SURROGATES Toolbox User's Guide
Version 3.0

Working at full speed!!!

Felipe A. C. Viana

Summer, 2011

To my wife Nádia and my daughter Bruna, whose unconditional love and encouragement have given me wings to fly higher than I ever imagined.

ACKNOWLEDGEMENTS

I am deeply grateful to Dr. Raphael Haftka for his guidance during the development of this toolbox and for his valuable comments on the documentation. I thank Dr. Tushar Goel for his collaboration until Version 1.1 (Release 2). I also thank Mr. Anirban Chaudhuri for helping improve the implementation of the efficient global optimization algorithm.

Revision history:

- July, 2011: SURROGATES Toolbox Version 3.0.
- July, 2010: SURROGATES Toolbox Version 2.1.
- March, 2009: SURROGATES Toolbox Version 2.0.
- February, 2008: SURROGATES Toolbox Version 1.1.
- February, 2007: SURROGATES Toolbox Version 1.0.

Please, cite the user's guide in any publication derived from the use of the toolbox:

FAC Viana, "SURROGATES Toolbox User's Guide," Version 3.0, 2011, available at `http://sites.google.com/site/felipeacviana/surrogatestoolbox`.

Contents

List of Tables

Chapter 1

SURROGATES toolbox

1.1 What is the SURROGATES toolbox?

The SURROGATES toolbox is a general-purpose library of multidimensional function approximation and optimization methods for MATLAB® and OCTAVE. The current version includes the capabilities shown in Table 1.1.

Table 1.1: SURROGATES toolbox capabilities.

MATLAB®/ OCTAVE capabilities[a]	Description
Experimental designs	Full factorial and variants of the Latin hypercube designs.
Surrogates	Gaussian process, kriging, polynomial response surface, radial basis neural network, and linear Shepard.
Error analysis and cross validation	Classical error analysis (coefficient of determination, root mean square error, and others), leave-one-out and k-fold cross-validation.
Surrogate-based optimization	Global sensitivity analysis, conservative surrogates (via safety margin), contour estimation, and variants of the efficient global optimization (EGO) algorithm.

[a] MATLAB® users might also have access to: central composite designs (which depends on the MATLAB® Statistics toolbox [1]), radial basis neural networks (which depends on the Neural Network toolbox [1]), and support vector regression (which depends on the SVM toolbox [2] — available in the SURROGATES toolbox).

1.2 Third party software

The SURROGATES toolbox uses the collection of third party software listed in Table 1.2.

Table 1.2: Third party software used by the SURROGATES toolbox. Please, acknowledge the individual components in any publication derived from the use of the SURROGATES toolbox. Suitable entries for the list of reference of your own work can be found in the Bibliography.

Software	Author(s)	Algorithm	Available at
GPML	Rasmussen and Williams [3]	Gaussian process	www.gaussianprocess.org/gpml
DACE	Lophaven et al. [4]	Kriging	www2.imm.dtu.dk/~hbn/dace
RBF	Jekabsons [5]	Radial basis function	www.cs.rtu.lv/jekabsons/regression
SVM	Gunn [2]	Support vector regression	www.isis.ecs.soton.ac.uk/resources/svminfo

1.3 Returning the favor

I strongly ask the user's community to give credit to the individual components and to the SURROGATES toolbox in any publication derived from the use of the toolbox. For example, when I publish my papers I usually have a paragraph like this:

> Table 1.3 details the different surrogates used during this investigation. The SURROGATES toolbox was also used for easy manipulation of the surrogates.

Table 1.3: Setup for the set of used surrogates. The GPML[3], DACE [4], MATLAB® neural networks [1], RBF [5], SURROGATES [7], and SVM [2] toolboxes were used to run the Gaussian process, kriging, radial basis neural network, radial basis function, linear Shepard algorithms, and support vector regression algorithms, respectively.

Surrogate	Details
gp	Gaussian process: Squared exponential covariance function.
krg	Kriging model: constant trend function and Gaussian correlation.
rbnn	Radial basis neural network.
rbf	Radial basis function: Multiquadric basis function.
shep	Linear Shepard model: Subroutine LSHEP from SHEPPACK [6]
svr	Support vector regression: Gaussian kernel function and $\epsilon-$insensitive loss function.

1.4 Required packages

The SURROGATES toolbox uses the collection of native MATLAB®/ OCTAVE packages listed in Table 1.4. Keep in mind that some packages may depend on functionality provided by other packages in order to function properly.

Table 1.4: Packages required by the SURROGATES toolbox.

Interpreter	Packages
MATLAB®	Statistics, Neural Network[a].
OCTAVE	Statistics[b] (which requires the "miscellaneous" package).

[a] Only for radial basis neural network models.
[b] Available at http://octave.sourceforge.net/packages.php

1.5 Mail group

If you would like to receive updates on the developments of the SURROGATES toolbox, you can join the "Felipe A. C. Viana" Google group.

It is easy! You can either:

1. Go to the "Felipe A. C. Viana" Google group and subscribe on-line.
2. Send me (felipeacviana@gmail.com) an e-mail with the following data: full name, e-mail address, and affiliation.

By doing that, you will be informed about the state of development of the toolbox, including updates on the code, research, etc.

1.6 Installation

Download the latest version from

`http://sites.google.com/site/felipeacviana/surrogatestoolbox`.

Unzip the file, open a MATLAB®/ OCTAVE terminal and go to the directory where the toolbox is; for example:

C:/users/felipe/projects/SRGTSToolbox
(there is no preference for where you will unzip it).

Next, type:

```
>> cd setup
>> srgtsInstall
```

At this point, the setup routine will help you to install the current version of the SURROGATES toolbox. The installation consists of:

1. Choosing between compiling or copying files for the GPML and SVM toolboxes (see details about third party software in section 1.2). The GPML `gpml_sq_dist` function computes a matrix of all pairwise squared distances between two sets of vectors. There are C and M versions of this function. When installing the SURROGATES toolbox, you can choose between
   (a) copying a pre-compiled file (I have tested it with MATLAB® 7.0 and OCTAVE 3.2.4), or
   (b) compiling the code for your machine (preferable, but may require you to chose a compiler — this is the default option), or
   (c) copying the MATLAB®/ OCTAVE version of this function.

The compiled versions are potentially faster than the MATLAB® / OCTAVE one.

2. The SVM `svmgunn_qp` function is a quadratic and linear programming optimizer. Unfortunately, I have succeeded compiling it only for MATLAB® (that is why SVM is not available in OCTAVE[1]). When installing the SURROGATES toolbox, you can choose between

   (a) copying a pre-compiled file (I have tested it with MATLAB® 7.0 and OCTAVE 3.2.4), or

   (b) compiling the code for your machine (preferable, but may require you to chose a compiler — this is the default option).

1.7 Uninstallation

Open a MATLAB® / OCTAVE terminal and go to the directory where the toolbox is. For example:

C:/users/felipe/projects/SRGTSToolbox

Next, type:

```
>> cd setup
>> srgtsUninstall
```

At this moment, the setup routine will help you to uninstall the current version of the SURROGATES toolbox.

1.8 Development and current version

The SURROGATES toolbox version 3.0 was developed using (meaning, it is likely to run in):

- MATLAB® Version 7.6.0 (R2008a).
- OCTAVE Version 3.2.4.

To check the installed version of the SURROGATES toolbox, open a MATLAB® / OCTAVE terminal and type:

```
>> srgtsVersion
```

[1]Help compiling it for OCTAVE would be very much appreciated. In case of success, please send an email to Felipe A. C. Viana (felipeacviana@gmail.com)

1.9 Help for functions

All in the SURROGATES toolbox start with the prefix "srgts." To get help on a specific function just type:

```
>> help functionname
```

1.10 Bug buster

If you happen to encounter a problem using the SURROGATES toolbox, feel free to report it by sending an email to Felipe A. C. Viana (felipeacviana@gmail.com).

A list of bugs and previously found problems can be found at:

http://sites.google.com/site/felipeacviana/surrogatestoolbox

(under Bug buster).

Chapter 2

List of functions

All functions in the SURROGATES toolbox start with the prefix "srgts." To get help on a specific function just type:

```
>> help functionname
```

2.1 Setup

Table 2.1: Setup functions.

Function	Details
srgtsInstall	Installs the SURROGATES toolbox in your machine.
srgtsUninstall	Uninstalls the SURROGATES toolbox of your machine.

2.2 Information about current version

Table 2.2: General information functions.

Function	Details
srgtsRoot	Returns a string that contains the directory where the SURROGATES toolbox is installed.
srgtsVersion	Displays information the current SURROGATES toolbox version.

2.3 Experimental designs and sample manipulation

Further reading about experimental designs can be found in [8–11].

Table 2.3: Experimental designs and sample manipulation functions.

Function	Details
srgtsDOEFullFactorial	Generates a mixed-level full-factorial design.
srgtsDOELHS	Generates a Latin hypercube design.
srgtsDOEMinDistCriterion	Calculates the minimum distance d_{min} criterion for a given experimental design.
srgtsDOEOLHS	Generates an optimized Latin hypercube design (using the algorithms found in [12, 13]).
srgtsDOEPHIpCriterion	Calculates the minimum distance ϕ_p criterion for a given experimental design.
srgtsDOESubSample	Selects points from a user-defined design according to a specific optimality criterion.
srgtsDOETPLHS	Generates a Latin hypercube design by using the translational propagation algorithm [14].
srgtsScaleVariable	Generates information the current SURROGATES toolbox version.

MATLAB® users will also find:

- srgtsDOECentralComposite: generates central composite design.
- srgtsDOELHSFilling: fills a user-defined design with a Latin hypercube design.

2.4 Surrogate techniques

2.4.1 Gaussian process

Further reading about Gaussian process can be found in [3, 15, 16].

Table 2.4: Gaussian process functions.

Function	Details
srgtsGPEvaluate	Predicts the response of a Gaussian process model.
srgtsGPFit	Fits the specified Gaussian process model using the GPML toolbox [3].
srgtsGPPredictionVariance	Returns the estimated prediction variance of a Gaussian process model.
srgtsGPPredictor	Returns the predicted response and the estimated prediction variance of a Gaussian process model.
srgtsGPSetOptions	Creates the SURROGATES toolbox option structure for Gaussian process models.

2.4.2 Kriging

Further reading about kriging can be found in [17–19].

Table 2.5: Kriging functions.

Function	Details
srgtsKRGEvaluate	Predicts the response of a kriging model.
srgtsKRGFit	Fits the specified kriging model using the DACE toolbox [4].
srgtsKRGPredictionVariance	Returns the estimated prediction variance of a kriging model.
srgtsKRGPredictor	Returns the predicted response and the estimated prediction variance of a kriging model.
srgtsKRGSetOptions	Creates the SURROGATES toolbox option structure for kriging models.

2.4.3 Polynomial response surface

Further reading about response surface can be found in [20, 21].

Table 2.6: Polynomial response surface functions.

Function	Details
srgtsPRSCreateGramianMatrix	Generates the Gramian matrix of a given experimental design.
srgtsPRSEvaluate	Predicts the response of a polynomial response surface model.
srgtsPRSFit	Fits the specified polynomial response surface model.
srgtsPRSMonomials	Returns the vector of monomial used in the polynomial response surface model.
srgtsPRSNumberOfCoefficients	Calculates the number of coefficients in a full polynomial response surface model.
srgtsPRSPredictionVariance	Returns the estimated prediction variance of a polynomial response surface model.
srgtsPRSPredictor	Returns the predicted response and the estimated prediction variance of a polynomial response surface model.
srgtsPRSSetOptions	Creates the SURROGATES toolbox option structure for polynomial response surface models.

2.4.4 Radial basis functions

Further reading about radial basis functions can be found in [22, 23].

Table 2.7: Radial basis functions.

Function	Details
srgtsRBFEvaluate	Predicts the response of a radial basis function model.
srgtsRBFFit	Fits the specified radial basis function model using the RBF toolbox [5].
srgtsRBFSetOptions	Creates the SURROGATES toolbox option structure for radial basis function models.

2.4.5 Radial basis neural network

Further reading about radial basis neural network can be found in [22, 23].

Table 2.8: Radial basis neural network functions.

Function	Details
srgtsRBNNEvaluate	Predicts the response of a radial basis neural networ model.
srgtsRBNNFit	Fits the specified radial basis neural networ model using the MATLAB® neural network toolbox [1].
srgtsRBNNSetOptions	Creates the SURROGATES toolbox option structure for radial basis neural networ models.

2.4.6 Linear Shepard

Further reading about the Shepard algorithm can be found in [24, 25].

Table 2.9: Linear Shepard functions.

Function[a]	Details
srgtsSHEPEvaluate	Predicts the response of a linear Shepard model.
srgtsSHEPFit	Fits the specified linear Shepard model.
srgtsSHEPSetOptions	Creates the SURROGATES toolbox option structure for linear Shepard models.

[a] M versions of the subroutines LSHEP and LSHEPVAL from SHEPPACK [6].

2.4.7 Support vector machine

Further reading about support vector machine can be found in [26, 27].

Table 2.10: Support vector machine functions.

Function	Details
srgtsSVCEvaluate	Predicts the response of a support vector classification model.
srgtsSVCFit	Fits the specified support vector classification model using the SVM toolbox [2].
srgtsSVCSetOptions	Creates the SURROGATES toolbox option structure for support vector classification models.
srgtsSVREvaluate	Predicts the response of a support vector regression model.
srgtsSVRFit	Fits the specified support vector regression model using the SVM toolbox [2].
srgtsSVRSetOptions	Creates the SURROGATES toolbox option structure for support vector regression models.

2.4.8 Weighted average surrogates

Further reading about weighted average surrogates can be found in [28–30].

Table 2.11: Weighted average surrogate functions.

Function	Details
srgtsWASComputeCMatrix	Computes the C matrix used in some weighted average surrogate schemes.
srgtsWASEvaluate	Predicts the response of a weighted average surrogate model.
srgtsWASFit	Fits the specified weighted average surrogate model.
srgtsWASPredictionVariance	Returns the estimated prediction variance of a weighted average surrogate model.
srgtsWASPredictor	Fits the specified weighted average surrogate model.
srgtsWASSetOptions	Creates the SURROGATES toolbox option structure for weighted average surrogate models.

2.5 Cross validation and error analysis

Further reading about cross validation and error analysis can be found in [29, 31–33].

Table 2.12: Cross validation and error analysis functions.

Function	Details
srgtsCrossValidation	Computes cross-validation measures for the given surrogate model.
srgtsErrorAnalysis	Calculates some metrics associated with the quality of the fit and prediction capability of the surrogate.
srgtsGetKfoldValues	Calculates the possible k-fold values for a certain number of points.
srgtsGetKfolds	Generates cross-validation sets of data following the “k-fold” strategy.

2.6 Surrogate-based optimization

Further reading about surrogate-based optimization can be found in [34–37].

Table 2.13: Surrogate-based global optimization functions.

Function	Details
srgtsEGO	Finds the minimum of a function of several variables using the efficient global optimization (EGO) algorithm by Jones et al. [38].
srgtsEGOKRGBeliever	Finds the minimum of a function of several variables using the kriging believer algorithm by Ginsbourger et al. [39].
srgtsMPPIEGO	Finds the minimum of a function of several variables using the efficient global optimization (EGO) and the multiple point probability of improvement, as detailed in Viana and Haftka [40].
srgtsMSEGO	Finds the minimum of a function of several variables using the multiple surrogate efficient global optimization (MSEGO) algorithm by Viana et al. [41].

Table 2.14: Surrogate-based contour estimation functions.

Function	Details
srgtsEGRA	Improves surrogate accuracy near the limit state using the efficient global reliability analysis (EGRA) algorithm by Bichon et al. [42].
srgtsEGRAKRGBeliever	Improves surrogate accuracy near the limit state using the kriging believer algorithm [43].
srgtsMSEGRA	Improves surrogate accuracy near the limit state using multiple surrogates [43].

Table 2.15: Infill criteria for sequential sampling functions.

Function	Details
srgtsExpectedFeasibility	Computes the expected feasibility at a point.
srgtsExpectedImprovement	Computes the expected improvement at a point given the present best solution.
srgtsMultiPointProbOfImprovement	Computes an approximated multiple point probability of improvement (neglecting correlation between the points).
srgtsProbOfImprovement	Computes the probability of improvement at a point given a target.

2.7 Conservative surrogates

Further reading about conservative surrogates can be found in [44, 45].

Table 2.16: Conservative surrogates.

Function	Details
srgtsDesignSafetyMargin	Designs the safety margin for a conservative surrogate given a target conservativeness.

2.8 Global sensitivity analysis

Further reading about global sensitivity analysis can be found in [35, 46].

Table 2.17: Global sensitivity analysis.

Function	Details
srgtsMCGlobalSensitivity	Calculates individual effects and total effects ([46]) using Monte Carlo simulations.

2.9 Heuristic optimization algorithms

Further reading about differential evolution can be found in [47, 48].

Table 2.18: Heuristic optimization functions.

Function	Details
srgtsOPTMDE	Minimizes a user-supplied function using the differential evolution algorithm of Price et al. [48].

Bibliography

[1] MathWorks contributors, *MATLAB The language of technical computing*, The MathWorks, Inc, Natick, MA, USA, version 7.0 release 14 ed., 2004.

[2] Gunn, S. R., "Support vector machines for classification and regression," Tech. rep., University of Southampton, UK, 1997, available at http://www.isis.ecs.soton.ac.uk/resources/svminfo/.

[3] Rasmussen, C. E. and Williams, C. K. I., *Gaussian Processes for Machine Learning*, The MIT Press, 2006.

[4] Lophaven, S. N., Nielsen, H. B., and Søndergaard, J., "DACE — a MATLAB kriging toolbox," Tech. Rep. IMM–TR–2002–12, Technical University of Denmark, Denmark, Aug 2002, available at http://www2.imm.dtu.dk/~hbn/dace/.

[5] Jekabsons, G., *RBF: Radial Basis Function interpolation for MATLAB/OCTAVE*, Riga Technical University, Latvia, version 1.1 ed., 2009, available at http://www.cs.rtu.lv/jekabsons/regression.html.

[6] Thacker, W. I., Zhang, J., Watson, L. T., Birch, J. B., Iyer, M. A., and Berry, M. W., "Algorithm 905: SHEPPACK: modified Shepard algorithm for interpolation of scattered multivariate data," *ACM Transactions on Mathematical Software*, Vol. 37, No. 3, 2010, pp. 1–20.

[7] Viana, F. A. C., *SURROGATES Toolbox User's Guide*, Gainesville, FL, USA, version 3.0 ed., 2011, available at http://sites.google.com/site/felipeacviana/surrogatestoolbox.

[8] Box, G. E. P., Hunter, J. S., and Hunter, W. G., *Statistics for Experimenters: an introduction to design, data analysis, and model building*, John Wiley & Sons, 1978.

[9] Simpson, T. W., Lin, D. K. J., and Chen, W., "Sampling strategies for computer experiments: design and analysis," *International Journal of Reliability and Applications*, Vol. 2, No. 3, 2001, pp. 209–240.

[10] Giunta, A. A., Wojtkiewicz, S. F., and Eldred, M. S., "Overview of modern design of experiments methods for computational simulations," *41st AIAA*

Aerospace Sciences Meeting and Exhibit, AIAA, Reno, NV, USA, Jan 2003, pp. AIAA–2003–0649.

[11] Montgomery, D. C., *Design and Analysis of Experiments*, John Wiley & Sons, 6th ed., 2004.

[12] Jin, R., Chen, W., and Sudjianto, A., "An efficient algorithm for constructing optimal design of computer experiments," *Journal of Statistical Planning and Inference*, Vol. 134, 2005, pp. 268–287.

[13] Bates, S. J., Sienz, J., and Toropov, V. V., "Formulation of the optimal Latin hypercube design of experiments using a permutation genetic algorithm," *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, AIAA, Palm Springs, CA, USA, Apr 2004, pp. AIAA–2004–2011.

[14] Viana, F. A. C., Venter, G., and Balabanov, V., "An algorithm for fast optimal Latin hypercube design of experiments," *International Journal for Numerical Methods in Engineering*, Vol. 82, No. 2, 2010, pp. 135–156.

[15] Mackay, D. J. C., "Introduction to Gaussian processes," *NATO Advanced Study Institute on Generalization in Neural Networks and Machine Learning*, NATO, Cambridge, MA, USA, 1998, pp. 133–165.

[16] Seeger, M., "Gaussian processes for machine learning," *International Journal of Neural Systems*, Vol. 14, No. 2, 2004, pp. 69–106.

[17] Stein, M. L., *Interpolation of Spatial Data: some theory for kriging*, Springer Verlag, 1999.

[18] Martin, J. D. and Simpson, T. W., "Use of kriging models to approximate deterministic computer models," *AIAA Journal*, Vol. 43, No. 4, 2005, pp. 853–863.

[19] Kleijnen, J. P. C., "Kriging metamodeling in simulation: A review," *European Journal of Operational Research*, Vol. 192, No. 3, 2009, pp. 707–716.

[20] Myers, R. H. and Montgomery, D. C., *Response Surface Methodology: process and product optimization using designed experiments*, John Wiley & Sons, 1995.

[21] Myers, R. H., *Classical and Modern Regression with Applications*, Duxbury Press, 2000.

[22] Smith, M., *Neural Networks for Statistical Modeling*, Von Nostrand Reinhold, 1993.

[23] Bishop, C. M., *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

[24] Shepard, D., "A two-dimensional interpolation function for irregularly spaced data," *23rd ACM National Conference*, ACM, New York, NY, USA, 1968, pp. 517—-523.

[25] Berry, M. W. and Minser, K. S., "Algorithm 798: high-dimensional interpolation using the modified Shepard method," *ACM Transactions on Mathematical Software*, Vol. 25, No. 3, 1999, pp. 353–366.

[26] Cristianini, N. and Shawe-Taylor, J., *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, 2000.

[27] Smola, A. J. and Schölkopf, B., "A tutorial on support vector regression," *Statistics and Computing*, Vol. 14, No. 3, 2004, pp. 199–222.

[28] Goel, T., Haftka, R. T., Shyy, W., and Queipo, N. V., "Ensemble of surrogates," *Structural and Multidisciplinary Optimization*, Vol. 32, No. 3, 2007, pp. 199–216.

[29] Viana, F. A. C., Haftka, R. T., and Steffen Jr, V., "Multiple surrogates: how cross-validation errors can help us to obtain the best predictor," *Structural and Multidisciplinary Optimization*, Vol. 39, No. 4, 2009, pp. 439–457.

[30] Acar, E., "Various approaches for constructing an ensemble of metamodels using local measures," *Structural and Multidisciplinary Optimization*, Vol. 42, No. 6, 2011, pp. 879–896.

[31] Kohavi, R., "A study of cross-validation and bootstrap for accuracy estimation and model selection," *Fourteenth International Joint Conference on Artificial Intelligence*, IJCAII, Montreal, QC, CA, Aug 1995, pp. 1137–1143.

[32] Jin, R., Chen, W., and Simpson, T. W., "Comparative studies of metamodelling techniques under multiple modelling criteria," *Structural and Multidisciplinary Optimization*, Vol. 23, No. 1, 2001, pp. 1–13.

[33] Meckesheimer, M., Booker, A. J., Barton, R. R., and Simpson, T. W., "Computationally inexpensive metamodel assessment strategies," *AIAA Journal*, Vol. 40, No. 10, 2002, pp. 2053–2060.

[34] Jones, D. R., "A taxonomy of global optimization methods based on response surfaces," *Journal of Global Optimization*, Vol. 21, No. 4, 2001, pp. 345–383.

[35] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P. K., "Surrogate-based analysis and optimization," *Progress in Aerospace Sciences*, Vol. 41, No. 1, 2005, pp. 1–28.

[36] Forrester, A. I. J., Sóbester, A., and Keane, A. J., *Engineering Design via Surrogate Modelling: a practical guide*, John Wiley & Sons, 2008.

[37] Forrester, A. I. J. and Keane, A. J., "Recent advances in surrogate-based optimization," *Progress in Aerospace Sciences*, Vol. 45, No. 1-3, 2009, pp. 50–79.

[38] Jones, D. R., Schonlau, M., and Welch, W. J., "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, Vol. 13, No. 4, 1998, pp. 455–492.

[39] Ginsbourger, D., Le Riche, R., and Carraro, L., "Kriging is well-suited to parallelize optimization," *Computational Intelligence in Expensive Optimization Problems*, edited by Y. Tenne and C. Goh, Vol. 2, Springer Berlin / Heidelberg, 2010, pp. 131–162.

[40] Viana, F. A. C. and Haftka, R. T., "Surrogate-based optimization with parallel simulations using the probability of improvement," *13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA, Fort Worth, TX, USA, Sep 2010, pp. AIAA–2010–9392.

[41] Viana, F. A. C., Haftka, R. T., and Watson, L. T., "Why not run the efficient global optimization algorithm with multiple surrogates?" *51th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, AIAA, Orlando, FL, USA, Apr 2010, pp. AIAA–2010–3090.

[42] Bichon, B. J., Eldred, M. S., Swiler, L. P., Mahadevan, S., and McFarland, J., "Efficient global reliability analysis for nonlinear implicit performance functions," *AIAA Journal*, Vol. 46, No. 10, 2008, pp. 2459–2468.

[43] Viana, F. A. C., Haftka, R. T., and Watson, L. T., "Sequential sampling for contour estimation with concurrent function evaluations," *Structural and Multidisciplinary Optimization*, Vol. under review, 2011.

[44] Picheny, V., Kim, N., Haftka, R. T., and Queipo, N. V., "Conservative predictions using surrogate modeling," *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Schaumburg*, AIAA, Schaumburg, IL, USA, Apr 2008, pp. AIAA–2008–1716.

[45] Viana, F. A. C., Picheny, V., and Haftka, R. T., "Using cross validation to design conservative surrogates," *AIAA Journal*, Vol. 48, No. 10, 2010, pp. 2286–2298.

[46] Sobol, I. M., "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and Computers in Simulation*, Vol. 55, No. 1–3, 2001, pp. 271–280.

[47] Storn, R. and Price, K., "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, Vol. 11, No. 4, 1997, pp. 341–359.

[48] Price, K. V., Storn, R. M., and Lampinen, J. A., *Differential Evolution: a practical approach to global optimization*, Springer Verlag, 2005.