretical work has been published in the literature that suggests that advanced algorithms might be available although they have only been demonstrated using test functions on a small number of compute nodes[2].

The design process involves running an extreme number of large computations or 'extreme ensemble' (on the order of thousands) in order to create a robust solution that will remain optimal under conditions that cannot be controlled ('uncertainties'). We call this process optimization under uncertainty. The ensemble is a list of runs generated by the optimization and uncertainty analysis algorithms that is dynamic in nature and is not deterministic. This means that the number of additional simulations is dependent on the results of the prior converged simulations.

In this paper we explore the computational design of a Formula 1 tire and brake assembly using large-scale, three-dimensional Reynolds-Averaged Navier-Stokes simulations on a high performance computing cluster. The purpose of designing the brake duct is to increase the amount of air captured by the duct while minimizing the total drag of the tire. This multi-objective optimization problem is tacked using a genetic algorithm which produces a Pareto front of best solutions. In addition, uncertainty analysis of 4 specific points on the Pareto front (minimum drag, maximum cooling, best operating point or trade-off, and baseline F1 tire geometry) are shown in the results section of this paper. Future work will include a study showing how uncertainties can be invasively incorporated in the optimization procedure, producing a probabilistic Pareto front rather than analyzing the sensitivity of the deterministic Pareto due to uncertainties. For such a study, there are approximately 400 simulations to perform per optimization cycle (i.e. generation). When the results of those 400 simulations are analyzed, an additional list of 400 simulations, each with a unique range of input parameters, are generated for the next generation in the optimization process. The values of the the input parameters for the next generation are not known a priori. The optimization procedure needs to account for uncertainties arising from variable inflow conditions as well as variability in the flexible tire geometry. This complex baseline geometry consists of 30 million mesh cells. In order to generate an optimal design under uncertainty the mesh is deformed locally, creating 5000 unique simulations to perform. Each simulation (or realization) will be run on our in-house cluster using 2400 cores; the full design process should take approximately 2 weeks to complete.

The second contribution of this paper is the development of a software platform able to reduce the total time needed to carry out an engineering design process such as the one described above. We have developed a simulations environment, hereafter referred to as Leland, that allows us to schedule the resources and to monitor the calculation ensemble and extract runtime information as well as simulations results and statistics on the fly. Leland is equipped with an auto-tuning strategy for selecting an optimal processor count. Moreover a fault tolerance strategy is implemented to ensure that a simulation or a processor stall is detected and does not impact the overall ensemble finish time. The results of this study show the actual computational time savings with efficient use of resources using Leland as opposed to submitting individual jobs on the cluster one at a time using traditional queue managers (e.g. Torque, SLURM, etc.).

2   ROBUST DESIGN ALGORITHM

The impact of uncertainties in the robust design process are characterized using the Simplex Stochastic Collocation (SSC) algorithm[3,4], which combines the effectiveness of random sampling in higher dimensions (multiple uncertainties) with the accuracy of polynomial interpolation. This approach is characterized by a superlinear convergence behavior, thus outperforming classical Monte Carlo sampling, although retaining its robustness. In the SSC methods, a discretization of the space spanned by the uncertain parameters is employed and the simplex elements obtained from a Delaunay triangulation of sampling points is constructed. The robustness of the approximation is guaranteed by using a limiter approach for the local polynomial degree based on the extension of the Local Extremum Diminishing (LED) concept to probability space. The discretization is adaptively refined by calculating a refinement measure based on a local error estimate in each of the simplex elements. A new sampling point is then added randomly in the simplex with the highest measure and the Delaunay triangulation is updated. The implementation of advanced algorithms to improve the scalability of Delaunay triangulation in higher dimensions, in order to circumvent the curse of dimensionality, has not been fully investigated as part of this study. There are proofs in the literature that show that Delaunay triangulation can achieve linear scaling with higher dimensions[5].

In this work we analyze a nontrivial multi-objective problem in which it is not possible to find a unique solution that simultaneously optimizes each objective: when attempting to improve an objective further, other objectives suffer as a result. A tentative solution is called non-dominated, Pareto optimal, or Pareto efficient if an improvement in one objective requires a degradation of another. We use the NSGA-II algorithm[6,7] to obtain the non-dominated solutions, therefore we analyze the more interesting solutions on the deterministic Pareto set in presence of uncertainty. This is done in order to prove the importance of taking in account the variability of several input conditions in the design process. For all these solutions the SSC is used to obtain a reconstruction of the objective function statistical moments, refining the simplexes until an accuracy threshold is reached.

3   DYNAMIC RESOURCE MANAGER - LELAND

The structure of Leland is based on a workflow through I/O sub-systems that represent the software applications (i.e. Sculptor, Fluent, Tecplot, Matlab etc.) involved in the process. This environment is designed to run natively on any high-performance computing (HPC) system, by integrating with the job-submission/ queuing system (e.g. Torque). Moreover, it does not require continuous management: once the analysis is initiated multiple simulations are submitted and monitored automatically. In Leland, a job is an instance of the entire multi-physics simulations, which might include grid generation, mesh morphing, flow solution and post-processing. The main objective of Leland is to set-up a candidate design as a job and to manage it until it is completed and to gather relevant results that are used to inform the optimization under uncertainty process. ROpt (robust optimum), shown in Figure 1a, is the engine behind this design environment. Given the design and/or uncertain input variables, ROpt continuously generates new design proposals (samples) based on the evolutionary strategy and/or analysis of the uncertainty space, until a convergence criterion is met.
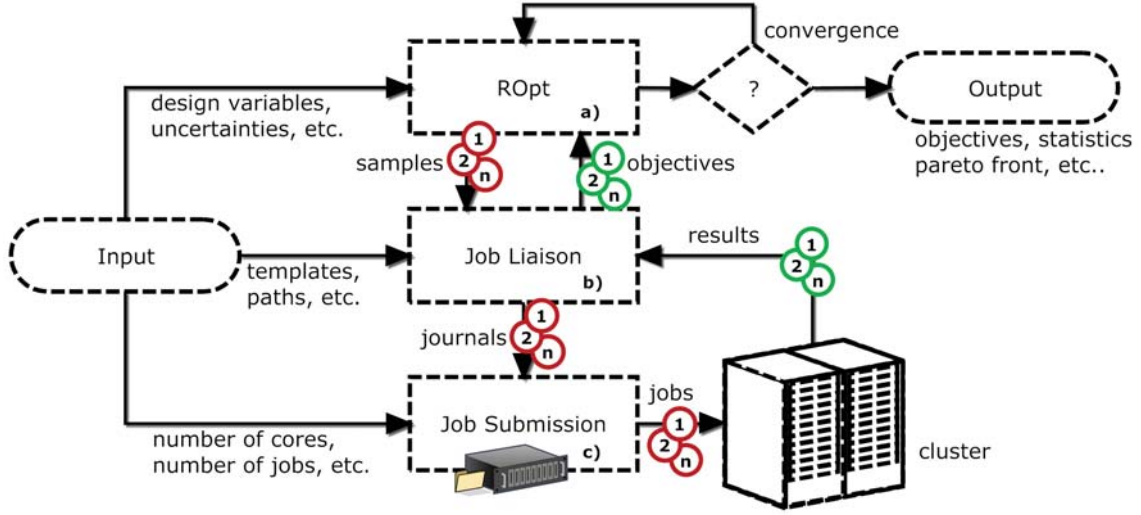
Figure 1: Leland flowchart

The Job Liaison, shown in Figure 1b, defines the characteristics of each single job and continuously monitors the progress of the simulations until completion in order to communicate the objective evaluations back to ROpt. It is the job of this module to continuously monitor for faults, stalls, or errors to ensure that the total runtime is not detrimentally affected by processor/memory failure.

The Job Submission engine, shown in Figure 1c, ensures that the correct number of jobs is always running on the cluster. The variables (number of cores, number of jobs, etc.) from the input file that are used to initialize the runs are dynamic, meaning they can be edited on the fly and the system will respond accordingly.

Leland has the ability to dynamically select the optimal number of processors to run per realization. This is achieved by auto-tuning. The user selects an optimal window of cores to use per realization prior to launching the full ensemble. The auto-tuning algorithm then samples the space by using a unique number of cores per realization in the ensemble. Once two or more realizations are complete the auto-tuning algorithm can start to construct an application specific speed-up curve (Figure 2). Speed-up is defined as the total time required to finish the simulation using 1 processor divided by the total time required to finish the simulation using p processors (see Equation 2).
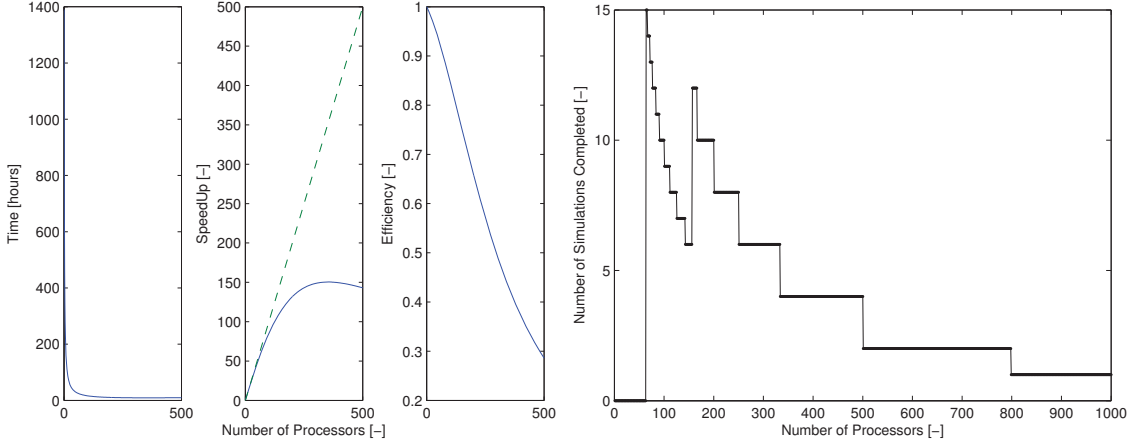
$$t_{total} = t_{serial} + t_{parallel} + t_{comm} \tag{1a}$$

$$= 5000 + \frac{5 \times 10^6}{p} + 40p \tag{1b}$$

$$\text{Speedup}(p) = \frac{t_{total}(1)}{t_{total}(p)} \tag{2}$$

$$\text{Efficiency}(p) = \frac{\text{Speedup}(p)}{p} \tag{3}$$

The speed-up curve in Figure 2 was generated by artificially replicating an HPC simulation. The time required to complete an HPC simulation is primarily a function of three factors i) portion of the code that is not parallelizable (t_{serial} in Equation

(a) Total time required to complete simulation as a function of the number of processors (left), Speed-up curve (middle), and Efficiency curve (right)

(b) Number of simulations that would be completed in a 24 hour window with 1000 available processors using exactly p processors for each simulation

Figure 2: Sample HPC simulation diagnostics

1) ii) portion of the code that is parallelizable ($t_{parallel}$ in Equation 1) and iii) the communication time between CPUs (t_{comm} in Equation 1). The serial portion of code in the example shown in Figure 2 is constant (5000 seconds) and not a function of the number of processors allocated to the job. The length of time required to complete the parallel portion of code in the example shown in the same figure is 5 million seconds divided by the number of processors used. Finally, there will always be some latency between CPUs and this is characterized by the communication time between nodes. The linear penalization we used in this example is 40 seconds per processor, but the latency slowdown could also be a more complex function related to the specific application.

Linear speed-up, also referred to as ideal speed-up, is shown as the green dotted line in the middle plot of Figure 2. An algorithm has linear speed-up if the time required to finish the simulation halves when the number of processors is doubled. It is common for fluid dynamic simulations to experience speed-down; this occurs when the total time required to finish the simulation actually increases with increasing processors. Leland has the ability to recognize the point at which speed-down occurs (at about 400 processors in Figure 2) and never use more than this number of processors. The rightmost plot in Figure 2 shows the efficiency (defined by Equation 3) curve for this artificial HPC simulation. The efficiency typically ranges between values of $0 \sim 1$ estimating how well utilized the processors are compared to the effort wasted in synchronization and communication. It is clear from this plot that the highest efficiency occurs with the lowest number of processors.

This speed-up curve will guide Leland's auto-tuning algorithm in assigning the optimal number of cores per realization (which may not be in the users original window). Since an ensemble of this size takes more than a few weeks on a large cluster, multiple job submissions need to be submitted to the local queuing system. These jobs are typically limited to 24 hour run times (or a wall clock time of 24 hours). Thus, it is essential that the auto-tuning algorithm recognizes how many hours remain prior to the job terminating due to the wall clock time and tries to increase the number of cores to finish as many realizations as possible within a

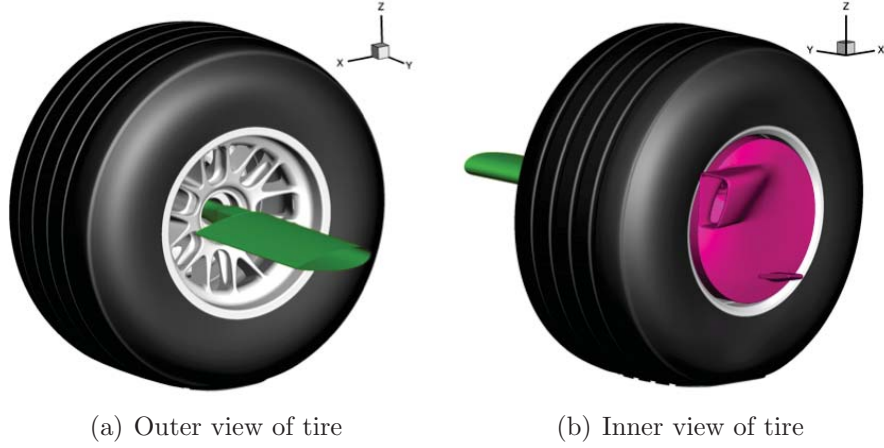(a) Outer view of tire   (b) Inner view of tire

Figure 3: Front right tire of the Formula 1 race car used in this study showing green airfoil strut used to secure tire to the experimental wind tunnel facility and the outer brake duct (magenta) used to cool the brake assembly

specific time frame.

4 APPLICATION DESCRIPTION

Leland is used to optimize the shape of a F1 tire brake duct (magenta color in Figure 3(b)), taking into account the geometrical uncertainties associated with the rotating rubber tire and uncertain inflow conditions.

The objectives are to minimize the tire drag [N] while maximizing the captured mass flow (kg/s) needed to cool the brake assembly. A computational mesh consisting of 30 million elements is considered for a fully detailed 3D wheel model (Figure 4). The simulations that require geometrical modification (either for optimization or uncertainty) are created using Sculptor, a commercial mesh deforming software from Optimal Solutions[8]. The software is used to generate multiple CFD mesh model variants, while keeping CAD and grid generators out of the design process loop, thus saving design time and costs substantially. The generated models are then used to compute the airflow around the tire geometry by a parallel CFD solver (Fluent v12.1.4). It is important to closely monitor the skewness of elements in Sculptor to ensure grid quality. If the deformation in Sculptor is too large, the CFD solver will diverge. The boundary conditions, computational setup, and experimental comparison for this case are outlined in separate studies[9,10,11,12].

4.1 Optimization Variables

A local mesh morphing software, Sculptor (v2.3.2), was used to deform the baseline Formula 1 brake duct (Figure 3). Specific control volumes were used to deform the brake duct in three dimensions, namely i) width of opening (Figure 5(a)) ii) height of opening (Figure 5(b)) and iii) protrusion length (Figure 5(c)). Each design variable was allowed to change by ± 1cm as shown in Figure 5.

4.2 Uncertain Variables

Multiple uncertain variables were tested to determine their sensitivity to output quantities of interest using a DOE (design of experiments) approach. Some of the uncertain variables were based on the inflow conditions (i.e. yaw angle, turbulent
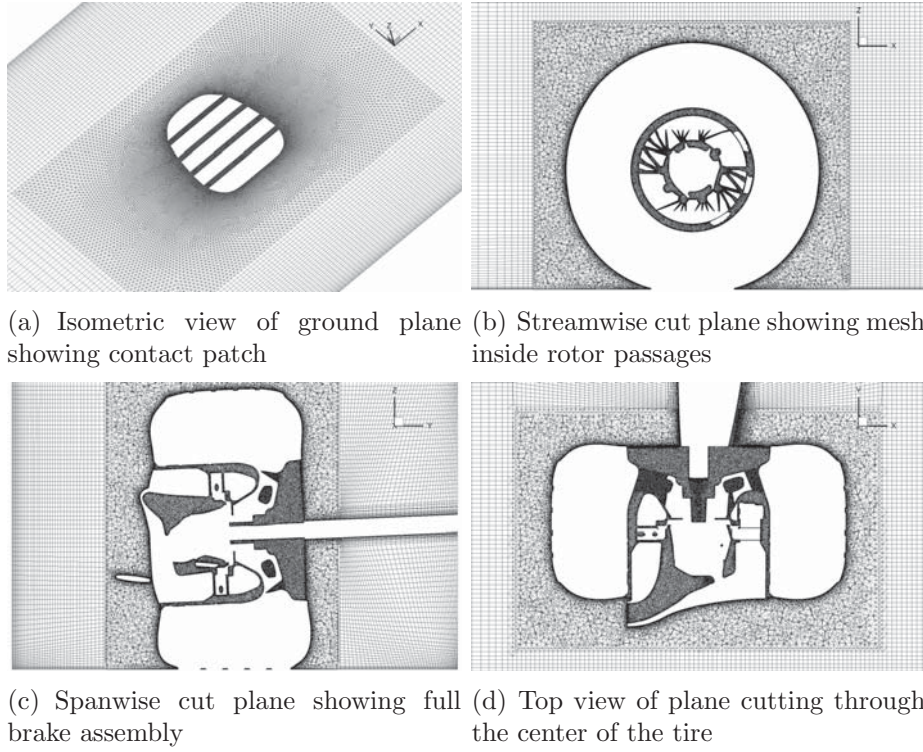
(a) Isometric view of ground plane showing contact patch

(b) Streamwise cut plane showing mesh inside rotor passages

(c) Spanwise cut plane showing full brake assembly

(d) Top view of plane cutting through the center of the tire

Figure 4: Four different views showing the Formula 1 tire mesh

(a) Brake duct width

(b) Brake duct height

(c) Brake duct length

Figure 5: Brake duct optimization variables

(a) Contact patch width

(b) Contact patch height

(c) Contact patch streamwise location

(d) Contact patch spanwise location

(e) Contact patch yaw angle

(f) Tire bulge radius

(g) Contact patch platform height

(h) Tire compression

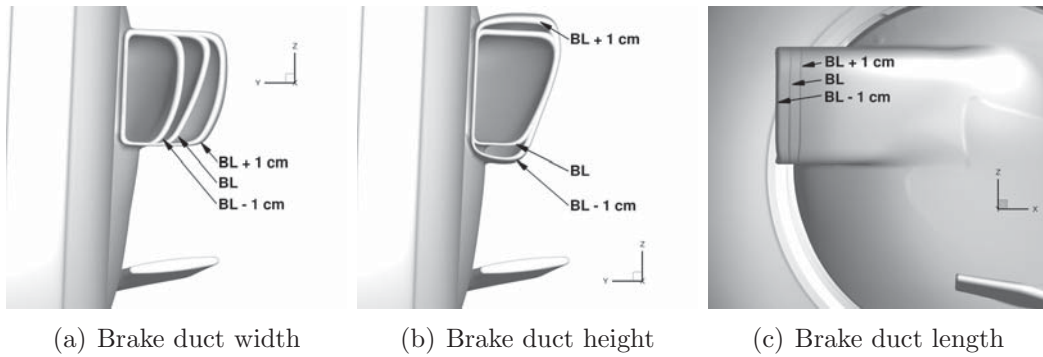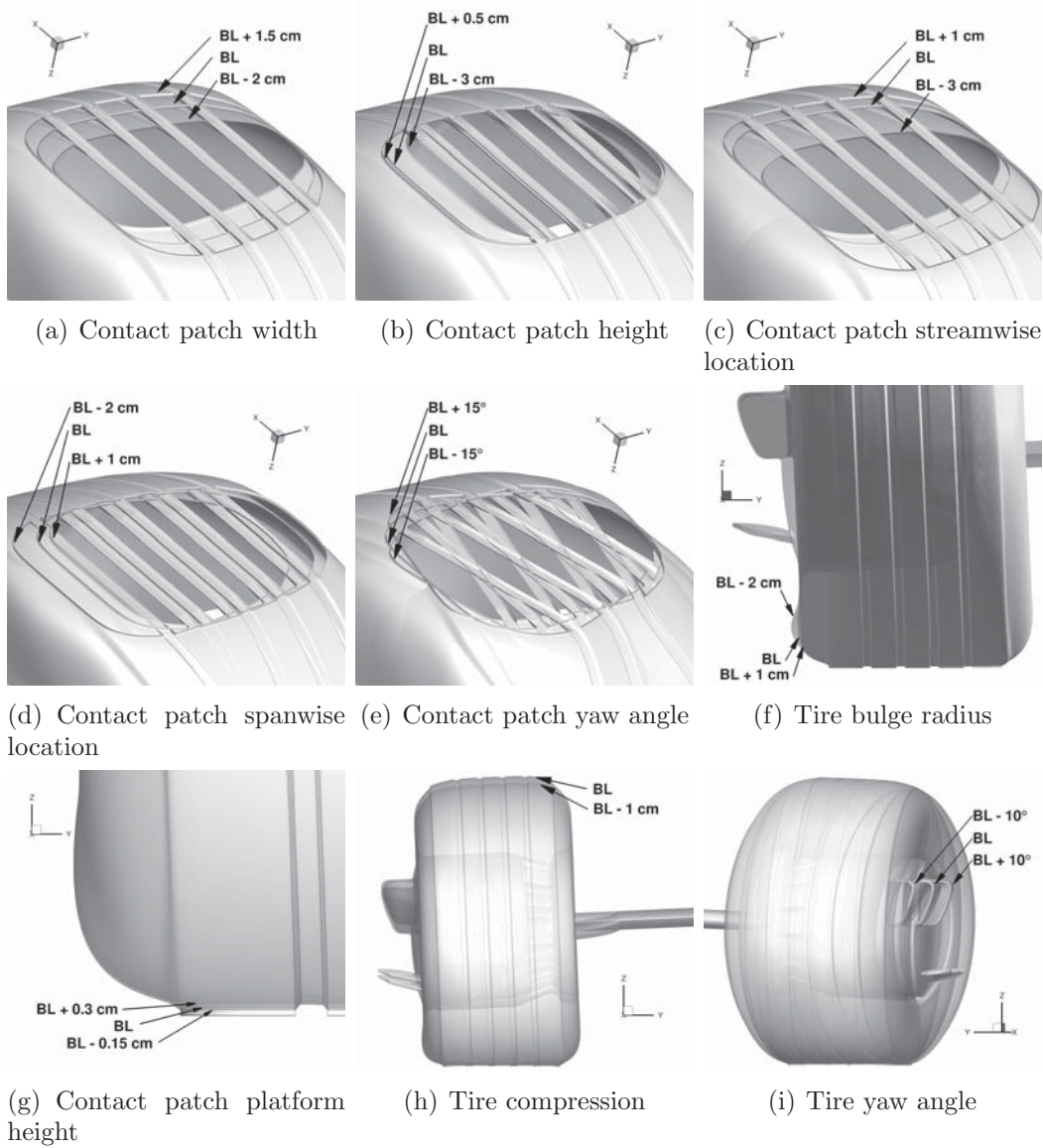(i) Tire yaw angle

Figure 6: Subset of uncertain variables tested for sensitivity in output quantities of interest

intensity, turbulent length scale) while others were based on geometric characteristics of the tire (i.e. contact patch details, tire bulge radius, camber angle). Figure 6 shows 9 geometric modifications that were performed. Each subfigure shows the minimum, baseline F1 tire geometry, and maximum deformation for each uncertain variable.

From the results of purely a one-dimensional perturbation analysis the turbulence length scale (on the order of 0m $\sim$ 2m) results in less than a 0.1% difference in both the mass flow rate through the brake duct and overall drag on the tire. Conversely, both the mass flow rate and tire drag are very sensitive to the turbulence intensity. The mass flow rate decreased by 7.8% compared to the baseline (less cooling) with 40% turbulence intensity, and the tire drag increased by 7.2% with 40% turbulence intensity. This analysis confirms that the car performance decreases with 'dirty' air compared to 'clean' air. The sensitivity of the output quantities of interest caused by the tire yaw angle is shown in the first row of Table 1. The remaining rows in Table 1 show the sensitivity of mass flow rate and drag force to geometric

Table 1: Mass flow rate into the brake duct and drag force on the tire sensitivity for 9 uncertain variables and 3 design variables

	Deformation	Mass Flow Rate Change	Drag Force Change
Tire Yaw Angle [°]	-10	1.93%	-1.03%
	10	-4.48%	6.12%
Contact Patch Width [cm]	-2	-0.65%	3.65%
	-1	-0.34%	1.87%
	1	0.39%	-2.10%
	1.5	0.72%	-4.24%
Contact Patch Height [cm]	-3	-0.28%	-1.79%
	-2	-0.21%	-1.11%
	-1	-0.12%	-0.47%
	0.5	0.08%	0.18%
Contact Patch Streamwise Location [cm]	-3	0.78%	-4.73%
	-2	0.40%	-1.92%
	1	-0.13%	0.57%
Contact Patch Spanwise Location [cm]	-2	0.22%	-1.34%
	-1	0.04%	-0.20%
	1	0.12%	-0.41%
Contact Patch Yaw Angle [°]	-15	0.66%	-6.08%
	-10	0.31%	-2.50%
	10	-0.02%	0.02%
	15	0.24%	-1.99%
Tire Bulge Radius [cm]	-2	0.08%	0.38%
	-1.5	0.06%	0.39%
	1	0.01%	-0.51%
Contact Patch Platform Height [cm]	-0.15	0.49%	-3.28%
	-0.13	0.46%	-3.12%
	-0.11	0.41%	-2.72%
	-0.09	0.35%	-2.44%
	-0.07	0.29%	-1.90%
	-0.05	0.22%	-1.31%
	-0.03	0.17%	-0.98%
	0.3	-0.21%	0.88%
Tire Compression [cm]	-1	-2.06%	-6.44%
Brake Duct Width [cm]	-1	4.14%	1.46%
	1	-13.66%	-0.43%
Brake Duct Height [cm]	-1	-5.32%	0.98%
	1	3.33%	0.12%
Brake Duct Length [cm]	-1	-1.83%	-0.11%
	1	-2.85%	0.13%

characteristics, specifically contact patch, tire bulge radius, tire compression, and brake duct dimensions.

In the end, the three most sensitive uncertain variables, namely the tire contact patch width, tire yaw angle, and turbulence intensity were selected for the optimization under uncertainty study. The tire contact patch width was able to expand and contract up to 1cm, the tire yaw angle varied between $\pm$ 3°, and the turbulence intensity varied between 0% $\sim$ 5%.

5   RESULTS

Formula 1 engineers are interested in primarily three factors related to tire aerodynamics i) overall tire lift and drag ii) cooling performance of the brakes and iii) how the tire airflow affects downstream components (wake characteristics). All three factors are tightly coupled which makes design quite complicated, especially when uncertainty in the flexible tire walls and upstream conditions can negatively effect the car performance.
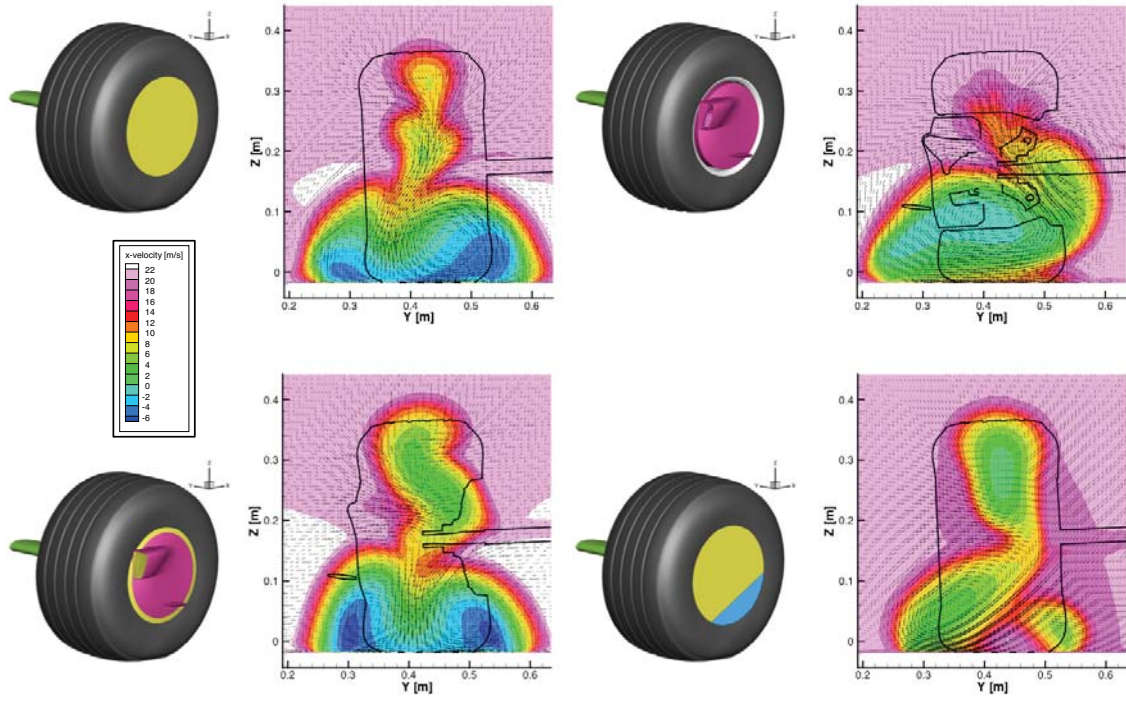
Figure 7: Wake sensitivity (shown by streamwise x-velocity contours for a plane located 1.12 wheel diameters downstream from the center of the tire) for a simplified tire with wheel fairings (top left), baseline F1 tire (top right), baseline F1 tire with blocked hub passages (bottom left), and simplified tire with artificial mass efflux from blue segment (bottom right)

Figure 7 shows the wake sensitivity caused by flow traveling through the tire hub and exiting from the outboard side of the tire. If the flow of air is not allowed to pass through the tire hub (the top left and bottom left images in Figure 7), there is no mass efflux from the outboard side of the tire and the wake is quite symmetric about the wheel centerline. The wake is dominated by a counter-rotating vortex pair and both the inboard (left) and outboard (right) vortex are of similar size. Alternatively, if the flow of air is allowed to pass through the tire hub the inboard (left) vortex becomes larger than the outboard (right) vortex causing wake asymmetry (the top right and bottom right images in Figure 7).

The results of the single parameter perturbations indicated previously show the mass flow rate through the brake duct and tire drag force are more sensitive to the brake duct width than the brake duct height or length (in the range of deformation between $\pm$ 1cm). The physical explanation of this result becomes evident when visualizing iso-contours of turbulent kinetic energy around the tire. Figure 7 shows the difference between a low width configuration (top) and high width configuration (bottom). The larger width of the brake duct causes a larger separation region immediately behind the brake duct in addition to higher turbulence levels in the shear layer immediately behind the inboard back edge of the tire.

The Pareto frontier showing the optimal brake duct designs under no uncertainty are shown in Figure 9. Ten generations, which equates to 450 simulations, were needed to eventually construct the Pareto frontier. Further details about the optimization strategy can be found in Table 2. This table reports the settings of the NSGA-II algorithm adopted to drive the main phases of the genetic algorithm: selection (e.g. mating pool, parent sorting)[6] and reproduction (e.g. crossover and
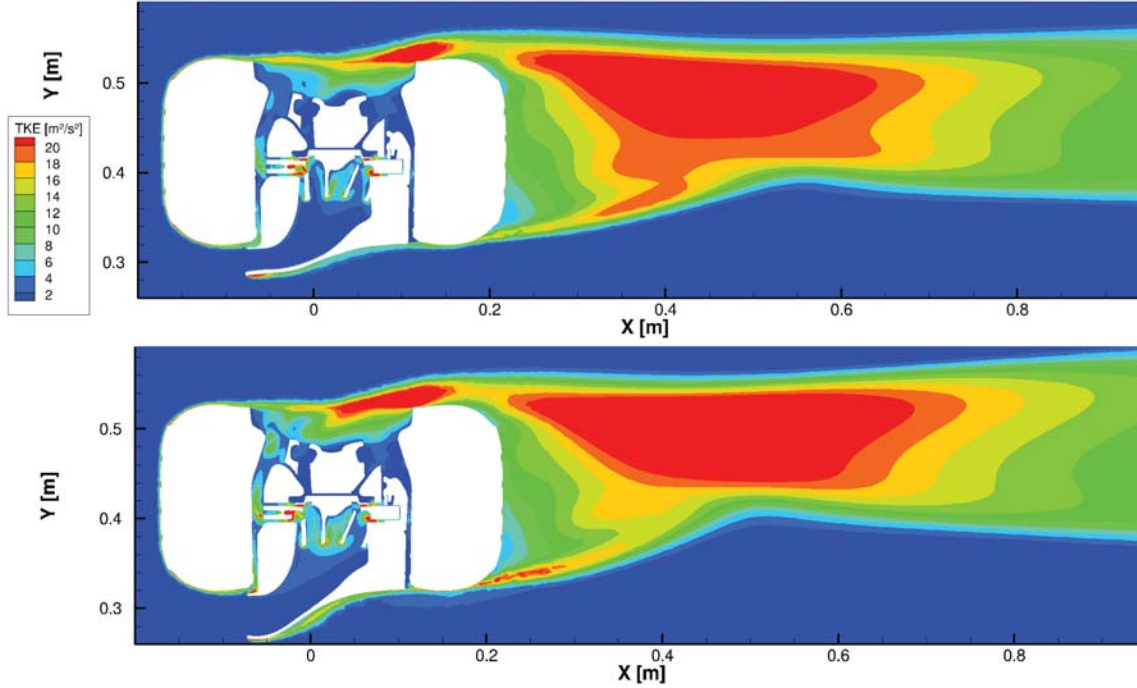
Figure 8: Turbulent kinetic energy contours for the minimum drag configuration (top) and maximum cooling configuration (bottom)

mutation)[6,7]. Leland was used to handle the job scheduling and management and as a result the time required to complete the 450 simulations was 2 days compared to about 4 days without using Leland, which requires submitting jobs manually to the job queuing system using a constant number of processors.

Table 2: Multi-objective optimization strategy

Parameter	Value
Population size [-]	50
Crossover fraction [-]	0.90
Mutation fraction [-]	0.10
Parent sorting	Tournament between couples
Mating Pool [%]	50
Crossover mode	Simulated Binary Crossover (SBX)[7]
Generations [-]	10

Among the Pareto set (see Figure 9), the design that achieves the highest mass flow rate is shown in blue and the design that achieves the lowest overall drag on the tire is shown in magenta. The green design is labeled as the trade-off design since this design tries to achieve the highest mass flow through the inlet of the brake duct while minimizing the total drag on the tire. The baseline geometry, reported in red, was shown not to be on the Pareto front in the deterministic setting.

In the previous results once the tire configuration and other input conditions are specified, the solution is uniquely determined without vagueness. On the other hand, when uncertainties are present, the results have to be expressed in a non-deterministic fashion either probabilistically or as ranges of possible outcomes. The approach we followed here using the SSC is strictly non-intrusive, in the sense that
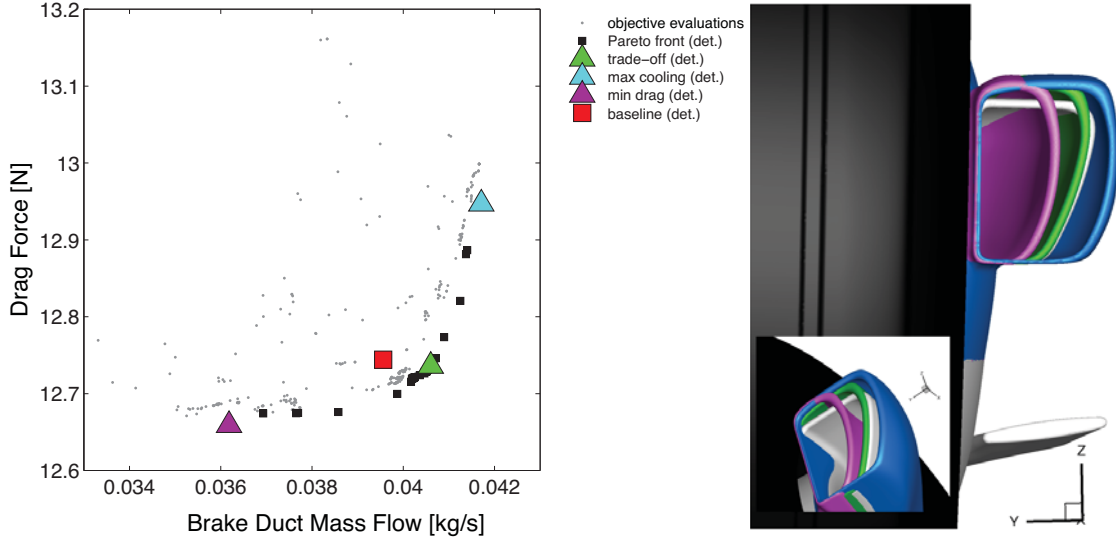
Figure 9: Deterministic Pareto front (left); the green, blue, magenta, and gray brake ducts in the subfigure on the right correspond to the trade-off, max cooling, minimum drag, and baseline configurations respectively

the existing tools are used without modifications, but the solution - or more precisely, their probability distributions - are constructed performing an ensemble of deterministic analyses. Further details about the uncertainty quantification strategy can be found in Table 3.

Table 3: Uncertainty quantification strategy

Parameter	Value
UQ algorithm[-]	SSC
Maximum number of SSC samples[-]	30
Convergence Threshold on refinement [-]	$1e^{-3}$
Number of Monte Carlo samples [-]	$1e^{3}$
Polynomial order of interpolation [-]	automatic up to 6

The variability of the four geometries described above (namely trade-off, highest mass flow, lowest drag, and baseline) as a result of the uncertainties in the the tire yaw angle, turbulence intensity, and contact patch width are shown in Figure 10. The variability of the minimum drag design is highest shown by the spread of magenta dots, followed by the maximum mass flow design shown by blue dots, trade-off design shown by green dots and baseline design shown by red dots. The colored dots in this figure represent the mean probabilistic values and the black lines represent ± 1 standard deviation of the probabilistic distribution. It is evident in this figure that the optimal designs, on average, move away from Pareto frontier, decreasing the overall performance of the race car.

A similar conclusion can be drawn by looking at the probability density of the drag force and the brake mass flow (Figure 11). The former shows a large excursion of both the position of the peak and the support, while the latter is only marginally affected. This directional sensitivity under uncertainty with respect to drag force might suggest that only the drag minimization could be treated as a probabilistic
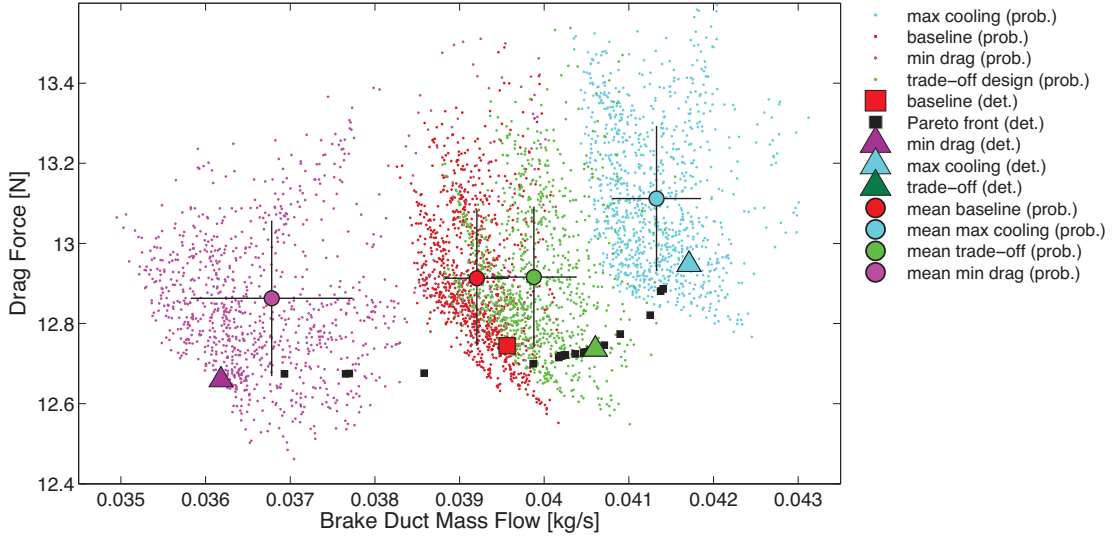
Figure 10: Pareto frontier for F1 wheel assembly showing the variability of the minimum drag (magenta), baseline (red), trade-off (green), and maximum cooling (blue) designs to uncertainty in the inflow conditions and flexible tire geometry.

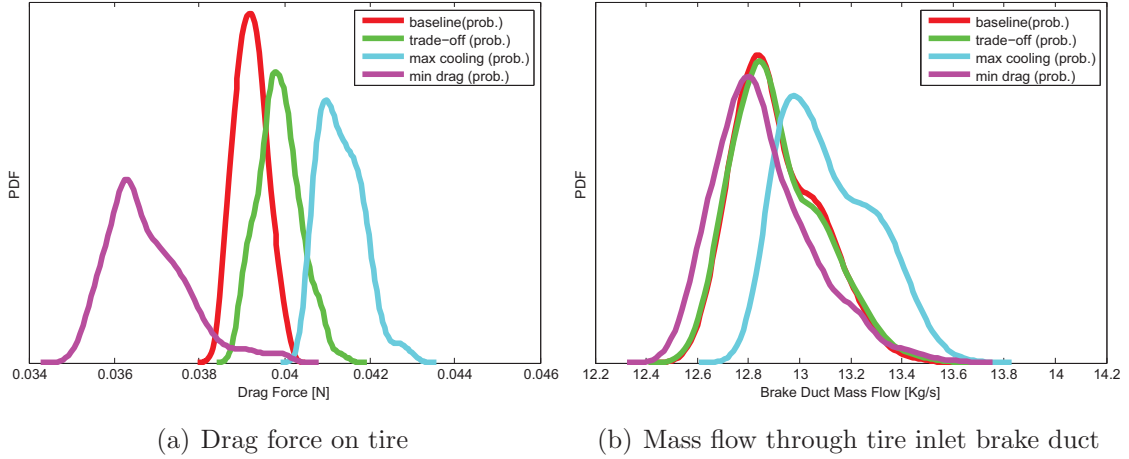(a) Drag force on tire

(b) Mass flow through tire inlet brake duct

Figure 11: PDF's of the output quantities of interest used for this study

objective, while the brake mass flow optimization can be handled using conventional (deterministic) optimization. Since the solutions identified above move away from the deterministic Pareto, the optimization process cannot be decoupled from the uncertainty quantification process. We plan to tackle the joint problem in a future study.

6   CONCLUSIONS

In this work we introduced an efficient method to perform massive ensemble calculations with application to a complex Formula 1 tire assembly optimization case. Special attention has been posed to the creation of an effective resource manager to handle the large number of computations that are required. Since the geometrical uncertainties associated with rubber tires and inflow uncertainties associated with upstream 'dirty' air have been shown to have impact on the dominating solutions, their presence have to be taken into account in the design process. The next step

of this study is to consider the presence of uncertainties invasively in the optimization procedure, generating a probabilistic Pareto front rather than analyzing the sensitivity of the deterministic Pareto due to uncertainties.

ACKNOWLEDGMENTS

REFERENCES

[1] Magoulès, F. *Fundamentals of grid computing: theory, algorithms and technologies*, volume 11 of Chapman & Hall CRC numerical analysis and scientific computing. CRC Press, (2009).

[2] Mahjoub, A., Pecero Sánchez, J., and Trystram, D. Scheduling with uncertainties on new computing platforms. *Computational Optimization and Applications* **48**, 369–398 (2011). 10.1007/s10589-009-9311-0.

[3] Witteveen, J. and Iaccarino, G. Simplex elements stochastic collocation for uncertainty propagation in robust design optimization. In *48th AIAA Aerospace Sciences Meeting, Orlando, Florida*, volume AIAA-2010-1313, (2010).

[4] Witteveen, J. and Iaccarino, G. Simplex elements stochastic collocation in higher-dimensional probability spaces. In *51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Orlando, Florida*, volume AIAA-2010-2924, (2010).

[5] Dwyer, R. Higher-dimensional voronoi diagrams in linear expected time. *Discrete and Computational Geometry* **6**(1), 343–367 (1991).

[6] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA - II. *Evolutionary Computation, IEEE Transactions on* **6**(2), 182–197 (2002).

[7] Deb, K., Sindhya, K., and Okabe, T. Self-adaptive simulated binary crossover for real-parameter optimization. In *In Proceedings of the 9th annual conference on Genetic and evolutionary computation (GECCO '07). ACM, New York, NY, USA, 1187-1194.*, (2007).

[8] www.gosculptor.com. *GoSculptor*. Optimal Solutions Software, LLC.

[9] Axerio-Cilies, J., Issakhanian, E., Jimenez, J., and Iaccarino, G. An aerodynamic investigation of an isolated stationary formula 1 wheel assembly. *Journal of Fluids Engineering* **In Press** (2011).

[10] Axerio, J., Iaccarino, G., Issakhanian, E., Lo, K., Elkins, C., and Eaton, J. Computational and experimental investigation of the flow structure and vortex

dynamics in the wake of a formula 1 tire. *Society of Automotive Engineers* .(2009-01-0775) (2009).

[11] Axerio-Cilies, J. and Iaccarino, G. Asymmetries in the wake structure of a formula 1 tire. In Proceedings of the Sixth International Symposium on Turbulence and Shear Flow Phenomena, N., K., K., E. J., R., F., A.C., H. J., V., J. A., and J., S. H., editors, volume 2, 523–528 (TSFP6 Organizing Committee, Seoul, South Korea, 2009).

[12] Issakhanian, E., Elkins, C., Lo, K., and Eaton, J. An Experimental Study of the Flow Around a Formula One Racing Car Tire. *Journal of Fluids Engineering* **132**, 071103 (2010).