Chapter 11
A Biomimetic SANET Middleware Infrastructure for Guiding and Maneuvering Autonomous Land-Yacht Vessels

Christopher Chiu and Zenon Chaczko

Abstract. This chapter elaborates an approach of guiding land-yachts according to a predefined maneuvering strategy. The simulation of the path as well as the controller scheme is simulated for a sailing strategy (tacking), where the scheme of the resultant path and the sailing mechanism is driven using Sensor-Actor Network (SANET) middleware infrastructure. The addition of obstacle avoidance and detection heuristics aids in the guiding process. By incorporating SANETs in the sailing craft, a range of sensory mechanisms can be employed to monitor and handle local obstacles in an effective manner, while data collated from the sensory environment can be transmitted to a base station node for monitoring conditions from a holistic dimension.

Keywords: Biomimetic Methodologies, Spring Tensor Analysis, Software Middleware, Autonomous Sailing Systems, Wireless Sensor Networks (WSN), Sensor Actor Networks (SANET).

11.1 Introduction

Autonomous vehicles using flow-based propulsion have been examined by Jouffroy [8] as a means to provide efficient navigation and control in variable conditions. A particular concern is that the control strategies necessary to assess stability and performance [8] cannot be solved using traditional Artificial-Intelligence techniques. Given the progression of technology advances in Sensor Actor Network (SANET) technologies [12], a distributed control approach using SANETS combined with sailing kinematics for path coordination and neural networks for obstacle avoidance can be embedded within the embedded system for monitoring and control [2].

In this chapter, a multi-paradigm approach is employed to monitor the environmental surrounds of the vessel, while control aspects assist in sailing maneuvers and

---

Christopher Chiu · Zenon Chaczko
Faculty of Engineering & IT, University of Technology, Sydney, Australia
e-mail: `{Christopher.Chiu,Zenon.Chaczko}@uts.edu.au`

trajectory planning. The use of a unified middleware framework to implement the controller functionality enables a uniform methodology to coordinate multiple sailing vessels; as well as provide localized awareness of vessels in the vicinity of a localized regional zone [2]. As the vessel navigates to its desired destination, real-time feedback of obstacles allows the SANET infrastructure to map the global terrain for the benefit of all vessels active in the environment [5]. This distributed approach to obtain a complete environmental map of the vessel's surroundings, provides redundancy in the situation where Global Positioning System (GPS) navigation signals are unavailable, or in the instance of outdated or unobtainable terrain maps of the region [2].

11.2 A Sailing Vessel as a Actor-Based Process

The actor-based paradigm, as elaborated by Rao and Georgeff [13], has been employed to incorporate the rule-sets for the sailing path maneuvers, and the tensor-analysis heuristics to incorporate obstacle avoidance [11]. In this system design, each software actor is associated with the sensors incorporated into the sailing craft (including the anemometer, GPS navigation sensor and ultrasonic detection sensors), and the motor controller mechanisms (to regulate steering and manipulate sail orientation). A core actor will represent the craft itself, such that a hierarchical structure of actor responsibilities exists within the sailing vessel model. The actor-based characteristics defined in the model are elaborated as follows [2]:

- **Centralized gateway:** To aggregate statistical data from all sailing crafts in a predefined region, and coordinate scheduled activities by the administrator or coordinator user:
  - **Weather forecasting:** Weather monitoring data collected from all sailing craft actors, along with meteorological data imported from external data sources; assist in forecasting potential low-pressure formations that would affect trajectory path planning.
  - **Terrain mapping:** Obstacle data accumulated from the crafts can be geotagged in a global map, thus allowing for the new mapping of undefined terrain or refreshing existing terrain maps for future expeditions.
- **Sailing vessel:** To incorporate various sensor data via a multi-modal approach [8, 9], communicate with neighboring vessels in a peer-to-peer fashion, and centralized gateway exchanges for collaborative data exchange [2]:
  - **Anemometer and barometer:** Wind direction and Speed distribution monitoring, along with atmospheric pressure measurement will monitor localized weather patterns and predict short-term weather forecasts.
  - **Global positioning system:** Satellite Navigation system to provide accurate positioning of vessels relative on Earth's coordinates; with sailing vessels incorporating GPS sensors are designated as anchor node vessels.

- **Ultrasonic and infra-red sensors:** For close-proximity obstacle detection so preemptive path navigation can be established; additionally vessels can incorporate digital video capture with embedded machine vision algorithms for enhanced obstacle recognition.

Harnessing the land yacht control dynamics as elaborated by Jouffroy [8], the purpose of the chapter is to examine the coordination of multiple craft irrespective of the environmental terrain. The coordination and control of autonomous systems have its origins in autonomous robotic control [12, 14], with the customization in place to account for land yacht kinematic parameters. The main study of investigation is to determine what optimum method can be used to steer a fleet of craft in uncertain conditions.

11.3 Heuristic Analysis for Autonomous Sailing Craft

Land-yachts by their nature need to perform complex maneuvers due in part to their main method of propulsion being a sail [6, 15]. This is of particular consequence especially when the craft needs to travel in the direction of the wind, because the vehicle quickly loses propulsion [9]. This no-sailing angular zone is resolved by maneuvering in a zig-zagged direction of the wind, where each angular turn of direction in the zig-zag formation is known as a tack [7]. The motion planning required for a tacking maneuver is important, because of the potential for competing interests between sailing in a safe manner to maintain propulsion, and trying to maintain a minimum trajectory [2].

The reason is because to ensure that a direct sailing path is made to the destination when traveling in the direction of the wind, the tack's angle should be made at the angle closest to the no-sailing zone [10, 16]. In other words, the zig-zag sailing formation should be as narrow as possible, but can never achieve an absolute linear path. However, tacking closer to the no-sailing zone angle also leads to a higher likelihood of propulsion loss, and thus stalling the craft. Therefore, an optimum balance must be made between achieving an optimum sailing trajectory, while simplifying the number of tacking maneuvers [8].

SANET middleware-based heuristics can be applied when a fleet of autonomously controlled craft need to be coordinated effectively, with each craft aware of its surroundings and neighboring vessels utilizing wireless radio-frequency communication [5, 10]. To develop an effective mechanism for trajectory mapping while simplifying tacking maneuvers, a trajectory mapping for sailing has been developed to linearize the trajectory itself. As shown in Figure 18.1 (*Left*), traditional trajectory mapping methods rely on local perception to maneuver around its neighbours, such as neural networks. Such a trajectory path places emphasis on intelligence at the source, without realizing the bigger picture: that the path can be potentially complex and meandering. By taking a SANET-based approach to the solution, we treat the environment as a global perception in Figure 18.1 (*Right*). This means that the

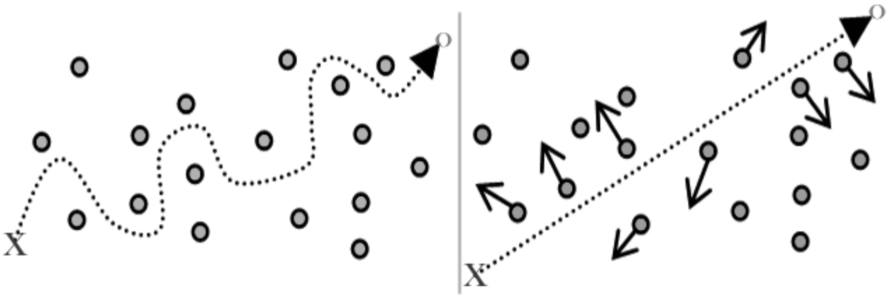


Fig. 11.1. Trajectory Mapping of the Destination Point using: (*Left*) Localized Awareness; (*Right*) Global Perception

neighboring vessels coordinate together to ensure a linearized path can be established by the sailing vessel, and final trajectory is executed as a result.

11.3.1 Application of Tensor Analysis for Trajectory Mapping

The method to determine a globalized trajectory mapping method has been inspired by protein fluctuation dynamics. As discussed in depth by Lin and Song [11], the premise of the spring tensor model is to determine conformational changes in proteins using second-order partial derivatives (Hessian matrices). Conformational change is the transition of macro-molecular structures in proteins as a result in a change of acidity, temperature, voltages and so forth. The spring tensor model is an enhancement of anisotropic modeling and Gaussian modeling methods, as while the former determines fluctuations of an atom's direction, the latter is better at determining the prediction of magnitudes of direction [11]. Thus by combining the two methodologies, the spring tensor model can be applied to a coordinated land-yacht system as follows:

- **Anisotropic modeling:**
  The determination of conformational variation or fluctuation in *direction* between elements.
  - *Adaptation:* This is suitable for determining how the interactions between neighboring land-yachts will result in the degree of directional fluctuation. The variation of potential direction will indicate what possible directions a vessel can travel if it is in proximity with a neighboring vessel.
  - *Meaning:* Smaller anisotropic values indicate a smaller potential to alter the direction, while larger values indicate a larger potential to alter the direction.

- **Gaussian modeling:**
  The determination of conformational variation or fluctuation in *magnitude* between elements.
  - *Adaptation:* This is appropriate to ascertain how interactions between land-yachts will result in the magnitude or total range of the fluctuation. The variation of potential magnitude indicates the possible maximum range the vessel can travel towards if it is in proximity with a neighboring vessel.
  - *Meaning:* Smaller magnitudes values indicate a smaller potential to alter the distance, while larger magnitudes indicate a larger potential to alter the distance of the vessel.

In the application of the Spring Tensor Model (STEM) by Lin and Song, the Go-like potential [2] is considered to take the non-native and native conformations as the input data [11]; for this instance these values are the difference in the land-yacht's Cartesian coordinates between time *n* and time *n+1*. These terms are divided into four terms as shown in the expression in Figure 18.2 (*Left*). The sum of the first term, V1, determines the radius of connectivity; the sum of the second term, V2, determines the bond angle; the sum of the third term, V3, determines the torsional interaction. What is of interest is examining the final term, V4, that determines non-local or global interactions.

$Y(X,Y_0) = \sum_{V1Bonds} + \sum_{V2Angles} + \sum_{V3Dihedral} + \sum_{V4NonLocal}$	(1.) The V(X,Y_0) values are sum of radial lengths, bonding angles and dihedral angles of consecutive objects for i and j. The non-local values are used.
$V_4 = \varepsilon\left[5\left(\frac{r_{0,ij}}{r_{ij}}\right)^{12} - 6\left(\frac{r_{0,ij}}{r_{ij}}\right)^{10}\right]$	(2.) The final non-local contact term is derived from the Go-like potential as discussed in Lin and Song's theoretical work.
$V_4 = -\varepsilon + \frac{120\varepsilon}{r_{o,ij}^2}\left(r_{ij} - r_{0,ij}\right)^2$	(3.) The Taylor expansion of the initial non-local contact term yields the following equation, where r_ij and r_0,ij are the consecutive long-term values for objects i and j.
$\frac{\delta^2 V_4}{\delta X_i \delta Y_j} = -\frac{240\varepsilon}{r_{o,ij}^2}\left(X_j - X_i\right)\left(Y_j - Y_i\right)/r_{ij}^2$	(4.) As the focus is on the equilibrium fluctuations, r_ijis equal or approximately equal to r_0,ijat equilibrium; thus the derivatives of V_4can be further simplified as shown.

Fig. 11.2. The Go-like Potential Expression and its Taylor Expansion of the Non-Local Contact Term [11]

The STEM model's fourth term is of interest as it examines the global interactions of the elements. The final term, examined in Figure 18.2 (*Right*), is shown with its Taylor expansion form. The final non-local derivation is adopted from Lin and Song's calculations, which is used as a point of reference in this research project [11]. Using the parameters stated by Clementi [2], the value of epsilon (ε) adopted is

0.36 as per conformation observations of macro-molecular protein structures using X-ray crystallography.

Although beyond the scope of this current research, the future task is to obtain a unique value of epsilon and Taylor expansion parameters suitable for a wireless SANET environment in land-yacht sailing contexts. These values are obtained through experimental observation of the sailing vessels as they interact in the physical environment, and determining the maximum thresholds of the land-yacht's direction and magnitude to make a tacking maneuver in different environmental conditions (i.e. finding the difference between calm versus stormy weather forecasts). It is noted that there is no fixed parameter values that can be used for all land-yachts, although a close approximation can be made for classes or category types of land-yachts that is sufficient for the majority of results. To give an example of this property, Class 2 land-yachts will have a different tacking angle to Class 5 land-yachts due to their differentiation in size, weight and the total surface area of the sail).

11.3.2 Developmental Approach and Methodology

For the design prototype, the STEM Hessian is calculated using the experimental methodology depicted in Figure 18.3; the excerpt source code is provided in Figure 18.4. The purpose of the experimental approach is to examine how the Tensor Analysis heuristic can be used in conjunction with swarming algorithms to optimize global trajectory mapping for the sailing craft.

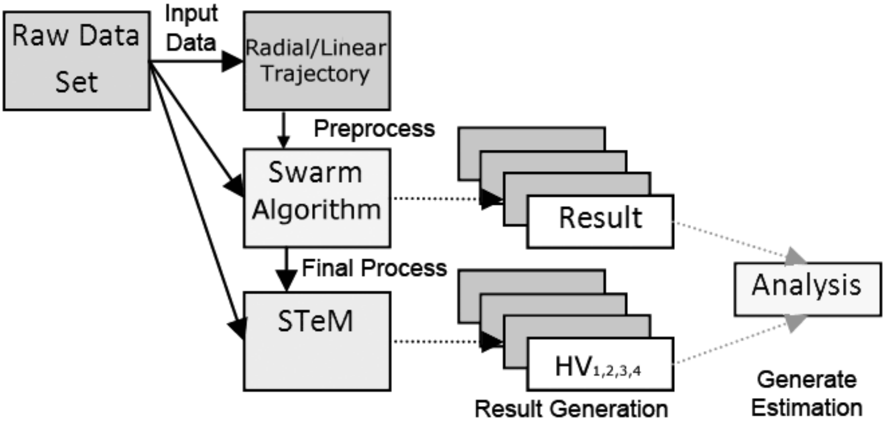


Fig. 11.3. Methodology of Analyzing Data Set using Tensor Analysis under Radial/Linear Trajectories and Particle Swarm Optimization

```
function preRunStemFunction()
global gHandles
clearOutput();
preProcessParamInput();
handles      = gHandles.handles;
dataMapMode  = get(handles.popup_DataMap,'Value');
maxIteration = str2double(get(handles.input_Iterations,'String'));
maxIteration = setMaxIteration(dataMapMode, maxIteration);
waitTime     = str2double(get(handles.input_WaitTime,'String'));
singMode     = 1;
diffMode     = 2;
inputData    = zeros(1,1);
for i=1:maxIteration
    switch dataMapMode
        case singMode
            [inputData, errInput] = getInputData(inputData, i);
            errTraj = getTrajData(i);
            if isDataInvalid(errInput, errTraj); return; end
            inputData = manipulateData(inputData, i, maxIteration);
            inputData = trajectoryEvent(inputData, i);
            runStemFunction(inputData, i, maxIteration);
        case diffMode
            diff = cell(1,2); k = 1;
            for j=i:i+1
                [inputData, errInput] = getInputData(inputData, j);
                errTraj = getTrajData(j);
                if isDataInvalid(errInput, errTraj); return; end
                inputData = manipulateData(inputData, j, maxIteration+1);
                inputData = trajectoryEvent(inputData, j);
                diff{k}   = inputData; k = k + 1;
            end
            runStemFunction(diff{2}-diff{1}, i, maxIteration, diff{2});
    end
end
```

Run Single Data Mode:
1. Manipulate Data
2. Add Trajectory Event
Run Heuristics

Run Dataset Difference:
1. Manipulate Data
2. Add Trajectory Event

Run Tensor Analysis

```
function hessian=fourthTerm(hessian,caArray,distance,numOfResidues,Epsilon,L_term)
% derive the hessian of the first term (off diagonal)
for i=1:numOfResidues;
    for j=1:numOfResidues;
        if abs(i-j)>3
            bx=caArray(i,1) - caArray(j,1);
            by=caArray(i,2) - caArray(j,2);
            bz=caArray(i,3) - caArray(j,3);
            distijsqr=distance(i,j)^4;
            % diagonals of off-diagonal super elements (1st term)
            hessian(3*i-2,3*j-2)        = hessian(3*i-2,3*j-2)-L_term*Epsilon*bx*bx/distijsqr;
            hessian(3*i-1,3*j-1)        = hessian(3*i-1,3*j-1)-L_term*Epsilon*by*by/distijsqr;
            hessian(3*i,3*j)            = hessian(3*i,3*j)-L_term*Epsilon*bz*bz/distijsqr;
            % off-diagonals of off-diagonal super elements (1st term)
            hessian(3*i-2,3*j-1)        = hessian(3*i-2,3*j-1)-L_term*Epsilon*bx*by/distijsqr;
            hessian(3*i-2,3*j)          = hessian(3*i-2,3*j)-L_term*Epsilon*bx*bz/distijsqr;
            hessian(3*i-1,3*j-2)        = hessian(3*i-1,3*j-2)-L_term*Epsilon*by*bx/distijsqr;
            hessian(3*i-1,3*j)          = hessian(3*i-1,3*j)-L_term*Epsilon*by*bz/distijsqr;
            hessian(3*i,3*j-2)          = hessian(3*i,3*j-2)-L_term*Epsilon*bx*bz/distijsqr;
            hessian(3*i,3*j-1)          = hessian(3*i,3*j-1)-L_term*Epsilon*by*bz/distijsqr;
            % Hii: update the diagonals of diagonal super elements
            hessian(3*i-2,3*i-2)        = hessian(3*i-2,3*i-2)+L_term*Epsilon*bx*bx/distijsqr;
            hessian(3*i-1,3*i-1)        = hessian(3*i-1,3*i-1)+L_term*Epsilon*by*by/distijsqr;
            hessian(3*i,3*i)            = hessian(3*i,3*i)+L_term*Epsilon*bz*bz/distijsqr;
            % update the off-diagonals of diagonal super elements
            hessian(3*i-2,3*i-1)        = hessian(3*i-2,3*i-1)+L_term*Epsilon*bx*by/distijsqr;
            hessian(3*i-2,3*i)          = hessian(3*i-2,3*i)+L_term*Epsilon*bx*bz/distijsqr;
            hessian(3*i-1,3*i-2)        = hessian(3*i-1,3*i-2)+L_term*Epsilon*by*bx/distijsqr;
            hessian(3*i-1,3*i)          = hessian(3*i-1,3*i)+L_term*Epsilon*by*bz/distijsqr;
            hessian(3*i,3*i-2)          = hessian(3*i,3*i-2)+L_term*Epsilon*bz*bx/distijsqr;
            hessian(3*i,3*i-1)          = hessian(3*i,3*i-1)+L_term*Epsilon*bz*by/distijsqr;
        end
    end
end
end
```

Calculation of Forth Term of Tensor Analysis Code (From Lin & Song)

Fig. 11.4. Source Code for Data Preprocessing and Sample of STEM Hessian Code

The experimental approach conducted is categorized according into the dataset processing method used. This includes pseudo-randomized data, grid-lattice structured data and a Particle Swarm Optimization approach:

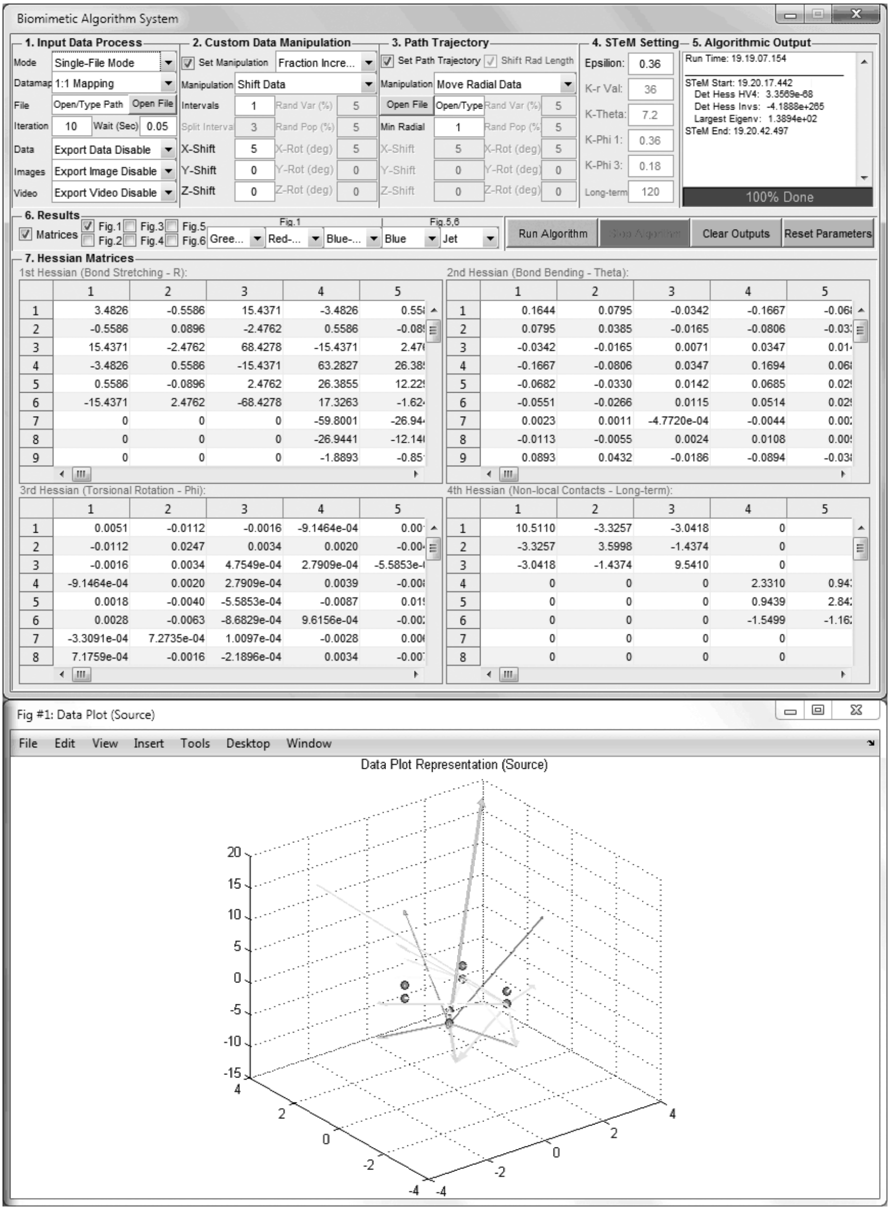
- **Random data generation**
  The data is randomly generated using a Pseudo-Random algorithm (Mersenne-twister method) as a control baseline to evaluate the other dataset layouts. The random data is subject to the limits of variation among the total population size using a pre-determined threshold value. Furthermore, dataset randomization limits ensure the environment is chaotic within reasonable bounds to control the experimental structure.
- **Grid-lattice layout**
  The data is generated according to a grid lattice plane, with each element equidistant from each other. This data structure is used to evaluate how radial and linear trajectories can impact on geometrically layered structures, and its impact on network structures that are evenly distributed in the environment. A layered lattice structure emulates the crystal structure found in carbon allotropes, such as graphite mineral.
- **Particle swarm optimization**
  The data is subject to Particle Swarm Optimization (PSO) as described by Clerc [3, 4], with random, grid-lattice datasets and future data sources being processed by PSO algorithms. The optimization of the data to obtain a global minimum or maximum value is limited by the total number of iterations that can be executed, the objection function used and the thresholds values for inertia and correction factors.

Implementation of the trajectory analytical system was developed by building upon the original STEM source code provided by Lin and Song [11]. The user frontend environment is primarily utilitarian: with the main panel in Figure 18.5 (*Left*) used to configure the experimental scenarios, and the visual display panel in Figure 18.5 (*Right*) showing the direction and magnitude of each element's long-term Hessian value. Apart from the provisioning of data manipulation and simulation of path trajectory functions, the analytical system allows the user to customize the Tensor Analysis parameters to evaluate a range of variations in epsilon and its eventual impact on the Hessian long-term values.

The prototype is developed in a step-wise fashion, where the execution of the simulation environment is based on the parameters defined in these steps. The environment is configured as follows:

- *Environment configuration:* Importation of the sail craft positions and their relative coordinates with one another;
- *External influences:* Introduction of environmental interference to the network structure, such as the initial detection of obstacles or foreign craft.
- *Internal influences:* Setup of the path trajectory to determine a craft's behavior as it encounters an obstacle. Execute results once final step is configured.

Additional data representation is made using surface map plots of the Hessian matrices as captured in Figure 18.6 (*Above*) for 20 elements with 3 coordinates



1. Input Data Process

Mode: Single-File Mode
Datamap: 1:1 Mapping
File: Open/Type Path
Iteration: 10 Wait (Sec): 0.05
Data: Export Data Disable
Images: Export Image Disable
Video: Export Video Disable

2. Custom Data Manipulation

☑ Set Manipulation: Fraction Incre...
Manipulation: Shift Data
Intervals: 1 Rand Var (%): 5
Split Interval: 3 Rand Pop (%): 5
X-Shift: 5 X-Rot (deg): 5
Y-Shift: 0 Y-Rot (deg): 0
Z-Shift: 0 Z-Rot (deg): 0

3. Path Trajectory

☑ Set Path Trajectory ☑ Shift Rad Length
Manipulation: Move Radial Data
Open/Type Rand Var (%): 5
Min Radial: 1 Rand Pop (%): 5
X-Shift: 5 X-Rot (deg): 5
Y-Shift: 0 Y-Rot (deg): 0
Z-Shift: 0 Z-Rot (deg): 0

4. STeM Setting

Epsilon: 0.36
K-r Val: 36
K-Theta: 7.2
K-Phi 1: 0.36
K-Phi 3: 0.18
Long-term: 120

5. Algorithmic Output

```
Run Time: 19.19.07.154
________________________
STeM Start: 19.20.17.442
  Det Hess HV4:  3.3569e-68
  Det Hess Invs:  -4.1888e+265
  Largest Eigenv:  1.3894e+02
STeM End: 19.20.42.497
```

100% Done

6. Results

☑ Matrices ☑ Fig.1 ☐ Fig.2 ☐ Fig.3 ☐ Fig.4 ☐ Fig.5 ☐ Fig.6
Fig.1: Gree... Red-... Blue-...
Fig.5,6: Blue Jet

7. Hessian Matrices

1st Hessian (Bond Stretching - R):

	1	2	3	4	5
1	3.4826	-0.5586	15.4371	-3.4826	0.55
2	-0.5586	0.0896	-2.4762	0.5586	-0.08
3	15.4371	-2.4762	68.4278	-15.4371	2.47
4	-3.4826	0.5586	-15.4371	63.2827	26.38
5	0.5586	-0.0896	2.4762	26.3855	12.22
6	-15.4371	2.4762	-68.4278	17.3263	-1.62
7	0	0	0	-59.8001	-26.94
8	0	0	0	-26.9441	-12.14
9	0	0	0	-1.8893	-0.85

2nd Hessian (Bond Bending - Theta):

	1	2	3	4	5
1	0.1644	0.0795	-0.0342	-0.1667	-0.06
2	0.0795	0.0385	-0.0165	-0.0806	-0.03
3	-0.0342	-0.0165	0.0071	0.0347	0.01
4	-0.1667	-0.0806	0.0347	0.1694	0.06
5	-0.0682	-0.0330	0.0142	0.0685	0.02
6	-0.0551	-0.0266	0.0115	0.0514	0.02
7	0.0023	0.0011	-4.7720e-04	-0.0044	0.00
8	-0.0113	-0.0055	0.0024	0.0108	0.00
9	0.0893	0.0432	-0.0186	-0.0894	-0.03

3rd Hessian (Torsional Rotation - Phi):

	1	2	3	4	5
1	0.0051	-0.0112	-0.0016	-9.1464e-04	0.00
2	-0.0112	0.0247	0.0034	0.0020	-0.00
3	-0.0016	0.0034	4.7549e-04	2.7909e-04	-5.5853e-
4	-9.1464e-04	0.0020	2.7909e-04	0.0039	-0.00
5	0.0018	-0.0040	-5.5853e-04	-0.0087	0.01
6	0.0028	-0.0063	-8.6829e-04	9.6156e-04	-0.00
7	-3.3091e-04	7.2735e-04	1.0097e-04	-0.0028	0.00
8	7.1759e-04	-0.0016	-2.1896e-04	0.0034	-0.00

4th Hessian (Non-local Contacts - Long-term):

	1	2	3	4	5
1	10.5110	-3.3257	-3.0418	0	
2	-3.3257	3.5998	-1.4374	0	
3	-3.0418	-1.4374	9.5410	0	
4	0	0	0	2.3310	0.94
5	0	0	0	0.9439	2.84
6	0	0	0	-1.5499	-1.16
7	0	0	0	0	
8	0	0	0	0	

Fig. 11.5. (*Above*) Tensor Analysis Application Frontend; (*Below*) An example of Trajectory Analysis Visualization

each. The x-coordinates are the examination of each element's coordinate relative to another in the y-coordinate. The maximum and minimum values are visually ascertained in the manner, and the decomposition plot reduces the Hessian to observe only one side of the diagonal. The figure shows how the Spring Tensor model determines the tensor trajectories at the diagonal, with the greatest measurement indicating an element having the greatest or strongest interaction with its adjacent neighbors. Inversely, the lowest measurement along the diagonal indicates an element have the least interaction with the neighbors.

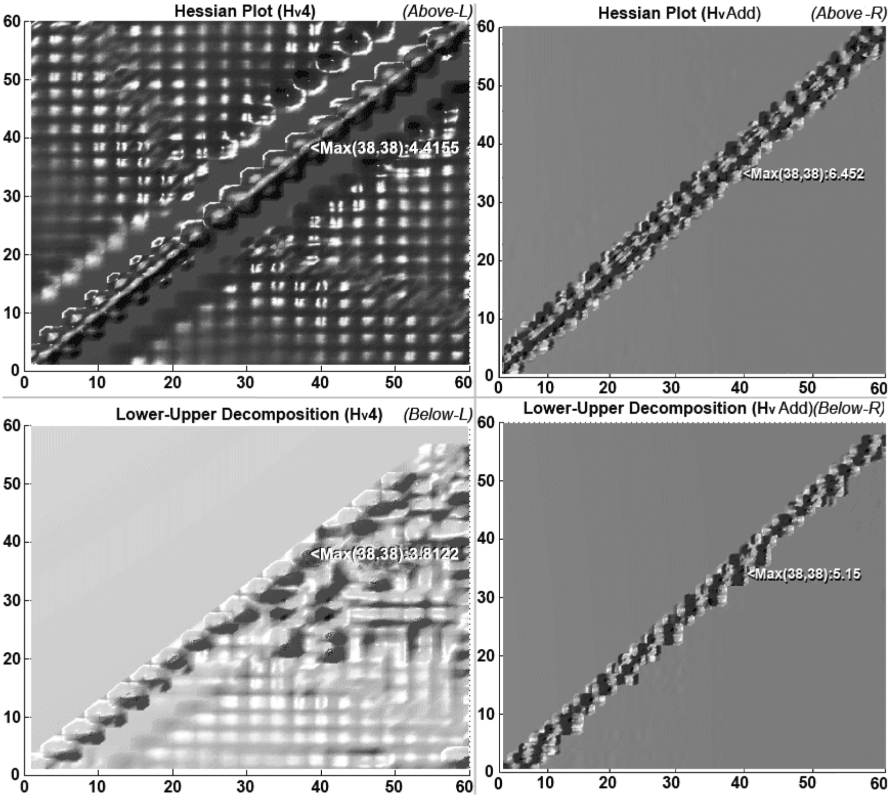


Fig. 11.6. Visualization of the (*Above-L*) Long-Term Hessian and (*Above-R*) Additive Hessian; Decomposition Matrix of the (*Below-L*) Long-Term Hessian and (*Below-R*) Additive Hessian.

Therefore, a strong interaction by one element can emphasize its significance relative to other elements; thus making it easier to identify hotspots such as when a land-yacht is in distress. Resource calculations are multiplied with the number of elements being processed; so using the example of 20 elements consisting of 3 coordinates each, a 60x60 long-term Hessian matrix is determined. With larger element values, the information space can magnify significantly resulting in resource constraints. As shown in Figure 18.6 (*Below*), lower-upper decomposition is used to find the maxima which results in a minor loss of information fidelity, but preserve memory resources by up to 50%.

11.4 Evaluation of the Tensor Analysis Framework

The experiment of the prototype framework examines the usability of STEM to determine projections with a range of different data sets. It is noted that the accuracy of the projections in land-yacht scenarios is not examined in this chapter, as this will be a future research task. The main objective of the prototype is to determine suitability, with reliability and efficiency being considered at a later stage of development.

11.4.1 Experiment of Heuristics

The following experimental parameters are established as follows:

1. The software platform consists of a Windows 7 Enterprise operating system running MathWorks MATLAB Release 2012a (Version 7.14). The MATLAB process runs on a C-based processing engine, with a Java UI frontend.
2. The hardware specification is a desktop computer with an Intel Pentium Dual-core Processor (2.10Ghz) and with 4GB of memory.

The experimental procedure executing the Tensor Analysis platform is elaborated below:

1. The data structures examined for the experimental case study:
   a. The first is using randomized data structures generated pseudo-randomly using the Mersenne-twister method. 50% of the data structure is subject to randomization at each iteration;
   b. The second is using a crystal lattice grid structure generated with each note equidistant amongst each neighbor. The lattice is shifted in a positive linear direction at each iteration;
   c. The third is a random data structure being processed with a Particle Swarm Optimization algorithm according to the spherical objective function at its maxima: $x^2 + y^2 + z^2 = r$
2. The experiment is run for 50 iterations for each agent population, with the following environmental bounds:
   a. The virtual land area is scaled down to 1m x 1m (Approximately scaled to 1:1000)
   b. Each land-yacht occupies an area of 10cm x 5cm (Craft occupies 0.5% of the total land mass)

c. A trajectory covers the diagonal path from the bottom left to the top right corner in iterative steps.

3. The population consist of the total number of land-yachts in the environment: 10, 20, 30, 40, 50, 60, 80 and 100 agents are examined in total.

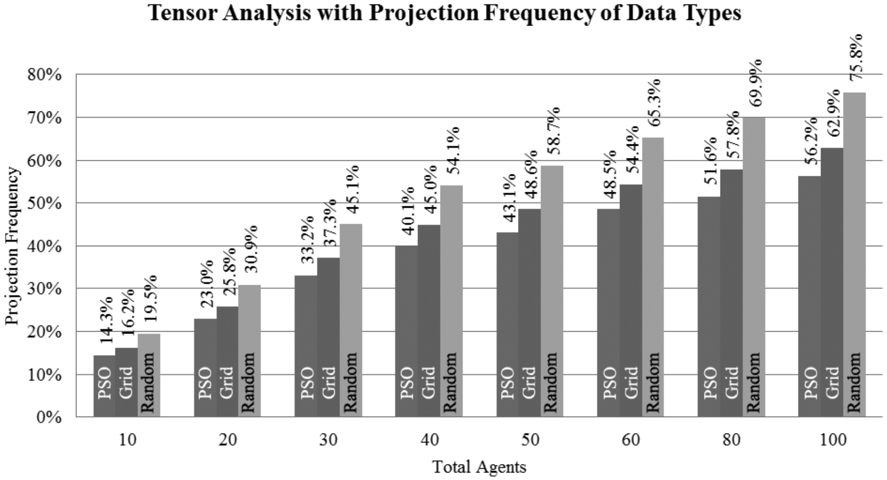


Fig. 11.7. Tensor Analysis with Projection Frequency of Various Data Processing Types

As observed in Figure 18.7, the purpose of the experiment is to measure the total number of projections made using the Tensor Analysis method. The results show that as the number of agents exists in the environment, the number of tensor projections increase. That is expected because the increase in population density will result in a larger number of interactions between land-yachts. However, it is important to take note of the projection frequencies between random data sets: grid lattice structures having an average 15% lower difference of projections compared to random data, and PSO with an average 18% difference of projections compared to random data. As the purpose of utilizing the STEM heuristic is to minimize the number of interactions of land-yachts as they cross through a SANET-connected field, the results demonstrate that a PSO method is suitable to extend the heuristic process further for environmental global awareness.

11.4.2 Analysis and Further Work

The difference in the observations about the data processing methods used in the experiment along with the final results, thus leads to the analysis of how structure impacts on the global perspective of a SANET-connected network:

- **Pseudo-random data**
  - Random coordinate structures are the least optimum method to distribute a network by its chaotic nature. Although randomized data will result in an evenly structured distribution of nodes in the network, making local observations on the data structure shows some elements cluster more tightly than others - though the global structure remains even.
  - Thus, the purpose of using a randomized data set is to provide experimental control. This is because it is a known data structure that is the least efficient in ensuring optimum resource distribution, considering multi-dimensional factors beyond physical geometry.
- **Grid-lattice structure**
  - From the physical observation, the grid lattice's equidistant distribution between nodes ensures points are evenly organized in a structured manner. However, equidistant lattices can only be organized evenly to its closest neighbors, and the distribution is linearly structured. As a result, the lattice structure loses its distributive quality when looking from a global perspective, because the observation of equidistance occurs only at the absolute latitude and longitude planes.
  - As an example, if a land-yacht traveling in the horizontal longitude plane needs to make a tacking maneuver by 25° clockwise, the lattice arrangement of neighboring land-yachts craft is lost with respect to the traveling yacht's perspective. In addition, equidistance is only a measure in the physical plane and does not translate to the non-physical plane - an example is the equitable distribution of resources and capability amongst all land-yachts.
- **Particle swarm optimized data**
  - PSO of the input data is a heuristic method to determine global minima or maxima values by adopting the biological collective behavior of swarming. The mathematical model devised to model this collective behavior is based on following the three main rules [3, 4]: to move in the direction as your neighbors, remain close as possible, but also avoid any collisions.
  - Particle swarming in robotic contexts is based on the principle that the autonomous actuators all move in a particular direction, but for the context of land-yacht trajectory control this is not the desired behavior. The current experimental model demonstrates how particle swarming can assist in global perception of land-yachts.
  - However, further process needs to be developed so that the swarming is required; only when a land-yacht needs to execute a path that is obscured by neighboring land-yachts. Further work needs to be developed to build a model that can balance the needs of the individual land-yacht agent, while harnessing the cooperative capability of SANET infrastructure space.

From the experimental observations, the results demonstrate the combination of Particle Swarm Optimization with Tensor Analysis can provide a two-fold effect on the global perception of the SANET field:

- **Tensor analysis**
  General summary: To determine competitive elements; obtaining the magnitude and direction of change.
  - Using STEM is an effective method to analyze how the effect of nearby land-yachts can interfere with the general trajectory of another land-yacht. In particular with further research, it will consider the effect of prioritizing a land-yacht's trajectory over the rest of the field. The dynamic between competitive and cooperative behavior is important especially for SANET-based infrastructures, as a quantitative measure can be established as to what optimum movement can be made by one land-yacht without affecting the whole field of yachts.
  - The increase in the projection frequency using a grid-lattice structure for the experiment is a good example of how qualitative measures of optimization (i.e. such as rational distribution of elements on a physical geographic basis) may not be compatible with global interests (i.e. maintaining stability of a global optima value or an objective function). In effect, tensor analysis provides a useful metric to determine how competitive behavior to establish a land-yacht trajectory. Thus, it eventually impacts on the globally-connected network of SANET-connected yachts in a positive or negative manner.
- **Particle swarm optimization**
  General summary: To establish cooperative behavior; finding how elements can coordinate activity.
  - Using PSO with STEM compliments the balance between competition and cooperation, as PSO seeks to coordinate a swarm towards a common goal or objective point. If one is to consider the point of achieving a global trajectory by a land-yacht, it is important to consider how the system would react. This is especially if all yachts moved synchronously as a result of one yacht's need to reach a direction. Furthermore, it needs to be considered how some yachts will need to perform complex tacking maneuvers, if they have no choice but to sail in the wind.
  - For this problem, future work will be the development solution to close the feedback loop between competition and cooperation. An example to apply this feedback loop is elaborated: when a land-yacht will move in one direction, it runs a PSO routine to make neighboring nodes guide a path away from the land-yacht intended trajectory path. Using STEM analysis, if an analysis of the potential change of magnitude and direction indicates a yacht may need to make a tight tack to avoid a nearby vessel. This will trigger a halting condition of the PSO routine until the changes of potential values reach satisfactory thresholds.

Table 11.1. Summary of Local Awareness and Global Perspective of a SANET-connected environment

	Local Awareness of Environment	**Global Perception of Environment**
Competitive Behavior	● *Main Reasoning:* Local-competitive awareness is the essence of competitive behavior, as it requires the land-yacht to be awareness of its surrounds at all times while it traverses through the SANET field. Such a behavior is considered to be preservation of self, because the existence of the agent requires it to be aware for it to look after itself. ● *Example of Occurrence:* When a land-yacht needs to perform a tacking maneuver to avoid an on-coming collision by a neighboring land-yacht or an obstacle that is suddenly detected without warning.	● *Main Reasoning:* Global-competitive perspective is required when a group of land-yachts need to perform a singular task. It is the unintended consequence when the behavior of one group competes with the actions of another group executing a predefined action. This behavior is considered to be the preservation of the team. ● *Example of Occurrence:* When one team of land-yachts need to approach a goal point while another team of land-yachts is traveling on a separate or alternate trajectory that is opposing the first team.
Cooperative Behavior	● *Main Reasoning:* Local-cooperative awareness from a cooperative perspective typically occurs when neighboring land-yachts cooperate to perform a common goal. This is considered to be local cooperation because each land-yacht assists its nearest neighbor to execute a common task on behalf of the primary land-yacht. ● *Example of Occurrence:* When one land-yacht requires the assistance of a neighboring land-yacht to provide safe passage in a terrain field that is uncharted, undefined or populated with obstacles in the environment.	● *Main Reasoning:* Global-cooperative perspective is achieving awareness on a distributed scale. It requires a balance between competing local interests and global objectives. Such a behavior is considered global cooperation because each land-yacht's existence is dependent upon ensuring all yachts cooperate for stability. ● *Example of Occurrence:* When a land-yacht seeks to travel in a trajectory, the neighboring land-yachts cooperate to move away from the trajectory path; thus resulting in the safe passage of the particular land-yacht.

An elaboration of the global and local concerns exhibited by a SANET land-yacht environment is described in Table 18.1. The goal to design a globally-aware system that performs cooperative behavior is the main objective in designing a land-yacht cooperative heuristic agent. Combining Spring Tensor Analysis with Particle Swarm Optimization endeavors to balance the need for cooperative behavior to reach an objective goal, with the metrics necessary to ensure global perception of the environment [11, 12]. This is because behavior that is excessively competitive results in individual agents leveraging over the needs of the group, while excessively cooperative behavior results in group needs overriding the actions of the individual.

The following recommendations are suggested for further examination to investigate the dynamics between competitive and cooperative behavior:

- **Alternate data reduction methods**
  - The use of the Spring Tensor Model is a computationally dependent activity, relying on memory to store the multi-dimensional matrices. Simplifying the matrices with other linear or non-linear approaches can assist in the simplification of the output Hessian matrices.
  - Using either non-linear or linear reduction approaches will result in a loss of data precision, so it is important to obtain a reduction method that achieves preserves data precision as practically as possible.
- **Integrate prototype framework in physical environment**
  - Once a suitable global perspective metric and cooperative behavior heuristic is determined for a land-yacht environment, the deployment of the prototype to real-life scenarios should be considered. The integration will need to consider the system architecture, the middleware infrastructure and the connectivity framework to interface the heuristic service with the physical world.
  - An organized integration strategy is important because the parameters can be easily modified to match theoretical calculations with observed behavior (such as obtaining the epsilon value for STEM or the objective function for PSO).
  - Furthermore, optimization heuristics can be evaluated to determine how uncontrolled variables affect the system environment. Hence, this resolves the practical implications of using a heuristic and the effect it makes on the physical integrity of the land-yacht.
- **Alternate processing methods for the spring tensor model**
  - The use of Hessians to determine the STEM model is an effective method by Lin and Song to determine anisotropic and Gaussian factors for protein models [11]. For a SANET land-yacht environment, a future recommendation is to investigate a method that enhances the projection process.
  - The use of other techniques such as Lagrangians or Single Value Decomposition should be considered to obtain metrics for global perspectives; with comparisons made between the information accuracy of using alternate methods.

- This accuracy will matter depending on the level of precision is required in a land-yacht guidance system. Finally in certain circumstances, one method may be advantageous over another given different environmental conditions placed on the system.
- Examples of such a condition would be the operation of a land-yacht in a high-altitude or a high wind area and the effect on the projection process.

The future work recommended for the STEM heuristic analysis can be applied to any context where a SANET infrastructure space is suitable. This is because the physical environment should be integrated using uniform standards, in terms of protocols and data structures used. The benefit of such an approach will mean the land-yacht controller can be harmonized with external systems that are value-adding for the system. An example would be the integration of a base emergency warning system, such that the detection of threats like a forest fire will dispatch a fleet of land-yachts to the heat source. Then, the STEM heuristic will be executed to determine the yacht maneuvers establishing possible magnitudes and directions to generate potential routes to the destination.

In summary, the results demonstrate that the data processing method used to determine a trajectory path makes an impact on the global environment. As a consequence, this impact will determine the potential for future interactions with neighboring vessels in the physical space. Heuristic approaches that model methods in biological organization are ideal for land-yacht trajectory mapping, because they provide a template for initiating the local awareness for the neighboring SANET-enabled land-yachts. Coupling with STEM allows a global analysis of the impacting effects that an organizational method has on the entire network.

11.4.3 Providing Representation and Context to Land-Yacht Systems

The STEM Code has been used in a land-yacht context to demonstrate how tensor analysis provides a global representation of the SANET network space. This is especially for the case in the magnitude and direction of elements, as a result of change in the physical environment. The visualization of the Long-Term Hessian shown as a 3D surface map in Figure 18.6 represents the Hessian values generated from the STEM Code; being interpreted by referencing the physical coordinate values and the Hessian projection values as a visualized directional plot in Figure 18.5. It is this visualization of the projection values that adds value to land-yacht control systems by assisting the coordinator in terms of coordinating and facilitating the actions of land-yacht agents. This assistance, either human controlled or by heuristic-algorithm, will respond to local impacts in the SANET infrastructure - such as the priority trajectory manipulation of a land-yacht due to the occurrence of a critical event.

Inspired by the conceptual behavior of social systems, the goal is to design a fleet of land-yachts that cooperatively organize when a prioritized traffic path needs to be established. Path prioritization is organized on a centralized level is the direct equivalent of having global perception with competitive behavior. An example would be one land-yacht's path trajectory leading to the halting of all other land-yacht's processes in its path. Thus to improve the way in which agent-based systems cooperate, Tensor Analysis provides an efficient and uniform approach to sharing the information space from a global perspective. Therefore only selective land-yachts are controlled if they are in the direct path of a land-yacht's trajectory.

11.5 Conclusion

In this chapter, the use of the Spring Tensor Model with Particle Swarm Optimization seeks to achieve the balance between global optimality and local environmental awareness in the development of a SANET land-yacht coordination system. The use of Spring Tensors captures the metric values necessary for projecting the magnitude and direction of change in the network, while Particle Swarm Optimization provides the navigational context to guide a land-yacht in the environment. This achieves the desire of achieving cooperative behavior from a global context, as complete visibility of the network structure is maintained throughout the optimization process to generate a trajectory in the SANET field. Although further research work is required to obtain parameter values that are suitable for analyzing cooperative land-yachts from a physical perspective, the current design prototype demonstrates the potential of utilizing Tensor Analysis for systems-engineering domains. These domains include systems requiring cooperative behavior in distributed, networked environments, such as cloud computing and traffic management systems.

References

1. Chiu, C., Chaczko, Z.: Steering a Swarm of Autonomous Sailing Vehicles implementing Biomimetic SANET Middleware. In: Proceedings from the Australian Conference on the Applications of Systems Engineering, pp. 85–86 (2012)
2. Clementi, C., Nymeyer, H., Onuchic, J.N.: Topological and Energetic Factors: What Determines the Structural Details of the Transition State Ensemble and En-route Intermediates for Protein Folding? An Investigation for Small Globular Proteins. Journal of Molecular Biology 298, 937–953 (2000)
3. Clerc, M.: Discrete Particle Swarm Optimization, illustrated by the Traveling Salesman Problem, New Optimization Techniques in Engineering, pp. 219–239. Springer (2004)
4. Clerc, M.: Particle Swarm Optimization, International Scientific and Technical Encyclopedia. Wiley ISTE (2006)
5. Clerc, M., Kennedy, J.: The Particle Swarm - Explosion, Stability, and Convergence in a Multi-dimensional Complex Space. IEEE Transactions on Evolutionary Computation 6(1), 58–73 (2002)
6. Fossen, T.I.: Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles, Marine Cybernetics (2002)
7. Harland, J.: Seamanship in the Age of Sail. Naval Institute Press (1984)

8. Jouffroy, J.: A Control Strategy for Steering an Autonomous Surface Vehicle in a Tacking Maneuver System, Systems, Man and Cybernetics - SMC Conference (2009)
9. Kanayama, Y., Hartman, B.I.: Smooth Local Path Planning for Autonomous Vehicles. In: Proceedings of the IEEE Conference on Robotics and Automation, Scottsdale, Arizona, USA (1989)
10. Kiencke, U., Nielsen, L.: Automotive Control Systems. Springer Publishing, Germany (2000)
11. Lin, T.L., Song, G.: Generalized Spring Tensor Models for Protein Fluctuation Dynamics and Conformation Changes. In: Computational Structural Bioinformatics Workshop, Washington D.C., USA (2009)
12. Low, K.H., et al.: Task Allocation via Self-organizing Swarm Coalitions in Distributed Sensor Networks. In: 19th Artificial Intelligence Conference, pp. 28–33 (2004)
13. Rao, M., Georgeff, P.: BDI-agents: From Theory to Practice. In: Proceedings of the First International Conference on Multi-agent Systems - ICMAS (1995)
14. Ritter, H., Schulten, K., Denker, J.S.: Topology Conserving Mappings for Learning Motor Tasks: Neural Networks for Computing, American Institute of Physics. In: 151st Conference Publication Proceedings, Snowbird, Utah, pp. 376–380 (1986)
15. Xiao, C.M., Austin, P.C.: Yacht Modeling and Adaptive Control. In: Proceedings of the IFAC Conference on Manoeuvering and Control of Marine Craft, Aalborg, Denmark (2000)
16. Yeh, E.C., Bin, J.C.: Fuzzy Control for Self-steering of a Sailboat. In: Proceedings of the Singapore International Conference on Intelligent Control and Instrumentation, Singapore (1992)