Optimising Yacht Routes under Uncertainty

Toby Allsopp
Andrew Mason www.esc.auckland.ac.nz/Mason
Andy Philpott www.esc.auckland.ac.nz/Philpott
Department of Engineering Science, University of Auckland, New Zealand
January 2000

Abstract

We consider the problem of finding a route that minimises the expected sailing time between two points on the ocean under uncertain weather conditions. This has applications in long distance offshore yacht racing. The uncertainty in the weather is modelled by a branching scenario tree that captures the serial correlation inherent in the evolution of weather systems over time. The stochastic solution method extends a deterministic dynamic programming approach to include the weather scenario as a state variable, yielding a stochastic dynamic programming algorithm. Careful attention to implementation details yields an approach that optimises with uncertainty while maintaining acceptable solution times on a PC. This paper summarises the work presented in Allsopp (1998).

Introduction

Weather routing is the process of determining an optimal route to sail between two given points, starting at a given time, based on predictions of the weather. The focus of this work is long, offshore yacht races, such as the Whitbread (now Volvo) "Round The World" race, in which an optimal route is one that takes the minimum time.

Many racing yacht skippers currently use deterministic yacht routing software packages to help with their route planning decisions. (The bibliography below gives a selection of routing papers.) This software performs what we refer to as *deterministic* weather routing in that it only considers one possibility for the weather and produces an optimal route based on the assumption that the predicted weather will occur exactly as forecast. However, races such as the Whitbread take place over large areas of ocean and can take months to complete. It is a very difficult problem to predict the weather for this length of time with any accuracy, so an approach based on a single forecast can yield solutions that perform badly when implemented under real weather conditions. In this work, we consider the possibility of different weather conditions evolving in the future, and produce routes that perform well under all of them, on average. This requires solving a *stochastic* weather routing problem.

Yacht Modelling

The speed of a yacht is determined by many factors, including wind strength, true wind angle, current, waves, and sail settings. These can be broken down into environmental factors, such as wind and current, and controllable factors, such as sail choice and trim. It is assumed that, for any given environmental conditions, the controllable factors will be set so as to maximise the speed of the yacht. Given this assumption, the speed at which a yacht sails is dependent upon the wind strength and the boat's heading relative to the wind. The maximum speed at which a yacht can sail for a given true wind angle and true wind speed is difficult, if not impossible, to determine analytically. This information is instead generated by numerical velocity prediction programs and on-the-water measurements and takes the form of a discrete set of triples giving true wind angle, true wind speed, and boat speed. Interpolating between known values can then give predicted maximum speeds for any true wind angle and true wind speed. Figure 1 is a typical 'polar plot' showing this data.

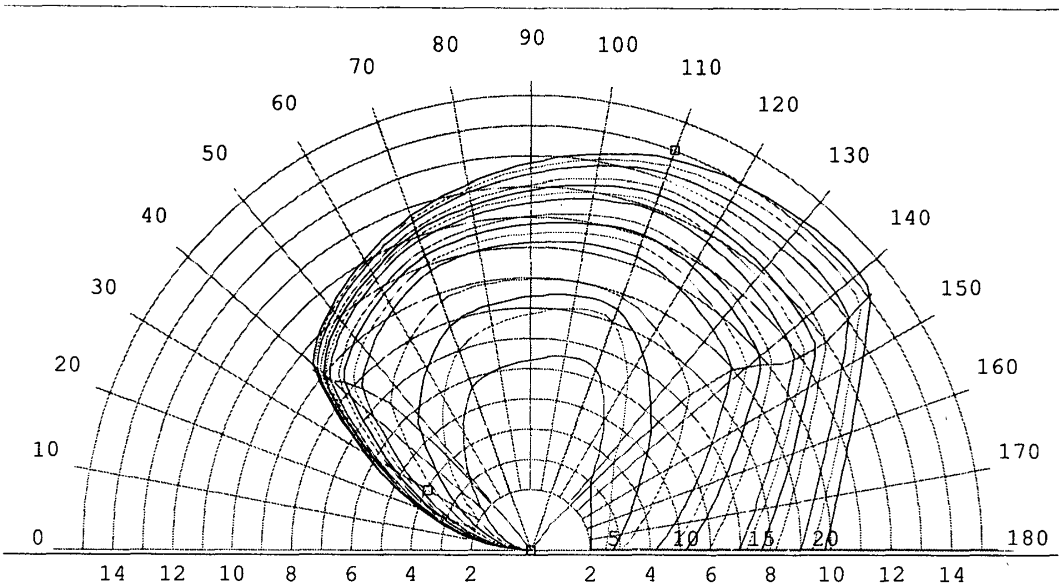

***Figure 1** Velocity prediction polar plot for true wind angles from 0° to 180° and true wind speeds from 0 to 20 knots. Note that the speed drops to zero as the boat heads closer to the wind. This plot was produced using the Kiwitech analysis software (KiwiTech 2000).*

Because the wind direction and speed varies with time t and location $\mathbf{x}$, we model it as a vector field $\mathbf{w}(\mathbf{x},t)$. Figure 2 shows the "wind barbs" view (Dey, 1998) sailers typically use when viewing wind fields. A typical weather forecast gives a set of discrete wind fields at evenly spaced time intervals. In a traditional deterministic forecast, there is only one wind field forecast provided for each time. To calculate predicted yacht speeds, the discrete wind fields are interpolated in time and space and then fed as input to a 'polar plot' lookup function.

In addition to the wind field, we also need data on the water currents $\mathbf{c}(\mathbf{x},t)$ as this impacts on the boat's effective velocity. It is relatively easy to get very accurate forecasts for currents, and so we assume that full deterministic current information is available. By combining wind and current information, we can now determine the boat's effective velocity at any point and time for any chosen heading.

Deterministic Routing

In deterministic routing we seek the route to sail from the start to the finish that minimises the time taken, assuming perfect knowledge of the weather. The parameters that define the problem are start and finish positions $\mathbf{x}_{\text{start}}$ and $\mathbf{x}_{\text{finish}}$, start time t_{start}, and wind $\mathbf{w}(\mathbf{x},t)$ and current $\mathbf{c}(\mathbf{x},t)$ at each location $\mathbf{x}$ and time t. If we discretise the race area into a set of directed arcs and nodes, we can then determine the time $c_{\text{arc}}(i, j, t(i))$ required to travel from node i to node j assuming a start time of $t(i)$ at node i. This time calculation takes into account the sideways motion of a yacht known as leeway which causes the boat direction to vary from its heading. It also allows for tacking as the boat heads to windward.

To determine the optimal path in our discretised network, we use the following dynamic programming recursion. This recursion includes an explicit (discretised) time state to handle the start-time-dependent travel times.

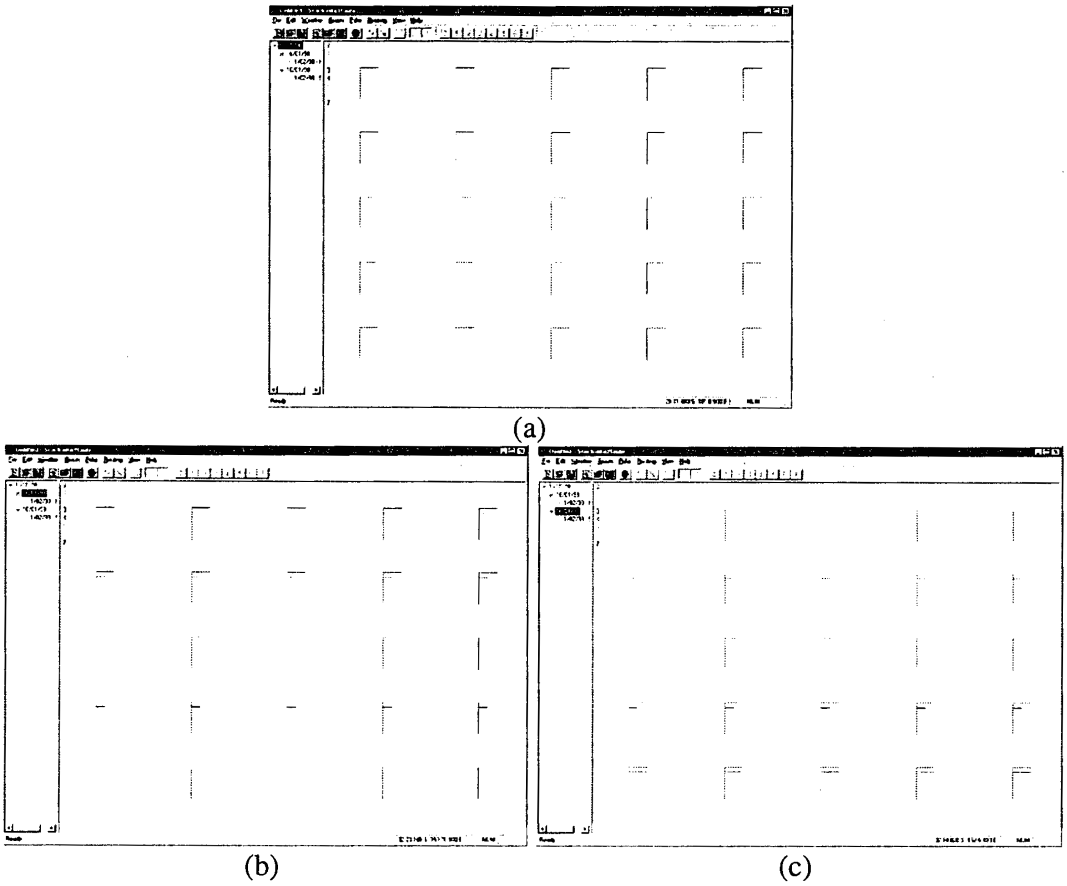

***Figure 2** Example of a wind field demonstrating how the weather evolves over time from (a) through (b) to (c). The lines indicate wind direction; the wind speed is shown by the number and size of the barbs on the lines.*

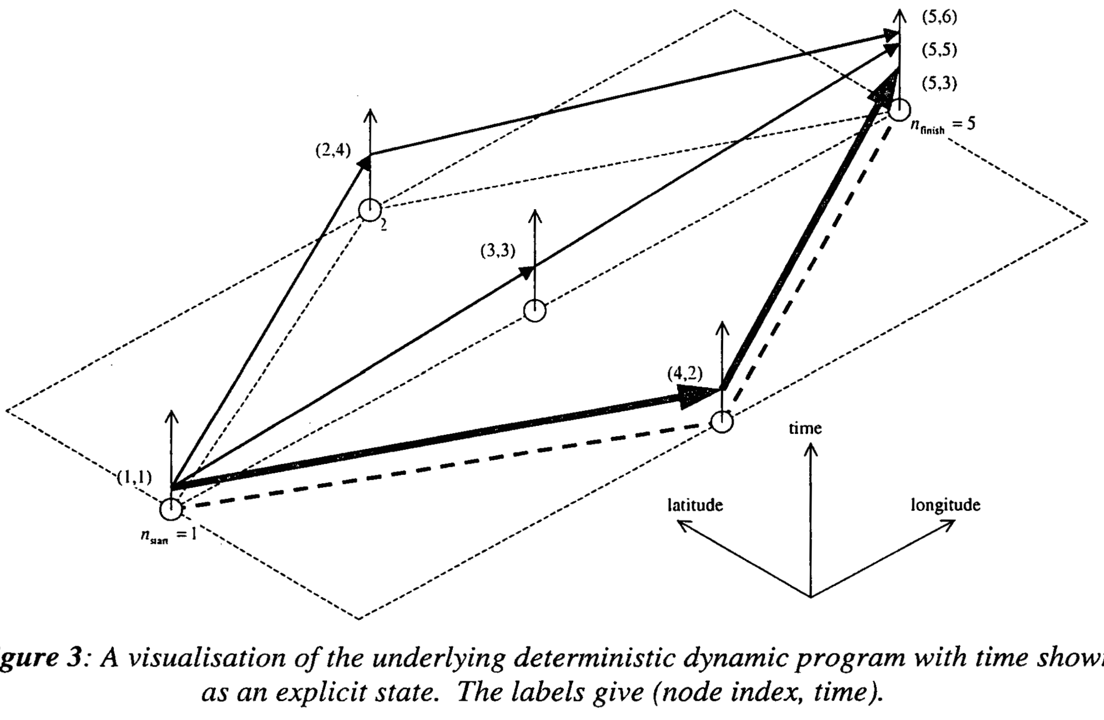

***Figure 3**: A visualisation of the underlying deterministic dynamic program with time shown as an explicit state. The labels give (node index, time).*

$$f^*(i,t)=\begin{cases}0, & i=n_{\text{finish}}\\ \min\limits_{j\in\Gamma_i}\left[c_{\text{arc}}(i,j,t)+f^*(j,t+c_{\text{arc}}(i,j,t))\right], & \text{otherwise}\end{cases}$$
$$j^*(i,t)=\arg\min_{j\in\Gamma_i}\left[c_{\text{arc}}(i,j,t)+f^*(j,t+c_{\text{arc}}(i,j,t))\right],\quad i\neq n_{\text{finish}} \tag{1}$$

In this expression, Γ_i is the set of successors of node i, $f^*(i,t)$ is the time taken for the optimal sequence of decisions from the node-time pair (i,t) to the finish node and $j^*(i,t)$ is the successor of i on the optimal path when arriving at node i at time t. The quantity being minimised at each node-time pair (i,t) is the time to reach the finish node, n_{finish}. The state-space for the underlying shortest path in this dynamic program includes an explicit time dimension as is illustrated in Figure 3.

Weather Modelling

To accommodate uncertainty in the weather, we need a model that describes how this uncertainty behaves. The simplest stochastic model treats the wind speed and direction at a point as a random variable with a known distribution. However, this model incorporates no information about what has happened in the past when, in fact, weather is strongly serially correlated. To overcome this, we use a discretised branching scenario model in which the weather unfolds according to an underlying tree. A branch in the scenario tree structure represents the arrival of new information and a node the actual information. A possible representation of weather using this structure is shown in Figure 4. The node at time t_1 represents the current weather situation. The two nodes at time t_2 represent two possible outcomes for the wind direction and/or strength at that time; which of these possibilities eventuates is assumed to be revealed at time t_2 according to some known set of probabilities. At some later time t_3 each t_2 weather outcome itself splits into two different weather outcomes, again with known associated probabilities. So called ‘ensemble’ forecasts of this type are now becoming available from a variety of sources.

We note that at time t_1 there is only one possibility for the wind, that is all four scenarios are indistinguishable. We say that all 4 scenarios belong to the same *bundle* {1, 2, 3, 4}. At time t_2 this bundle of four scenarios branches into two bundles of two scenarios each and at time t_3 all four scenarios are distinguishable. We define $\mathbf{B}(t)$ to be the set of bundles at time t. In the example above, $\mathbf{B}(t)=\{\{1,2,3,4\}\}$ for $t_1\le t<t_2$, $\mathbf{B}(t)=\{\{1,2\},\{3,4\}\}$ for $t_2\le t<t_3$, and $\mathbf{B}(t)=\{\{1\},\{2\},\{3\},\{4\}\}$ for $t\ge t_3$. We also define $B(s,t)$ to be the bundle to which scenario s

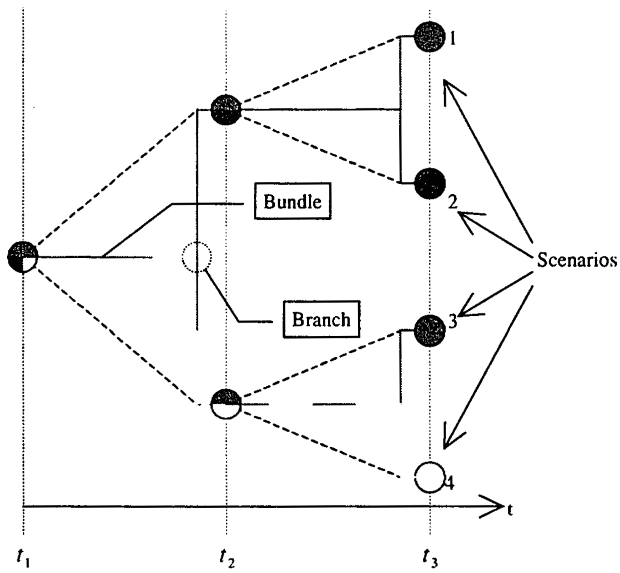

***Figure 4** Example scenario structure with three branches and four scenarios. Where scenarios are identical, they are treated as one ‘bundle’.*

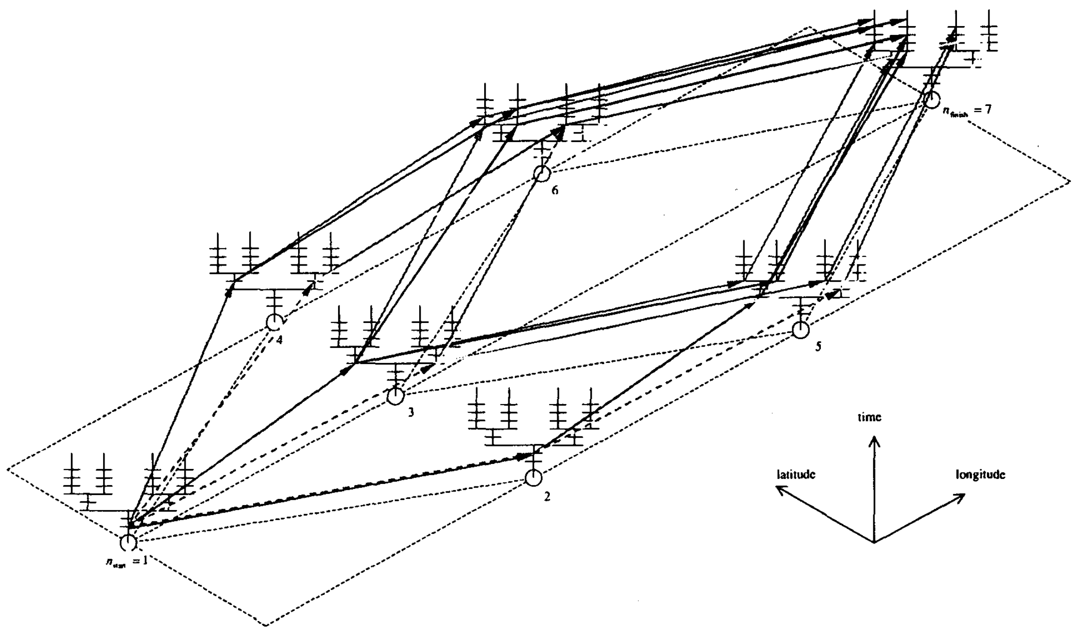

Figure 5 All possible arcs and node-time-scenario triples for an example with four scenarios.

belongs at a certain time, t. In the example above, $B(1,t)=\{1,2\}$ for $t_1 \le t < t_2$.

Stochastic Solution Method

An important feature of stochastic solutions is that they must be 'non-anticipative' in the sense that the route up to some time t_i must not depend upon scenario information that becomes available after time t_i. One approach, known as progressive hedging (Rockafellar and Wets, 1991), is to initially relax this requirement, build an optimal path for each scenario, and then modify these paths so that they satisfy the non-anticipative requirements. We take an alternative approach in which stochastic dynamic programming (eg see Bertsekas, 1995) is used to solve the problem.

A number of changes need to be made to the earlier deterministic dynamic program to incorporate the stochastic elements. In order to accommodate the uncertainty, it is necessary to know, in addition to the position and time, the location in the scenario tree. This requires the addition of another state variable that identifies the branches that have been taken. It is convenient to identify the branch history by specifying a scenario. This is depicted in Figure 5, where the tree at each node shows discretised time periods within the scenario tree of Figure 4. For this problem, we use the actual scenario, and not the scenario bundle, as the state variable; this allows correct interpolation of the weather between successive wind fields. (Note that this allows for the weather to vary and impact on the boat speed before that becomes apparent to the decision maker as the realisation of a scenario.)

From the node-time-scenario triple (i,t,s) we know that the only scenarios that can possibly eventuate are those in the same bundle as s at time t, that is $s \in B(s,t)$, where $B(s,t)$ is the bundle of scenarios that contains scenario s at time t.. Thus, we want to choose the successor to node i, $j^*(i,t,s)$, that gives the least expected time to go, where the (conditional) expectation is taken over the scenarios $s \in B(s,t)$. Let $t^*(i,j,t,s)$ be the expected optimal time to go from node i, at time t, in scenario s, when passing through node j next, given by:

$$t^*(i,j,t,s) = \sum_{s\in B(s,t)} \frac{p_s}{p_{B(s,t)}} \begin{bmatrix} c_{\text{arc}}(i,j,t,s)+ \\ f^*(j,t+c_{\text{arc}}(i,j,t,s),s) \end{bmatrix}, \tag{2}$$

where $p_{B(s,t)} = \sum_{s\in B(s,t)} p_s$ is the probability of the bundle containing scenario s at time t and $p_s / p_{B(s,t)}$ is the conditional probability of scenario s at time t given that it is known that one of the scenarios in the bundle containing scenario s will eventuate. Modifying (1) to choose the successor j^* to node i that minimises the expected optimal time to go gives the stochastic forward-looking recursion:

$$\begin{aligned} f^*(i,t,s) &= \begin{cases} 0, & i = n_{\text{finish}} \\ \min_{j\in\Gamma_i} t^*(i,j,t,s), & \text{otherwise} \end{cases} . \\ j^*(i,t,s) &= \arg\min_{j\in\Gamma_i} t^*(i,j,t,s), \quad i \neq n_{\text{finish}} \end{aligned} \tag{3}$$

This recursion forms the heart of the dynamic program we use to obtain our stochastic routes.

Implementation

If this software is to be used on board during races, it is important that solution times are kept to a minimum. This is particularly difficult for a dynamic program such as ours which has a large number of states and stages. There are a number of techniques we use to reduce running times. Firstly, we sort the successors of each node by distance from that node and use bounding to achieve speed-ups of approximately 60%. We also exclude successors based on angle spacing and distance respectively. Using these together achieves speed-ups of approximately 90%. We also use an adaptation of a Pijavskii-Shubert algorithm (Pijavskii, 1972; Shubert, 1972) for global optimisation to achieve a small further speed-up.

Results

To demonstrate our results, we compare the optimal solutions produced by the deterministic and stochastic solution methods under uncertain weather conditions. Consider the example of uncertain weather shown in Figure 6. Here the initial wind is a uniform 8 knot northerly. The weather then branches into two scenarios, one in which the wind is stronger towards the north and the other in which it is stronger toward the south. The probabilities of the two scenario are 0.75 and 0.25, respectively. In both cases the wind direction remains from the north. If we knew which of these two scenarios was going to eventuate, we could use a deterministic routing algorithm to obtain a minimal time route. However, such a route will perform badly if the other scenario eventuates. Figure 7 shows four routes for these weather conditions. The top-most and bottom-most routes are the deterministic minimal time solutions for scenarios 1 and 2, respectively. The splitting central routes are the stochastic solution. Note that they coincide at first and then diverge once it is known which scenario will eventuate.

The routes shown here are intuitive; the deterministic routes head towards the strongest wind, that is the route assuming scenario 1 takes a northerly course and that assuming scenario 2 takes a southerly course. The stochastic solution heads slightly north of the straight-line minimal distance route initially. When it becomes apparent which scenario is eventuating, the route is in a position to take advantage of either scenario. The position is better for scenario 1

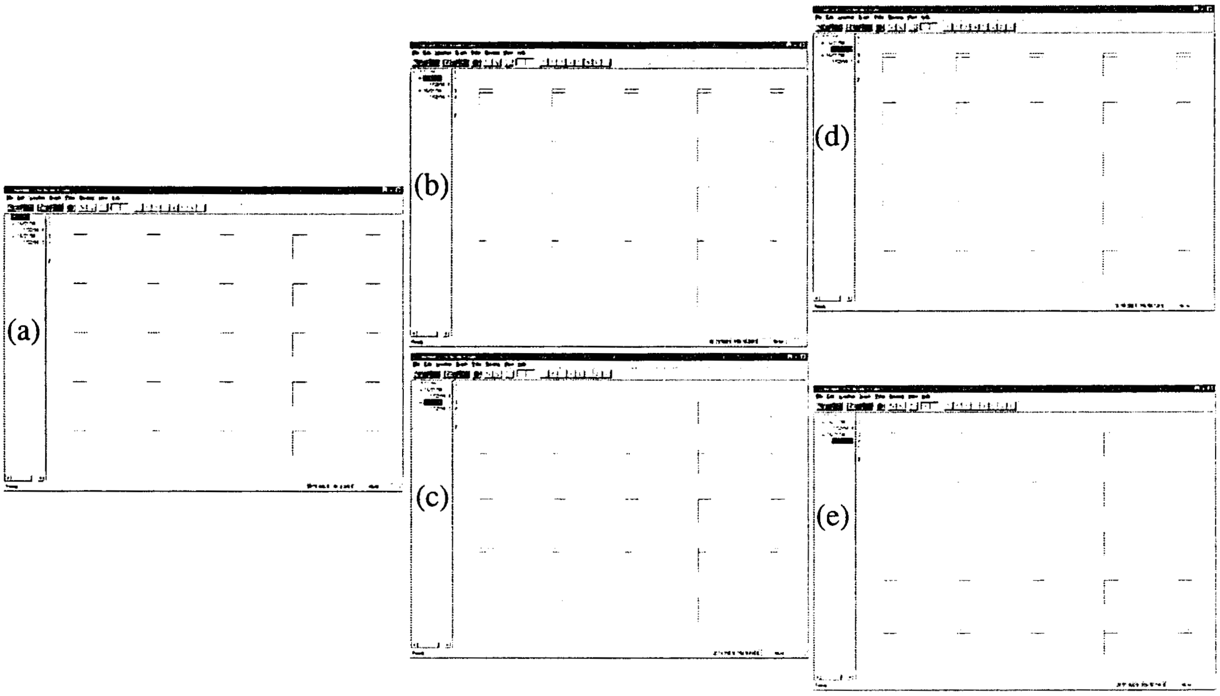

Figure 6 *Example of the evolution of the weather over time in two scenarios. Scenario 1 proceeds (a) to (b) to (d) while scenario 2 proceeds (a) to (c) to (e). From (a), the weather evolves to (b) (scenario 1) with probability 0.75 and to (d) (scenario 2) with probability 0.25.*

because this scenario is more likely to eventuate than scenario 2. The expected travel time for each of these routes is shown in Table 1.

The expected optimal route time shown in this table is simply the expectation of the optimal route times under each individual scenario. We can see that the stochastic solution has the best expected time, at just over two hours less than the next best, the deterministic solution assuming scenario 1. The expected time for the deterministic solution assuming scenario 2 is another four hours behind this. The reason the deterministic solution assuming scenario 1 does so much better than that assuming scenario 2 is that scenario 1 is more likely than scenario 2 and therefore has more influence on the expected time.

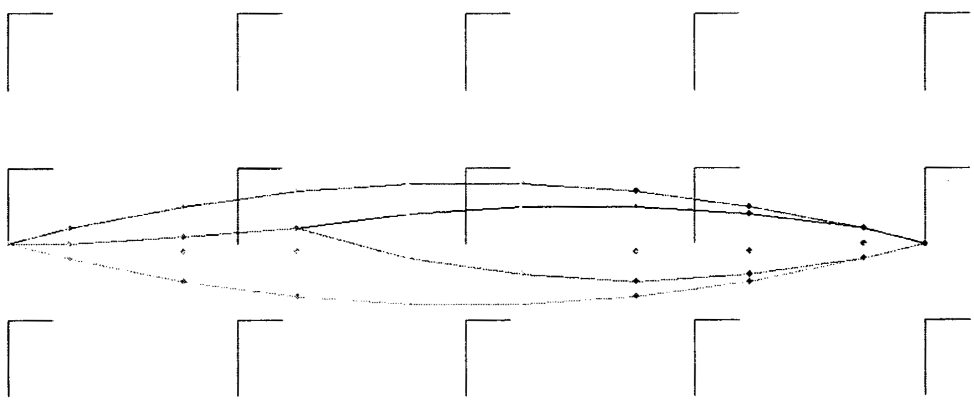

Figure 7 *Deterministic (top-most and bottom-most), and stochastic (central splitting) solutions.*

Description	Optimal route time under scenario 1	Optimal route time under scenario 2	Expected optimal route time
Deterministic scenario 1 (upper-most route)	4days 04:59:13	4days 18:21:04	4days 08:19:41
Deterministic scenario 2 (bottom-most route)	4days 15:38:23	4days 03:17:20	4days 12:33:07
Stochastic (splitting route)	4days 05:42:42	4days 07:06:27	4days 06:03:38

Table 1 *Optimal route times for the routes shown in Figure 7.*

Conclusions

We have developed a technique for computing minimal expected time routes under uncertain weather conditions. To accomplish this we have discretised both the area of ocean and the time over which we consider sailing and applied a stochastic dynamic programming algorithm. We have modelled the uncertainty in the weather by a branching scenario tree in order to capture the serial correlation that exists due to the way weather evolves over time. Careful attention to implementation details has reduced running times to about 5-10 minutes, making this system suitable for its intended application of on-yacht use during racing.

References and Bibliography

Allsopp, T. (1998), Stochastic Weather Routing For Sailing Vessels, A thesis submitted in partial fulfilment of the requirements for the degree of Master of Engineering, Department of Engineering Science, The University of Auckland, New Zealand

Bijlsma, S. J. (1975) *On Minimal-Time Ship Routing*. Koninklijk Nederlands Meteorologisch Instituut, Mededelingen en verhandelingen, No. 94

Bertsekas, D. P., (1995) *Dynamic Programming and Optimal Control* Volumes 1 and 2, Athena Scientific, Belmont, Mass.

Carryer, M. (1995) *Optimal yacht routing strategies using progressive hedging*.Fourth Year Project, Department of Engineering Science, University of Auckland, New Zealand

Dey, C. H. (1998) *NCEP Office Note 388 – GRIB (Edition 1)* [online]. Avaliable: ftp://nic.fb4.noaa.gov/pub/nws/nmc/docs/gribed1/ (August 24, 1998)

KiwiTech (2000), now Raytheon Marine Ltd; see http://www.raymarine.com/

Papadakis, N. A., & Perakis, A. N, (1990) Deterministic minimal time vessel routing. *Operations Research, 38*(3), 426-438

Pijavskii, S. A. (1972) An algorithm for finding the absolute extremum of a function. *USSR Computational Mathematics and Physics, 12*, 57-67

Rockafellar, R. T., Wets, R. J.-B. (1991), 'Scenarios and policy aggregation in optimization under uncertainty', *Mathematics of Operations Research* 16(1), 1-29.

Shubert, B. O. (1972) A sequential method seeking the global maximum of a function. *SIAM Journal of Numerical Analysis, 9*, 379-388

Zoppoli, R. (1972) Minimum-time routing as an N-stage decision process. *Journal of Applied Meteorology, 11*, 429-435