SIMTECT 2009

> SIMULATION > CONCEPTS, CAPABILITY & TECHNOLOGY

CONFERENCE PROCEEDINGS

> 15-19 JUNE 2009
> ADELAIDE CONVENTION CENTRE, ADELAIDE, SA, AUSTRALIA

www.simtect.com

SimTecT 2009 is held under the auspices of the Simulation Industry Association of Australia (SIAA) Ltd ABN 13 087 862 619

CAE

South Australia.

MINING INDUSTRY

CUBIC

Editor
Dr Elyssebeth Leigh
University of Technology, Sydney

Published by the Simulation Industry Association of Australia
ABN 13 087 862 619

ISBN 0 9775257 6 7

For further information, contact:

Simulation Industry Association of Australia
PO Box 226
Lindfield, NSW, 2070
Australia

www.siaa.asn.au

Email: exec@siaa.asn.au

SimTecT 2009 Paper Review Committee

Dr Elyssebeth Leigh (Papers Coordinator)	Senior Lecturer, University of Technology, Sydney
Ms Emily Andrew	Director, NCS Modeling & Simulation, Raytheon Company
Dr Jonathan Binns	Researcher, MT&E, Australian Maritime College
Mr Anthony Cramp	Defence Scientist, Defence Science and Technology Organisation
Mr Alfred Devivi	Senior Software Engineer, Northrop Grumman
Mr Gary Eves	Practice Lead, QinetiQ Consulting
Mr Per Gustavsson	Vice Manager, Ericsson Microwave Systems AB
Mr Peter Hill	Director, Simcon Services
Mr Nick Howden	Business Development Manager, CAE Professional Services
Dr Heath James	Modelling and Simulation Discipline Lead, Defence Science and Technology Organisation
Mr John Marychurch	Solutions Architect, Thales Australia
Dr Michael McGarity (Paper Review Coordinator)	Manager, Decision Support Services, CAE Australia
Prof N K Mehta	Indian Institute of Technology Roorkee
Mr Roger Mulligan	Consultant
Mr Daniel Munro	Associate, Booz & Company
Prof Saeid Nahavandi	Chair in Engineering, Deakin University
Dr John Olsen	Lecturer, School of Mechanical and Manufacturing Engineering, University of New South Wales
Dr Simon Parker	Principal Research Scientist, Defence Science and Technology Organisation
Mr Craig Pepper	Technical Lead, Boeing Defence Australia
Mr Shane Rogers	Senior Manager, Systems Analysis Laboratory, Boeing Defence Australia
Dr Peter Ryan	Principal Research Scientist, Defence Science and Technology Organisation
Dr David Stratton	Leader, Distributed Simulation Laboratory, University of Ballarat
Mr Philip Swadling	Chief Engineer – Simulation, Thales Australia
Dr Andreas Tolk	Senior Research Scientist, Old Dominion University, USA
Mr Grant Tudor	Director, Army Simulation Wing
Mr Charles Turnitsa	Virginia Modeling Analyses & Simulation Center, Old Dominion University, USA
Dr Susannah Whitney	Cognitive Scientist, Defence Science and Technology Organisation

Development and Use of a Computer Controlled Sailing Simulation

Dr. Jonathan R. BINNS; Mr. Robert MAHER; Dr. Christopher K.H. CHIN; Prof. Neil BOSE
National Centre for Maritime Engineering and Hydrodynamics, Australian Maritime College, University of Tasmania
j.binns@amc.edu.au
Dr. Mark HABGOOD
University of Melbourne and Virtual Sailing Pty Ltd
mhabgood@unimelb.edu.au
Dr. Chris MANZIE
University of Melbourne
manziec@unimelb.edu.au

Abstract. The Virtual Sailing (VS) developed and produced ride-on sailing simulators continue to be used for research and educational purposes. This paper reports on a collaborative project between VS, the Australian Maritime College (AMC) and the University of Melbourne (UoM) which has led to the development of a computer controlled simulation (CCS) incorporated into the existing human-in-the-loop (HIL) sailing simulator.

The CCS can be run on a laptop networked to the main simulator. It uses the high level numerical language Matlab communicating over a TCP/IP network to display the final results in real time. This robust setup permits its implementation on a variety of platforms and development in a number of paths. The user can sail against the CCS as another on-screen opponent, providing a link back to the HIL simulation that is the core of the active simulators.

A number of numerical techniques were investigated to integrate state variables with respect to time. The methods have been analysed and an adaptive step size method has been proposed. A Proportional-Integral-Derivative (PID) controller has been implemented relating error in the vessel's heading angle to the required rudder angle for correction. The PID controller combined with efficient simulation and incorporation to an existing HIL simulator provides a powerful tool for future development and deployment of simulation technologies. Applications of this technology are described in the areas of design evaluation, sail training and manoeuvre prediction.

1. INTRODUCTION

The use of simulation in the sailing industry has grown from the initial purpose of design evaluation (Keuning, Vermeulen, & deRidder, 2005; Masuyama, Fukasawa, & Sasagawa, 1995) and optimisation (Philpott & Mason, 2002) to become a multi-purpose tool for training at all levels (Binns, Bethwaite, & Saunders, 2002; Binns, Hochkirch, DeBord, & Burns, 2008).

The complexity of the underlying simulation can be extremely high due to unsteady aerodynamic and hydrodynamic forces (Gerhardt, Flay, & Richards, 2008; Imas, Buley, Baker, & Ward, 2008; Masuyama & Fukasawa, 2008), although sufficient accuracy can be obtained from relatively low complexity simulations (Binns et al., 2002). The challenge in simulating any plant or device is to reduce the order of the model to a level whereby the key response characteristics are maintained to give a desired level of simulation fidelity, while the overall complexity of the model is lowered to allow efficient real-time computation and servicing of peripherals (such as graphics display, motion control and interfacing). This paper describes an application of CCS and a methodology for maximising the computer resources available to perform the necessary calculations.

The most fundamental numerical calculation required of an inertial based dynamic simulation is to perform a numerical integration to step through time, such as that described by Richardt, Harries, & Hochkirch (2005). A wide variety of such numerical integration schemes have been produced by mathematicians for use by engineers for some time (Davis, 1986). Each method has inherent trade-offs of computational speed and accuracy which depend on the equations they are solving. For this paper four methods have been used as described in Kiusalaas (2005).

The method of time integration chosen uses an adaptive step size, such that the time step is contracted at times of predicted numerical instability and expanded when the system of equations is relatively close to a steady state solution. If comparable in CPU time to a constant step size; adaptive step size produces a system capable of freeing considerable CPU time.

2. THE UNDERLYING SIMULATION

The simulation developed in this paper has been focused on the Laser standard sailing dinghy developed by Bruce Kirby during the 1970s. The reason for the focus on this particular sailing dinghy is due to its great popularity and the availability of dinghies.

The particulars of the laser standard sailing dinghy are shown in Table 1.

Table 1: Particulars of the laser standard sailing dinghy.

Length overall (m)	4.23
Beam (m)	1.37
Sail Area (m^2)	7.06

Hull Weight (kg)	59

For the purpose of simulation, the laser sailing dinghy was simplified to four core components: 1.the sail - this produces aerodynamic forces that drive the vessel; 2. the hull form - producing a hydrodynamic force and moment; 3. the centreboard - which creates a hydrodynamic force and moment; and 4. the rudder - which creates a hydrodynamic force and moment.

The centre of gravity is at the location shown in Table 2. It is taken to be the centre of geometry of the vessel with the zero point being the centreline of the transom on the load waterline. The centres of effort for the three lifting surfaces are shown in Table 3.

Table 2: Centre of gravity position.

	X	Y	Z
Centre of Gravity (m)	1.884	0	0

Table 3: Centre of effort for sail, daggerboard and rudder.

	X	Y	Z
Sail	1.734	0	2.07
Daggerboard	1.474	0	-0.41
Rudder	-0.466	0	-0.22

The first simplification applied to the simulation was in the application of the inertial based system. For this purpose second order terms with respect to yaw velocity, trim angles and products of inertia were all assumed to be small, resulting in greatly simplified equations of motion (Binns, 2005). The resulting boat centred coordinate system applied to this simulation is shown in Figure 1. The four degrees of freedom used (surge, sway, yaw and roll) are labelled in this figure.

For the Matlab implementation of the simulation the sail, centerboard and rudder were all modelled as simple lifting surfaces, such that a given angle of attack resulted in a predicted lift and drag in flow fixed coordinates. The hull was modelled with a velocity dependent drag component and a roll dependent righting moment. Additional yawing and rolling moments resulted from the offsets in force application points listed in Table 3.

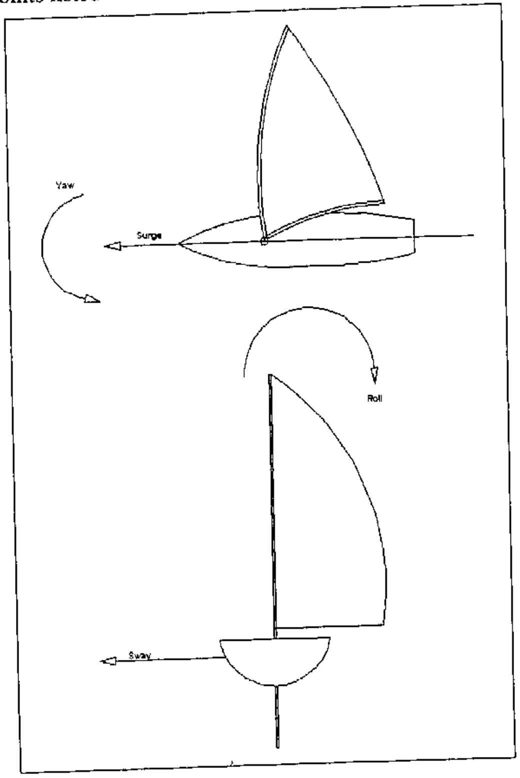

Figure 1: Four degrees of freedom coordinate system.

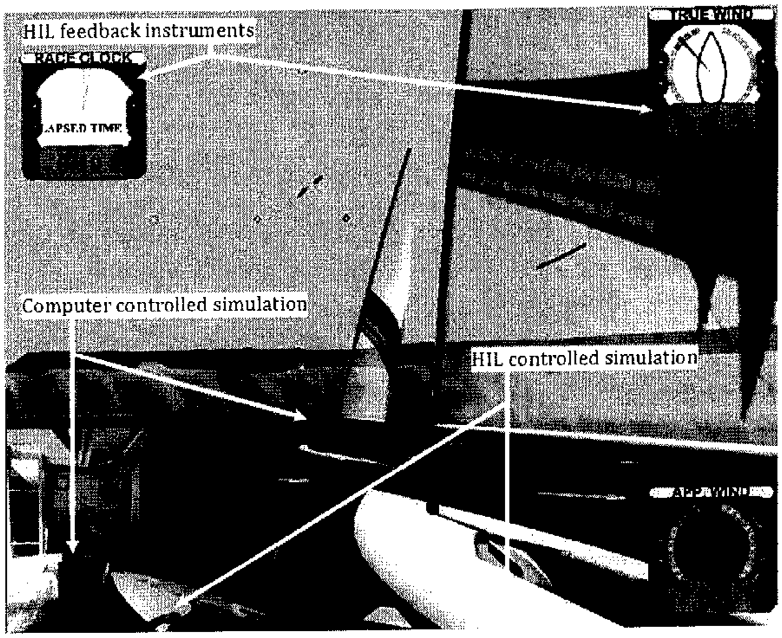

Figure 2: Main picture shows 3D graphic representation of HIL simulation following the CSS. Inset shows the operator using the HILS, the laptop in front of the user is running the CSS.

3. ELEMENTS OF THE COMPUTER CONTROLLED SIMULATION

For all simulation results presented within this paper the wind direction has been held fixed. Therefore a constant true wind angle results in a constant compass heading.

3.1 Interface with the existing simulator

The first element required to make use of the computer controlled simulator (CCS) was to interface it with the existing human-in-the-loop simulator (HILS) by displaying the results in real time. For this purpose standard Winsock TCP/IP communications were used within Matlab, wrapped within a dynamic link library (.dll). This arrangement permitted the easy interfacing of a standalone CCS over an Ethernet network as depicted in Figure 2.

3.2 Trajectory decision module

Two simple algorithms were coded into the trajectory decision module. Firstly the desired heading was alternated by 90° every 40 s, that is the CCS tacked every 40 s. The second trajectory algorithm employed, altered the requested heading such that the dinghy maintained a constant true wind angle until directly abeam from the next desired location. At this point the PID heading setting was rotated to point to the next desired location. In this way the CCS progressed its way tacking into the wind. Although this element is extremely simple; it does provide the investigation with an answer concerning the possibility of future developments.

3.3 PID Controller

The Matlab based simulation applied a proportional–integral–derivative (PID) control feedback loop to determine the rudder response to correct a course to the desired heading and a proportional (P) control feedback loop to trim the sails. The PID control module in this program controls the rudder angle which is based on the error between the actual heading and desired heading of the vessel.

Due to the large magnitude of the error that is created when the dinghy is required to perform a manoeuvre such as changing tack, the rudder angle of the vessel was limited to ±45 degrees.

Figure 3 shows the yaw angle and yaw velocity of the vessel simulating a beat to windward (the first trajectory mode) at an initial true wind angle of – 45 degrees and tacking through 90° every 40 s. The vessel is sailing in a true wind speed of 10 knots.

The response of the vessel when required to tack every 40 s is repeatable throughout the manoeuvre. The yaw velocity of the vessel can be seen to be changing quickly to around 0.6 rad/s before settling back to 0 rad/s without excessive overshoot or excessive settling time.

This system response is believed to be a good "idealised" model of a helmsman. A quick method of introducing a more "human" response without the addition of significant computational resources could be to introduce greater steady state errors by the reduction of the integral controller. That is, a better reflection of a human controlled model is possible by using a proportional-derivative (PD) module to control the helm of the vessel, thus reducing the order of the system by one and re-introducing a steady state error.

In addition to removing the integral control from the controller, the proportional value would also need to be adjusted.

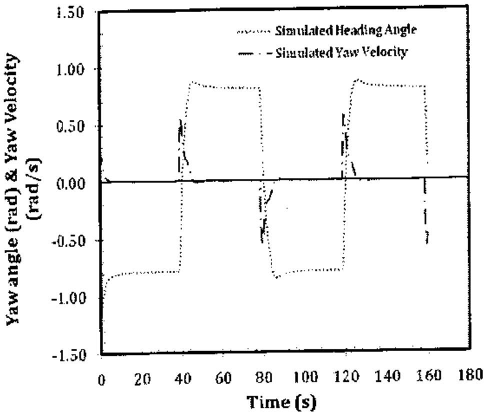

Figure 3: PID system response in Yaw.

4. TIME INTEGRATION SCHEMES

It has always been a major aim of the VSail simulation projects that standard desktop computing should be used to run the simulation components. This is to ensure mainstream take-up of the technology, which is heavily cost driven. Therefore, real time operation of the simulation has always been maintained by using efficient and simplified algorithms. The development of the CCS was no exception, requiring that standard Matlab be used to produce results at real time.

To achieve real time output from Matlab four different integration algorithms were investigated: 1. a 4th order Taylors method; 2. a 4th order Runge-Kutta method; 3. a 5th order Runge-Kutta method; and 4. an adaptive step size 4th/5th order Runge-Kutta method. The adaptive step size method was programmed such that numerical stability was assessed by the comparison of the 4th and 5th order methods. If the numerical stability criterion was not met, the step size was reduced. If the stability criterion was exceeded, the step size was increased (thus saving computational time).

To test these methods a basic mass/damper/spring (MDS) system was analysed. This basic system was chosen for analysis because a sailing yacht resembles a simple MDS system in roll and an analytical solution is easily obtainable for a simple MDS system for establishing errors and convergence rates. For example a solution to the MDS system can be obtained by considering the analytical response of the system with an initial unit of spring displacement and the same system can be predicted by the numerical methods mentioned above. A comparison between the numerical and analytical results provides an approximation of the error.

A full description of the study has been given in Maher (2008), a summary of which follows. The error and convergence of the methods can be analysed by plotting a log-log graph of the difference between the instantaneous numerical result and the analytical (or continuous) result, such as shown in Figure 4. In this figure the error has been calculated using Equation (1),

$$E(t) = \left| y_{ANALYTICAL}(t) - y_{NUMERICAL}(t) \right|, \qquad (1)$$

where $E(t)$ is the instantaneous error at time t, and $y_{ANALYTICAL}(t)$ and $y_{NUMERICAL}(t)$ are the analytical and numerical results respectively.

For this figure a final convergence rate of 5th order can be clearly seen on the right hand side of the graph.

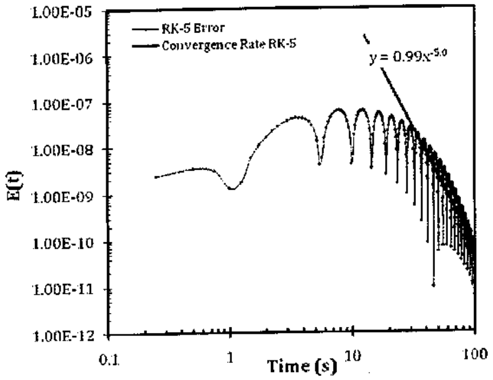

Figure 4: The absolute error of 5th order constant step size Runge-Kutta integrations with respect to time from initial release and the convergence rate of the power series on a log-log scale.

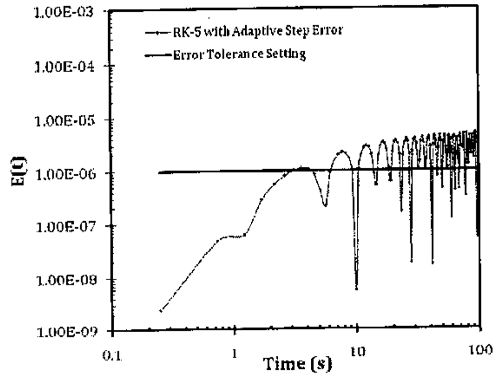

Figure 5: Absolute error of 4th/5th order adaptive step size Runge-Kutta integration with variable step size with respect to time from initial release and the convergence rate of the power series on a log-log scale.

It was found that the 4th order convergence rate of the Taylors method could only be obtained by decreasing the time step to 1/25th of the Runge-Kutta methods, therefore this method was not pursued as it would appear that it requires a significantly smaller time step to attain equivalent convergence rates.

The Runge-Kutta methods all converged to the stated rates at the same time step. However, the adaptive step size Runge-Kutta method displays a useful engineering property of constant error with time as shown in Figure 5. For computational efficiency, this method therefore shows promise, as desired accuracy is easily set. A real-time simulation, however, requires a constant time step, which may appear to preclude the use of adaptive step size control. To overcome this problem a series of logic steps were established which adapted the step size until the required output time would have been exceeded. At this point the step to the next output time was used, that is a time step slightly smaller than that predicted for numerical stability was used to ensure that real-time output was maintained.

4.1 Adaptive step size implementation

The adaptive step size 4th/5th order Runge-Kutta method described above requires significantly more calculations per iteration step, it saves computing time by expanding out the time step, hence requiring less iterations, at times within the simulation when the system is close to steady state within acceptable tolerances. To test the appropriateness of the adaptive step size control the full simulation was used to provide a sample manoeuvre sequence lasting 100 s. The 5th order Runge-Kutta and the adaptive step size methods were then tested throughout the manoeuvre. To perform this test the Profiler function of Matlab was used to time the code.

The computer specifications used for the test are detailed in Table 4.

Table 4: Computer specifications.

CPU Type	Intel Core (TM) 2 CPU T7200 @ 2.00 GHz
RAM	997 MHz, 0.99 GB
OS	Microsoft Windows XP, SP2
Interface	400 Mbps Firewire

From this test it was found that the adaptive step size method required 96 seconds of CPU time to complete the 100 second simulation with interfacing to the main simulation and production of one position plot. This setup therefore permitted real time operation. For this method the time step had to be reduced to 0.04064 s at the most point if highest estimated numerical instability.

The next analysis was performed on a 5th order Runge-Kutta procedure with a constant 0.04064 s time step. It has been assumed that by equating the time step of the 5th order Runge-Kutta method with that of the minimum time step used in the adaptive step size method the two methods will maintain a 4th and 5th order convergence to the analytical answer. The analysis of the constant step size 5th order Runge-Kutta method showed that 122 seconds of CPU time was required to perform the 100 second simulation, which precludes real time operation given the existing hardware. For this system to work at real time some operations would have to be farmed out to other hardware items. It was found that to complete the calculations within the same time (that is 96 seconds) the time step needed to be increased to 0.053 s for the 5th order Runge-Kutta method.

This test demonstrates that for the specific sailing manoeuvre examined on the specific hardware used the adaptive step size technique can be seen to produce a simulation with comparable computational performance as the constant step size 5th order Runge-Kutta method, that is they both take a similar amount of CPU time to complete the given simulation. The adaptive step size method does, however, provide the additional robust feature of maintaining a constant relative error.

5. APPLICATIONS OF THE CCS

5.1 Use of adaptive step as a Dynamic Response Index

If one of the equations of motion is exhibiting a large difference between 4th and 5th order approximations, the dynamic response in that equation is becoming faster and its step size can be expected to become smaller so the program can maintain a constant error tolerance. As such in addition to maintaining error limits, independent stepping of the equations of motion allows for a dynamic response assessment to be calculated.

Each of the equations' step size in this simulation code is independent and thus dynamic response can be detected by examining the optimised "next step" size which is selected by the adaptive step mechanism. For example, if the dynamic system requires large gradients in yaw, the next step size for yaw will reduce until it is within acceptable error limits before the program completes the requested time step. If the instability is persistent or greater on the next loop then the step size will be reduced again on the next step and again on the step after that until the program grinds to a halt or (if under a transitional condition) the model changes to a more stable (that is closer to steady state) condition. This phenomenon is very helpful when debugging as it allows the user to determine the specific equation of motion that is causing the instability by examining step sizes. As such a dynamic response index has been added to the program.

The size of the next step has given rise to a concept of a dynamic response index. The dynamic response index is the next step size for each of the equations of motion. The higher the value returned for these parameters the closer the dynamic system is to steady state in that direction.

An index has also been created from the "applied step size". This is the value of the smallest "next step" that the adaptive 4th/5th Runge-Kutta method has calculated in one cycle. That is, if the vessel is most unstable in yaw then the yaw equation "next step" size will be the smallest for one cycle and will be stored. However if the vessel is most unstable in roll then the roll equation "next step" size will be the smallest for the cycle and is stored.

This index is believed to be a tool that if developed, could be used as an indicator of the performance of a design with respect to its dynamic response. The logic behind this conclusion is as follows. If it is assumed that the error propagated by the numerics of the simulation is constant for simulations of similar vessels then any change in the response index is due to a the system stepping closer or further away from a steady state solution. As such the dynamic response index for two or more similar vessels of different configurations can be compared and any difference in their response index should correlate directly with changes in their real world response. This is believed to be a tool which could be useful in the selection and optimisation of appendages for use on sailing vessels as it provides a quantitative measure for dynamic response.

An example of a dynamic response index is shown below in Figure 6. In this simulation the vessel is sailing to windward and tacking every 40 s. As can be seen, from t = 0 to t = 40 s the step size has been increased from 0.08 s per step to 0.23 s per step. At t = 40, the vessel initiates a tacking manoeuvre and the step size is adjusted to 0.06 s per step to allow for instability. The step size begins to increase again until around t = 45, where the step size starts to decrease again sharply from 0.13 s per step to 0.11 s per step before commencing to rise again. The cause of the decrease in step size at t = 45 s is not entirely understood. However, it is clear that this drop in step size corresponds with the completion of the tacking manoeuvre.

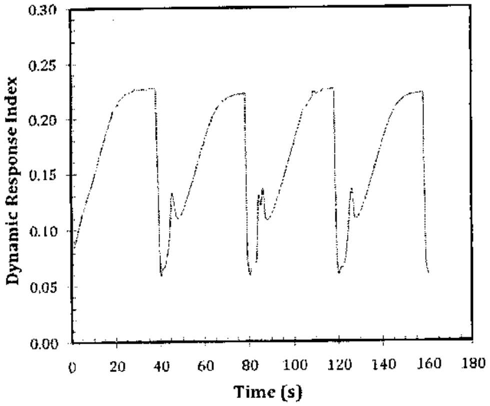

Figure 6: Dynamic response index with respect to time.

5.2 The CCS as a consistent opponent

As detailed in Section 3.3, the PID controller has produced idealised steering. Very small overshoot and steady state errors are visible. When coupled to the main human-in-the-loop simulator, the CCS can be raced, providing a consistent opponent. The addition of this feature increases the application of the simulator in the refinement of specific racing manoeuvres.

5.3 Forward prediction of dinghy position

The results presented in Section 4.1 indicate that the simulation requires further simplification or greater error tolerance to operate faster than real time. However both of these adjustments are possible; in addition increasing hardware performance decreases CPU time taken to perform calculations. The CCS could therefore be operated faster than real time. Once achieved the CCS could then be used to provide predictions of possible future dinghy movements. In sail racing this has particularly valuable application to the area of pre-start manoeuvres. In a pre-start manoeuvre the objective is to cross the start line at a specified time with maximum speed (Binns et al., 2008), therefore the ability to predict future boat positions could be used to provide present information concerning the need to increasing or decreasing dinghy velocity to arrive at the starting position on time.

6. CONCLUSIONS

The development of a computer controlled simulation (CCS) has been coupled to an existing human-in-the-loop (HIL) sailing simulator. To achieve this, a PID controller and a simple decision algorithm have been added to an underlying simulation coded within Matlab. The results of the CCS have been incorporated into the HIL simulator through a TCP/IP network interface, transmitting results in real time.

Although a constant output time is required for real time operation, the method of stepping to that time is open. An adaptive step size method has been used and compared to constant step size methods. The adaptive step size method has the very desirable property of a predetermined error bound and has been shown to use similar CPU time requirements for typical sailing manoeuvres.

A number of future uses of the CCS have been proposed. These uses include: adoption as a dynamic response index; providing a consistent opponent; and the forward prediction of dinghy position and speed.

Future work on the CCS will focus primarily on optimisation of the trajectory decision module such that it reflects the control of a human skipper with greater accuracy. The most likely path to this result will begin with a comparison of the CCS path to that of a human skipper, which will require considerable work in itself, as well as providing valuable methodologies for comparing such dynamic data.

REFERENCES

Binns, J. R. (2005). *Extreme Motions of Modern Sailing yachts - Quantifying Re-righting.* Australian Maritime College, Launceston.

Binns, J. R., Bethwaite, F. W., & Saunders, N. R. (2002). Development of a more realistic sailing simulator, *The 1st High Performance Yacht Design Conference* (pp. 221-228). Auckland, NZ: RINA.

Binns, J. R., Hochkirch, K., DeBord, F., & Burns, I. A. (2008). The development and use of sailing simulation for IACC starting manoeuvre training, *The 3rd High Performance Yacht Design Conference.* Auckland, NZ: RINA.

Davis, G. d. V. (1986). *Numerical Methods in Engineering and Science.* London, UK: Chapman and Hall.

Gerhardt, F., Flay, R., & Richards, P. (2008). Unsteady aerodynamic phenomena associated with sailing upwind in waves, *The 3rd High Performance Yacht*

Design Conference (pp. 148 - 157). Auckland, NZ: RINA.

Imas, L., Buley, G., Baker, B., & Ward, B. (2008). CFD-Based hydrodynamic analysis of high performance racing yachts, *The 3rd High Performance Yacht Design Conference* (pp. 130 - 137). Auckland, NZ: RINA.

Keuning, J. A., Vermeulen, K. J., & deRidder, E. J. (2005). A generic mathematical model for the manoeuvring and tacking of a sailing yacht, *The 17th Chesapeake Sailing Yacht Symposium* (pp. 143-163). Annapolis, USA: SNAME.

Kiusalaas, J. (2005). *Numerical Methods in Engineering with Matlab*. New York, USA: Cambridge University Press.

Maher, R. (2008). *Computer Controlled Sailing Simulation.* Unpublished Bachelor of Engineering (Naval Architecture) Final Year Thesis, Australian Maritime College.

Masuyama, Y., & Fukasawa, T. (2008). Tacking simulation of sailing yachts with new model of aerodynamic force variation, *The 3rd High Performance Yacht Design Conference* (pp. 138 - 147). Auckland, NZ: RINA.

Masuyama, Y., Fukasawa, T., & Sasagawa, H. (1995). Tacking simulation of sailing yachts – numerical integration of equations of motion and application of neural network technique, *The 12th Chesapeake Sailing Yacht Symposium* (pp. 117-131). Annapolis, USA: SNAME.

Philpott, A., & Mason, A. (2002). Advances in optimization in yacht performance analysis, *The 1st High Performance Yacht Design Conference* (pp. 229-236). Auckland, NZ: RINA.

Richardt, T., Harries, S., & Hochkirch, K. (2005). Manoeuvring simulations for ships and sailing yachts using FRIENDSHIP-Equilibrium as an open modular workbench, *International Euro-Conference on Computer Applications and Information Technology in the Maritime Industries*. Hamburg, Germany.