Ocean Engineering 134 (2017) 129–139

Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng

A real-time strategy-decision program for sailing yacht races

F. Tagliaferri[a,*], I.M. Viola[b]

[a] School of Marine Science and Technology, Newcastle University, United Kingdom
[b] Institute for Energy Systems, School of Engineering, The University of Edinburgh, United Kingdom

ARTICLE INFO

Keywords:
Yacht race
Tactics
Routing
Dynamic programming
Artificial neural network
Coherent risk measure

ABSTRACT

Optimal decisions for a skipper competing in a match race depend on a number of factors, including wind speed and direction variations, behaviour of the opponent, sea state, currents, racing rules. Expert sailors are able to combine observations on these various factors and process them to take optimal decisions. This study presents an attempt at emulating this decision process through a computer code that can be used in real time to advise on race strategy. The novelty of the proposed method consists in combining various approaches for the multiple factors affecting the decision process.

The wind variability is modelled with the use of neural networks, to produce a short-term wind forecast. The willingness of the sailor to risk is modelled using coherent-risk measures. Experimental results are used to quantify the loss of speed due to the presence of a nearby opponent. Finally, all these factors are combined through dynamic programming to compute an optimal course, based also on information on the current and yachts performance. The program is tested modelling the last 13 races of the 34th America's Cup, and results show that the route computed is close to the shortest possible route computed assuming perfect knowledge of sea conditions.

1. Introduction

A yacht race is a competition where two or more boats race each other to complete a certain course in the shortest time. Traditionally, the problem that a sailor has to solve is addressed as an optimisation problem consisting in going from point A to point B in the shortest possible time, under certain constraints given by the dynamics of the yacht and racing rules. This approach however doesn't really capture the competitive aspect of a race. In fact, the real aim of a sailor is not to get to the finish line as fast as possible, but rather to get there before their opponent(s). Moreover, the speed of a sailing yacht is highly dependent on the behaviour of the wind. A sailor doesn't have perfect knowledge of the future wind patterns, and therefore the problem must be addressed as a stochastic problem, based upon probability distributions of the wind behaviour. This paper presents the development of a routing algorithm aimed an enhancing the probability of winning during a race between two yachts, by considering multiple issues of stochastic wind changes, presence of the opponent, and risk management.

1.1. Background on yacht racing

The speed of a sailing yacht depends on the wind speed and the course wind angle (TWA, the supplementary angle between the wind velocity and the boat heading). Fig. 1 presents an example of boat speed (BS) as a function of the TWA for a given wind speed in a polar diagram. A polar plot of this kind, which may include different curves associated to different wind speeds, is the conventional way of presenting the boat speed, and although the actual BS can depend on other factors (such as waves and crew), it is considered as a characteristic of a yacht.

As shown in the plot, the highest values for the BS are achieved when sailing at a TWA of approximately 90° (on a *beam reach*). Conversely, when the TWA tends to zero, BS tends to zero. Therefore, when sailing upwind (for instance, from a downwind mark to an upwind mark), the most effective route consists in a zig-zag in the wind direction, sailing at a TWA of 35–50° (*close hauled*). In this case, a skipper's aim is to maximise the speed in the upwind direction, which means to find the TWA such that the projection of the boat velocity on the upwind direction is a maximum. The corresponding velocity is referred to as Velocity Made Good (VMG) and is shown in red in Fig. 1.

The VMG can be defined also for downwind sailing. In fact, as shown in Fig. 1, even if the velocity is not null when the TWA is 180°, the maximum projection on the downwind direction for this example is obtained at angles of approximately 150°. However, the optimal angle for downwind sailing can have significant variations depending on the yacht geometry.

* Corresponding author.
E-mail addresses: francesca.tagliaferri@ncl.ac.uk (F. Tagliaferri), i.m.viola@ed.ac.uk (I.M. Viola).

http://dx.doi.org/10.1016/j.oceaneng.2017.02.026
Received 14 January 2016; Received in revised form 13 January 2017; Accepted 22 February 2017
0029-8018/

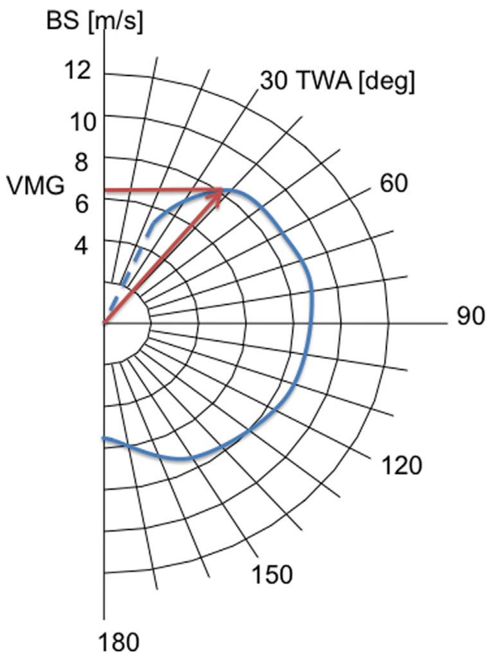

Fig. 1. Example of polar diagram.

Yacht races are held in many different formats and levels: in the case of a *match race* only two boats face each other, while in a *fleet race* the number of participants can be very high. Fig. 2 shows an example that appears in most races, where a yacht has to sail between two marks aligned with the wind direction. In the example shown, the yacht is sailing upwind, and as previously noted the VMG gives the best angle to sail at. This means that there will be a need for a certain number of changes of direction, called *tacks*, two of which are underlined by red arrows. The tacks lead to a time loss, as when changing direction the yacht is not sailing at maximum VMG. However, they may also lead to an advantage when a tack is performed to react to a change in wind direction. A fundamental question for a sailor is when it is the right choice to tack.

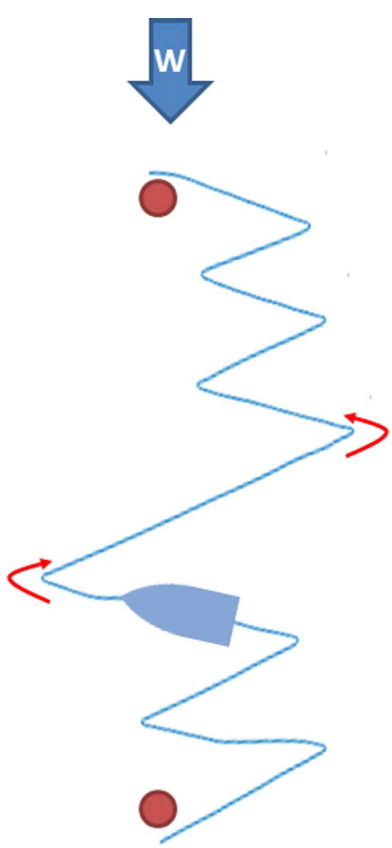

Fig. 2. Example of upwind leg.

1.2. State of the art on yacht racing strategy and contribution of present work

Most of the academic research in the field of competitive sailing is dedicated to analysing the fluid dynamics of sailing yacht, with the aim of maximising their performances. However, in recent years more attention has been given to the topic of racing tactics, and to the development of Race Modelling Programs (RMP).

Philpott and Mason (2001) and Philpott (2005) addressed the problem of finding the optimal route by introducing the use of dynamic programming. The race area is discretised and the wind is modelled as a discrete-time Markov Chain. At each step, the only available information about the wind is the current state and a probability distribution for the next state. The outcome of the algorithm is a policy aimed at minimising the expected time to complete the race leg. DP is shown to be an effective way to address the problem and is used in many subsequent studies, including Dalang et al. (2014), Tagliaferri et al. (2014) and the present work. DP has also been extensively investigated for decision-making in different applications, including energy (Clement-Nyns et al., 2010), logistics (Hall and Potts, 2003), medicine (Sahinidis, 2004). Markov Chains have been used also by Ferguson and Elinas (2011), in a study focussed on finding optimal routes for inshore racing, in presence of landmasses influencing the wind. In this case, the environment is considered fully observable (i.e. the wind behaviour is known), while the yacht's dynamics is uncertain.

Philpott et al. (2004) also addressed the issue of interactions between yachts. In this work an RMP is implemented to assess virtual competitions between yachts of different designs, to quantify the advantage of different candidates in a race scenario. The physical interactions are modelled by developing a theoretical model of a wind shadow region, using a penalty which penalises the downwind yacht and rewards the upwind yacht. Other studies have developed a fleet race simulator and used a lifting line method to compute the covering and blanketing effects in a fleet race (Spenkuch et al., 2008, 2010, 2011). Subsequently, Aubin (2013) and Richards et al. (2012) have provided experimental data showing the changes in the wind speed and direction observed by a yacht when sailing close-hauled to another yacht. This experimental data can now be used to compute the speed changes of a boat due to the relative position of her competitor.

The risk attitude of the sailors was recognised to be a crucial factor in the work by Scarponi et al. (2008), who introduced risk modelling in terms of payoff matrices allowing the sailor to choose the option of delaying a tack. Tagliaferri et al. (2014) included the risk model in the decision algorithm, and showed that rather than finding a strategy that minimises the expected time to complete a race leg, a strategy aimed at maximising the probability of completing before the opponent can improve the probabilities of winning. This can be achieved by having a risk-seeking behaviour when losing, and a risk-averse behaviour when leading the race. In this case, the attitude of the sailor depends on their relative position with respect to the opponent.

One of the most recent and complete work on yacht routing is represented by the computer program described in Dalang et al. (2014), produced in collaboration with the Alinghi team for the 2007 America's Cup. In this study, a program for the computation in real-time of the best route is presented. The program is extensively tested both through simulations and on the water, and represents the most advanced published work on real-time routing for yacht races. The wind is modelled as a Markov Chain based on past measurements, however no information on wind forecast is used. In the present work, we combine some of the aforementioned techniques to produce a program able to compute an optimal strategy during a match race between two boats. More specifically, the contribution beyond Dalang et al. (2014) consists in the use of a short-term wind forecast (the forecasting methodology, based on Artificial Neural Networks, is described in Tagliaferri et al. (2015)), in the inclusion of speed penalties based on the experimental results by Aubin (2013) and Richards et al. (2012), and of the risk model presented in Tagliaferri et al. (2014) and its optimisation. In this paper we discuss the advantages given by each of these features and their combination.

2. Method

2.1. Wind forecast

The changes in wind influence not only the yacht speed but also its optimal direction. Therefore a wind direction forecast is used to determine the most suitable discretisation of the race area used for the computations. The wind forecast is based on Artificial Neural Networks (ANN) as described in Tagliaferri et al. (2015). An ANN-based forecast is a statistical method that uses past measured values to produce a prediction of future values.

A wind dataset of 34 racing days collected during the 34th America's Cup, which was held in S. Francisco in summer 2013, is used (Americas Cup Event Authority, 2013). Wind speed and direction are recorded at a frequency of 5 Hz and averaged over the race area (i.e. the wind differences across the race area are unknown). The recorded data, denoted by x_1, x_2, … in Fig. 3, is smoothed by using a moving average over 6 min, and then sampled every 30 s. The new time series obtained, denoted as w_1, w_2, … is used as input for the ANN model. At the time t of the race, the input vector is made of the current (w_t) and past values ($w_{t-30\ \mathrm{s}}$, $w_{t-60\ \mathrm{s}}$,…,$w_{t-9\ \mathrm{min}}$) for a total of 9 min. This choice of data pre-processing is the result of the optimization described in Tagliaferri et al. (2015).

The input data points are processed by computational units called neurons, which are organised in a network structure. Input neurons and output neurons represent the units receiving input and providing output, and are represented by rectangles in Fig. 3. In the case presented, each input neuron is associated to one element of the input vector and the two output neurons provide the forecast wind for 30 and 60 s ahead, respectively. The processing neurons, also called “hidden”, represent a non-linear function. The connections between neurons represent numerical weights that are multiplied by the output of the previous neuron in the network. Two hidden layers of 20 neurons each are used. The output of the ANN model is a two-step-ahead forecast, which is used as a new input to the ANN to produce a longer-term forecast covering the expected duration of the upwind leg. The wind is assumed to be constant across the whole race area, and to vary only with time. The ANN is trained using the Levenberg-Marquardt back propagation training algorithm, in which the weights are repeatedly adjusted to obtain the best fit of the training set. The training is carried out using 75% of the dataset, while the remaining wind data are used for testing.

The full validation of the assembled code will be possible only in the long term through full scale tests. As a first step, to gain the users' confidence in the predicted strategy, Tagliaferri et al. (2015) showed that the ANN forecast applied to the direction of the wind recorded during the 34th America's Cup has an average error of 1.71 degree and 3.01 degrees for one and two minutes ahead, respectively. The authors argued that, for the racecourse of the 34th America's Cup, the ability to forecast wind shifts greater than 3 degrees allowed taking the correct tactical decision. On this basis, they showed that the ANN forecast enabled the correct tactical decision 97% of the times.

2.2. Race area

Fig. 4 shows the boundaries of the race area used in this work; these mimic the typical race course area of the 34th America's Cup. The distance between the starting point and the upwind mark is of 5000 m, and the width of the area is 3000 m. The course is assumed to be aligned with an average initial wind direction which is kept constant for the entire race. The area as shown in the Figure is delimited by ideal laylines, but in some cases the actual routes go beyond those lines. There is a limited tolerance (100 m) on the side boundaries for ease of grid computation.

The problem is modelled as a shortest path problem defined on a set of nodes connected in a lattice. The set of nodes is not fixed, but their position depends on the wind forecast. Before formally describing the process of grid definition, an example to motivate this choice will be shown. Let us consider the final phase of the upwind leg when the boat is reaching the mark in the case of a gradual wind shift towards the left. Fig. 5 shows the optimal route towards the mark with two different underlying grids. In the left grid, the optimal route does not go through the nodes defined by the grid, therefore a certain approximation in the DP algorithm is needed.

Conversely, the grid on the right shows an exact superposition of the route and one of the lines constituting the grid. Ideally, the nodes defined by the grid should correspond to the reachable points on the racing area. Of course the racing area is a continuum, so every point

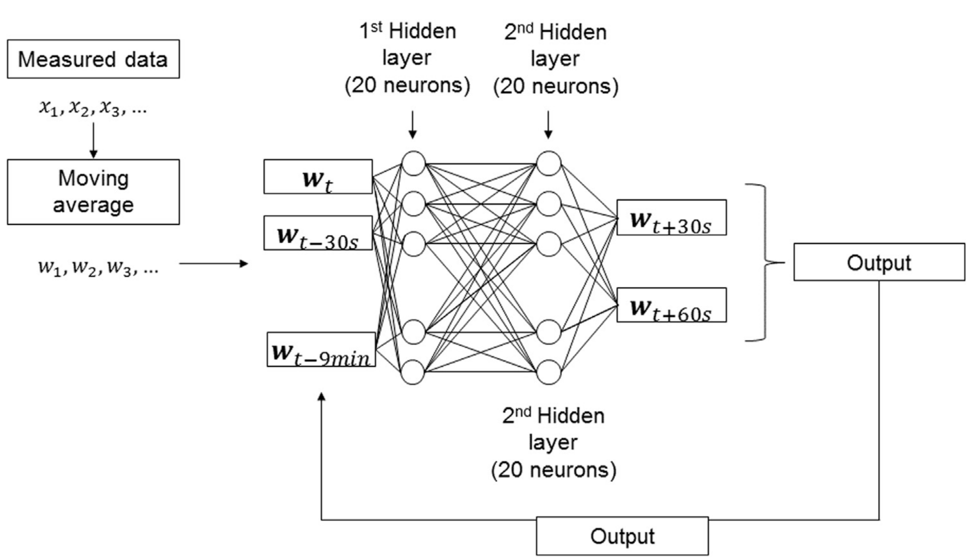

Fig. 3. Summary of ANN wind forecasting procedure.

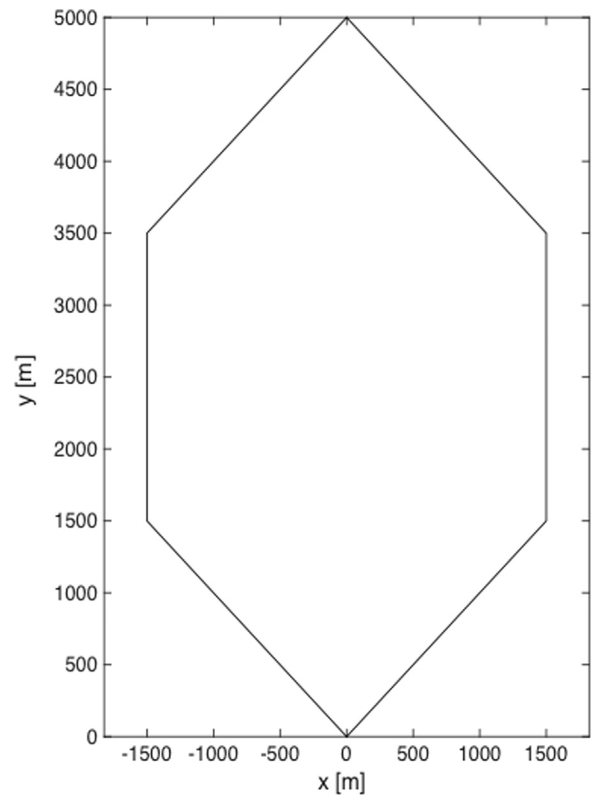

Fig. 4. Racing area.

within the race boundaries is always reachable, but the discretisation should be developed so that, if the yacht is in a given node belonging to the set of nodes defined by the grid, then the neighbour nodes should be reachable from that node. This property is not satisfied by the left grid in Fig. 5, but it is satisfied by the right grid, as shown by the red path followed by the yacht to reach the upwind mark. A curvilinear grid that matches the optimal route can be drawn if the future wind evolution is known. The grid defining the lattice used in this work is therefore based on the wind forecast, in order to predict the possible reachable points. The grid is then recomputed every time step. The main assumption underlying the construction of the grid is that, in the absence of tactical interactions due to the presence of a competitor, a skipper will always sail at maximum VMG.

Spatial variations are currently not considered in the current model because these are not measured by any instrument on board, and are known only through visual observation of the racing area. This is a limitation of the current model. Indeed, in some conditions, such as those when a new wind is coming through the racing area, neglecting the spatial wind variation could lead to a completely wrong strategy. However, if a spatial distribution of the wind was known, this could be implemented in the program. This would not affect the dynamic programming computation (the computation of the strategy), but only the construction of the grid points and it would increase the amount of data to be stored. In fact, instead of having one time series for wind speed and wind direction covering the whole race area, different wind time series from different sources on the race area would be needed. Moreover, the computational cost of each time step would increase due to the additional interpolation needed on each grid point.

Fig. 6 shows how to build the subsequent grid points given an initial node of coordinates (x,y). If the forecast wind when reaching the point (x,y) is represented by the wind w, then the possible reachable points in a given time step dt has coordinates (x_{1L}, y_{1L}), (x_{1R}, y_{1R}) depending on

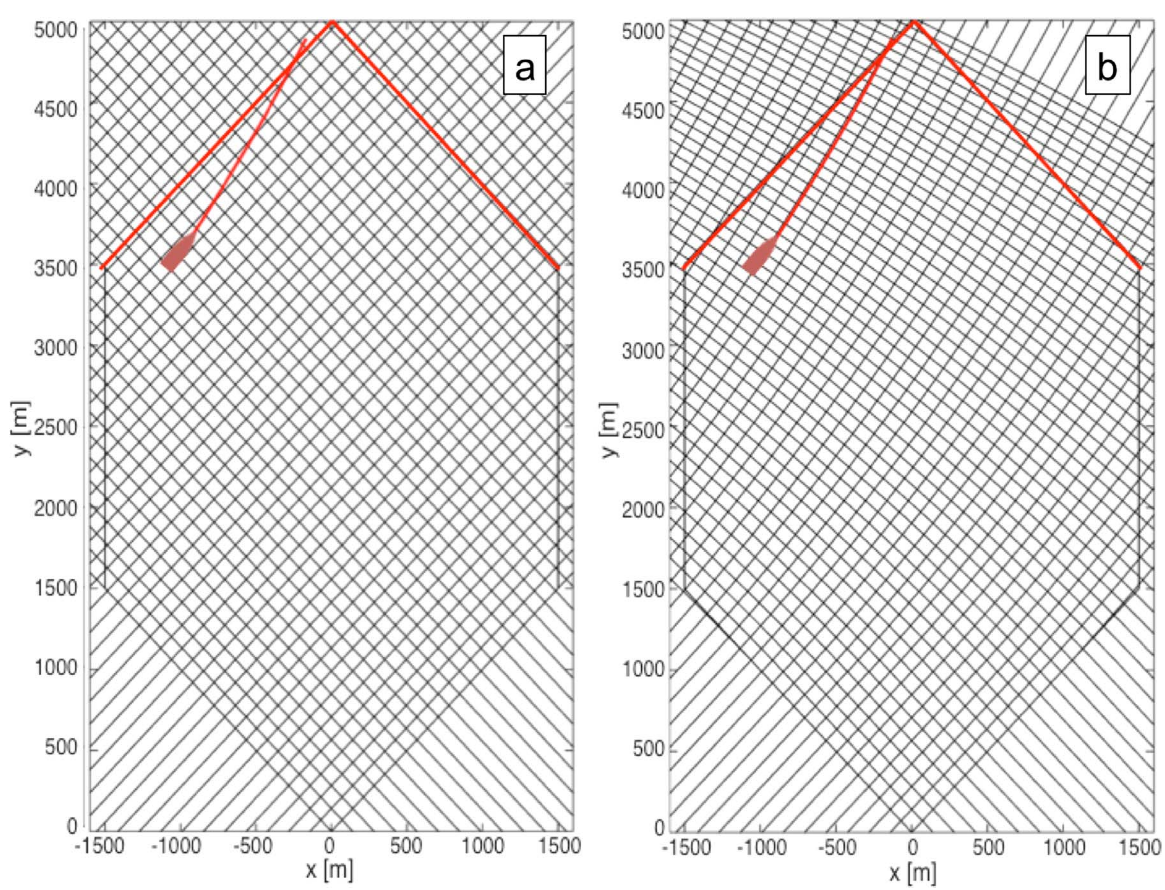

Fig. 5. Comparison between grid with fixed spatial steps (a) and with wind-dependent steps (b).

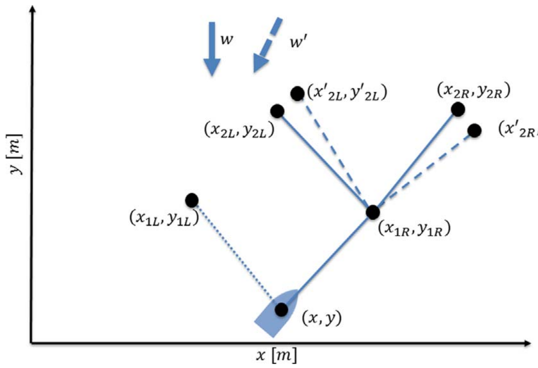

Fig. 6. Construction of grid points.

the current tack. Let us assume that the boat is on a port tack. Then in a period of time of $2dt$ the points (x_{2L}, y_{2L}), (x_{2R}, y_{2R}) can be reached. w is the wind which is *expected* at the moment when the grid is generated. A subsequent forecast could predict a different wind (e.g. w' in Fig. 6), in which case the reachable points become (x'_{2L}, y'_{2L}), (x'_{2R}, y'_{2R}). This is why the grid construction is updated at every step.

If every node generated two subsequent nodes, the size of the grid would grow exponentially at each iteration. Rather than building the grid point by point, the grid is therefore built by defining a set of lines and then considering their intersections as the nodes constituting the graph underlying the DP algorithm defined in the following Section. Fig. 7 shows the construction of the initial grid. A set of M_0 evenly spaced point is defined on the x axis, where M_0 depends on the desired grid resolution.

At step one of the computation the following operations are performed:

1. Yacht position: $(x_0, y_0) = (0, 0)$
2. Generate wind speed and direction forecast
3. Compute grid lines
4. Compute lines intersections. These points constitute the DP nodes.
5. Store grid points in matrices G_x,G_y

The distance between grid points depends on the chosen time step, which is also the period at which the optimal route is recomputed. This is the time step used in the computation of the optimal strategy and does not necessarily correspond to the time step used for the wind forecast. In this work we use dt=5 s, which corresponds to the maximum time, in the wind conditions addressed, needed to cover two boat lengths. The chosen time step must be higher than the computational time needed for recomputing the optimal course. For instance, with dt=5 s, the computational time is less than 2 s on personal laptop with a 1.6-GHz Intel Core i5-5250U processor. A smaller dt would lead to a higher computational time and, therefore, the smallest dt depends on the computational resources available. However, the results presented hereafter show that the time step allowed by a standard laptop are satisfactory.

The grid is built starting from the current position of the boat. The objective of the boat is to round the upwind mark clockwise. The mark itself is not necessarily a point of the grid. However, by construction the mark will lay between four grid nodes defining a grid cell. The leftmost node is considered the arrival node at each iteration. At each step k of the computation the grid is re-computed. The current position of the boat, (x_k, y_k) becomes the initial node. The equivalent of the initial points laying on the x axis are now a set of points evenly spaced (according to wind speed as for the first step) laying on the line of equation

$$y = \tan(CWA)(x - x_k) + y_k \tag{1}$$

2.3. Dynamic programming algorithm

Let us consider a dynamic system evolving according to the following Eq. (2):

$$x_{k+1} = f_k(x_k, u_k, \omega_k),\ t = 1, \ldots, N - 1 \tag{2}$$

where k represents a discrete step, x_k and x_{k+1} represent the state of the system at steps k and $k + 1$, respectively, u_k represents a decision, also called *control*, and ω_k is a random variable influencing the evolution of the system, characterised by a certain probability function p_k.

The step index may refer to an increment over time or space, and the increment doesn't need to have fixed amplitude. Usually the initial state x_0 is fixed. All the variables defined take values in some determined interval or space; in particular, for a given state of the system x_k, the set of admissible controls $\mathcal{U}_k(x_k)$ is defined as the set containing all the possible decisions that can be taken at that stage. For instance, in financial problems, $\mathcal{U}_k(x_k)$ may be the set of all the possible assets that it is possible to buy or sell. In sailing applications, x_k can represent the state of a yacht on the race area (in this case x_k can be the vector constituted by the yacht's coordinates and the observed wind, assuming values on a limited subset of $\mathbb{R}^n$), u_k the CWA followed by the skipper, with $u_k \in \mathcal{U}_k \subseteq [0, 360)$, and ω_k the unknown wind evolution between step k and step $k + 1$. The position of the yacht at step $k + 1$ is then a function of those three variables.

A control, or a *policy*, is a finite sequence $U = u_0, \ldots, u_{N-1}$, where $u_k = u_k(x_k)$ is a function of the current state of the system, and all the $u_k \in \mathcal{U}_k(x_k)$ for all x_k. In the following, $\mathcal{U}$ will denote the set of the admissible policies. The aim of DP is to find an admissible policy $U = u_0, \ldots, u_{N-1}$ that minimises a cost function which can assume the

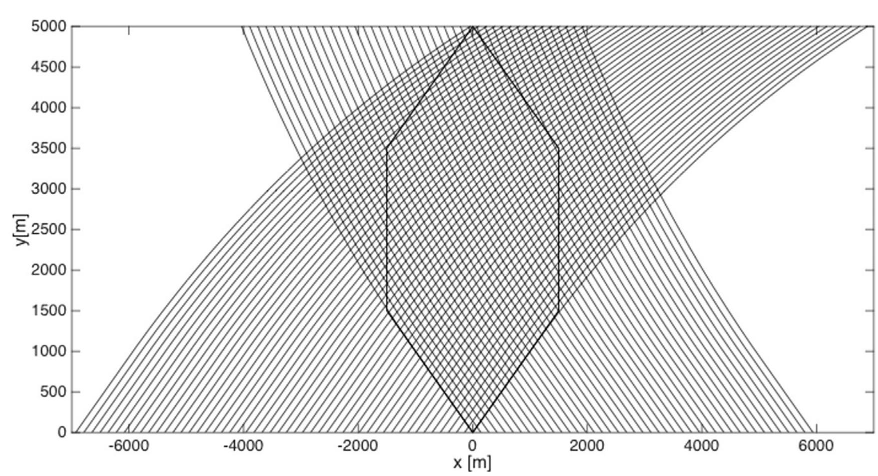

Fig. 7. Example of grid construction.

generic form as expressed in Eq. (3):

$$C(U, \omega) = \sum_{k=0}^{N} c_k(x_k, u_k(x_k), \omega_k) \tag{3}$$

where $\omega = [\omega_0, \ldots\omega_N]$, subject to the system constraint specified in Eq. (2).

In sailing, this cost corresponds to time:

$$T(U, \omega) = \sum_{k=0}^{N} t_k(x_k, u_k(x_k), \omega_k) \tag{4}$$

where t_k represents the time needed to sail from state x_k to state x_{k+1}.

For this class of problems, the cost function is known at every stage. Unfortunately, in practical applications (including sailing) the cost function is only known in terms of a probability distribution, and rather than minimising a cost the aim is to minimise its expected value. In this case, the stochastic version of dynamic programming is used. Going back to the general description, a solution for the problem is then a policy U^{opt} such that

$$\mathbb{E}(C(U^{opt})) = \min_{U\in\mathcal{U}} \mathbb{E}(C(U, \omega)) \tag{5}$$

We assume that the minimum in Eq. (5) is well defined. A discussion of this aspect can be found in Bertsekas (2007). According to the principle of optimality, an optimal solution has the property that, considering the subproblem starting at stage M, then the subpolicy $(U^{opt,M} = (u_M^{opt}, u_{M+1}^{opt}, \ldots, u_N^{opt})$ is optimal for that subproblem. This principle has its justification in the following derivation.

$$\begin{aligned}
\min_{U\in\mathcal{U}} \mathbb{E}(C(U, \omega)) &= \min_{u_0,\ldots,u_N} \mathbb{E} \sum_{k=0}^{N} c_k(x_k, u_k(x_k), \omega_t)\\
&= \min_{u_N}\left[\min_{u_0,\ldots,u_{N-1}} \mathbb{E} \sum_{k=0}^{N} c_k(x_k, u_k(x_k), \omega_t)\right]\\
&= \min_{u_N}\left[\min_{u_0,\ldots,u_{N-1}} \mathbb{E}c_N(x_N, u_N(x_N)) + \mathbb{E} \sum_{k=0}^{N-1} c_k(x_k, u_k(x_k), \omega_t)\right]\\
&= \min_{u_N}\left[\mathbb{E}c_N(x_N, u_N(x_N)) + \min_{u_0,\ldots,u_{N-1}} \mathbb{E} \sum_{k=0}^{N-1} c_k(x_k, u_k(x_k), \omega_t)\right]\\
&= \min_{u_N}[\mathbb{E}c_N(x_N, u_N(x_N))] + \min_{u_0,\ldots,u_{N-1}}\left[\mathbb{E} \sum_{k=0}^{N-1} c_k(x_k, u_k(x_k), \omega_t)\right]
\end{aligned} \tag{6}$$

This derivation leads to defining a recursive algorithm for finding an optimal policy, proceeding backwards and solving subsequently truncated sub-problems. The process described in Eq. (6) must not mislead the reader in thinking that the optimal solution can be found by minimising the sub-functions c_k. A locally optimal decision may in fact lead to a subsequent stage of the system from which only "bad decisions" can be taken. Although the expected values in Eq. (6) are linear, and therefore the sums defining the function C can be decomposed, the interdependence between stages of the problems is hidden in the sets of possible decisions u_k. The functions c_k represents the time needed to sail between two different points of the grid. If a tack is performed, a penalty is added, which is computed by assuming that the yacht will sail for 20 s at 80% of VMG. This procedure is based on the analysis of tacking patterns for the AC72 catamarans.

2.4. Risk management

A probability distribution for the future wind behaviour is included in the form of a Markov model, used to compute the expected values in Eq. (5). In a Markov model, a probability value for a future event is assigned, depending on the current state s_i (with $i = 1, \ldots, n$) of the n bins used to discretise the wind speed and direction. The conventional representation of these probabilities is through a square matrix, where each row represents a current state, each column a future state, and each cell contains the probability of jumping from the current state to a future state. All the rows of the matrix are normalised and represent a discrete probability distribution. The elements of the matrix are computed from recorded data using a maximum likelihood estimator. The generic element p_{ij} of the matrix is computed by counting how many times a value in the interval s_i is followed by one in the interval s_j in the recorded wind speed time series, divided by the number of occurrences of values in interval s_i.

This model is also the basis for the risk model. In fact, by using coherent risk measures, the transition matrix for the Markov process is multiplied by a transformation matrix which has the function of shifting the probabilities of favourable/unfavourable events. A complete description of this procedure can be found in Tagliaferri et al. (2014).

The optimal transformation is found among a set of matrices heuristically selected according to the following principles. A boat skipper who is losing will seek risk. If she adopts a minimum expected finish time strategy against another skipper who minimises his expected time to finish, then she will tend to make the same decisions (unless the boats see very different winds) and lose the race almost certainly. She will instead seek different wind conditions from the competitor, being optimistic about the possible advantageous wind shifts and assigning a higher probability to these outcomes (i.e. lifting shifts). Being optimistic about random outcomes increases risk, as well as incurring some loss in expected performance. A sailor who is losing will seek risk. This corresponds to increasing her confidence of a lifting wind shift while discounting the likelihood of a heading wind shift.

Transition matrices that can be used to represent the two attitudes are shown in Fig. 8. Advantageous shifts (cells below the diagonal when the skipper is to the left of the opposition, and cells above when on the right) happen with higher probability than in the risk-neutral case. The remaining probabilities in each row are reduced to add to one. The transition matrices are represented by using a grey scale, where darker colours represent higher probabilities.

This method was first introduced by Tagliaferri et al. (2014), where the two arbitrary matrices in Fig. 8 were used in combination to the routing algorithm proposed in Philpott and Mason (2001). This led to an improved probability of winning of 62% compared to a boat using a constant Markov Chain matrix. In this work, the risk model is applied to the above-described routing algorithm and the probability matrices are optimised in order to maximise the probability of winning. The two arbitrary matrices are repeatedly multiplied by a matrix that shifts the probabilities towards more extreme risk-seeking and risk-averse behaviours. The selected transformation is the one that leads to the highest probability of winning, and the optimal transformation is shown in Section 3.3.

2.5. Boats' interaction

A set of rules aimed at avoiding collisions between the boats and at respecting the racing rules are implemented. In particular:

1. If the two boats meet, then the boat on a port tack increases the TWA, passing behind the other boat.
2. A boat cannot tack if this leads to its track crossing the opponent's under a certain fixed safety distance.

The safety distance is defined noting that the boats are modelled as points. The longitudinal safety distance is 11 m, the side distance is 7 m. These safety distances are arbitrarily set according to the AC72 catamaran dimensions (22 m long, 14 m wide), and can be tuned to suit other yacht classes. The computations for the manoeuvre of bearing away and passing behind the opponent's boat is carried out by adding a node to the set of reachable nodes. This temporarily modifies the assumption that a boat always sails at maximum VMG. In the example shown in Fig. 9, the red yacht expects to meet the

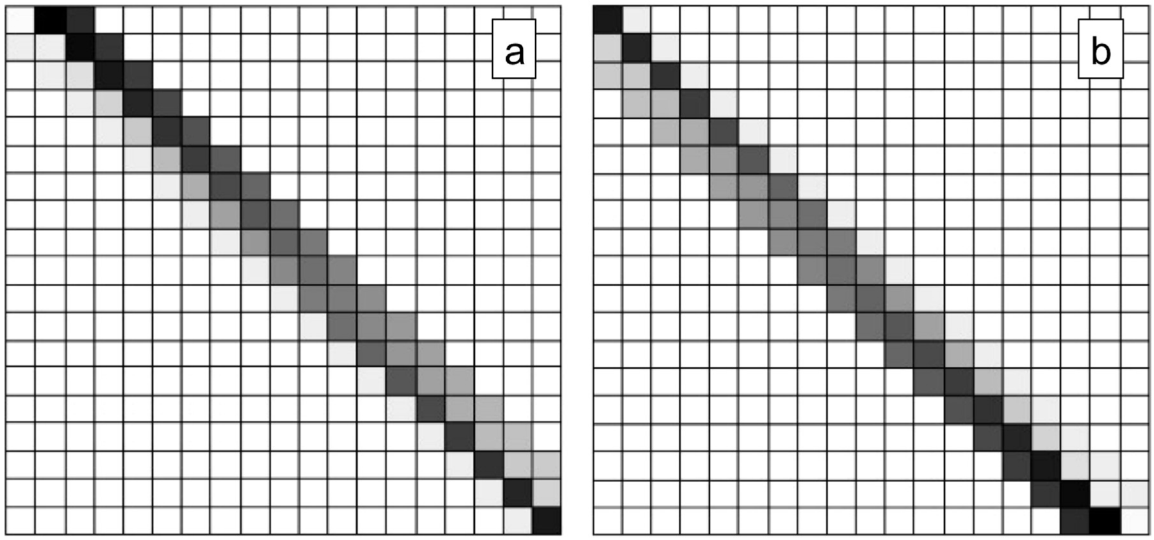

Fig. 8. Modified transition matrices for a risk-seeking skipper. Advantageous wind shifts occur with higher probability than disadvantageous ones. (a) Yacht on the left-hand side of competitor and (b) yacht on the right-hand side of competitor.

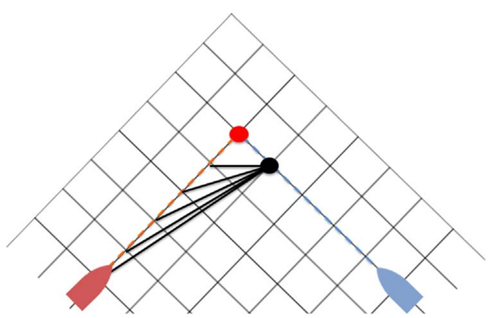

Fig. 9. Example of grid modification when two boats meet.

opponent at the node indicated by the red dot. The black node is therefore added to the set of the reachable points.

The model for physical interactions between the two yachts competing in a match race is based on the experimental results of wind tunnel tests presented in Richards et al. (2012) and Aubin (2013). Let BS^* be the boat speed in presence of an influencing yacht. The first step of the interpolation is to compute the AWS and AWA as a function of TWS, TWA, BS. Let A be the yacht of interest, B the influencing yacht, and d_{AB} be the vector distance between the two yachts. The corrected BS is then computed as in Eq. (7):

$$BS^*(TWS,\ TWA,\ BS,\ d_{AB}) = =BS - f_{int}(AWA(TWS,\ TWA,\ BS),\ AWS(TWS,\ TWA,\ BS)) \tag{7}$$

where f_{int} represent the interpolating polynomial for the BS variations based on the coefficients presented in Aubin (2013).

The opponent is assumed to follow a strategy aimed at minimising the expected time to complete the race. This means that he is expected to follow a reasonable route that depends on the forecast wind. To compute an optimal strategy, it may be possible to forecast the future position of the opponent, to take into account that the two boats might meet further in the future. However, in order to properly take into account such events, the computation of a probability distribution is required. In fact, let's assume that with the wind conditions forecast at the beginning of the race the two boats can compute an optimal strategy which will lead them to meet in proximity of the upwind mark. This event will actually happen with a probability which is equal to the probability that the wind realisation is exactly the one forecast at the beginning and that the boats actually follow the computed strategy. The further this event is in the future, the closer this probability is to zero. In the current software implementation, the future window is set at one minute, which is the time frame at which the wind forecast has an average error lower than 2° (Tagliaferri et al., 2015) and because a yacht is expected to perform not more than one tack in one minute. An important hypothesis, not necessarily corresponding to reality, is that no yacht will take a decision that leads to a higher expected time with the aim of slowing the other yacht down.

3. Results

This Section is organised as follows. First, results of a series of simulated races are presented, where a boat following the decisions suggested by the algorithm is racing a boat which has perfect knowledge of the future wind. Subsequently simulations are presented to demonstrate the effect of the two key aspects of novelty of this work: the use of an ANN forecast and of a risk management strategy. Finally, an example of a complete race between two boats that are both driven by the complete algorithm is presented Table 1.

3.1. ANN forecast vs perfect wind knowledge

The algorithm is tested using recorded wind scenarios from the 2013 edition of the America's Cup held in San Francisco. A strategy based on the ANN forecast is compared with a strategy which assumes perfect knowledge of the wind behaviour. The results of the simulated races are summarised in Table 2, where 13 upwind legs are simulated using the initial minutes of the last 13 races of the dataset. The data

Table 1
Simulated races with San Francisco wind dataset.

Race	1	2	3	4	5	6	7	8	9	10	11	12	13
T_{perf} [s]	603	743	618	743	642	818	597	661	654	712	748	684	697
T_{ANN} [s]	608	747	634	781	648	821	608	672	659	715	761	693	712
Difference [%]	0.8	0.5	2.5	4.8	0.9	0.4	1.8	1.6	0.7	0.4	0.3	1.2	2.1

Table 2
Simulated races with San Francisco wind dataset.

Race [deg]	1	2	3	4	5	6	7	8	9	10	11	12	13
T_{const} [s]	624	788	663	798	662	832	627	692	688	759	794	741	767
T_{ANN} [s]	608	747	634	781	648	821	608	672	659	715	761	693	712
Difference [%]	2.6	5.5	4.5	2.2	2.2	1.3	3.1	3.0	4.4	6.1	4.3	6.9	7.7

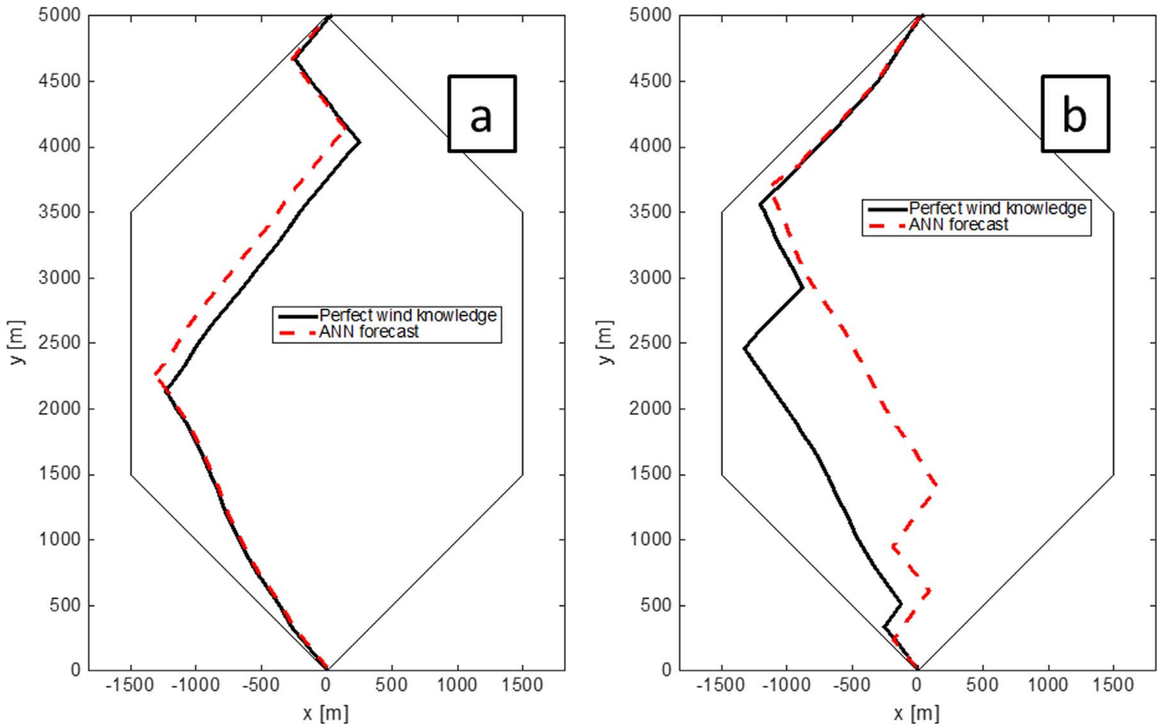

Fig. 10. Routes computed using forecast and assuming perfect wind knowledge for Race 2 (a) and Race 4 (b).

relative to the other races of the dataset was used for training the ANN. The second row (T_{perf}) shows the time to complete an upwind leg with perfect knowledge of the future wind velocity, while the third row (T_{ANN}) shows the time to complete an upwind leg using the ANN to forecast the future wind velocity. The average difference between the different times to completion between a boat with a perfect knowledge of the wind and a boat which uses the ANN forecast is 10.7 s, i.e 1.4% of the completion time using the ANN. A representative example is given in Fig. 10(a), corresponding to Race 2. The black trajectory is the one computed by the algorithm having perfect knowledge of the future wind, while the red dashed one is computed by the algorithm using the ANN forecast. In this example, the difference between the two strategies is limited to a slight delay of the second tack when the ANN forecast is used.

One of the worst cases is shown in Fig. 10(b), corresponding to Race 4. The ANN-based boat sails closer to the centre of the sailcourse at the beginning of the race due to the incorrect forecast of a wind shift. However, the error is soon recovered and in the final part of the race the two strategies become almost indistinguishable.

3.2. ANN forecast vs no forecast

In order to evaluate the benefit of the ANN forecast, which represent one of the two key advances compared to previous published results Dalang et al. (2014), the algorithm is tested computing the optimum course of a boat which assumes that the wind speed and wind direction do not change during the race. The same data set as in Section 3.1 is used. The results of the simulated races are summarised in Table 2, where T_{const} is the time to complete an upwind leg assuming a constant wind velocity. The average difference between the times to completion between a boat using the ANN for wind forecasting and a boat assuming a constant wind velocity is 29 s, i.e 4% of the completion time using the ANN.

3.3. Optimal risk model

The contribution of risk management is the second key element of novelty of this paper and, therefore, we investigate how this contributes to enhance the optimum strategy. We consider two boats racing each other. Boat A is following a risk-neutral strategy, meaning that no changes due to risk-management are added to the computation, while boat B is using the following procedure. At every step of the simulated race, if B is more than a time period T_{switch} behind A, she uses the risk-seeking, optimistic matrix for the relevant side of the course. If A is more than T_{switch} s behind B, then B uses the risk-averse, conservative matrix. For this case, $T_{switch} = 15$ s, which is the time needed to travel ca. 6 boat lengths. The time difference and the matrix transformations are arbitrarily fixed, and the results obtained confirm the results presented in Tagliaferri et al. (2014). Fig. 11(a) shows differences between the arrival times of boats A, which does not manages the risk, and B, which adapts the strategy depending on the relative position of the boats. When this time difference is positive, it means that A wins the race. Conversely, if the time difference is negative, B wins the race.

This set of results confirms that a risk seeking attitude can constitute an advantage for a skipper who is losing the race. In fact, even if when boat B loses she loses by more seconds than what A does when he loses, B wins 67% of the races. This is consistent with the 62%

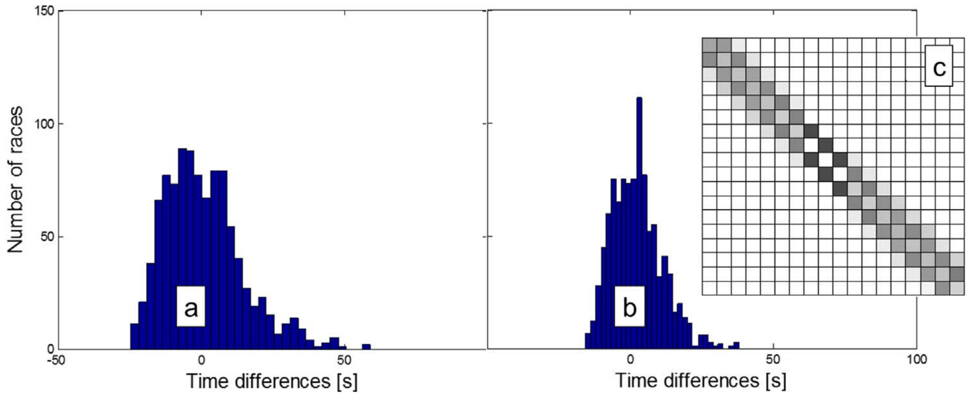

Fig. 11. Histograms of time differences for risk-neutral strategy vs optimistic-pessimistic combination (a) and optimal optimistic-pessimistic combination (b) based on an optimal processing matrix (c).

that was achieved by Tagliaferri et al. (2014), where the same matrices and T_{switch} were used in combination with a different routing model.

Here we optimise the amount of risk, i.e. how much the new Markov matrices differ from the original one, and the time at which the attitude is changed, i.e. the time difference between the two boats that triggers the attitude switch. The optimum risk attitude is investigated by comparing strategies obtained using matrices that have been multiplied for the transformation matrix multiple times. The best outcome is obtained with the use of the matrix shown in Fig. 11(c), and by setting T_{switch} = 10 s. The optimised risk model leads to the distribution in Fig. 11(b), which corresponds to a win for boat A in 74% of the cases, and is obtained by post-multiplying the risk-neutral matrix for the square of the original transformation. This optimisation was carried out over a limited set of possibilities, and it must be the subject of further research.

3.4. Example of complete race

In this Section a complete race between two boats that follow the optimum course and have an optimum management of risk is presented. The wind history from Race number 13 of the AC dataset is used. Fig. 12(a) shows the grid corresponding to the wind realisation. The grid shown is coarser than the one computed by the algorithm for clarity.

Both boats A (blue in Fig. 12) and B (red in Fig. 12) start on a starboard tack (sailing towards the left) and boat B is on the left of boat A, at a distance of 25 m (distances between the two boats are magnified in the figures for clarity). Fig. 12(a) shows the beginning of the race and the grid represents how the wind was forecast at that time. Both boats begin the race sailing on a starboard tack. As a significant increasing shift towards the right is forecast, the best strategy consists in approaching the mark on a starboard tack. However, it should be noted that the mark must be rounded clockwise and therefore, approaching the mark from the right will require an additional tack to round the mark. Boat A chooses to sail towards the left of the race area up to where, with only one tack, she can reach the right-hand-side layline. This would be the optimal choice for B as well in the absence of A. Unfortunately the more the wind shifts towards the right, the more B finds herself in her area of unfavourable aerodynamic influence (Fig. 12(b)). B cannot tack until A tacks because the two boats are too close. When eventually A tacks (Fig. 12(c)), B is free to tack as well, but she chooses to wait in order to perform the tack outside of the area where she would still be slowed down because of the presence of A.

In Fig. 12(d) both boats are initially sailing on a port tack, and A is leading. A reaches the layline and tacks to sail towards the mark. B adopts a risk-seeking behaviour, and instead of waiting to reach the layline as well she tacks hoping for a favourable (albeit unlikely) wind shift. Fig. 12(e) shows the end of the race. Although B has managed to avoid A to gain more advantage, she is still slightly behind.

Fig. 12(f) shows an alternative realisation for this example, where the strategy is computed without taking into account the presence of the opponent. In this case, boat B postpones the second tack until she reaches the racing area right boundary, but this then results in finding herself following the opponent and having no chances to overcome for whichever wind shift.

4. Conclusions

This paper presents a methodology for computing an optimal strategy for a sailing match race. This methodology improves the state of the art by combining for the first time a real-time wind forecasting and a risk management interaction between yachts. The aim of the method is to compute a strategy that improves the probability of winning the race with respect to strategies aimed at minimising the expected time to complete the race. As an example, for an upwind leg of the 34th America's Cup, the completion time of a boat which uses the proposed forecast is only 10.7 s longer than a boat which has perfect knowledge of the wind. The risk model is based on coherent risk measures in order to investigate whether a change in risk attitude can improve the probabilities of winning a race. An optimistic attitude is associated to a losing skipper, and a pessimistic, conservative one to a winning skipper. The risk model is optimised using parameters relative to the distance between boats and the anticipated future wind changes. The results suggest that there is a threshold defining the moment when it is advisable to seek more risk and that not always the risk-seeking and risk-averse behaviours correspond to optimistic and pessimistic anticipations on the wind.

The proposed method is implemented in a computer program capable to compute the optimum route in real-time during a race. The contribution of the various features analysed is quantified by comparing strategies computed with and without the ANN forecast and the risk management. Results show that taking into account the ANN-based forecast can reduce the expected time of almost 30 s in a simulated race leg, corresponding to a 5% improvement, and that using different risk attitudes can improve the probability of winning by up to 74%.

4.1. Future work

The main improvement that could benefit this work is a validation of the proposed algorithm with full-scale on-water data. This obviously presents some challenges in terms of costs and experiment design. The wind model could be improved to include spatial variations of wind speed and direction. This would not require major changes in the routing algorithm, as spatial variations can easily be implemented in the grid construction. The algorithm could include a tactical model for

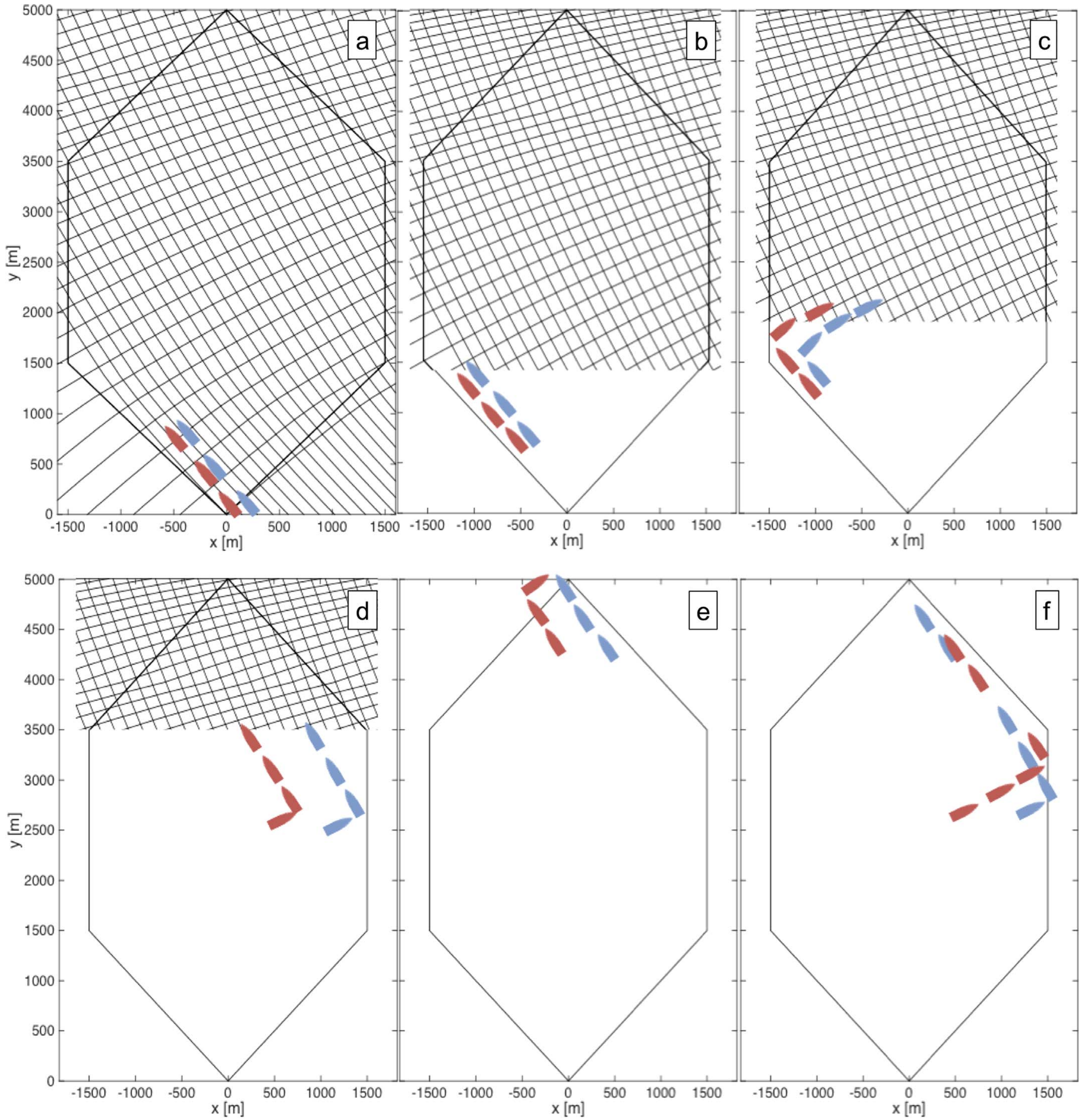

Fig. 12. Simulation results for an upwind leg.

round marking, which is a critical phase of any race, especially when yachts are close to each other. The challenges associated to mark rounding include a VPP which can capture the changes in boat speed while rounding a mark, and the racing rules regulating mark rounding should be implemented. However, if the scope of the program is to assist the sailors to take fast decision during the race, the value of including round marking is limited because it is unlikely that the program would be interrogated during such a complicated manoeuvre. The use of coherent risk measures has shown a great potential for risk modelling. However, due to the high dimensionality of the model to be optimised, further investigation needs to be carried out to define the optimal risk-seeking and risk-averse behaviour for a sailor.

Acknowledgements

This project has received funding from the European Union's Seventh Programme for research, technological development and demonstration under grant agreement No PIRSES-GA-2012-318924, and from the Royal Society of New Zealand for the UK-France-NZ collaboration project Sailing-Fluids.

References

Americas Cup Event Authority, 2013. AC34 Race Data Export Format Specification, 1.03.
Aubin, N., 2013. Wind tunnel Modelling of Aerodynamic Interference in a Fleet Race (Master's thesis). Ecole Centrale de Nantes and Universite de Nantes.
Bertsekas, D.P., 2007. Dynamic Programming and Optimal Control, vol. I. Athena Scientific Belmont, MA.
Clement-Nyns, K., Haesen, E., Driesen, J., 2010. The impact of charging plug-in hybrid electric vehicles on a residential distribution grid. IEEE Trans. Power Syst. 25 (1), 371–380.
Dalang, R.C., Dumas, F., Sardy, S., Morgenthaler, S., Vila, J., 2014. Stochastic optimization of sailing trajectories in an upwind regatta. J. Oper. Res. Soc. 66 (5), 807–821.
Ferguson, D.S.S., Elinas, P., 2011. A markov decision process model for strategic decision making in sailboat racing. Adv. Artif. Intell. 6657 LNAI, 110–121.
Hall, N., Potts, C., 2003. Supply chain scheduling: batching and delivery. Oper. Res. 51 (4), (566–584+674).
Philpott, A., 2005. Stochastic optimization in yacht racing. In: Ziemba, W., Wallace, S. (eds.), Applications Of Stochastic Programming, SIAM.
Philpott, A., Mason, A., 2001. Optimising yacht routes under uncertainty. In: Proceedings of the 15th Chesapeake Sailing Yacht Symposium, Annapolis, MD.
Philpott, A.B., Henderson, S.G., Teirney, D., 2004. A simulation model for predicting yacht match race outcomes. Oper. Res. 52 (1), 1–16.
Richards, P., Le Pelley, D., Jowett, D., Little, J., Detlefsen, O., 2012. Wind tunnel investigation of the interaction between two sailing yachts. In: Proceedings of the 18th Australasian Fluid Mechanics Conference, Launceston, Australia, 3–7 December 2012.
Sahinidis, N., 2004. Optimization under uncertainty: state-of-the-art and opportunities. Comput. Chem. Eng. 28 (6–7), 971–983.
Scarponi, M., Shenoi, R., Turnock, S., Conti, P., 2008. Interactions between yacht-crew systems and racing scenarios combining behavioral models with vpps 150, 11–18.
Spenkuch, T., Turnock, S., Scarponi, M., Shenoi, A., 2008. Lifting line method for modelling covering and blanketing effects for yacht fleet race simulation. In: Proceedings of the 3rd High Performance Yacht Design Conference 2008. Auckland, New Zealand. pp. 111–120.
Spenkuch, T., Turnock, S., Scarponi, M., Shenoi, A., 2010. Real time simulation of tacking yachts: how best to counter the advantage of an upwind yacht. In: Procedia Eng., vol. 2, pp. 3305–3310 (http://dx.doi.org/10.1016/j.proeng.2010.04.149).
Spenkuch, T., Turnock, S., Scarponi, M., Shenoi, R., 2011. Modelling multiple yacht sailing interactions between upwind sailing yachts. J. Mar. Sci. Technol. 16 (2), 115–128.
Tagliaferri, F., Philpott, A.B., Viola, I.M., Flay, R.G.J., 2014. On risk attitude and optimal yacht racing tactics. Ocean Eng. 90, 149–154.
Tagliaferri, F., Viola, I.M., Flay, R.G.J., 2015. Wind direction forecasting with artificial neural networks and support vector machines. Ocean Eng. 97, 65–73.