A Process for Evaluating Parametric Models for Mechanical Systems Simulation : the Case of a Sailboat

Emilien Lavigne, *MerConcept, emilienlavigne@merconcept.com*
Benoît Piquemal, *MerConcept, benoitpiquemal@merconcept.com*
Adeline Bourdon *Université de Lyon, CNRS INSA-Lyon, LaMCoS UMR 5259 F-69621 Villeurbanne France, adeline.bourdon@insa-lyon.fr*
Simon Chesne, *Université de Lyon, CNRS INSA-Lyon, LaMCoS UMR 5259 F-69621 Villeurbanne France, simon.chesne@insa-lyon.fr*
Goulven Guillou, *Lab-STICC/UBO, goulven.guillou@univ-brest.fr*
Jean-Philippe Babau, *Lab-STICC/UBO, babau@univ-brest.fr*

Introduction

Due to the emergence of modern sensors and embedded computation capacity, more and more Cyber Physical Systems (CPS) are developed. Into a CPS, the controller (the Cyber part) has to consider various situations (imposed by the Physical System and its environment). The CPS development requires large simulations in order to ensure safety and efficiency of the controlled physical system. Realism is a key challenge to make the simulation pertinent. Realistic simulation requires dedicated models to emulate different behaviors of the controlled system and its environment.

The targeted application domain of this paper is racing sailboat. A racing sailboat is a complex mechanical prototype, constantly evolving. So, defining realistic physical models is complicated, and sometimes impossible. Moreover the behavior of a sailboat depends on the environmental conditions strongly. But because of the intrinsic unpredictability of the marine environment, the environmental conditions are also difficult to evaluate and characterize, and so the behavior of the sailboat remains unknown.

In the other hand, due to an intensive usage of sensors, it is possible to get a large amount of data describing the behavior of the system. So, for the targeted domain, an interesting alternative to model the physical system is the use of parametric models. A parametric model is able to mimic an observed behavior of a system by analyzing measured input and output of the system.

In this paper, we propose to study parametric models to simulate a racing sailboat. We first investigate the choice of an adapted parametric model (autoregressive or state-based). Then we discuss the impact of the data reference (absolute or relative) and of the time discretization. What is the influence of the number of measured data on the parametric model accuracy?

The following section gives the main principles of parametric models and the main sailboat characteristics. Then the approach is presented before to be evaluated on data measured on a racing sailboat. The last section concludes the paper and gives some perspectives.

Background and Case Study

Parametric models

A parametric model is a mathematical model which describes the behavior of a system by using a finite number of parameters. Such a model is able to compute one or more outputs, by considering one or more inputs. In this paper, we use two families of MIMO models [1]: the state-space models and the autoregressive models. The first one is based on a set of state variables, contained in a state vector which represents the system. In case of linear systems, the behavior can be modeled by four matrices (input, output, state and feedthrough matrices), characterized by a set of coefficients (the parameters of the model). The second family is based on a set of transfer-functions, a function describing the contribution of one input on one output. The parameters of the model are the coefficients of the transfer-functions. In both cases, models have an order: it corresponds to the order of the transfer function for the autoregressive approach, and to the length of the state vector for the state-space approach.

The identification of parameters requires a set of input and output data, measured on the real system: the parametric model is computed to mimic the observed behavior of the system. The parameters are determined by minimizing the gap between the measured outputs and those obtained by simulating the parametric model.

Sailboat

In this paper, we apply parametric model to sailboat modeling. When sails are trimmed, a sailboat reacts to its environment (wind and sea) and to a direction controller. The control law aims to regulate the course of the sailboat, in order to steer it along a fixed heading. In this case, the command delivered by the controller is the rudder angle. For environment, in this work, as a first approach, we assume that only the wind has an influence on the sailboat. Sea is supposed flat with limited waves and without current. Depending on the inputs (wind and rudder angle), the behavior of the sailboat determines the outputs: here the speed, the yaw (the direction), and then the position and the trajectory of the sailboat.

These different inputs and outputs can be obtained thanks to different sensors embedded on the sailboat, here a racing multihull sailboat.

Among the input data, the wind can be described by apparent or true wind. The apparent wind is the wind felt by an observer on the sailboat and is a relative wind. The true wind is the wind felt by an immobile observer. Apparent and true wind are usually described through a polar notation which gives the speed (AWS and TWS respectively for Apparent Wind Speed and True Wind Speed) and the angle (AWA and TWA respectively for Apparent Wind Angle and True Wind Angle) of the wind relative to the bow axis of the sailboat. However a Cartesian conversion is always possible and very useful in some cases.

For outputs, a fast GPS (10Hz) provides the geographical position and the Speed Over Ground (SOG) (a classical speedometer does not work on a racing multihull as the hulls are regularly above the sea surface) and the Course Over Ground (COG). These data also allows evaluating the True Wind Direction (TWD) which is a geographical direction.

Apparent wind data is measured at the masthead of the sailboat. And true wind is obtained by subtracting the sailboat speed vector (SOG) to the apparent wind.

Controlled process modeling

In this section, we present how to model a controlled system by reusing real input and output data.

Global system

First, we present the global model of the system. As usual in control, it is divided in three main parts (colored rectangles surrounded by dotted rectangles in fig. 1): the controlled system, the controller and the environment. The environment is connected with both the controlled system and the controller: the environment impacts the behavior of the system and the controller uses some environment measures to perform adequate commands. At the end, the controller acts on the system to apply specific commands.

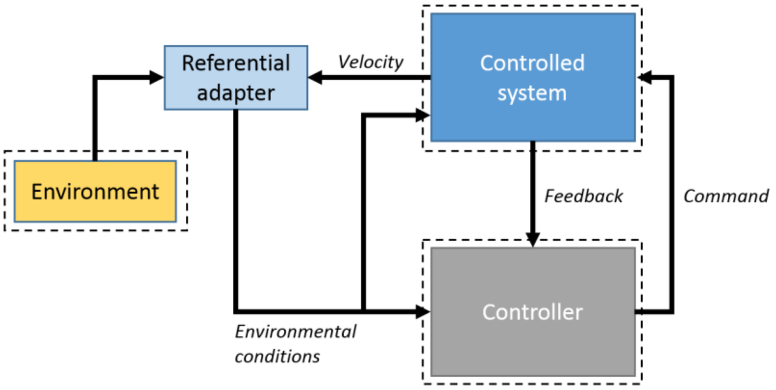

Figure 1 – A generic decomposition of control system

Control may be based on an absolute reference (point of view of an immobile observer compared to the ground), or on a relative reference (the referential of the system itself). In the latter, environmental conditions are viewed relatively to the system, whereas in the first case, there are described in the absolute referential. Consequently, we add a “Referential adapter” to adapt input data to the chosen reference.

To evaluate properties of the control (safety, performance …), it is fundamental to simulate each part of the system by proposing adequate models. The choice of models for each part of the system is a key challenge of CPS simulation.

In the targeted domain, the environment is disturbed, uncertain and unpredictable. A model of environment is then difficult to explicit. For example, up to now, high frequency (more than 1 Hz) wind model are not realistic enough. Most existing models are climatic models or models related to energy production [2, 3].

The controller is also difficult to model. In the targeted domain, autopilots are based on industrial and confidential policies. And in case of human control, especially for racing sailboats, it is difficult to exhibit the control law followed by the skipper, often acting instinctively [4].

It is also complicated to simulate the controlled system dynamic which depends on many parameters, often unknown for racing prototypes (the domain of the paper). The intensive calculus (like finite elements or CFD simulation) gives a first approximation. In real, the equations of the motion do not accept trivial or analytic solutions.

Because *a priori* models are difficult to obtain, the main idea of this work is to simulate the dynamic of the controlled system by using parametric models, instead of solving the equations of motion. Once the parametric model is determined, its main interest is to allow predicting the behavior of the controlled system with very light calculus.

As noted above, it is necessary to have datalog to determine the coefficients of the parametric model, and the data must be sufficiently representative to contain the system’s entire dynamic.

Controlled system modeling

In this study, we consider only linear parametric models, which are easier to determine and faster to calculate. However, in many physical systems, the response to output data is not linear, but rather quadratic or polynomial. To overpass those nonlinearities, it is possible to introduce additional inputs, with derived data as their squares or their cubes.

Two families of parametric models are available to simulate MIMO systems. At first sight it is difficult to determine which one is the best: an autoregressive model is relevant to obtain each contribution of each input for one given output, and a state-space model is more accurate if those contributions are coupled. Moreover the system can be simulated with two different references (absolute or relative). Physically, the two methods are strictly equivalent (particularly if the system is a linear system), but the accuracy of a parametric model, which involves numerical computation, might change with those different approaches.

If we combine the choice of the model's family and the choice of the reference, we get four possibilities that give four potential models for the controlled system. An hybrid solution may be also considered, where some outputs are computed with a kind of model and the others with another. Moreover, if the outputs of the first model are used as the inputs of the second, it leads to a large number of possibilities. At the end, there is a necessity to find a criterion to evaluate the performance of each model. A classical criterion is the norm between the simulated and measured data, for a given output. It estimates the gap between the real behavior of the system and the one predicted by the model with the same input data.

To compare different model candidates, it is important to determine a representative set of data (inputs and outputs) which allows characterizing the behavior of the system in different conditions. The classical approach in identification theory [5] is to split the set of data in two parts: the first one is used to compute the coefficients of the model, and the second one is used to evaluate the performance of the model, by comparing the simulated and the measured outputs. However, in the case of uncertain environment, during a given time period, the conditions can evolve and the computed model, which is valid for a certain period, is wrong elsewhere. To avoid this problem, instead of splitting the data in two equal parts, it is interesting to consider different periods of time for which the behavior of the system is similar for computation and validation. However, the duration of these periods must be sufficient for capturing all the dynamic of the system.

Sailboat modeling

For a sailboat control, the global architecture of the system is given by figure 2. The system is the sailboat itself, and the controller is the autopilot. As the sea is here considered as neutral, only wind provides environmental conditions, and the referential adaptor is a simple vector calculus to get the apparent wind (AWS and AWA) from the real wind (TWS and TWD) and the sailboat velocity (SOG and COG).

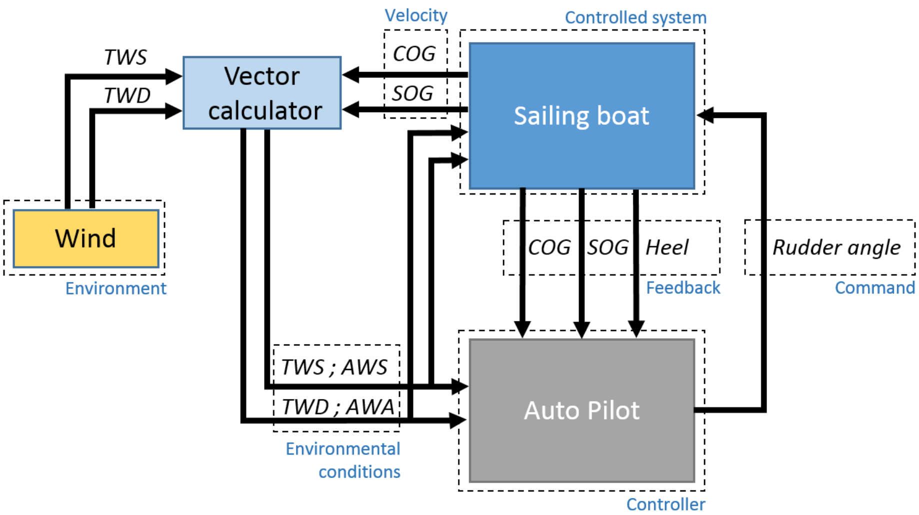

Figure 2 – The architecture of the sailboat control system

For experiments, we have chosen the Matlab language and toolbox [6]. Thanks to the Matlab identification toolbox, parameters of a model can be computed from a given data set. And the toolbox proposes an option to select the auto-computation of the best model order for a given situation.

Inputs and Outputs

The data set used for this work is a time series of 5000 points. The sample time is 1 second. The set corresponds to 1h23' of navigation. The figure 3 presents the characteristics of the logged true wind (speed and direction). The sailboat's sensors can only measure the apparent wind. Data presented in figure 3 have been reconstructed in real time by the sailboat computer.

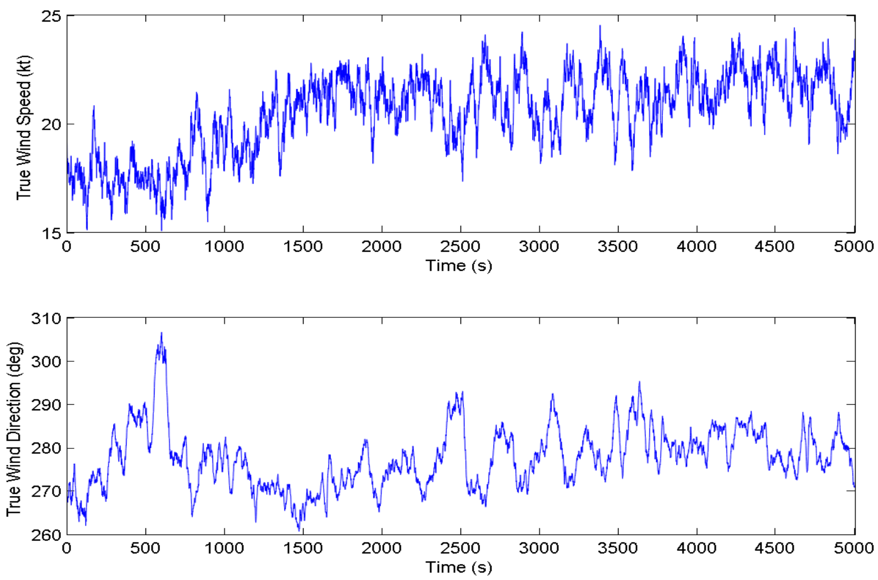

Figure 3 - Real wind of the data set

Even if the goal of a parametric model is to ignore the physical meaning of the coefficients, it is interesting to wonder what kind of equations can describe the simulated behavior, because it may give information for the order of the model, and the potential nonlinearities. For the case study, it can be assumed that the motion of the sailboat is govern by a classical second-order-dynamic equation, but the effect of the wind is not linear: simplest models of fluid mechanics give the lift of the sail as a function of the squared speed of the wind. To keep a linear model, we add an input channel with the wind speed directly squared.

For the case study with absolute referential, the inputs are TWS, TWS^2, TWD and Rudder angle. For those with relative referential, inputs are AWS, AWS^2, AWA and Rudder angle. In all cases, outputs are COG, SOG and Heel.

Model choice

We evaluate now which model can reproduce efficiently the behavior (here the motion) of the sailboat. Due to lack of space, we consider here only the four models generated by the two options: choice of a referential (absolute or relative) and choice of a model family (state-space or autoregressive). Hybrid and multi-model solutions are not discussed here. The characteristics of the four models we compare are listed below:

- A : absolute, state-space;
- B : absolute, autoregressive;
- C : relative, state-space;
- D : relative, autoregressive;

To compare performances of those four models, we propose two C1 and C2 criteria. C1 is the norm between the simulated and the measured output COG. This norm is given by the following equation in order to be independent from the length of the data set.

$C1 = \left(1 - \frac{\|COG_m - COG_s\|_2}{\|COG_m - mean(COG_m)\|_2}\right) \times 100$ with COG_m the measured COG and COG_s the simulated one. A value of 100% indicate to identic signals.

The second criterion C2 is the distance (in nautical mile) between the end of the simulated trajectory and the measured one. It evaluates the capacity of the model to give the correct trajectory. For each kind of models, the outputs are simulated for the full dataset, then the COG and SOG are integrates to get the simulated trajectory. Results are presented in figure 4.

Model name	C1 (%)	C2 (NM)
A	36.6	0.8261
B	4.39	1.3089
C	-2.95	1.4157
D	-110	1.9528

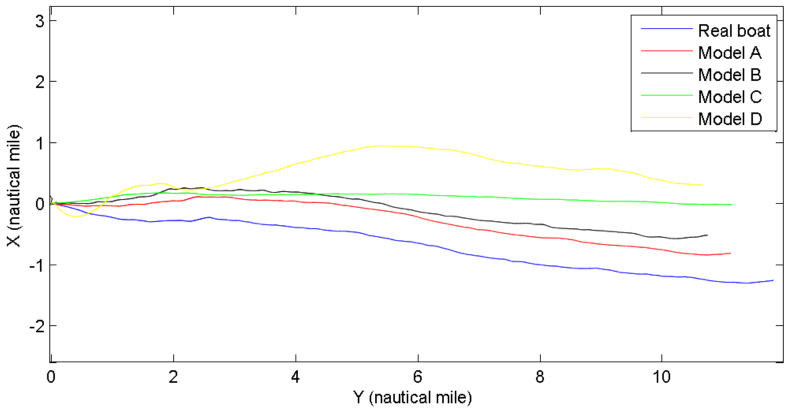

Figure 4 – Numerical criteria for comparison of models and simulated trajectories of the sailboat

We can see that models based on the absolute reference are more accurate than those based on the relative one, for the two criterions. The superiority of the absolute reference is not really surprising: the racing multihulls are able to go faster than the wind, so the apparent wind is very dependent of accelerations and gyration of the sailboat. For family of models, the state-space model appears as better than the autoregressive one. In the following, we only consider model A, which appears as the best of the fourth for the considered data.

Impact of window size

The goal of this section is to determine if it is possible to increase the performance of a parametric model, by changing the size of subsets of data used to compute the model's coefficients. In the previous section, the set is split in two equal parts of 2500 points each. First part is used to compute the model, and the second is used for validate it, mainly with the two criteria. We propose now to split the data set in n equal sets. If coefficients are computed on the p^{th} set (p belongs to [1 ; n]), the n-1 other sets can be used to evaluate the model accuracy. Sweeping p for all values between 1 and n gives n*(n-1) combination of evaluation.

The evaluation has been done for the model A. For each situation, the C1 criterion is computed. C1 is chosen because it does not depend of the size of the data set, contrary to C2. The figure 5 presents the results using box plots, one for each value of n between 2 and 7.

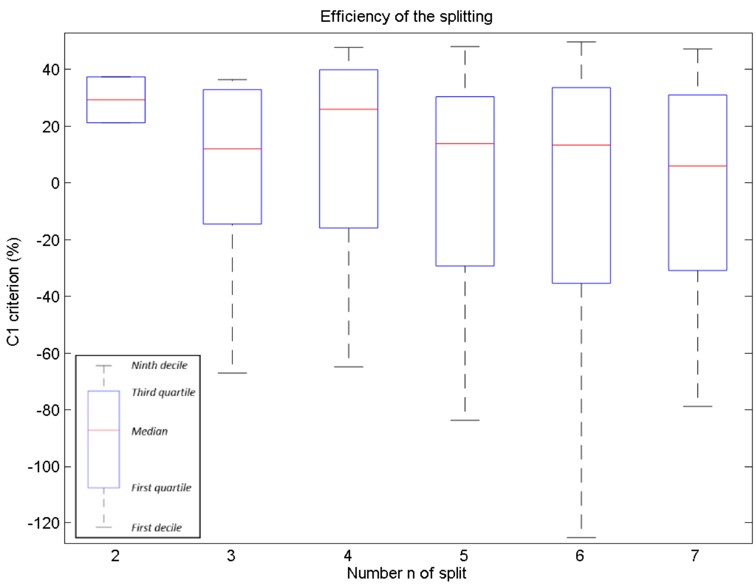

Figure 5 – C1 criterion for different values of n

The figure shows two interesting values of n providing better results: 2 and 4. For both, the mean is high, so does the first decile. Results for n = 4 are quite particular: the majority of results are pretty good, and only a few values are really bad. It can be explained by analyzing the measured wind. We may observe two different regimes on the wind speed: during the first 1300 points, the speed regularly increases from 17 to 21 knots, and then it oscillates around 21 knots, with average amplitude of 2 knots. The direction graph presents the same trend: a strong variation of 35° in the first period, and then an oscillation around 280°. The first 1300 seconds (nearly the first quarter of the set) contains a different regime of wind. The model computed with this first quarter is bad to predict the behavior of the sailboat in the other regime, whereas models calculated with the three other quarters are quite good to predict the behavior in the three last quarters. This explains the presence of few bad values among very good ones.

On the other hand, the results for n = 2 are good too, even if the low dispersion is explained by the number of elements in the series (only 2). The detailed values contains interesting information : the C1 criterion is 37.1 for the model computed in the first part and evaluated in the second one, and it is only 20.9 for the model computed in the second part and evaluated in the first one. This difference can be interpreted because of the presence of different regimes: the first part contains the two different wind regimes, so the model computed on it can “catch” the behavior of the sailboat on both, and it can simulate the behavior on the second set of data, which contains the second regime. Conversely, the model computed with the second part of data only catches the behavior of the sailboat in the second regime. That’s why it is worse to simulate this behavior during the first regime at the beginning of the first part.

To conclude on the influence of time discretization, a time period containing a very particular regime is powerful to emulate only similar regimes, whereas a large period, containing different regimes, leads to a more polyvalent model, maybe less accurate. The discretization work consists then in finding a trade-off between number of splits and model accuracy.

Related works

As mentioned in [7], Cyber-Physical Systems (CPS) present several issues in term of design, performance and quality of service. There are numerous challenges in CPS design, in term of architecture modeling, simulation, tooling, frameworks, and validation. This paper addresses the challenge about realistic simulation.

About simulation, we share with [8] the idea that the dynamic of CPS evolves very fast, and design of CPS requires handling the complexities posed by temporal variations and designing situation-specific control actions.

In [9], the author proposes a technique for translating the analytical dynamics of a physical system, and especially mechanical ones, into running simulation codes. The approach is interesting because the gap between analytical modeling and the simulators can significantly, impedes the development of CPS. In our paper, this problem is solved differently: parametric models are computed only by identification, so the CPS can be simulated without any analytical model, and the parametric model can be integrated directly into simulation tools. The gap with real system may be reduced, but on the other hand the parametric model does not embed physical meaning.

In [4] the author proposes a multi-agent architecture for modeling and implementing a safe and efficient autopilot. For tuning the controller, a virtual environment has been built simulating sea, wind and sailboat. The user may trim the virtual sails, define different weather scenarios (sea state and wind), and clone the sailboat in order to compare two different pilots and finally evaluate which is the best. The sailboat model is based on classical mechanics equations but for efficiency reasons the model is quite basic and doesn't capture all the dynamic of the sailboat. Therefore the behavior of the virtual sailboat doesn't match with a real one and makes difficult the evaluation of the autopilot.

Most of the key challenges of this paper are shared with the author of [10], whose goal is to simulate the behavior of an IACC mono-hull. [10] aims at the development of a physical simulator for sailors training. The simulated behavior should be as close as possible as the real one. [10] proposes to use a combination of analytical models for the dynamic, resulting from computational fluid dynamics (CFD) and wind tunnel experiment, and real scale experiments. Some identifications methods are used during these large scale experiments, but only to identify values of coefficients of the proposed analytical model of sailboat's dynamic. Moreover, some specific sailing navigation has been done especially for the identification. The parametrical models discussed in this paper may be less accurate than the one exposed in [10]. But they do not need heavy CFD calculus, and they can be computed from data recorded on a "classical" navigation, without monopolizing the sailboat for a measurements campaign.

Conclusion

This paper proposes an approach to set up a parametric model for a racing sailboat. Experiments on a representative set of data shows the interest of considering an absolute reference and a state-space model. The time discretization is interesting to capture specific behaviors but presents some limits to cover multiple regimes of a system.

We are now working on hybrid modeling (autoregressive and state-space) and on automating parametric model evaluation and its initialization. We are also considering more and more data (new data like the sea state and new measurement campaign) to improve the obtained models.

References

[1] P. Stoica, M. Jansson "*MIMO system identification: state-space and subspace approximations versus transfer function and instrumental variables*" The Bell System Technical Journal, Vol 48 (5), 1101-1113, 1969.

[2] J. Mann "*Wind Field Simulation*". Probabilistic engineering mechanics 13 (4), 269-282, 1998.

[3] M. Shinozuka, G. Deodatis "*Simulation of Multi-Dimensional Gaussian Stochastic Fields by Spectral Representation*". Appl. Mech. Rev 49(1), 29-53, 1996.

[4] G. Guillou. "*Architecture multi-agents pour le pilotage automatique des voiliers de compétition et extensions algébriques des réseaux de Petri*". PhD thesis, Brest(UBO), November 2010.

[5] I. Landau, A. Besançon-Voda, "*Identification des systèmes*". Paris : Hermes Science Publications, 2001

[6] http://www.mathworks.com/

[7] S. K. Khaitan, J. McCalley. "*Design Techniques and Applications of Cyberphysical Systems: A Survey*" IEEE Systems Journal, Vol 9(2), 350-365, July 2014.

[8] G. Hackmann, W. Guo, G. Yan, Z. Sun, C. Lu, and S. Dyke, "*Cyber-physical codesign of distributed structural health monitoring with wireless sensor networks*" IEEE Transactions on Parallel and Distribute Systems, Vol 25, 63-72, Jan 2014.

[9] Y. Zhu et al., "*Mathematical equations as executable models of mechanical systems*" in Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems, 1–11, Stockholm, Sweden, April 12 - 16, 2010.

[10] J. R. Binns, "*The development and use of sailing simulation for IACC starting manoeuvre training*" 3rd High Performance Yacht Design Conference, Auckland, 2-4 December, 2008.