MODEL-BASED SYSTEM ENGINEERING COMPONENTS OF ANTENNAS

L. Foged [(1)], B. Bencivenga [(1)], A. Potenza [(1)] , M. Bandinelli [(2)] , F. Milani [(2)], S. B. Sørensen [(3)],
H.B. Nielsen [(4)], Oscar Borries [(4)], M. Sabbadini [(5)]

[(1)] SATIMO Italy, Via Dei Castelli Romani 59, I-00040 Pomezia, Italy,
Email: lfoged@satimo.com, bbencivenga@satimo.com, apotenza@satimo.com

[(2)] IDS, Via Flaminia 1068, 00189, Roma, Italy, Email: f.milani@ids-spa.it, m.bandinelli@ids-spa.it

[(3)] TICRA, Læderstræde 34, 1201 Copenhagen, Denmark, Email:sbs@ticra.com

[(4)] DTU Informatics, Build 321, 2800 Lyngby, Denmark, Email:hbn@imm.dtu.dk

[(5)] ESA/ESTEC, P.O. Box 299, AG Noordwijk ZH, The Netherlands. Email: Marco.Sabbadini@esa.int

ABSTRACT

Very sophisticated antenna models are available today for the accurate prediction of antenna performances, in particular those included in the European Antenna Modeling Library in which GRASP, ADF and SATSim are key components [1, 2]. However these models are not directly suitable for Phase 0/A and early Phase B activities in a Model-Based System Engineering context where fast and reliable predictions, with a known uncertainty margin, are required in order to avoid over- or under specification and reduce the risk of under or overdesign in later phases.

The Model-Based System Engineering tool (MBSE) is a proof-of-concept demonstrator suitable for the quick assessment of antenna performances in pre-phase studies with applications in the area of Earth Observation, Science and Exploration.

1. INTRODUCTION

Rather sophisticated antenna models are available for accurate prediction of antenna performances, in particular those included in the European Antenna Modelling Library (RD[1]). However these models are not suitable for Phase 0/A and early Phase B activities (up to SRR) in a Model-Based System Engineering context, where fast and reliable predictions, with a known uncertainty margin, are required in order to avoid over- or under specification and reduce the risk of under or over-design in later phases.

Existing antenna models usually require the complete and detailed definition of antenna configurations, which are typically unavailable at this stage of the development programme. Furthermore they usually are either rather time-consuming to use or computationally intensive, thus not compatible with fast trade-off cycles. At the same time, the very simple models based on first principles traditionally used for system-level studies are too coarse to provide satisfactory input for antenna-level assessments. The Model-Based System Engineering tool (MBSE) is a proof-of-concept demonstrator for a set of high-level antenna models suitable for the quick assessment of antenna performances in pre-phase studies. The system includes both High Level Models (HLM) and Low Level Models (LLM), equivalent to Fast Tools (SATSim, GRASP and ADF). As end-user the ESA's Concurrent Design Facility (CDF) has been specifically targeted but the demonstrator maintains full flexibility in the architecture and openness towards future extension, also for other end-users.

Within MBSE the user performs both online and offline tasks. An online task is performed when the user is involved in the definition of an antenna solution within a concurrent engineering facility. The time to respond is an important parameter so all functions are optimized for this use. The offline tasks are more time consuming activities like definitions of new modules, creation of new databases, further in-depth investigation of a specific design solution by the EAML tools provided with the system. The EAML tools provided by the system are derived as accelerated version of the proprietary tools Grasp (TICRA), SatSim (SATIMO), ADF (IDS). The tools have been optimized for speed and applicability for the MBSE system.
This paper is an introduction to the philosophy and architecture of the MBSE tool and reports on the ongoing and future developments.

2. MODELS DESIGN

The system includes both High Level Models (HLM) and Low Level Models (LLM), equivalent to Fast Tools (SATSim, GRASP and ADF). The former are sizing and performance models based on formulas and algorithms, while the latter are used in combination with the extractor algorithm (common to all the types of antennas) and are useful when an accurate pattern shape in different scenarios is needed, in order to calculate some parameters.

2.1. High Level Models

Two types of High Level Models are available: Sizing Models and Performance Models. The sizing models are basic models, based on simple formulas, allowing the derivation of initial guesses for the antenna dimensioning and main performances (gain, beamwidth), given one or more of the same parameters. Such models require the minimum possible amount of information.

The performance models are refined models allowing the synthesis of more complete antenna configurations, from several possibly conflicting requirements and producing complete performance data based on accurate predictive models. The performance models are predictive, i.e. actually based on a mathematical representation of the physics phenomena; they address individual antenna components and are structured in such a way to allow their combined use to assemble different, possibly new, antenna configurations. The high-level models cover different antenna classes: small antennas (patch, horn, helix antennas), array and reflector antennas. All high level models have been tested and validated against accurate numerical models and error bounds have been derived. In this way the user is informed about the accuracy of the results.

Models can belong to one of the following types:

- Direct
- Partial
- Invertible
- Numerical

Direct: the model consists of either simple formula, algorithm which can be directly accessed since the I/O parameters set is unambiguously defined.

Partial: the model is a collection of formulas acting on the same parameters set, but each of them as its own I/O parameters list, denoted as configuration. Each configuration somehow represents a direct model. When a meta-model is involved in the computation, the system can address each specific model by selecting the relevant parameters configuration.

Invertible: the model consists of either a simple formula, algorithm for with any parameters can play the role of output parameter.

Numerical: the model consists of some interpolation coefficients strictly related to the used interpolation technique. When the system accesses such models, it has to call the proper interpolating function. A special mechanism is provided by which a Direct model can be inverted with respect to one of the input parameters. The inversion is performed by a numerical search procedure and the resulting inverted function is represented as a Numerical model based on kriging interpolation, described in Section 4.

2.2. Low Level Models

Three different Low Level Models are integrated into MBSEA: SATSim, GRASP and ADF.

A. The Astigmatic beam Tracer SATSim is used to evaluate the antenna performances (E, H fields) on freespace and on realistic design scenario. SATSim is able to use both analytic source and simulated/measured source antenna. The system has a data repository containing a set of primitive geometries (cube, parallelepiped, plate, cylinder) to be used to create to be used to create scenes of satellite models. The approach based on the usage of SATSim to evaluate antenna performances in real scenario, perfectly meets the MBSE system specification in terms of speed and output generation.
B. A PO (Physical Optics) tool GRASP-S (GRASP Subset) is also included in the MBSE-A tool. The accuracy of the output is the same as GRASP, but the number of options concerning geometry and feed models is restricted (Gaussian beams and Hybrid Mode feed). Automatic convergence test is implemented to ensure accurate and fast evaluation of the PO integral. The program is able to analyze single reflector parabolic reflectors and also, if needed, dual Cassegrain / Gregorian antennas. Like the High Level Models, the input to the Low Level Models is given by very few parameters to make them easy to use. As an example an offset single reflector antenna can be specified by just three geometrical parameters (see Fig. 2-1) and two electrical parameters, i.e. the feed edge taper and the frequency.

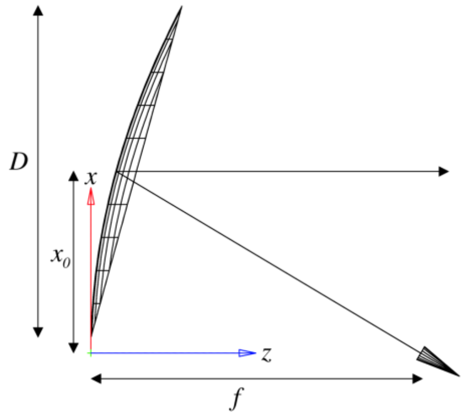

Fig. 2-1 – Single reflector offset geometry defined by three parameters: D=Diameter, f=Focal length, x_0=Offset distance.

Using such basic parameters it is possible to compute a radiation pattern from which a large number of electromagnetic performance parameters can be extracted, e.g. beamwidth, cross-polarization level, side-lobe level, efficiency, etc.

C. An electromagnetic simulator environment as Antenna Design Framework – Electromagnetic Satellite (ADF) integrates a code 3DAMxLAD that is able to perform full-wave analysis, based on MoM methods, of any kind of direct radiating arrays and phased arrays. In order to reduce the response time, a number of approximate closed-form or simplified numerical methods are used:

1. Closed form expressions,
2. Synthetic methods for array performances calculation

The approach based on the usage of analytical formulation to evaluate array performances, perfectly meets the MBSE system specification in terms of speed and output generation.

3. SOFTWARE ARCHITECTURAL DESIGN

Like any interactive system, the MBSEA system is composed of two main functional items:

- **The user interface**, which provides the means to exchange data with the system (models and parameters selection).

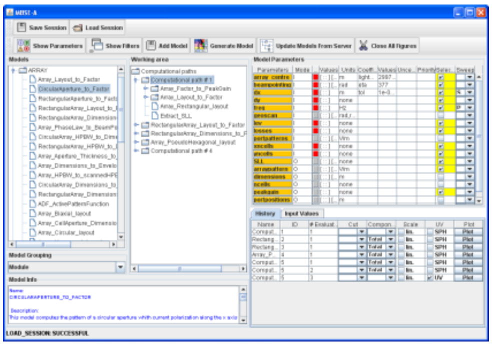

Fig. 3-1 – MBSEA user interface

- **The system kernel**, which consists of the system management and computational core.

Briefly, the system is composed of:

Interface & Data Displayer: it represents the outer layer of the system. It also provides the plotting functionalities supported by the MatLab environment.

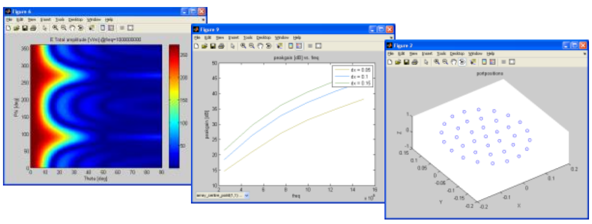

Fig. 3-2 – MBSEA typical outputs: field pattern, performances vs. geometry, antenna layout

Interface to CDF: provides a bridge module for the Excel-based data exchange between the MBSEA system and the CDF.

System kernel: it comprises the following items:

- **Model Manager**: it provides the system with the means to properly address and evaluate each model (model repository access). Whenever required, it determines the computational path starting form a list of models and required I/O parameters.
- **Data Manager**: it checks the input data against the model and parameters constraints. It collects and manages the parametric analyses data and manages all models data.
- **Model Generator**: it accounts for creating new numerical MBSEA models based on external data (measurements, external tools output data). <u>The created numerical models are stored in the model repository and can be used for future evaluations.</u>
- **Uncertainty estimator**: it is responsible for evaluating the uncertainty of the combined models.

Data repository, which comprises a:

- **Stable data storage area**: for models (Models repository), external antenna data (Antenna data repository for data coming from outside the MBSEA environment) and saved working session data (Sessions Data storage). <u>User can at any time update and add new models with no need to recompile the system.</u>
- **Temporary storage area**: for data produced during a work session. It is mainly represented by the central memory.

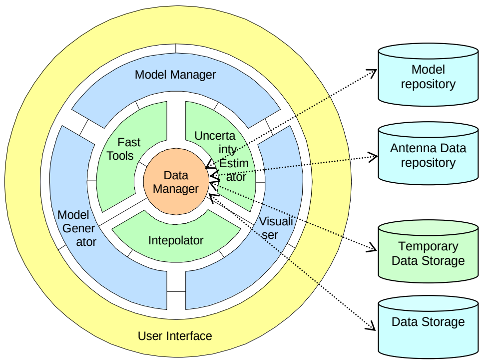

Fig. 3-3 – High Level architecture

The GRASP-S (GRASP subset) is programmed in Fortran and linked as a dynamic link library. In the MBSE-A system this library is loaded into the MatLab system from which the Fortran functions can be called directly by MatLab functions.

4. KRIGING METHOD

Kriging is a statistically based method of interpolating datasets, and is also known as Gaussian Process estimation. Originally conceived by Danie Krige in 1951 [3], its use in modelling computationally expensive functions is due to Sacks et. al. [4].

The kriging predictor is based on a two-part model, typically dubbed the "regression" and "correlation" models respectively. The regression is a standard linear least squares regression, thus allowing for any linear model that can be used in this context. With this fit, the correlation model can then be considered as a local adjustment, causing the predictor to interpolate the samples.

Using $\mathbf{f(x)}$ and $\boldsymbol{\varphi}\mathbf{(x)}$ to represent the realization of the regression and correlation models respectively at a single, multi-dimensional point $\mathbf{x}$, the predictor can be expressed as

$$y=\mathbf{f(x)}^T \boldsymbol{\beta} + \boldsymbol{\varphi}\mathbf{(x)}^T \boldsymbol{\gamma}$$

where $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are vectors determined during the calculation of the kriging model. Details on the calculation of the model are found in [5], while details on the derivation of the Kriging predictor can be found in [6,7].

Kriging offers a number of significant advantages compared to traditional interpolation methods such as splines or other piecewise polynomial methods:

- Kriging scales well with the dimensionality of the design space - in contrast to splines, where more than two dimensions become unmanageable.
- The predictor is based on any scattering of points, and thus works with both irregularly and unevenly distributed points. Consequently, there are no special concerns, contrary to e.g. the condition problems seen when applying splines to uneven point scatterings.
- The end-user need not have special knowledge of the method, since the user input requires very little specialization to the current dataset. The regression model is typically chosen as a low order polynomial model, and the correlation model is almost always Gaussian. Thus, the user simply supplies a dataset, and possibly the desired order of the regression model. Furthermore, the model is quite robust, meaning that as long as the input is syntactically correct, a reasonable model will almost always be the result.
- Finally, due to its solid foundation in statistics and stochastic processes, a statistical estimate of the expected error can be computed, providing a measure of quality of the predictor as a function of the design space. This allows subspaces where the predictor is poor to be identified automatically, and consequently improved by acquiring additional samples.

Despite this rather long list of advantages, kriging suffers from a few disadvantages as well. Most notably, it is more computationally intensive. In particular, the computations involved in creating of the model far exceeds that of piecewise polynomial methods. For more details, timings can be found in [8, page 25]. Furthermore, the theory behind the kriging framework is somewhat complicated, particularly due to it relying on knowledge of both statistics, stochastic processes and numerical analysis. Thus, for those wishing to dive deeper into the theory behind kriging, the initial process of understanding the concepts can be overwhelming - as mentioned, however, merely using kriging to achieve a model of some dataset fortunately requires very little knowledge of the method.

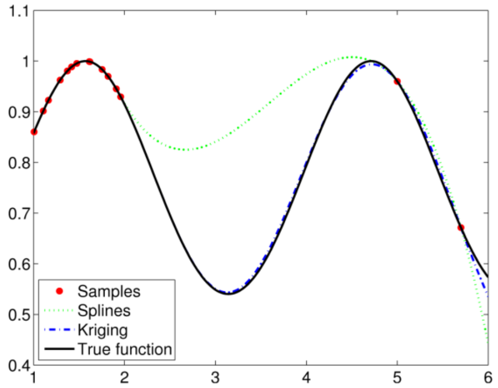

Fig. 4-1 – Comparison between splines and kriging for interpolating non-equidistant points on a sine curve.

4.1. Applications in MBSE

For the MBSE project, the advantages offered by kriging suits the requirements very well. Of special importance is the ability of kriging to handle data points in more than two dimensions and to interpolate irregularly and unevenly distributed points. It is also important that kriging can identify the regions in the interpolation space in which more data points are needed in order to obtain a specified accuracy.

The implementation is based on the DACE software package [9], written in MatLab. Several improvements to the package were made during the MBSE project, primarily to increase the computational speed and stability of the package to allow its use as a subroutine rather than a stand-alone package. Functions that efficiently expand the model, exploiting the structure of the matrices to avoid recalculating factorizations were also developed. Several of these functions will be released into the public domain at a later time as part of the DACE 3.0 package.

The *automodel* function is the primary interface between DACE and the rest of the MBSE. It can be used if a model in the MBSE system (e.g. antenna gain as a function of a number of geometrical parameters) has to be represented numerically. In the first call of *automodel* an initial set of data values of the MBSE model is supplied, from which *automodel* calculates an initial kriging model and also identifies the areas in the interpolation space where more data are needed.

Hereafter *automodel* requests more data from the given MBSE model and calculates a new improved kriging representation. The process is repeated until the requested accuracy is obtained. In this way an accurate kriging representation of the given MBSE model is created using relatively few data points.

5. CONCLUSION

The MBSE-A system, latest addition to the European Antenna Modelling Library, enables antenna engineer to quickly answer the requests of system engineers in the early phases of mission developments (phase 0/A) while relying on consolidated and robust models. It also offers a systematic approach to the mapping of high-level requirement flowing down from system design to antenna requirements. The availability of fast version of the same tools allows both a smooth transition from back-of-an-envelope antenna sizing to detailed analyses performed with popular space antenna design tools. The system is ideal for use win in a concurrent engineering context and its structure paves the way for the extension to cover the need of highly concurrent preliminary design activities (phase B).

6. REFERENCES

1. M.Sabbadini et al, “European Antenna Modelling Library: an open platform for space antenna engineering”, 30th ESA Antenna Workshop, May 2008
2. P.E.Frandsen, M.Sabbadini, “An introduction to the Electromagnetic Data Exchange language”, March 2008
3. D. G. Krige, "A statistical approach to some basic mine valuation problems on the Witwatersrand", Journal of the Chemical, Metallurgical and Mining Engineering Society of South Africa, pp. 119-139, 1951.
4. J. Sacks et. al., "Design and analysis of computer experiments", Stat. Sci., pp. 409-435, edition 4, 1989.
5. Nielsen et. al., "Aspects of the MATLAB toolbox DACE", Technical Report IMM-REP-2002-13, IMM, DTU, 2002, available from http://www.imm.dtu.dk/~hbn/publ/TR0213.ps
6. Forrester et. al., "Engineering Design via Surrogate Modelling", Wiley and Sons, 2007, chapter 3.
7. Santner et. al., "The Design and Analysis of Computer Experiments", Springer, 2008.
8. O. Borries, "Surrogate Modelling using DACE", Report, IMM-B.Sc.-2010-08, IMM, DTU. Available from the Orbit database, http://tiny.cc/2xeaa.
9. DACE, a MATLAB Kriging software package. Available online from www.imm.dtu.dk/~hbn/dace/.