An Individual Based Model of the Lobster Fishery in St. John Bay, Newfoundland, Canada

research instituut voor kennissystemen bv

Layout: Guy Engelen, RIKS bv
Text: Jennifer Whalen, Roger White, Memorial University Newfoundland, Canada
Guy Engelen, Inge Uljee, RIKS bv
Illustrations: Guy Engelen, RIKS bv
Published by: RIKS bv

Product information

The **LOBSTER IBM-MODEL** has been developed for the National Institute for Coastal and Marine Management (RIKZ), P.O. Box 20907, 2500 EX Den Haag, The Netherlands, under contract nr. **RKZ-1137** by RIKS bv in close collaboration with Memorial University Newfoundland, Canada.

For more information you are requested to contact RIKS bv.

The **LOBSTER IBM-MODEL** is available on CD-Rom which can only be obtained from Drs. Hans Hartholt, project coordinator at the National Institute for Coastal and Marine Management (RIKZ), P.O. Box 20907, 2500 EX Den Haag, The Netherlands.

The latest information regarding the further development of the **LOBSTER IBM-MODEL**: new versions of the software and/or written documentation, will be made available from the following web-site: http://www.riks.nl/projects/Lobster-IBM.

An Individual Based Model of the Lobster Fishery in St. John Bay, Newfoundland, Canada

Jennifer Whalen[1], Inge Uljee[2], Roger White[1], Barbara Neis[1], Guy Engelen[2]

[1] Memorial University, St. John's, Newfoundland, Canada
[2] Research Institute for Knowledge Systems (RIKS)

June 2004

Research Institute for Knowledge Systems
P. O Box 463
6200 AL Maastricht
The Netherlands
www.riks.nl

RIKS
research instituut voor kennissystemen bv

Contents

1 Introduction 7
1.1 The *Coasts Under Stress* Major Collaborative Research Initiative 7
1.2 The St. John Bay Lobster Fishery Project 7
2 Project Methodologies 9
2.1 Fieldwork 9
2.2 Results of St. John Bay fieldwork 11
2.3 Eastport Peninsula Fieldwork 13
3 Critical factors and processes in the St. John Bay lobster fishery 15
4 Structure of the St. John Bay model 17
4.1 Introduction 17
4.2 Individual Based Model of the St. John Bay lobster fishery 19
5 Working with the St. John Bay model 35
5.1 Calibration of the model 35
5.2 Features of the Lobster Fishery Model 35
5.3 Illustrative Simulation Experiments 38
6 Bibliography 47
7 User Manual 49
7.1 Introduction 49
7.2 Getting started 49
7.3 Opening a .SIM file 50
7.4 The Geonamica – Lobster Fishery Model application window 50
7.5 The Lobster Fishery Model window 52
7.6 Viewing simulation output 57
7.7 Customizing map output 60
7.8 Saving simulation results 62
7.9 Printing simulation results 63
7.10 Exiting the Lobster IBM-Model 64
7.11 If you experience problems 64
8 The Menu System 65
8.1 File Menu 65
8.2 Edit Menu 68
8.3 View Menu 69
8.4 Simulation Menu 71
8.5 Options Menu 73
8.6 Window Menu 76
8.7 Help Menu 77
ANNEX A: Geonamica® DSS Generator 79
ANNEX B: Pointer Shapes 85

1 Introduction

1.1 The *Coasts Under Stress* Major Collaborative Research Initiative

The project described in this report is part of a major collaborative effort between two Canadian universities—Memorial University of Newfoundland and the University of Victoria, British Columbia—in addition to several other universities and partner agencies in Canada, the United States, and the Netherlands. The primary goal of Coasts Under Stress (CUS) is to identify and understand the ways in which changes in society and the environment of these coastal areas have affected, or will affect, the health of people, their communities and the environment over the long run. The deeper understanding will be directed at improving policy affecting these and similar areas. The Social Sciences and Humanities Research Council of Canada (SSHRC) and the Natural Science and Engineering Research Council of Canada (NSERC) jointly fund the CUS MCRI project, with additional funding from participating universities and partners in government, business, non-governmental organizations and First nation groups.

1.2 The St. John Bay Lobster Fishery Project

This project focuses on the lobster fishery of St. John Bay, on the west coast of the Great Northern Peninsula of Newfoundland. The lobster fishery of St. John Bay is very important to the local economy. Local lobster habitat is among the best in the province and there are several hundred harvesters from several communities who fish these waters. In this area, as in others, the closure of the cod and other ground fish fisheries in the early 1990s was associated with increased effort directed at lobster, and this caused serious declines in local lobster stocks. As a result of the declines in the northern cod stocks, traditional cod fishermen transferred lobster licenses from the Bonne Bay area (south of St. John Bay) into St. John Bay during the mid to late 1980's. The addition of these new harvesters, as well as the fact that many harvesters began fishing lobster the entire season have produced a situation of encroachment and crowding on traditional territories, expansion of these territories, and utilization of areas in the Bay that were once untouched. The department of Fisheries and Oceans Canada (DFO) policy now prevents license holders from fishing outside of St. John Bay. Policies also prevent them from acquiring crab and shrimp licenses, and from accessing reasonable cod catches. As a result, they are trapped inside a very crowded St. John Bay lobster fishery. Increased fishing pressure is threatening local lobster stocks and undermining individual lobster landings, thereby threatening the social and economic future of harvesters.

The aim of the St. John Bay Lobster Fishery Project is to investigate the changing structure of the St. John Bay lobster fishery, and then, on this basis, to develop a formal model of the lobster fishery in St. John Bay that can be used to explore management options, regulations and the impact of possible environmental changes. The model uses a Cellular Automata based approach for modelling the lobster population and its movements, and implements an Individual Based Model for modelling the dynamic behaviour of harvesters. The model is designed to be as generic as possible in so that it can be applied, with only minimal modifications, to other areas. Some experiments are made to test the model's usefulness as a decision support tool in policy formation and planning for the lobster fishery. The modelling phase of the project involves a close collaboration with the National Institute for Coastal and Marine Management (RIKZ) and the Research Institute for Knowledge Systems (RIKS), both in the Netherlands.

2 Project Methodologies

2.1 Fieldwork

Conversations with the local fisheries observer, information from scientific reports generated from that area (i.e. Craig. T. Palmer, 1992), LEK (local ecological knowledge) taken from face-to-face interviews, and onboard observation are used to inform the development of an individual based computer simulation model of interactions between harvesting strategies, groups of harvesters, management initiatives and local lobster ecology.

St. John Bay lobster harvesters come from communities as far south as Port aux Choix and as far north as Eddies Cove East (Fig. 1). The communities between Port aux Choix and New Ferolle are located within the Bay itself and most of these harvesters fish from their home communities. The traditional cod harvesters who have been harvesting lobsters in the Bay since the mid 1980's move into cabins located in the communities on the Bay or on islands offshore. These groups of harvesters live away from their home communities, which are often an hour and a half drive away. In general harvesters from each community located within St. John Bay share the same wharf and occupy similar lobster fishing areas. However, since the amount of experience with the lobster fishery that harvesters possesses varies, different fishing strategies coexist within these communities.

A random sample of 50 lobster harvesters was generated from a list obtained from the department of Fisheries and Oceans. In the spring of 2002 forty-three face-to-face interviews were conducted from this sample. The entire fifty interviews could not be completed due to time constraints, since the research was being conducted during the lobster fishing season and some harvesters did not have enough time to participate. These participants included harvesters fishing in different areas of the Bay and harvesters who have been fishing the Bay for over 30 years as well as the newer entrants who have been fishing in the Bay only since the collapse of the cod fishery in the early 1990's. These interviews were designed to provide information on demographics, fishing experience, licensing (other licenses held), economic information on the importance of the lobster fishery to their incomes, information on fishing areas and the number of other boats on these areas, strategies for setting and moving traps, how weather affects strategies, and conservation attitudes of local harvesters. The original questionnaire had to be reworked as a result of the responses obtained from the interviews and also as a result of new data requirements that became apparent during the formulation of early versions of the model. Important factors such as catch and wind speed and direction were identified and additional questions had to be asked so that it was clear how these factors affected the strategies. It was also clear that we needed a map of the areas harvesters were talking about.

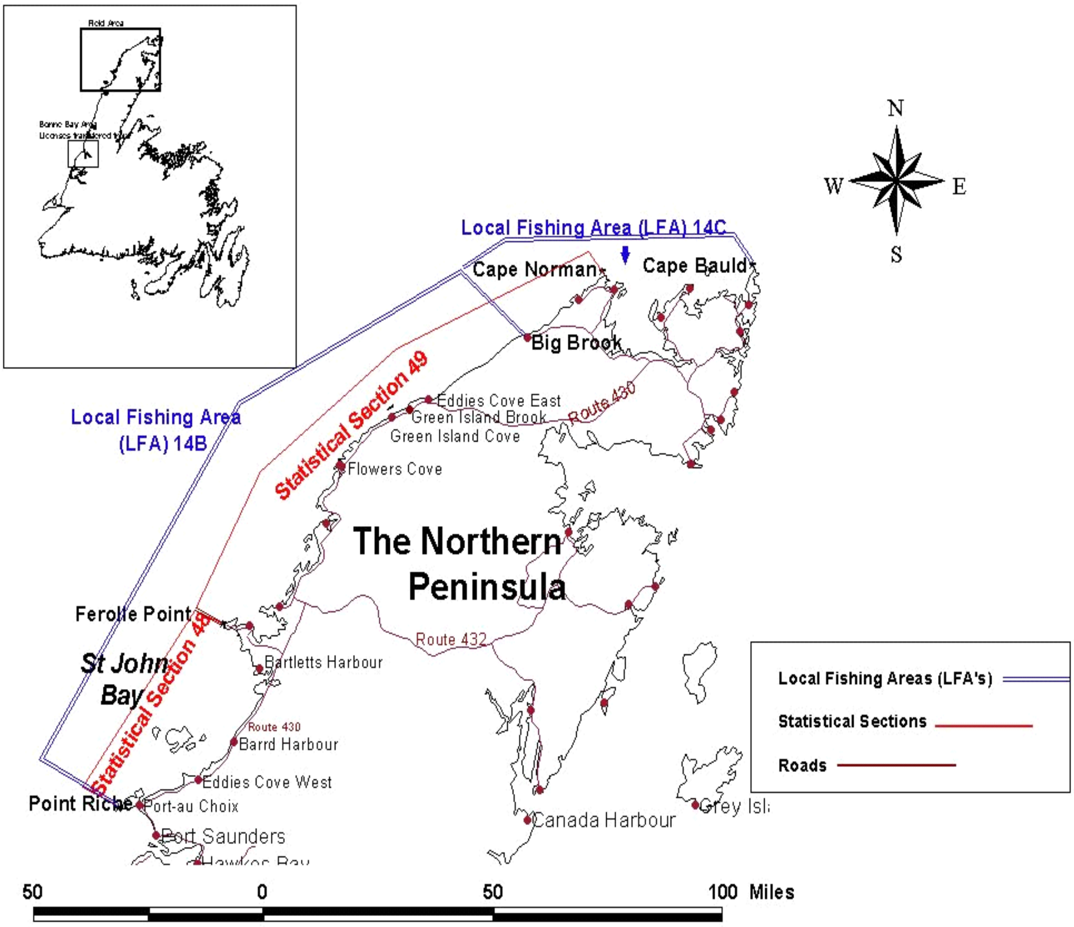


Figure 1. *Local Fishing Areas on the Great Northern Peninsula, Newfoundland.*

In addition to these face-to-face interviews, onboard observation also comprised a portion of the fieldwork at this time. The researcher went lobster fishing with twelve lobster harvesters who fished in different areas of the Bay. During these trips the local names of the areas being fished, the sequence of areas and the routes taken to these areas were mapped. Notes were taken on any changes in the areas fished and harvester's explanations on why these changes had taken place. In addition questions were asked about why the harvesters would move their traps and where they would move them. Observations were also made as to how much the harvesters were paying attention to what others were doing. At the end of each trip each harvester was asked to draw the routes we had taken and the areas we had fished on a Mylar (water proof) copy of the map. The harvesters were also asked how wind speed and direction would affect the sequence of areas visited and the routes taken.

More detailed, face-to-face interviews with approximately four 'expert' harvesters were also conducted at this time. These experts were chosen from peer recommendation from the forty-three previous face-to-face interviews. The rest of these interviews were completed on a return trip to the study area in the fall of 2002. The purpose of the expert interviews was to identify the changes that had occurred in the fishery, including more detailed information on fishing strategies at different points in their fishing careers, their explanations for changes in their fishing strategies, and their observations of trends in the abundance, size range, productivity, and distribution of the lobster resource and in the crowding on fishing areas,. Reflections on the consequences of different management strategies for harvesters' options and for the resource were collected, along with accounts of the conservation initiatives they participate in and their observations on the significance of these initiatives. During the return trip to the area an additional three expert interviews were completed. These participants fished in different areas of the Bay compared to the previous four participants. This was important because it was useful to see the different patterns of changes (crowding, abandonment of areas, expansion into new areas, and catch rates) in the different areas. After reviewing the previous four interviews and beginning to develop the model of the fishery, it was apparent that some additional questions needed to be added to the questionnaire to clarify unclear issues. For example, the reasons why people started to buddy up (i.e. fish two licenses out of the one boat) needed to be better understood.

2.2 Results of St. John Bay fieldwork

After reviewing the maps from the face-to-face interviews and the onboard observation, patterns of intensification and expansion were identified. The intensification of fishing effort started in the mid 1980's. New entrants into the fishery caused the number of boats on nearly all lobster-fishing areas to increase at this time, one exception being the area in the far north around Ferolle point. This increase in effort eventually meant that some traditional areas were fished out and abandoned. This meant that the fishers needed to expand their fishing territories, causing overlap in many areas between traditional community territories. This expansion is seen from the southern portion of the Bay to Bartletts Harbour in the north. Expansion has also been reported farther offshore around the many islands present in St. John Bay. In the past four or five years this expansion and intensification has produced extreme crowding in the middle section of the Bay utilized by the harvesters from Barr'd Harbour and the Whale Islands, causing a large scale shift towards the northern portions of the Bay. This situation has left the harvesters in the northern sections of the Bay feeling trapped in an increasingly crowded fishing area.

A number of factors were found to be important influences on the activities and strategies followed by lobster harvesters. Findings regarding these factors can be summarized as follows.

Ecological factors (wind speed/direction and water temperature):

- Affect whether or not the harvesters will actually go fishing.
- Affect the sequence of areas they will travel to, and routes taken to get to these areas.
- Affect the amount of time spent harvesting lobster because if it gets too windy they will have to return home.
- Affect the water temperature, which in turn affects lobster movement and therefore catch rates.

Catch on different time scales:

- While setting traps lobster harvesters tend to think in the long term—they consider where catches were good last year, a few years ago and all the way back to the beginning of their career.
- Once the traps have been set for a few days the decision seems to be based more on short term considerations—specifically on where catches were best the last few days.
- Harvesters seem to take catches for all lines into consideration and compare how one area of lines is doing against the others.
- If the catch is down in all areas harvesters tend just to move the lines into deeper or shallower water to try to find where the lobsters are.
- Catch thresholds are the number of lobsters the harvesters need to catch on one line of traps in order to return the line to the same location. Catch thresholds have changed over the 30 year study period, for example a harvester 30 years ago would have moved a line with 8 lobsters on it because the line he had in another area was producing 14 or 15 lobsters.
- Presently the catch thresholds are much smaller and can vary depending on the time of season. For example harvesters have reported a pattern (that has changed in the past few years) of high catch rates earlier in the season; therefore the catch threshold would be lower towards the end of the season. They stressed however, that this seasonal pattern was dependent on how early the season opened and on weather conditions.

Dependence on other species:

- Cod was important to many of the harvesters in the past.
- Herring was important but did not affect the effort put on the lobster and was only used as a supplement to income. Herring was used as bait for the lobster fishery.

Watching others:

- Some harvesters, especially those new to the fishery, seemed to watch where other people were setting and moving traps and followed them accordingly.

Listening to others:

- Harvesters would not say much about their catches until after the season was over.
- Most harvesters reported that they would not put much faith into what others were telling them because it is a very competitive fishery and every harvester has to look out for their own best interests.
- Higher levels of communication were noted between families and close friends. However, there is still a sense of looking out for one's own best interests and not giving away too much vital information. For example a harvester may not even tell his own brother or best friend exactly the location of the lines that are producing the best catches.

Economic considerations:

- Lobster earnings represent an important part of most harvesters' incomes.
- In the past cod represented the most important source of income of some harvesters, especially the newer entrants.
- The expenses associated with the fishery including rising gas and bait prices, as well as repairs to expensive motors and hauling devices, have driven some harvesters to buddy up in order to save on expenses and fish more efficiently.
- Declining lobster stocks and subsequent losses of crucial income has driven many harvesters to take their wives fishing instead of hiring share men. This is to keep all earnings associated with the fishery in the one household.

Changes in management rules:

- A trap reduction imposed in 1996 caused the harvesters to lower the number of traps they were using on a line.

2.3 Eastport Peninsula Fieldwork

As one of the primary aims in this project is to develop a model that is as generic as possible, other research was conducted on the Eastport Peninsula of Newfoundland, in order to develop a model that could handle both cases. In comparison to St. John Bay, the lobster fishery and the lobster stocks on the Eastport Peninsula in northeast Newfoundland are healthier. In response to the cod collapse the harvesters of Eastport started utilizing their lobster licenses throughout the full season, but there was no major influx of large numbers of new harvesters, and they took steps to prevent encroachment on their grounds. In addition, some Eastport harvesters have been able to obtain crab licenses to supplement their income.In Eastport problems with non-harvester poaching and disregarding the rules were tackled with the establishment of the Eastport Peninsula Lobster Protection Committee (EPLPC) in 1995. The Committee took steps to conserve the lobster resource as harvesters were encouraged to police themselves, get other harvesters to abide by the rules, v-notch berried females, complete logbooks and at-sea sampling programs, and implement two closed zones around Round and Duck Islands as well as an exclusive fishing zone around the Eastport Peninsula. They have formed a partnership with the Department of Fisheries and Oceans of Canada (DFO) and are seeking Marine Protected area status for the closed islands.The situation in the Eastport stock is starting to look up with reports of better catches and an increase in small and spawny lobsters. Reports from St. John Bay are quite the opposite with most individual catches down, and large declines in total catch in the Bay. In addition the small and spawny lobsters are reported to be declining as well. In St. John Bay v-notching, the filling out of logbooks, and at-sea-sampling were also carried out for a few years but the funding for these initiatives was terminated as of 2002. When the harvesters of St. John Bay were asked if they thought an arrangement like the one present at Eastport would work in St. John Bay, many thought anything was worth a try. Several people pointed out that the fundamental difference between the two areas was that the number of harvesters in St. John Bay is much higher than in Eastport, and thus it would be very difficult to get everyone to agree on new management initiatives.

While developing the structure of the model both situations were kept in mind. This should make it easier for applying (with minor alterations) the model to other localities in the future. Interviews were conducted on the Eastport Peninsula in the winter of 2002 and the interview schedule was used as a basis for the St. John Bay interview schedules.

3 Critical factors and processes in the St. John Bay lobster fishery

Wind speed and direction are critical factors in the St. John Bay lobster fishery. If the wind is too strong (usually around 30 knots/hour) the harvesters will not go fishing, unless the wind speed decreases later in the day. If the forecast is for strong winds they will likely check the traps in the areas that are farthest from the wharf to ensure that they won't miss them if the wind does get stronger and they have to return home for safety reasons. The harvesters can check traps in areas closer to shore in higher wind speeds. The direction of the wind affects the sequence of areas they will check as well as the route taken because the harvesters prefer to fish with the 'wind to their backs'. Harvesters also reported that after colder winds (for example winds from the north) the lobsters don't tend to move as much and are harder to catch. They also reported that when the water is warmer the lobsters tend to move into shallow water and they follow the lobsters into shallower water when this occurs. If the season opens early, like it has done for the past few years the water is typically not that warm and the majority of the lobsters are caught in the deeper water between 15 and 10 fathoms.

Catch is another important factor involved in the lobster fishery. When setting traps at the beginning of the season the harvesters reported thinking back to areas they have caught lobsters over their entire career. Some traditional harvesters reported leaving a few lines on grounds that had not fished well in several years just to make sure the lobsters did not return to the area. They seem adamant about not missing any lobsters they could possibly catch. After the first few days of fishing the harvesters start making decisions on moving their traps. This is based largely on the more recent catch history. Each harvester has a catch threshold (beyond which they will move their line of traps) that can change over the course of his or her career and over the season. The catch threshold depends on the overall catch of all the harvester's lines. The harvester tends to compare catch per line of all his fishing gear and moves the lines that are catching the least numbers of lobsters. If no lines are fishing well the harvester tries out different depths (both deeper and shallower) in order to try to figure out where the lobsters are. The decision to move lines seems to depend on both the individual's catch threshold and the overall catches of the harvester's lines.

The **dependence on other fisheries** affects the amount of effort that a harvester puts into the lobster fishery. The number of weeks the lobsters are being fished, number of times per week the lobster traps are checked, and the number of traps that are being checked characterize the amount of effort. In some instances the amount of effort taken away from the lobster fishery is dependent on when the cod fishery opens. Even if there is a temporal overlap many harvesters can still devote the same amount of effort to the lobster fishery. The abundance and price of all species fished affect which species will attract most of the fishing effort. Harvesters will devote most effort to the species that will be most profitable. Most harvesters said that the herring fishery did not affect the effort put on the lobster fishery because both fisheries could be carried out at the same time. From onboard observations it was clear that the herring nets could be hauled early in the morning and there was still plenty of time to check their lobster traps.

Communication between harvesters is another factor affecting fishing strategy. When the new entrants came to St. John Bay they watched closely where the traditional harvesters were setting and moving their traps. They consequently followed these harvesters for a few years until they felt confident enough to explore new grounds and fishing strategies of their own. During the season harvesters are pretty vague about the information they share with other harvesters because they do not want others to find out where their best fishing areas are. If this type of information is widely known then there will be a tendency for other harvesters to move into these areas, resulting in lost catches for the harvester who is already fishing in the area. The information that fishers share with each other therefore is not taken into great consideration when making decisions about where to put traps. However, if a harvester is fishing and he can see another harvester moving several lines into an area he is often tempted to put a few lines there as well. Higher levels of trust and communication are seen between families and close friends. In many cases a harvester is fishing with a wife, brother, or uncle, or they are buddied up. In this case they share all information. However, close family members and friends who fish from different boats do not share all the areas where they are getting the good catches.

Since the harvesters are trying to make the best living possible from the fishery, **economic considerations** are important factors in determining strategy. The lobster fishery is an important component of total income for most who participate in the fishery. With declines in the cod stocks, and the present quotas smaller than they once were, lobster has become more important to many of the traditional cod harvesters over the past 15-20 years. Improvements in technology, such as increased motor size and introduction of hydraulic motors that haul the traps onboard have increased the expenses of the harvesters. The increases of gas and bait prices have also added to the harvester's expenses. In many instances this has caused harvesters to buddy up and fish two licenses (850 traps) out of one boat. The harvesters also said that fishing is a lot easier when you have a buddy with you, even if you do have twice as many traps to haul. The economic considerations have forced some harvesters to bring their wives fishing with them. There is not enough money in the fishery anymore to support two households, so they try to keep the income coming from one boat in one household.

Changes in **management rules** can also affect the strategies used in the lobster fishery. For example when the trap limit was cut from 600 to 425 traps per license in 1996 this caused harvesters to change the number of traps they were using on a line. Many of the harvesters went from fishing 8-10 traps per line to fishing 6 or 7 traps per line. Reportedly this allowed them to still cover roughly the same areas as they did before.

4 Structure of the St. John Bay model

4.1 Introduction

A simulation model takes a real life situation, in this case a lobster fishery, and translates it into a computer environment. The model structure reflects the structure of the lobster fishery in general. It is relatively generic; it is made specific to a particular case by means of the input data and parameters which characterise the application, in this instance the lobster fishery of St. John Bay, Newfoundland. The harvesters are the basic units of the simulation, and boat agents represent each of the harvesters involved in the lobster fishery. Information collected from scientific literature and information found during the fieldwork is used to ground the simulation agents in the empirical data patterns found in St. John Bay. In this application the simulation runs through every day of the lobster season for the 40 year period starting in 1972, and in some what-if experiments, from 2002 to 2012. The boats set lines and return each day to check and possibly move their lines from one area to another as they search for, follow, and catch the lobsters.

The model can be used to gain a deeper understanding of the observed dynamics of the lobster fishery in the Bay. Beyond that, it can be used to perform what-if experiments to explore the effects of possible changes in environmental conditions, social structure, or management policy. Two such experiments are described in this report.

4.1.1 Cellular Automata Based Modelling

Modelling has been in existence for many years and many different types of models are available. Two of the newer modelling approaches being used presently are cellular automata (CA) and individual based models (IBM's).

The CA approach to modelling has a rule-based dynamic that allows high spatial detail to be retained in the final product. CA models are generally defined as a grid of cells, each cell having a discrete state that can change as a function of cell states within a neighbourhood using various rules within the model, and all cells are updated simultaneously at each iteration. Tobler (1979) was the first to highlight the fact that CA models are computationally simple and yet retain high spatial detail, making them a good choice for modelling spatial dynamics. In the case of typical CA models of land use dynamics, cells acquire states representing land uses, e.g. housing or agriculture. As the model runs the cells change state in accordance with transition rules, and as a result various land use patterns appear and evolve.

In the model of the lobster fishery presented here, harvesters (boats) are modelled in an Individual Based Model (IBM), and that model runs on top of a linked CA model of the lobster population. More specifically, the CA is used to model the spatial dynamics of the lobsters, and thus the pattern of the lobster population and its transformation over time. One of the limitations of the traditional CA approach has been the use of discrete cell states, which would not be useful for the model of lobsters being proposed here. However Wu and Martin (2002) developed a CA that explores the possibility of having continuous cell states. They proposed cells that have densities instead of discrete states. This is more appropriate for the lobster model because it can represent the number of lobsters in each cell and how that changes. In this project we also recognize the importance of non-homogeneity in the cell space. We use bottom depth and bottom type to define the suitability of cells, which in turn affects the number of lobsters (population density) within each cell.

4.1.2 Individual Based Modelling

Individual Based Modelling uses the individual—in this case a boat—as the basic unit in the model with each individual having as many characteristics (or attributes) as needed. Each individual then reacts to changing local circumstances during the simulation according to a list of rules. In this project we are interested in the individuals' behaviours and strategies under different scenarios, and what collectively they are doing. A variety of characteristics and strategies govern their behaviour. The mechanisms controlling a harvester's strategies and decision-making operate on the level of the individual. By using the IBM approach we can study the individuals as individuals, but also as groups or as a collectivity, showing their reactions to new management initiatives, or the effects of changing social structure.

Some bio-economic models of lobster fisheries have been developed (for example see Sutinen and Gates, 1995; Milon *et al.*, 1999). SIMLOB (Sutinen and Gates, 1995) is a bio-economic model that is used by scientists in Maine for making conservation recommendations. These models focus on the bio-economics of the fishery but do not encompass individual decision-making processes. Therefore these models do not predict what effects management recommendations will have on the harvester's decision making.

Little work has been done using IBM's to study fisheries. Bousquet *et al.* (1999) produced such a model to simulate fishing 'households' of the central delta of Niger. The simulations were used to study the transition from individual behaviour to general behaviour (group households) to see if variability at the individual level and variability in the environment could be linked to variability that is characteristic of society. Households of fishermen differ according to their ethnic group, number of people occupying them, origin, etc.

4.1.3 Why modelling as opposed to traditional statistical methods?

Statistics has been used extensively as a tool for studying fisheries. However, statistical techniques essentially describe the situation at the time the data was collected, but they reveal little about the underlying processes, nor do they uncover the interactions of individuals, or the consequences of those interactions. While statistical approaches do prove useful for characterising the situation being modelled, by giving summary measures such as average age of harvesters, or showing correlations between observations, they cannot be used to uncover causalities and to understand strategies and underlying process of behaviour. In the larger context the use of an IBM allows the user to detect patterns of behaviours at the community or entire Bay level that emerge from the behaviour of individuals and the relationships and communication among individuals.

4.2 Individual Based Model of the St. John Bay lobster fishery

4.2.1 Overview

The information obtained from the fieldwork component of the project was used to develop an individual based simulation model calibrated to the St. John Bay lobster fishery. In this model every boat in the lobster fishery is modelled as an individual. Each boat is assigned characteristics that will influence the decisions it makes. These boats are the core component of the model. Additional boat agents are added to the model at appropriate times during the simulation. Within the model sets of rules based on information found during the field work guide the agent's strategies and behaviour. At the start of every season the boats set their lines in their individual areas. As each day passes each boat checks its lines and decides to move them or to leave them where they were. The catch values are recorded for each of the boat's fishing areas. If the catch is very low the agents can decide either to move lines to their own best areas or into shallower water, or to move lines on the basis of information from other boats.

The spatial distribution of the lobster population is modelled on a daily basis. At the beginning of each season a population of lobsters is distributed over the Bay in a way that reflects water depth, bottom type, and a specified level of patchiness and randomness. Lobsters then migrate from cell to cell as the season progresses, and as they are caught, the population is reduced. It is important to note, however, that lobster population dynamics as such is not modelled, as that was not the aim of this project. In other words, the population of lobsters at the beginning of each season is established independently of the population at the end of the previous period. Thus it is not possible in this model to see long term feedback effects between lobster catches and lobster stocks.

The various components of the model are described in the following sections.

4.2.2 St. John Bay lobster fishery IBM and its components

The model consists of several interlinked components. The boats component is the major one (see Figure 2). Other components include a season component, a management component, a sea floor component, a catch component, and a lobster component. Data from interviews and other sources were used to supply the information needed to develop and parameterize these various components.

Information on harvester characteristics, behaviour and strategies is the basis for transition rules that drive the model during the simulation.

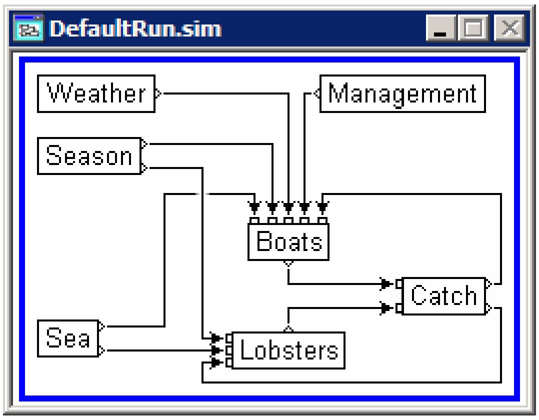
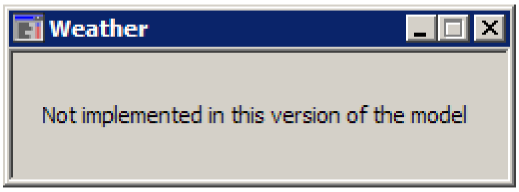


Figure 2. *Interface of Lobster fishery model and its components*

4.2.3 Weather Component

The weather component of the model was not implemented in this first version of the model due to the inconsistencies in wind speed and direction figures in the literature.

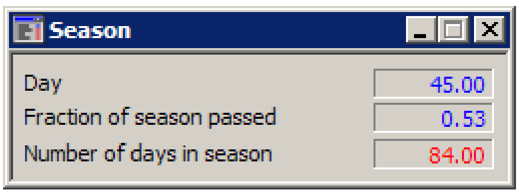


The wind speed factor prevents boats from fishing on days when it is too windy to safely go out at sea. The amount of wind (knots/hour) that will deter a boat agent from fishing is called the 'wind limit' and is stored as input data for each boat. If the wind speed is too high the harvesters report that they stay in closer to shore (inshore areas) and do not go to the areas farther offshore. At present code exists that omits the agents offshore areas if the wind speed is too high. This rule is already coded in the model and the 'wind limit' values are already in the boat characteristics array so that this can be implemented easily in the future.

Wind direction affects the particular sequence of areas a boat visits on a particular day. This would also have required sets of these areas sequence—one for each boat for every wind direction. The time and resources required for this feature meant that implementation was not possible within the constraints of this project.

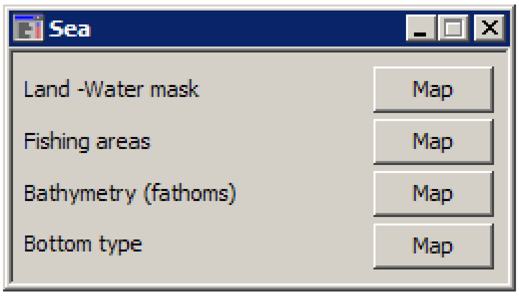
4.2.4 Season Component

The model runs through each day of the season for the 30-year study period starting in 1970. The length of each season was estimated based on data from Lobster management plans. The first day of the season depends largely on ice and weather conditions in the Bay. In general the season went from 12 weeks (84 days) at the beginning of the study period, to about 10 weeks



(70 days) in the mid-1980s, to 8 weeks (54 days) in 1997. These values can be changed within the model.

4.2.5 Sea floor component



Using a nautical chart (no. 4680) of the area, a digital representation of the sea floor of St. John Bay was produced. The bathymetry of the sea floor was reproduced as a digital bathymetric model, shown in Figure 3. The bathymetry was done in fathoms since these are the units present on the chart being used and also because this is usually how the harvesters refer to depth values. Since the attraction of lobsters to a particular cell is highly dependent on the bottom type, a representation of the ocean floor bottom type is also present in this component. Each bottom type is assigned a probability value from 0 to 1, where bottom types with values closer to 0 are less attractive to lobsters and bottom types with values closer to 1 are more attractive. For example bottom types that are rock or gravel get probability values of 0.9 and 0.99 respectively and bottom types of mud and sand are assigned values of 0.45.

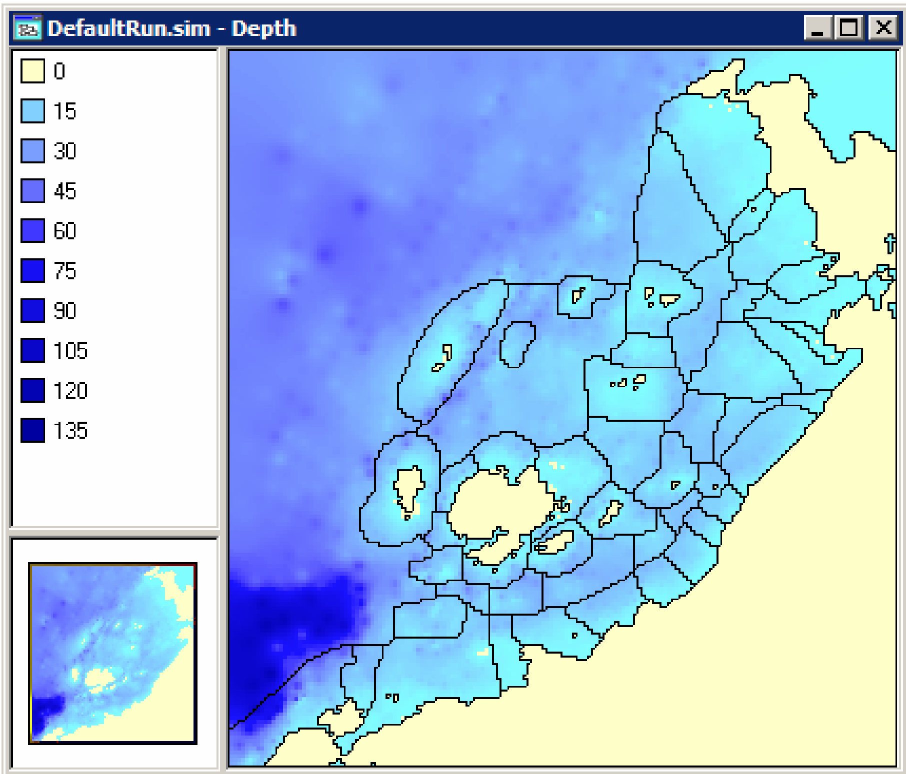


Figure 3. *Digital bathymetric model of St. John Bay*

The region modelled is divided into 54 areas (Fig. 4). These are used in the initialization of the model to specify the areas initially fished by each boat. They are also used to define community territories.

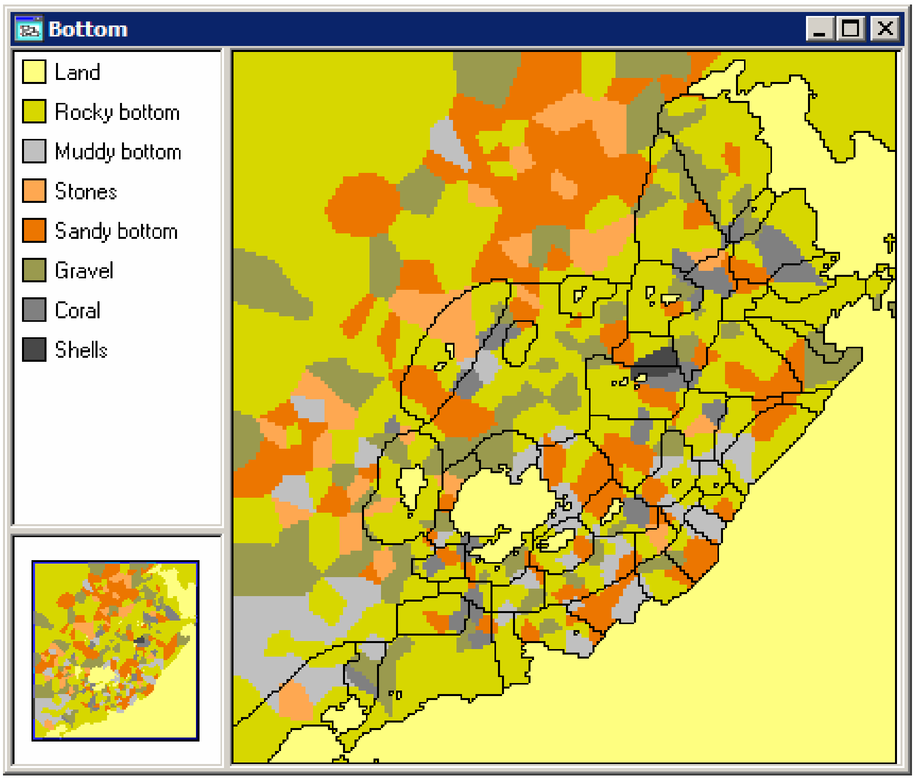


Figure 4. *Bottom type as represented in the Lobster fishery Model*

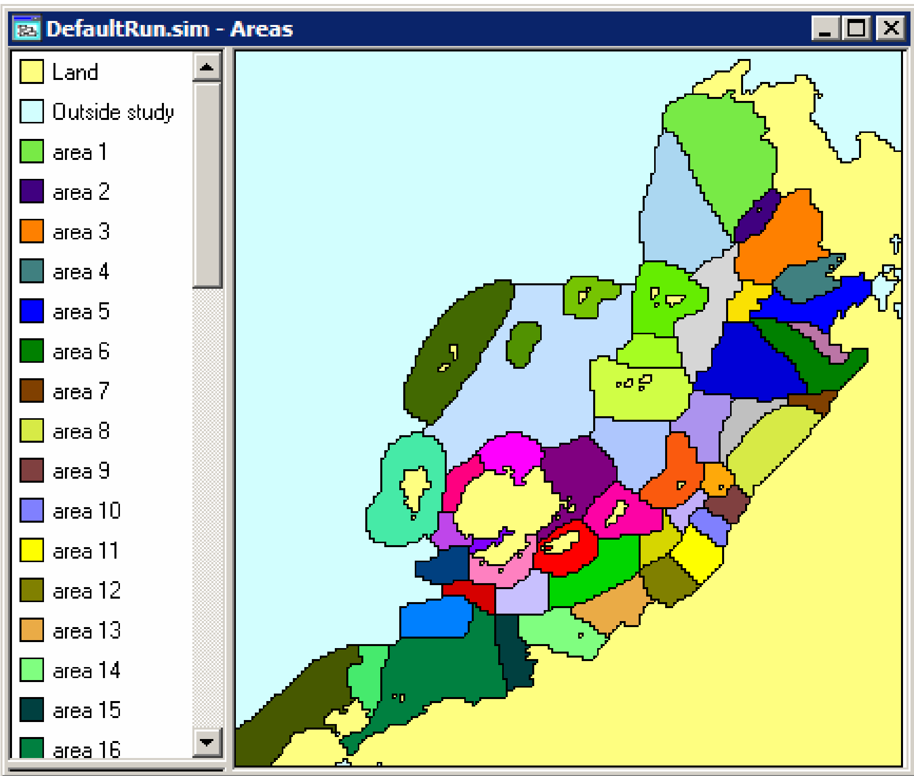


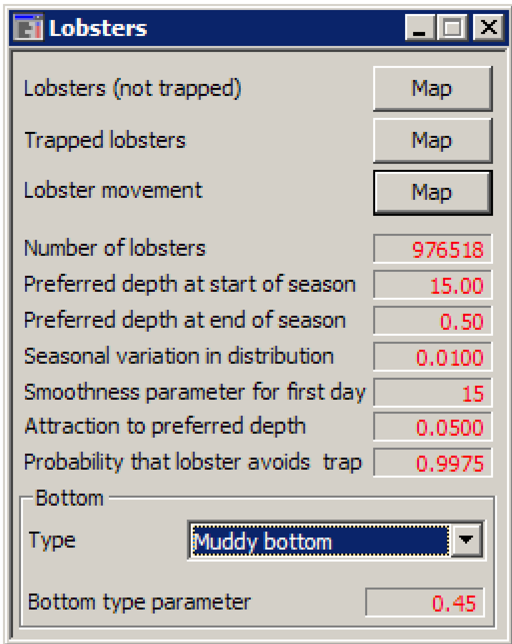
Figure 5. *Fishing areas of the St. John Bay lobster fishery*

The imposition of community territories, with boats from a community restricted to fishing within the territory assigned to that community, is one possible management option. Including specification of community territories in the sea floor module enables what-if experiments concerning this option. The territories defined for this project are shown in Figure 5. Areas were associated with communities on the basis of the composite of all interview maps from fieldwork.

4.2.6 Management component

The management component was developed in part with information from the Canadian Department of Fisheries and Oceans (DFO); for example trap limits for each year of the simulation are handled within this component. The primary function of the management component is to permit the model user to perform what-if experiments on the effects of management options on the fishery. For example, experiments can be performed on the effects of mandating a minimum number of traps per line, or of introducing community fishing territories. Closing areas to fishing is another possible management option.

4.2.7 Lobster Component



The lobster module must first provide a total population of lobsters available for catching at the beginning of each season. Data on lobster landings obtained from DFO was used as a basis to supply the lobster model with relatively accurate numbers of legal size lobsters. The Fisheries Resource Conservation Council of Canada (FRCC, 1995) report on lobster states that although landings in Newfoundland did not show a sustained increase from the 1970's like other areas in Canada, there was a peak in the early 1990's. In St. John Bay the peak year was in 1989 but landings were also very high in 1985, the year after the influx of new harvesters into the Bay. The landings data fluctuate but tend to decline fairly steadily after 1990. To mimic this we input data values for the total legal size lobster population every year, with year to year variations and a population decline after 1990. More generally, the total lobster population at the beginning of the season for each year of the simulation is entered by the user. This population of legal size lobsters is then distributed by the model around the Bay, and a migration routine generates daily movements during the season.

A simple cellular model of the spatial dynamics of the lobster population captures what is known empirically of the changing daily and seasonal distribution of lobsters. Specifically, the lobster population is attracted more to certain bottom types than to others; prefers certain depths depending on the date; is patchy; and varies from year to year. Spatial variance of measures such as densities in biological species have been used to quantify the degree to which organisms are aggregated. Indexes of the variance to mean ratio were used as a measure of aggregation, whereby a ratio greater than 1 would represent a patchy spatial distribution of species (Horne and Schneider, 1995). In additional studies the ratio of variance to mean was a function of the scale of measurement. One would expect that within an area, abundance or density will change from year to year but that there would be little variation in overall distribution (personal communication Ennis, 2003). Any shifts in distribution would be related to habitat quality, i.e. at low levels of population abundance marginal habitat areas would very likely have much lower densities than they would at high levels of abundance whereas density would not change to the same extent in areas of prime habitat. It is also possible that some lobsters either continuously or periodically at some time of the year occupy marginal habitats during periods of low abundance. Wahle and Steneck (1991) have reported that the patchiness of lobster distributions off the coast of Maine have been related to habitat type. Within most coastal areas, large stretches of soft, muddy bottom would be practically void of lobsters but all stretches of rocky bottom would most likely have some lobsters of some sizes. Smaller animals would predominate on gravel-cobble bottom (providing shelter) but would also be found on large cobble-boulder bottom where larger lobsters reside (Wahle, 1988 and 1990, and Wahle and Steneck, 1991). Within the lobster model all bottom types have been assigned attractiveness values based on this information, keeping in mind that we have only legal size lobsters in this model. The attraction of a lobster to a particular cell depends on depth and bottom type.

Within the model there are two types of patchiness of lobster distributions: a patchiness due to attraction to certain depths, and a patchiness due to random concentrations. The degree of attraction of the lobsters to a specified preferred depth can, if it is low, produce patterns of lobster distribution that are spread out over the entire Bay or, at the other extreme, it can produce a pattern of lobsters that stick to specific depth bands, resulting in a linear pattern of lobsters over the entire Bay. The parameter controlling this phenomenon does not change from year to year. The second type of patchiness does not depend on depth; instead it is random, with the patches changing location from year to year.

Lobsters move into shallower water as the water gets warmer and the season progresses (the daily movement routine in the model). As temperature increases during springtime, lobsters become more active and probably detect the temperature gradient as they move around more or less randomly and gradually end up in shallower water as the fishing season progresses; but even during summer, the population still occupies a range of depths, i.e. they are not all in the shallowest depths (Ennis, 1984a, Ennis et al, 1989). Over most of St. John Bay the bottom slopes quite gently and lobsters have to move greater distances to achieve the desired shift in depth or temperature.

In this model lobsters are distributed over the Bay at the beginning of each season, and then migrate from cell to cell during the course of the season. Both the initial distribution and the daily movements depend on the preferred depth on that day of the season, as well as on the bottom type.

First, the preferred depth *dt* at day *t* of the season is calculated as:

$$dt = ds + t\left[\frac{(ds - df)}{tt}\right]$$

where: *dt* is the preferred depth on day *t* of the season,
ds is the preferred depth at the start of the season ($d_s > d_f$, $d_f > 1$),
df is the preferred depth at the end of the season,
t is the day in the season,
tt is the number of days in the season.

The attractiveness of a cell, m_j, for lobsters at day *t* is given by:

$$m_j = q_j \exp\left(-c\left|d_j - dt\right|\right)$$

where: q_j is the attractiveness of the bottom type in cell *j* ($0 \leq c \leq 1$),
c scales the degree of attraction to the preferred depth,
d_j is the actual depth of cell *j*.

The higher the value of *c* the more lobsters move to the preferred depth, creating a negative exponential distribution; the lower the value of *c* the more the lobsters remain dispersed around the preferred depth.

To generate the initial seasonal lobster distribution, the map of St. John Bay is divided into four quadrants. These quadrants are then subdivided into four quadrants and so on to a total of four levels, such that the smallest quadrant is $1/256^{th}$ of the entire map. At each level the proportion of lobsters that will be placed in each quadrant is established as a product of:

(1) the square of the proportion of lobsters that went into the quadrant the previous year (this introduces some temporal autocorrelation into the regional lobster populations,

(2) the proportion of appropriate 200m cells (i.e. those of the right begin depth for lobsters) in all four quadrants that lie in this quadrant, and

(3) a random number. The random number is drawn from a Gaussian distribution with fixed mean (0.25) and standard deviation s_t, where s_t ($0 \leq s_t \leq 1$) is a parameter that can change over time as specified by the user.

The lobsters in quadrant (*g,i*) are initially distributed among the individual cells in proportion to *m*, the attractiveness values of the cells. The subroutine to allocate lobsters follows the formula:

$$N_j = Lob_{g,i} * \frac{m_j}{\sum_{j \in (g,i)} m_j}$$

where: N_j is the number of lobsters in cell j, and:

$$Lob_{g,i} = Lob_{g-1,j} \frac{\left(\frac{rr_{g,i}}{\sum_i r_i} \right)^2 * \left(\frac{d_{g,i}}{\sum_i d_{g,i}} \right)}{\sum_i \left[r_i * \left(\frac{rr_{g,i}}{\sum_i r_i} \right)^2 * \left(\frac{d_{g,i}}{\sum_i d_{g,i}} \right) \right]}$$

where: $Lob_{g,i}$ is the number of lobsters in quadrant j of grid level g, where j contains quadrant i at next level of grid; At the highest level, $Lob_{g,i}$ = $Lob_{0,0}$ = total lobster population in current season. [(g,i) is shorthand for (g0) or (g1) or (g1,g2) or (g1,g2,g3) or (g1,g2,g3,g4), according to level, with each gj = 1...4 for j>0]

$d_{g,i}$ is the number of 200m cells in the right beginning depth in quadrant i at level g.

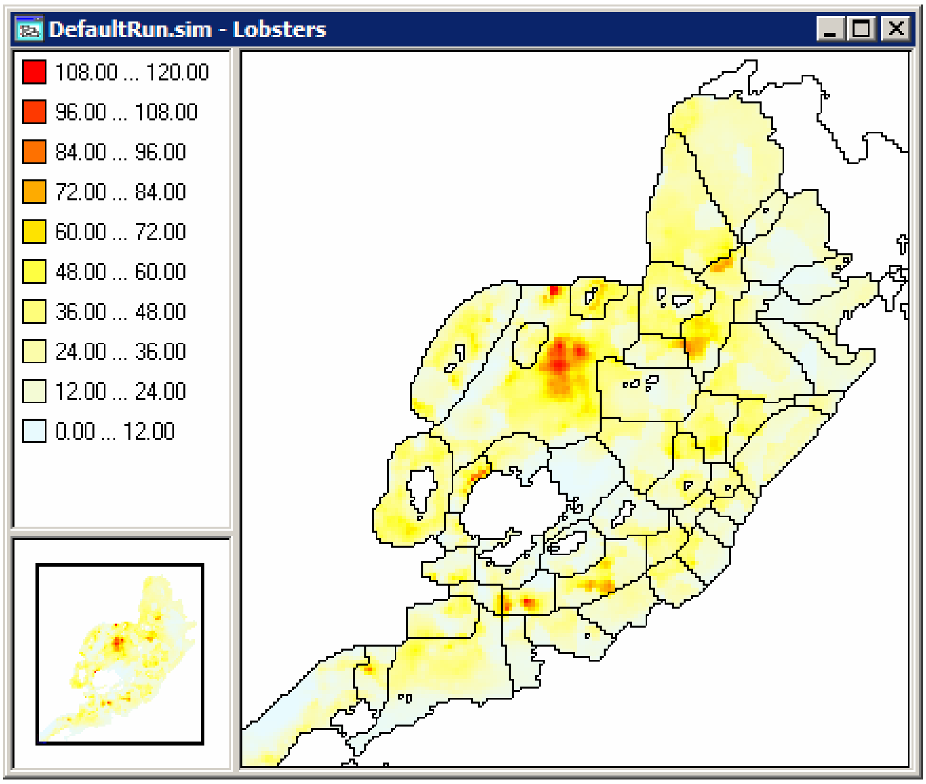


Figure 6. *Map of lobster distribution with high seasonal variability in the distribution of lobsters (s_t = 0.1)*

The parameter s controlling patchiness determines the degree of variation in the lobster distribution from year to year. If s_t = 0 the lobsters will be in the same places at the beginning of each season, with no seasonal variation. For larger values of s_t there is a larger variability in where the lobsters are at the start of each new season. As s_t approaches one the population becomes very patchy, with the population almost entirely concentrated in a few sections of the Bay, and this changes dramatically from season to season. Lower values of s_t represent the situation during the earlier period of the fishery when the harvesters would have been fishing several year classes of lobsters, and lobster patches would have been more plentiful and therefore the population more dispersed. A higher value of s_t characterizes the current situation of a lobster fishery that largely fishes one year class, which therefore has greater patchiness, with more year to year variation. In this application s_t is given a trend of increasing values, causing lobsters to become more patchy as the simulation runs through the years, thus simulating the actual situation.

Once the lobsters are placed in their initial cells the lobster daily movement routine is first run for a specified number of iterations to establish the distribution to be used on day one of the season. This is done in order to eliminate the boundary effects due to the quadrant procedure. The daily movement routine then migrates the lobsters from cell to cell during the season. The daily movement is calculated as follows.

For all cells *j*, the updated lobster population, N_j is:

$$N_j = \sum_{nj} N_k * P_{k,j} - \sum_{nj} N_j * P_{j,k} - K_j$$

$$P_{j,k} = \frac{m_k}{\sum_{nj} m_k}$$

where: K_j is the number of lobsters caught in cell *j*; $K_j = 0$ at t = 1
$P_{j,k}$ is the proportion of lobsters moving from cell *j* to cell *k,*
nj indexes the 5 cells of the von Neumann neighbourhood of cell *j.*

Thus the lobster population decreases over the season as a result of the boats catching the lobsters. The lobster catch routine within the model is implemented before the daily movement subroutine and before the harvesters check their lines. For one lobster in a cell, if there is one trap, q = the probability of not catching it; q is an input parameter.

For T traps in a cell, q^T is the probability of not catching it at all, in any of the traps, and $p = 1 - q^T$ is the probability of catching the lobster. Then for N lobsters in a cell, the probability of catching k ($0 \le k \le N$) lobsters is given by the binomial distribution:

$$p_k = \frac{n!\,p^k q^{N-k}}{k!(N-k)!}$$

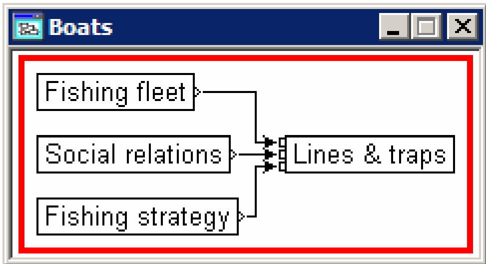
A uniform random number generator chooses a value of k—call it K_j— from the p_k distribution. K_j is the number of lobsters caught in all traps in the cell *j*. It is not necessary that K_j be integer. The catch is distributed equally among all traps in the cell. Thus if more than one boat has lines in the cell, the catch is distributed among all the boats in proportion to the number of traps each boat has in the cell:

$$K_{i,j} = K_j \left[\frac{T_{i,j}}{\sum_i T_{i,j}} \right]$$

where: $T_{i,j}$ = number of traps on lines of boat i in cell j.

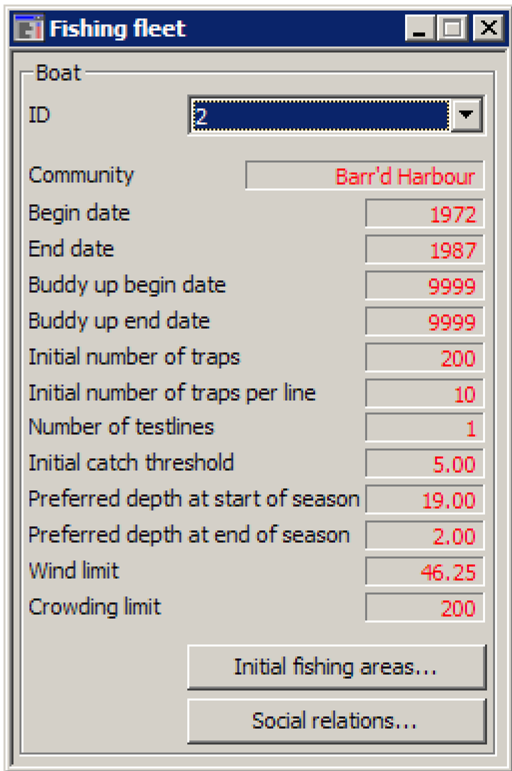
While the probability of catching a lobster when there is only one lobster and one trap is very low, the probability of catching a lobster when there are n lobsters and T traps is much higher. Also when the lobster is not caught today it may be caught tomorrow. The lobster module allows the user to see maps of lobster distribution (see Figure 6) and lobster movement as these change through the season and through the entire simulation.

4.2.8 Boats Component



This is the major component of the model. Each boat is represented as an agent. The model deals with the agents in random order to avoid boats first in the list having an advantage. In essence, the characteristics of boats can be grouped under three categories: characteristics of the fleet, the social relations among the fishermen, and the strategy to set and move lines. Based on these characteristics, the lines and traps are set in particular cells in fishing areas and lobsters are caught (or not). In the following, the strategies and actions of the agents are described.

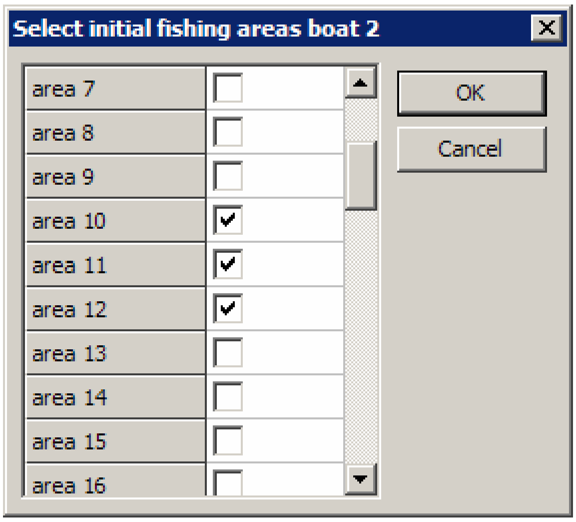
Agents decide how many lines they will use



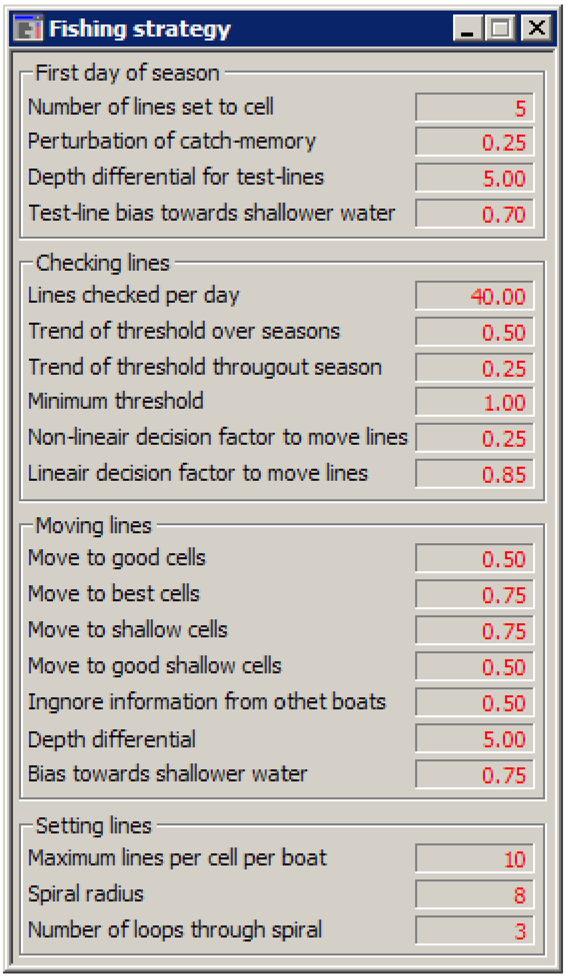
Each agent is initialised from the fleet database with the value for trap limit (the legal limit on the number of traps that may be set per license) to determine how many traps they will set. If there is no trap limit for that year (imposed by the management module) they will take the value from their input data. If there is a trap limit imposed for that year the agent will fish the allowed number of traps. The agent also reads in the number of traps he will use on each line. These two values allow the agent to calculate the number of lines he will use. Harvesters report that they change their number of traps per line over the years, thus maintaining roughly the same number of lines. This strategy allows them to cover the same amount of ground as they did with larger number of traps. This strategy is implemented within the model's logic. The number of traps will double if a boat decides to buddy up (i.e. two licenses are used on one boat); the timing of such events is included in the input data.

Agents decide which areas to fish and the proportion of lines to set in each area

Each agent decides which areas he will be fishing. For the first year this information is found in the input data; in subsequent years it is possible that there will be new areas, since all areas fished in past seasons are considered at the start of the season. During the course of a season, an agent may enter new areas not previously fished, either through local movements around a cell that take him over the boundary into a new area, or as a result of information received from other agents. Each new agent entering the fishery after the first year is assigned areas already used by agents from the same community. This simulates the new harvesters following others, a strategy that was reported by harvesters during interviews. The 'follow-others' strategy means that the new harvesters will follow harvesters with more experience and set their lines in the same areas.



The lines the agent sets are divided equally among all his areas for the first year of the simulation. After the first year the proportion of lines the agent sets in each area depends on the previous year's catch in that area. The agent adds weight to areas with little or no catch in memory so that areas with small catches can still possibly have lines set in them; areas with no catch in previous years will likely be abandoned.



Agents decide the number of test lines and regular lines they will set

The agent reads in the number of test lines from the input data; the rest will be regular lines they will set in each area. The test lines represent lines that are used to find lobsters if they are not where the agent expects them to be; they are set in shallower or deeper water than the regular lines. Each boat sets one to three test lines in each area fished. With more test lines the agent has a higher probability of finding new areas to fish. Regular lines are those that are set in cells of the appropriate depth for that day in the season. If the number of lines the agent is going to set in a particular area is not greater than the number of test lines, all lines in that area will be set as test lines.

Agents select cells within each area fished in which to set regular and test lines

A begin preferred depth value for the beginning of the season is recorded in the input data. A depth band is created around the begin depth by adding a parameter called 'begin depth differential' (bdd) to calculate the upper range of begin depth, and by subtracting bdd to get the lower range value. Use of a range is important because of error in the digital representation of the Bay and because it increases the number of suitable cells. The preferred depth range changes as the season progresses as described above in the Lobster Component. Within an area the agent randomly selects cells that fall within the depth range. The agent continues until all his lines are set in each area. The test lines are set in the same manner, except that lines must be set in cells that fall *outside* the preferred depth range. There are two test depths --one shallower and one deeper than preferred depth. A user specified parameter 'Test-line bias towards shallower water' determines the proportion of test lines to be set in shallower water and deeper water respectively.

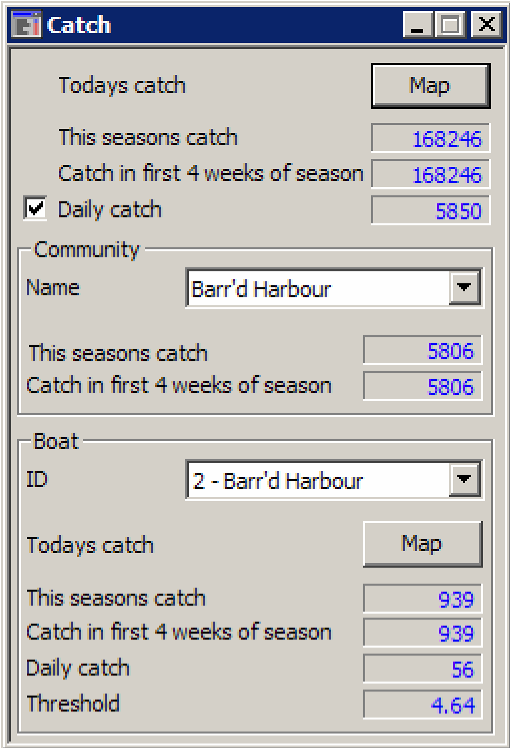
Generally, the model proceeds through the algorithm for each boat until all boats have their lines (both test lines and regular lines) set on the first day of each season. Each cell the agent chooses to set lines in must also satisfy a no-overcrowding criterion. Each individual decides whether or not the cell is 'too crowded', by comparing the number of lines in the cell to a number found in the input data for each boat. The 'too crowded' characteristic is the number of lines in one cell that an agent can tolerate before it becomes too crowded and he moves. Agents either have a too crowded value of 200 (meaning the agent is less tolerant to crowding and would likely move away from the crowd a little bit to set or move his lines) or a value of 250 meaning he is more tolerant and would try to squeeze his lines into a more crowded area. In addition, the agent may not set all his lines directly in the cell he has chosen; alternatively there is a parameter controlling the number of lines that get set in this cell and the number of lines that get set in the surrounding cells. This can allow the agent to spread out his lines and use more trial and error.

Catch Threshold and Soak Period

Each agent has a threshold catch per line below which he will not leave lines in the cell; this is specified in the input data for each boat. Since interview data suggests that thresholds decline as the season goes on and also decline over the study period, parameters have been included to allow the user to determine how much the threshold will decline ($\delta 8$, trend during the season; $\delta 7$, trend over all seasons). Catch threshold is calculated as catch per day of soak period, the soak period being the number of days since the line was last checked. An agent will normally check all lines each day but if there are too many lines, that is not possible, and so some will be checked on the next day.

Agents check their lines

On every day of the season after the first, agents check their lines and record their catch values. Each agent calculates the average catch per line in each cell. If the agent's catch per line per day of soak period is lower than the threshold value for that day he will move all lines from the cell. If it is equal to or higher than the threshold, he will keep some or all lines in that cell. Depending on how good the catch is in the cell with the best catch that day, he moves a number of lines that is a function of the best catch. The parameter δ1 determines the proportion of lines that will move. The larger δ1 the greater the proportion of lines that will move; for δ1 = 0 all lines will stay.



Choosing a cell to set lines in

When lines are moved they are not all moved into the target cell. Rather, a probability distribution is used to put varying numbers of lines into cells in the vicinity of the target cell, including that cell itself. Beginning with cells immediately adjacent to the target cell, and extending out to a maximum radius of 8 cells, lines are set in cells that satisfy certain criteria: the cell must be within the proper depth zone, otherwise it is rejected; also it must satisfy the no overcrowding criterion mentioned above. After a line has been set in the cell the agent sets the soak period equal to 1, records the cell type as test or regular, and keeps a record of the number of lines in the cell.

Agents move their lines into 'good cells'

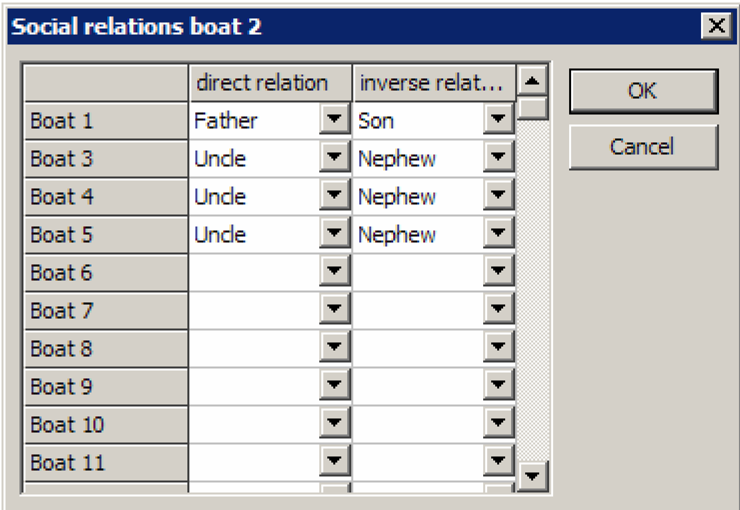
When an agent wants to move lines he searches the list of cells he has checked that day for the cells with the highest catches exceeding the catch threshold. These are the target cells, and a parameter δ2 ('move to good cells') controls the proportion of lines that will be moved into the vicinity of these cell(s), as opposed to the proportion that will be moved into shallower water, or moved on the basis of information from other agents. The smaller the parameter value the larger the number of lines that will move into the vicinity of the target cells. A second parameter, δ3 ('move to best cells'), controls the way the lines are distributed among these 'good cells'. The larger the parameter value the more the distribution of lines will be skewed in favour of the 'best good cells'. If δ3= 0 all lines will be divided equally among the 'good cells'.

Agents move lines to shallower water

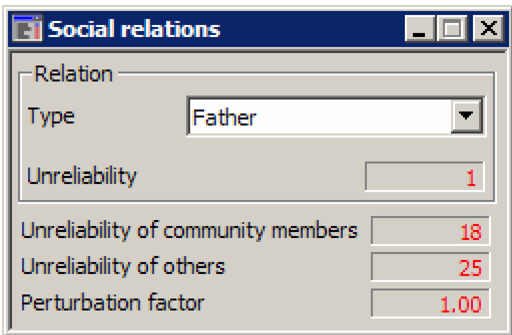
The lines that are not moved to the vicinity of 'good cells' either go to shallow water cells that had good catches in the previous year, or they move to cells on the basis of information obtained from others. A parameter δ12 ('move to shallow cells') controls the split between these two options. For higher values of the parameter, more lines will go to shallow cells; for δ12 = 1 no lines are moved on the basis of information from other agents. During the first week of the season any lines not moving to the vicinity of good cells will move to shallow water, since no information is available during this period. In the first year agents move shallow water lines into cells in the right depth band in the initial fishing areas, with a maximum of five lines per cell.

To locate shallow target cells, the agent looks at all cells in the appropriate (shallower) depth band that had average daily catches above the current threshold during the previous year. The parameter δ4 ('move to good shallow cells') controls the distribution of lines among these cells. The larger the parameter value the higher the proportion of lines that will go into the vicinity of the best of these cells. If δ4 = 0 the lines will be divided equally among the cells. If there are not sufficient shallow cells meeting the catch criterion to hold all the lines to be placed in shallow cells, the remaining lines are added to the lines to be located in cells based on information shared by other agents.

Agents move lines on the basis of information from other agents

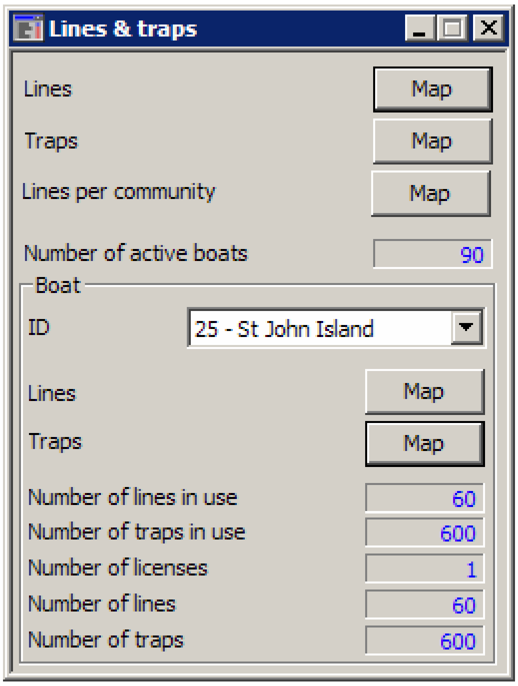


After the first week boats can seek information from other boats in the simulation. Boats seek information from the most reliable sources first, and failing to get useful information, move on to less reliable sources. The information sought is data on the location of the cell currently yielding the best catch, as well as that catch.



The input data for each boat includes a list of other boats operated by relatives and friends. These sources are listed in order of reliability—for example a father is more reliable than a cousin. Other sources, less reliable, are other boats operating out of the same community, and, least reliable, boats operating out of other communities.

In the information sharing routine, catch is expressed as the sum of the catch per soak day for the last 5 days. Unreliability of the information is represented by a random perturbation of the catch data, not the cell location. The reliability is calculated by multiplying a reliability factor characterising the source (lower values represent higher reliability) by a perturbation term with a fixed mean and a standard deviation given by parameter δ9 ('perturbation factor'). When an agent receives information from a relative or friend, the catch of the relative's best cell is perturbed. If this perturbed catch is greater than 5 times the agent's best catch per day from cells checked the current day, the agent moves lines to the vicinity of that best cell. When the agent receives information from another (unrelated) agent in the community, the community's best catch (not the agent's) is perturbed and if this perturbed catch is large enough, he moves lines to the community agent's best cell. Note that this is not necessarily the cell with the community's best catch (the one that was perturbed). When the agent is informed by someone from another community, the best average-catch-over-an-area gets perturbed and if this perturbed catch is good enough, the agent goes to the best cell in the areas fished by that community. Note that this cell does not have to lie within the best area (the one that was perturbed).



Agents go home

The agents return home and record catch information at the end of each day.

5 Working with the St. John Bay model

5.1 Calibration of the model

The numbers of lobsters in the simulation reflect empirical data on lobster landings for the years in the study period. The model was calibrated using both (1) long term catch data (1972-2002) for the Local Fishing Area and (2) available data on individual catches. Both data sets were provided by the Canadian Department of Fisheries and Oceans. The daily catch records consisted of logbook entries of 8 harvesters who fished in St. John Bay during the 2000 and 2001 fishing seasons. Data from a small sample for two years only out of 30 is clearly not an adequate basis for a reliable calibration; however, it is expected that additional logbook data from the 1990's will ultimately be made available, which will enable us to refine the calibration in the future. The average catch per boat and the percentage of catch caught in the first 4 weeks of the season were compared to simulation results. Exact calibration is not a well defined concept for Individual Based Models, but the calibration used here does reasonably replicate what is known of the dynamics of the actual lobster fishery in St. John Bay (see Table 1).

Table 1. *Comparison of Model calibration results with empirical data.*

Year	Average catch of a harvester in 2000 (lobsters)		Percentage caught in first 4 weeks	
	Empirical data	Model results	Empirical data	Model results
2000	1401	1886	64%	67%
2001	2220	1948	67.5%	66%

5.2 Features of the Lobster Fishery Model

The lobster fishery model has a number of features designed to make it easier to use.

The lobster module allows the user to view active maps of the lobster population during the simulation (Fig. 6). These show the user where the lobsters are from day to day as they move around and are caught, as well as from year to year. The module also allows the user to change parameter values affecting both the distribution of lobsters and lobster landings.

The boats component allows the user to see maps of the distribution of all lines in the Bay each day of the season (Fig. 7); the sequence of these maps thus shows the shifting location of lines as harvesters attempt to follow the lobsters. These maps can also display the distribution of traps. In addition, it is possible to display lines (but not traps) by community (Fig. 8), with lines belonging to agents based in each community shown in a different colour. Finally, the user has the option of viewing the lines (or traps) belonging to just one boat, in order to follow the detailed behaviour of any agent. For every day in the simulation, for any selected agent, the number of traps and the number of lines being fished, as well as the total number of lines possessed are shown in the 'Lines & traps' dialogue window.

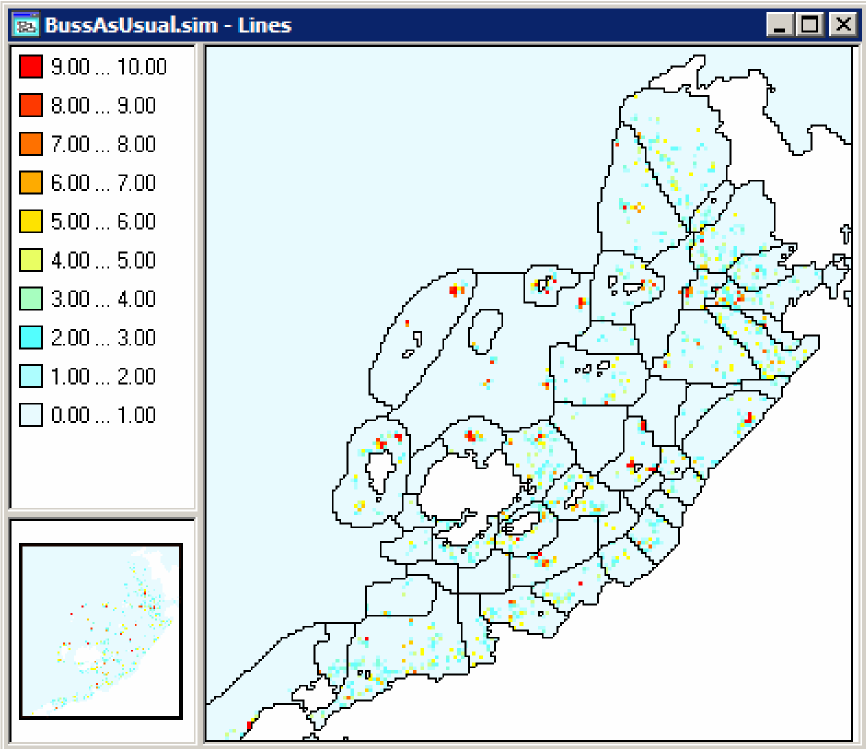


Figure 7. *Number of lines (for all agents) per cell on day 4, 1972*

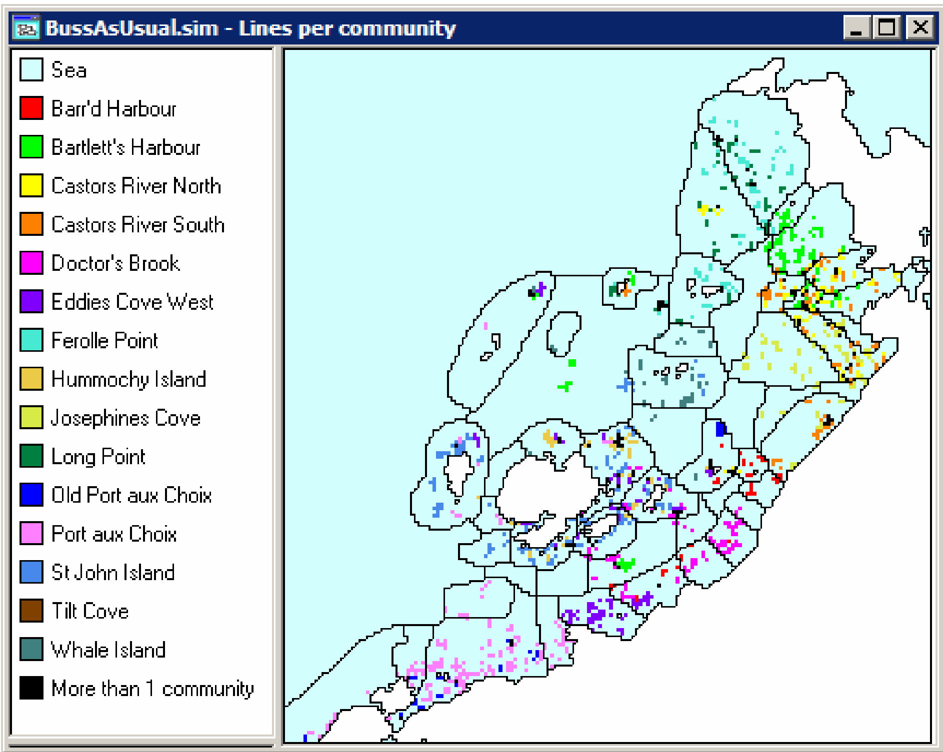


Figure 8. *Number of lines in each cell, by community, on day 4, 1972*

In the 'Catch' dialogue window a running sum of the total season's catch (for all agents) and the total daily catch is shown. The catch in the first 4 weeks of the season is also shown in this dialogue window. The user can also pick any one individual boat and view its daily catch as well as its total catch for the season to date. The daily catch for all agents is also mapped as shown in Figure 9 below. As is the case with the lines map, the user can also view a map of an individual boat's catch as well. These maps show the total catch (number of lobsters) per cell.

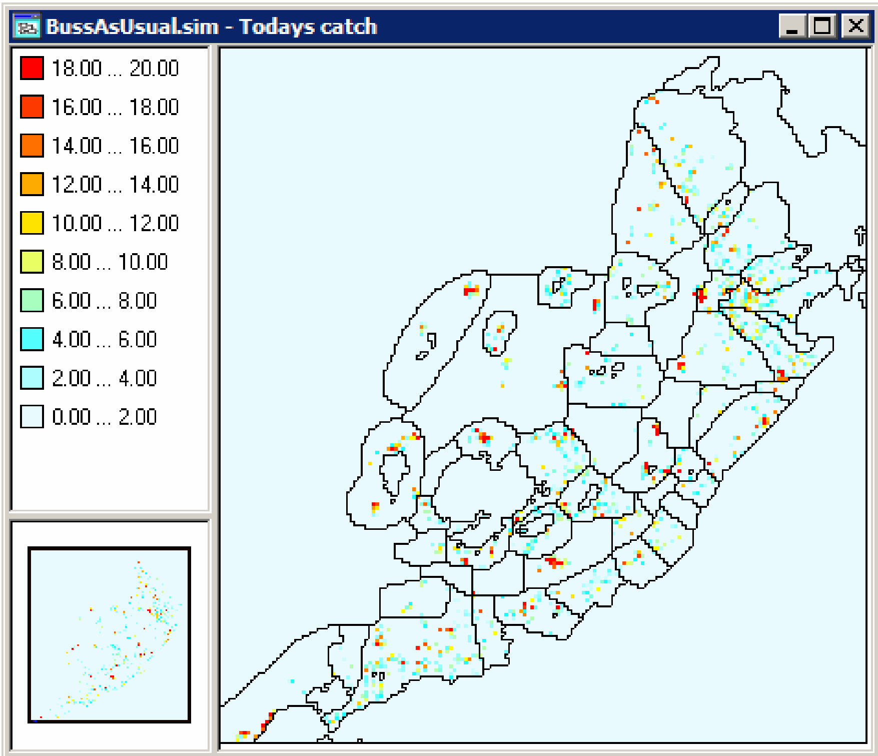


Figure 9. *Total daily catch for all agents on day 4, 1972*

The 'Management' dialogue window allows the user to implement trap limits in any year of the simulation. It can also be used to run simulations whereby the agents are restricted to community territories (one proposed management initiative) beginning in a specified year. The management dialogue window also allows the closing of any of the fishing areas. Maps of the fishing areas and the community territories can be opened from the 'Sea' dialogue window.

Output data generated for each day and agent simulated is sent to an Excel file that is dynamically linked to the model. This file contains the following information:

1. For each community, as well as for all communities combined, for each year:
   - the number of active boats;
   - the number of active licenses;
   - the total catch;
   - the average catch per boat and average catch per license;
   - the total catch in first 4 weeks of season;
   - the average catch per boat and per license in first 4 weeks of season.

2. For each boat for each year:
    - the number of licenses;
    - the number of traps;
    - the number of lines;
    - the number of traps per line;
    - the total catch for season;
    - the catch in first 4 weeks of season.

3. For each boat, for each area for each year:
    - the total catch and catch in first 4 weeks of season.

In addition, much more detailed output is written to a separate text file. For each agent in each area, for each day of each season the following is available:

- the number of lines checked;
- the catch of the day;
- the number of lines with a catch ≥ the threshold catch;
- the number of lines at the end of the day.

5.3 Illustrative Simulation Experiments

In order to illustrate the potential uses of this model as a tool for enhancing the development of fisheries management policy, two simulation experiments are described. The first explores the possible effects of implementing a community territory management scheme, whereby harvesters from each community may only fish in the areas assigned to their community, thus losing access to the entire Bay. The second examines the effect on catches of communication among harvesters under varying conditions of lobster distribution.

5.3.1 Experiment 1: Community territories

Information on traditional community territories was collected during fieldwork in the area in the spring and summer of 2002. In St John Bay, according to informants there were specific areas where harvesters from each community traditionally fished. These areas did not overlap as much as they do today: with the increase in lobster harvesters the situation is now one where harvesters fish in any part of the Bay.

An experiment was performed to see what would have happened if the Department of Fisheries and Oceans Canada (DFO) had formalized and enforced the informal territories that existed prior to the influx of new harvesters in the mid 1980's. If people were only allowed to fish in a defined community territory, how would it affect their landings? Would harvesters from certain communities be more successful in terms of catch than harvesters from other communities? Would variability of individual agent's catches from season to season increase or decrease? In order to answer such questions two simulations are run. The first is a business as usual scenario, with individual trap limits but no restrictions on areas fished: each person owning a lobster license has access to all areas within the Bay. The other assumes a community territory management strategy beginning in 1972. In this scenario the agents fishing from the different communities are assigned areas they are allowed to fish (shown in Figure 10 and Table 2). Community areas are defined on the basis of where most people from each community fished in the early years of the fishery.

All other input data and parameter values were held equal in both simulations.

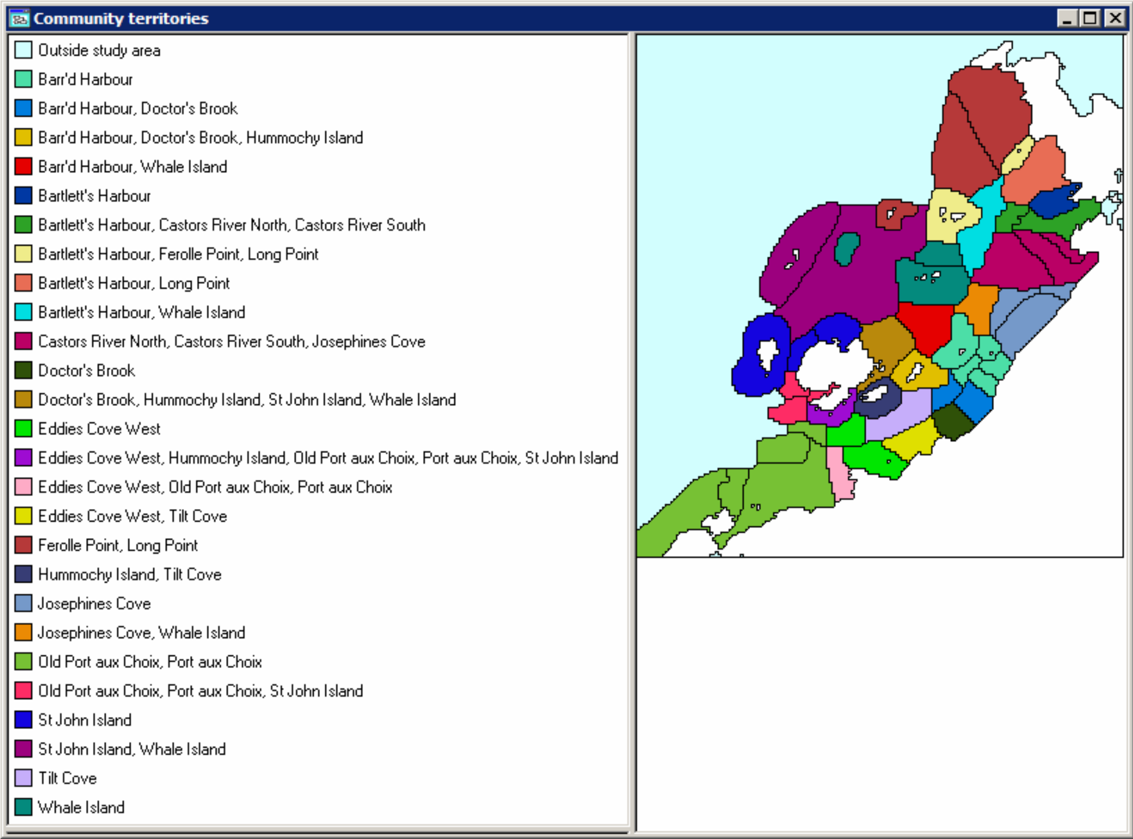


Figure 10. *Community Territories implemented in Experiment 1*

Table 2. *Areas comprising individual community fishing territories*

Barr'd Harbour	10	11	12	29	30	31	42	44	51	
Bartlett's Harbour	3	4	5	6	32	35	48			
Castors River North	6	7	32	40	52					
Castors River South	6	7	32	40	52					
Doctor's Brook	12	13	25	29	51					
Eddies Cove West	14	15	16	24	41					
Ferolle Point	2	3	35	36	45					
Hummochy Island	24	25	28	29						
Josephine's Cove	7	8	9	40	43	47	52			
Long Point	2	3	4	35	36	45				
Old Port aux Choix	16	17	18	19	20	22	23	24	39	49
Port aux Choix	16	17	18	19	20	22	23	24	39	49
St John Island	20	21	22	23	24	25	26	27	38	46
Tilt Cove	14	28	50							
Whale Island	25	33	34	37	38	43	44	46	48	

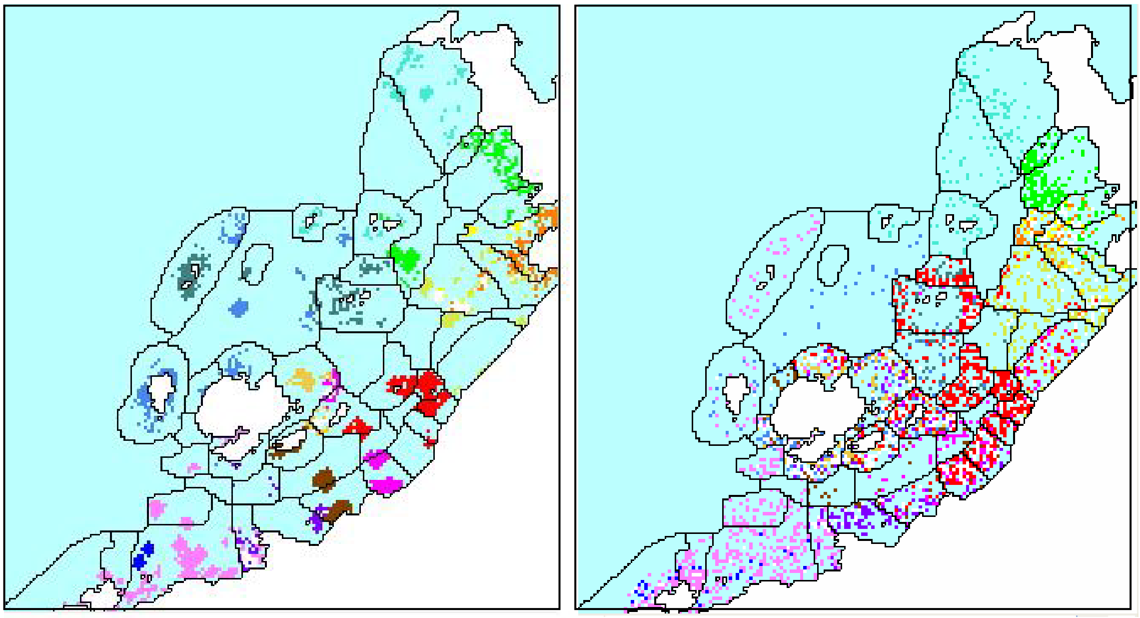


Figure 11. *Lines by community, day 2, 1997. Left: community territories scenario. Right: business as usual scenario (trap limits only).*

By visual comparison of Figure 11 it is easy to see that in their impact, these are two very different management scenarios. In the case of the business as usual scenario (Fig. 11, right) lines are spread widely across the Bay, and there is much intermixture of lines from different communities. On the community territories scenario map (Fig. 11, left) on the other hand, the lines are clustered in dense patches distributed over the entire Bay and there is very little mixing between communities, even in areas shared by communities. Comparison of the number of lines in each cell (Figure 12) also shows a strong contrast between the two management scenarios.

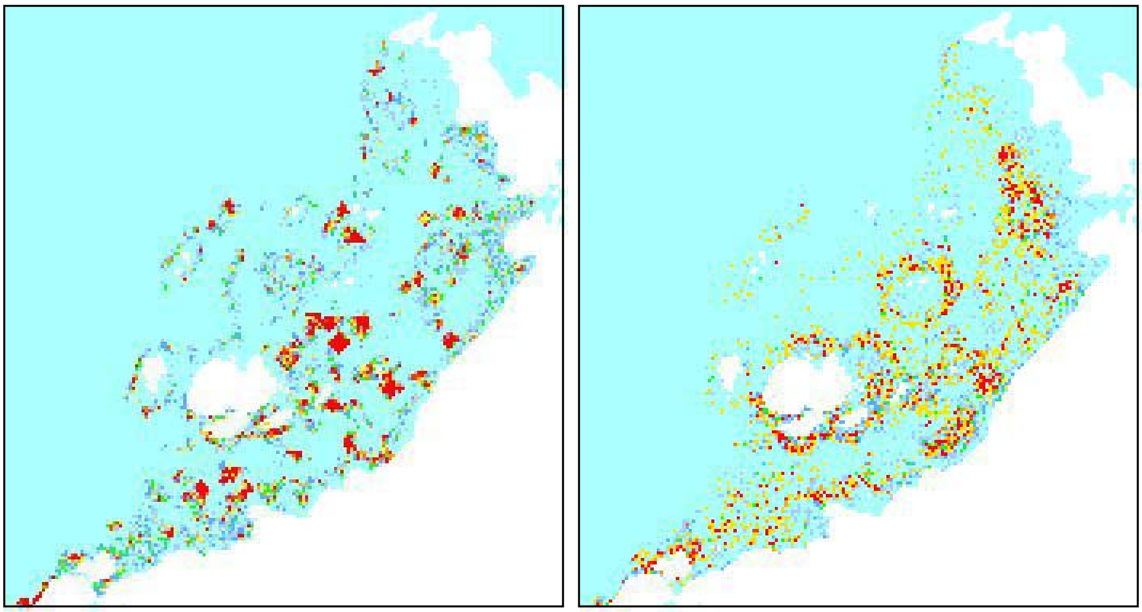


Figure 12. *Number of lines per cell (all agents) on day 25, 1997. Left: community territories scenario. Right: business as usual scenario (trap limits only). The colour codes are presented below*

The distribution of lines shown in the map on the right of Figure 12 is more evenly spread out over the Bay, while in the map on the left of Figure 12 most lines are in areas of dense concentrations of lines (areas in red), in a much more patchy distribution. Do these different distributions lead to a difference in catch under the two scenarios? If the lines are spread out then agents should have a better chance of finding lobsters. However, if the agents happen to find the lobsters early and they have all their lines concentrated in patches, they may have higher catches. Is there more variability in catch from year to year when community territories are enforced, and does the degree of patchiness in the distribution of lobsters systematically affect the catches?

To answer these questions four scenarios were examined, representing the two situations of no community territories and community territories, each under conditions of low and high seasonal variability in the distribution of lobsters. For each scenario, the model was run from 2003 to 2012 with 4 different random seeds, so in total 40 years were recorded.

Table 3. *Mean catch per boat and standard deviation for baseline and community territory scenarios under low and high seasonal variability of lobster distribution*

	Low Seasonal Variability		High Seasonal Variability	
	No community territories	Community territories	No community territories	Community territories
Mean	2568	2334	3842	3250
Std. Deviation	71	50	436	553

The results in Table 3 show that community territories result in lower mean catches under conditions of both low and high seasonal variability of lobster distribution, although in the case of high variability the difference, while much larger, was not significant at the 0.05 level because the variance was so much greater.

It is also interesting to examine the effect of the implementation of community territories on individual communities. Do some communities systematically benefit and others not?

The three communities that consistently showed a positive difference (community average catch higher under community territory scenario) were St John Island, Whale Island and Ferolle Point. The seven communities that consistently showed a negative difference were Bartletts Harbour, Castors River North, Castors River South, Barr'd Harbour, Tilt Cove, Doctors Brook, and Hummocky Island. All other communities showed variations over the years sampled.

Table 4. *Comparison of the differences between average community catch values under the baseline (no community territories) and community territories scenarios for selected years and communities*

Community	Difference of Mean Catches: Community Territories – Baseline Scenarios				
	1976	1986	1996	2002	Trend
Whale Island	+870	+1701	+1070	+1047	Always +
Ferolle Point	+380	+1649	+1097	+130	Always +
St John Island	+520	+2856	+2106	+1600	Always +
Bartlett's Harbour	-723	-729	-877	-473	Always -

The fact that some communities consistently showed an increase in community catch as a result of community territories raises several questions. Do these communities typically have high concentrations of lobsters in their community territories? Are there typically fewer boats from other communities in these territories under the community territory scenario? To explain the differences among community catches, it is helpful to examine the map for the distribution of lobsters in Fig. 13 (the distribution is the same for both scenarios) and the maps of the location of community lines under the two scenarios for day 27 in 1976 (Fig. 14).

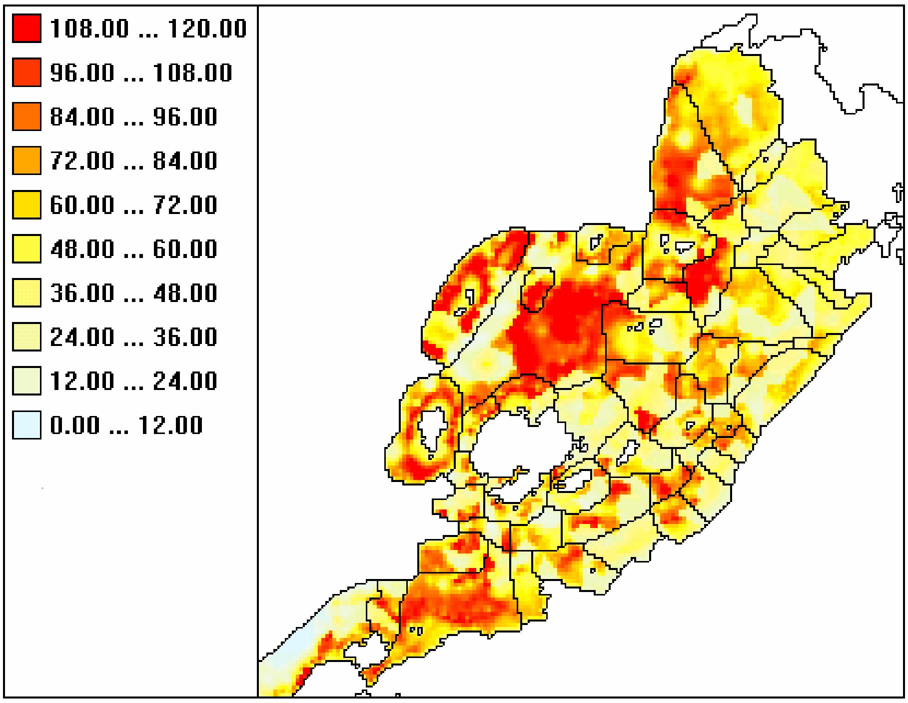


Figure 13. *Distribution of lobsters, day 27, 1976*

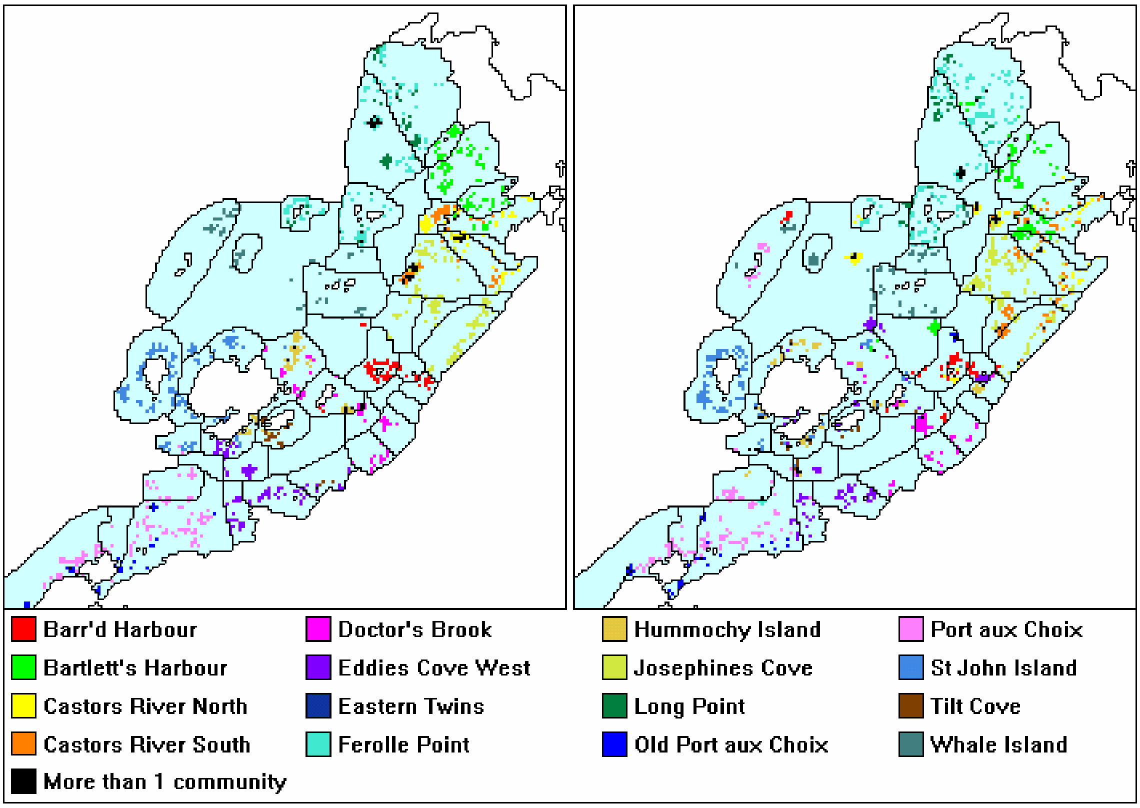


Figure 14. *Distribution of community lines, day 27, 1976.*
Left: under the community territories scenario;
Right: under the baseline scenario

Under the community territories scenario in 1976 Ferolle Point boats share their territories (four fairly large areas) with only four to six other boats, from Long Point. These areas have high concentrations of lobsters. Whale Island boats have lines fishing on six different areas and all of these have high concentrations of lobsters. Only two of the areas fished by boats from Whale Island are shared with boats from another community, and those set only a few lines, so competition in minimal.

Examining the case of St. John Island, under the community territories scenario boats from this community have lines in nine areas surrounding the Island (Fig. 14, Left). Although four of their areas are being shared with boats from other communities—specifically Port aux Choix, Eddies Cove West, Hummocky Island, and Whale Island, with a maximum of 20 boats—they concentrate their lines in areas where boats from these other communities are not fishing. The focus is mainly on areas to the north and west of the island where there are high concentrations of lobsters (Fig. 13). Under the baseline scenario of no community territories, St John Island boats share nine areas with boats from Hummocky Island, Eddies Cove West, Port aux Choix, Tilt Cove, Doctors Brook, and Barr'd Harbour (Fig. 14, Right).

In the case of Bartletts Harbour, where there is always a decrease in catch after the community territory scenario is implemented, the lower catches seem to be related to the lack of concentrations of lobsters. They do not have any other boats sharing their areas in the community territory scenario. Bartletts Harbour lines are more spread out and cover more area under the baseline scenario. Consequently if the community territories were not in place these harvesters would be able to move around more widely to find lobsters.

5.3.2 Experiment 2: Communication among harvesters

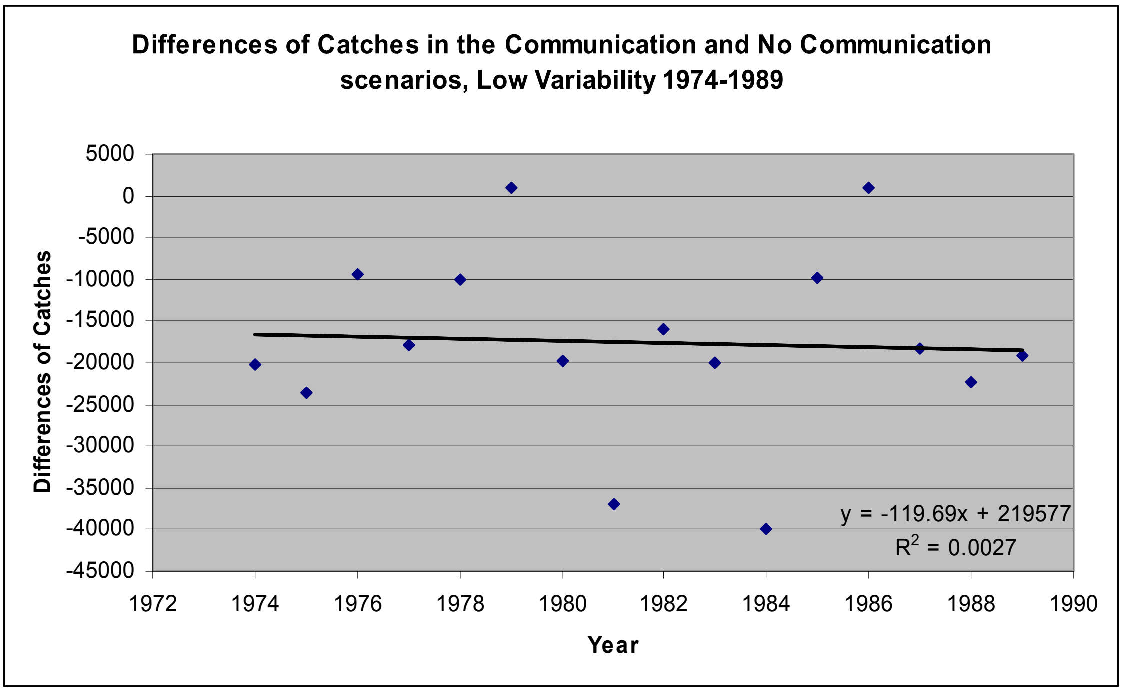
Experiments with the model in which catches are examined under conditions of both low and high variability in the distribution of lobsters show that catches are higher when the distribution is highly variable from one year to the next. This effect can be seen in comparing the mean catches under the two variability scenarios shown in Table 3 above. One reason that this may be so is that it is easier to find lobsters under the high seasonal variability scenario, because high seasonal variability is associated with high patchiness of the lobster population. With lobsters highly concentrated in limited areas they are easy to catch—if the patches are located. In this situation it may well be the case that communication among harvesters will help to raise catch values, since once a patch is found, other harvesters will be able to move their lines there immediately.

So, does communication help? The model is run twice under conditions of high and increasing variability in the lobster distribution—once with no communication among harvesters, and once with a high level of communication ("gossip"). The results are shown in Table 5, and suggest that communication does indeed help harvesters find and catch lobsters when their location is patchy but highly variable.

Table 5. *Total yearly catch values under conditions of high seasonal variability: Without and with communication among boats*

Year	No Communication	Communication	C2 – C1
Mean	569746	582545	+12799
Std. deviation	191982	169316	

The effect of variability can further be seen by running the model with low but slowly increasing variability during the 1972 – 1989 period, followed by rapid increase of variability beginning in 1990. Examining the differences in five year running averages of the total catch values shows clearly that under conditions of low variability communication does not increase total catch—indeed it has an adverse effect. But under conditions of high variability, the effect is positive, and becomes stronger as the degree of variability increases after 1990 (Fig. 15).



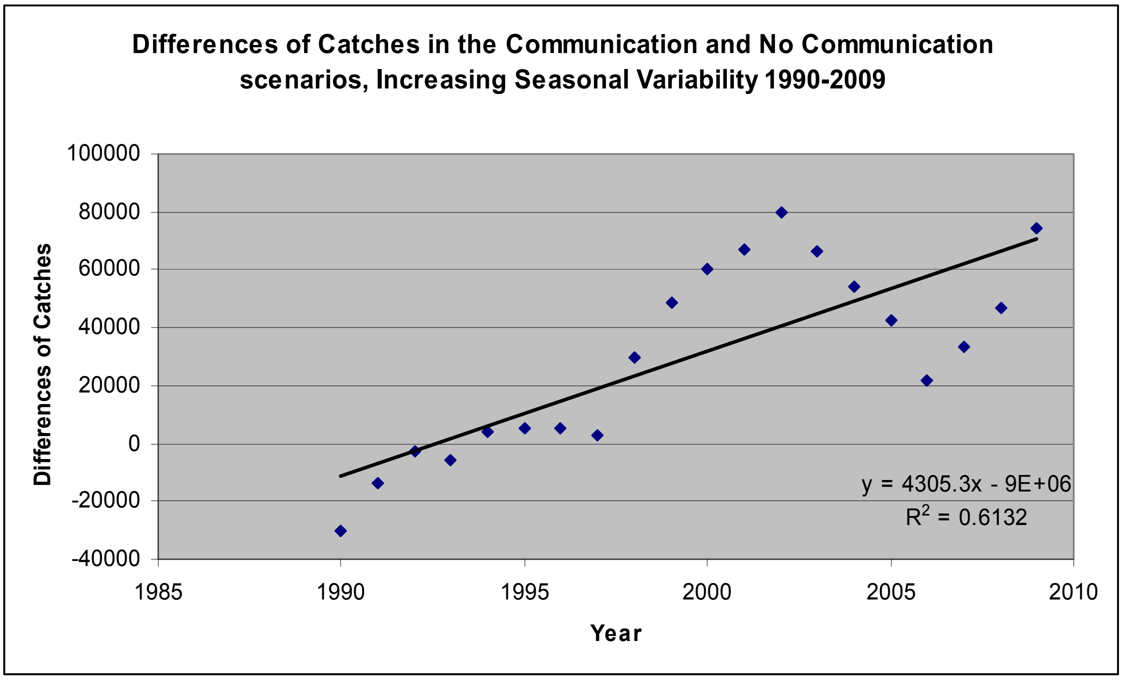


Figure 15. *Differences in catch between No Communication and Communication scenarios.*
Top: low variability in lobster distribution;
Bottom: increasing variability in lobster distribution.

6 Bibliography

Bousquet., F., Cambier., C., Mullon. C., Morand. P. & Quensiere (1994). Simulating fishermen's society. Chapter 7 In Nigel Gilbert & Jim Doran (Eds.) *Simulating Societies: the computer simulation of social phenomena*. London, UCL Press.

Ennis, G. P. (1984a). *American Fisheries Society* **113**: 336-338.

Ennis, G. P., Collins. P. W , and Dawe (1989). *Fisheries and Population Biology of Lobsters (Homarus Americanus) at St. Chads-Burndside, Newfoundland*. Canadian technical Report of Fisheries and Aquatic Sciences.1651.

Fisheries Resource Conservation Council (FRCC). (1995). *A Conservation Framework for the American Lobster. Report to the Minister of Fisheries and Oceans*, FRCC. Ottawa: 95.R.1.

Horne. John. K., Schneider, David. C. (1995). Spatial variance in Ecology. *Oikos* **74**:18-26.

Milon L.W. et al. (1999). *Bio-economic Models of the Florida Commercial Spiny Lobster Fishery*. Florida Sea Grant Report Number 117. FLSGP-T-99-002.

Sutinen, Jon. G. and Gates, John. M. (1995). *SIMLOB: The Resource and Harvest Sector Components of the North American Lobster (Homarus americanus) Market Model*, Final Report to the National Marine Fisheries Service, Northeast Fisheries Science Center, Woods Hole, MA.

Tobler, W. (1979). Cellular Geography. In S. Gale and G. Olsson (eds.) *Philosophy in Geography*, Dordrecht, Holland: Reidel Publishing Company, 379-386.

Wahle, R. A. (1988). Recruitment and body size-dependent habitat selection and predator impact on early benthic phase American lobsters (*Homarus americanus* Milne Edwards). American Zoology. **28**, 14.

Wahle, R. A. (1990). *Recruitment, habitat selection, and the impact of predators on early benthic phase of the American lobsters (Homarus americanus Milne Edwards)*, Ph. D. dissertation, University of Maine.

Wahle, R. A., and Steneck. R. S. (1991). Recruitment habitats and nursery grounds of the American lobster *Homarus americanus:* a demographic bottleneck? *Marine Ecology Progress Series*, **69**: 231-243.

Wu, F. L., and Martin, D. (2002). Urban expansion simulation of Southeast England using population surface modeling and cellular automata. *Environment and Planning A,* **34** (10): 1855-1876.

7 User Manual

7.1 Introduction

In the following 2 chapters the practical use of the model is explained.

In the remainder of Chapter 7 the basic information required for installing and getting started with the **LOBSTER IBM-MODEL** is presented. It describes the basic layout of the user interface and tells you what buttons to select in order to open and run simulations**.**

Chapter 8 is meant as a reference chapter. It gives a short explanation of all commands in the menu structure of the **LOBSTER IBM-MODEL**.

7.2 Getting started

Installing the **LOBSTER IBM-MODEL** is simply performed by copying all the files from the CD-Rom into a dedicated directory on the hard disk. While using the model, all the files generated (Simulations, Legends, Animations,…) are written to dedicated sub-directories in this directory. It is most convenient to create a shortcut to the file named *Lobster.exe* and put it on your desktop.

Files that are overwritten as a result of the use of the model can be re-installed from the CD-Rom. Be careful not to loose your precious work and copy only these files that need to be replaced.

To remove the model and all its files is simply performed by deleting the sub-directory.

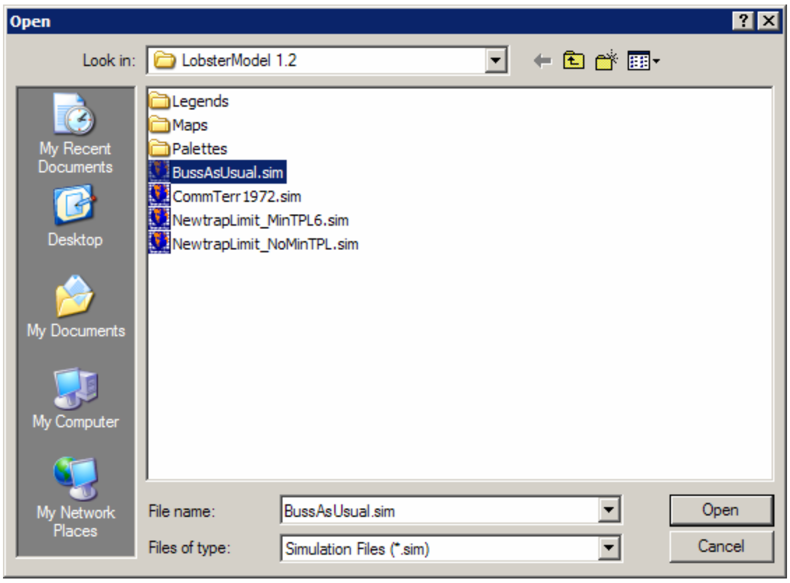
To start the model, press the icon *Lobster.exe* placed as a shortcut on your desktop (see above) or in the sub-directory containing the **LOBSTER IBM-MODEL**.

If the software was installed correctly, the application will be invoked and the *Geonamica – Lobster Fishery Model* application window will open.

The *Open* dialogue window will open and enable you to open an existing .SIM file. A .SIM file contains all the information required by the model to run a simulation.

7.3 Opening a .SIM file

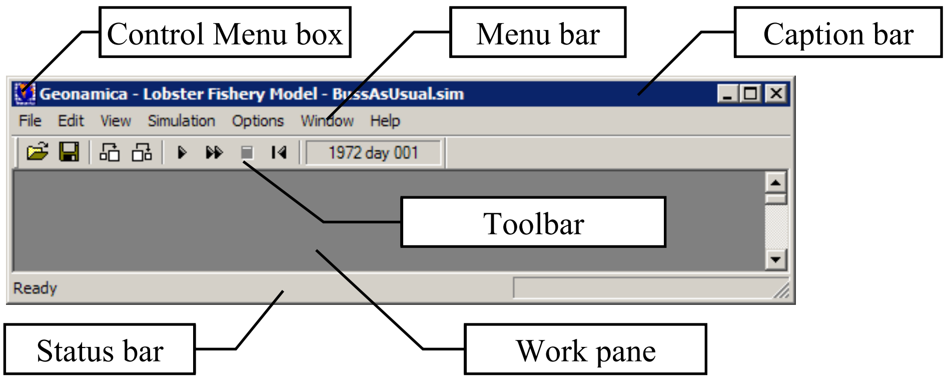
Find the correct file with .SIM extension in the *Open* dialogue window.



During the installation of the LOBSTER IBM-MODEL, a number of example .SIM files are copied in the same directory containing the application.

The LOBSTER IBM-MODEL is build according to the Windows standards. Hence, it is possible to find the file by browsing thought your own computer or another computer in your network. If you have found the correct file select it and click the Open button or double click on the icon of the file.

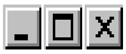
7.4 The Geonamica – Lobster Fishery Model application window



When you start the LOBSTER IBM-MODEL, you will enter immediately in the *Geonamica - Lobster Fishery Model* application window of the simulation environment. This is the window in which you will run your models. To that effect you can arrange input and output windows as you like. Before a simulation file has been opened, the window is empty except for its *Caption bar*, *Status bar*, *Toolbar* and *Menu bar*. The different components of this window will be described in the next paragraphs.

7.4.1 The Caption bar

The *Caption bar*, also called Title bar, of the application window shows the name of the application: *Geonamica – Lobster Fishery Model*. As soon as a simulation file is opened, the title will be extended with the name of this file. Left of the title is the Control menu icon containing the Control menu of the application window, which controls how windows are arranged on the desktop.



To the extreme right of the *Caption bar* three buttons are positioned that enable you to Minimize, Maximize and Close the application window.

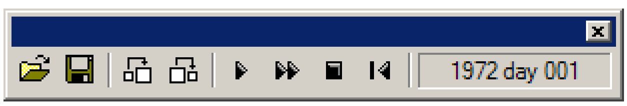
7.4.2 The Menu bar

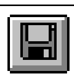
The *Menu bar* of the application window contains the main Menu of the **LOBSTER IBM-MODEL**. The commands are logically organised in the menu so that you will quickly become comfortable with the various functions of the program. The menus are summarised in the table below; subsequent chapters elaborate on the description. Chapter 8 gives an overview of all menu commands.

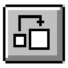
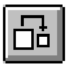
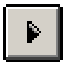
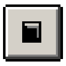
Use this menu	to …
File	manage your simulation files. The printing facilities are also located in this menu, and if you want to exit the program, you can do it from here.
Edit	edit maps, and parameters. All commands for editing are located here. Depending on the type of map that is active, the appropriate edit features are listed and accessible.
View	change the way a map is displayed. Commands to zoom in and out as well as commands to display information on the foreground or background of the maps are part of this menu.
Simulation	control the simulation.
Options	customize the workspace. Commands permit among others to display or remove the *Status bar* and/or *Toolbar*, and select types of output.
Window	manage your windows on the screen.
Help	To get access to the on-line Help of the model.

7.4.3 The Toolbar

The *Toolbar*, also known as Speed bar, gives faster access to some of the more frequently used commands that are also accessible via the menu.



Use this button	to …
	open a simulation file from the disk.
	save a simulation file to disk.

Use this button	to …
	zoom-in. Increase the size of the map on the display.
	zoom-out. Decrease the size of the map on the display.
	Step. Advance the simulation with one simulation step.
	Run. Advance the simulation till the next pause is reached (as set with the Pauses... command).
	Stop the simulation.
	Reset the simulation. State variables and maps take their initial values, parameters keep their actual values.

Besides the above set of buttons, the *Toolbar* also displays the simulation clock. Each time step, the clock is updated by a value of 1 day. It runs from 1972 day 001 to 2012 day 001 during a simulation.

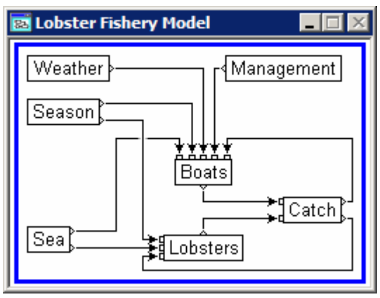
The Toolbar command in the View menu gives the option to display or hide the *Toolbar* in the *Geonamica – Lobster Fishery Model* application window.

7.4.4 The Status bar

The *Status bar* is displayed at the bottom of the application window. This zone provides information to the user on his actions while he is working with the application. The left area of the *Status bar* describes actions of menu commands as you use the mouse or arrow keys to navigate through menus. Also messages are shown describing the actions of *Toolbar* buttons when pressed. The right area of the *Status bar* indicates which of the following keys are latched down: the CAP Lock key, the Num Lock key, or the Scroll Lock key.

The Status Bar command from the View menu gives the option to display or hide the *Status bar* in the *Geonamica – Lobster Fishery Model* application window.

7.5 The Lobster Fishery Model window



The *Lobster Fishery Model* window is by all means the most essential feature of the *Geonamica – Lobster Fishery Model* application window. It shows an overview of the structure of the **LOBSTER IBM-MODEL** at the most synthetic level and enables access to the details of the model at this level but also at lower levels. You should learn to use it as a graphical explorer of the model. You can change neither the model structure, nor its graphical representation. Contrary to all other windows, it is not possible to close the *Lobster Fishery Model* window.

The **LOBSTER IBM-MODEL** has been implemented by means of the Decision Support System Generator **GEONAMICA**®. **GEONAMICA**®-models consist of *Model Building Blocks* (MBBs) that contain the code and/or data required to calculate and execute mathematical operations varying from a single operation (such as the sum of two numbers) to a complex set of interlinked operations (set of mathematical equations). Model Building Blocks are graphically represented in the user interface by means of a rectangle with the name of the MBB in it. They are connected to one another by means of *MBB-Connectors*. To improve the readability of the diagrams, some parts of the model have been grouped in so-called *SuperMBBs* and are shown as a single building block.

Appendix A contains some more details about **GEONAMICA**®

The representation of the model in the *Lobster Fishery Model* window has been created with the help of the following basic elements: *Model Building Blocks*, *MBB-Connectors*, *Connections*, and *MBB-Dialogue windows*.

7.5.1 Model Building Blocks *(MBB)*

Model Building Blocks are represented in the diagrams of the Overview pane by means of a rectangle with the name of the MBB printed in it.

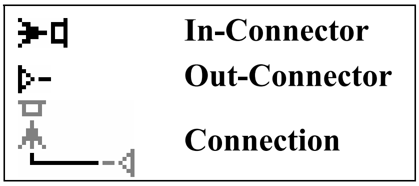
An active Model Building Block is represented in black in the diagrams. When the mouse moves over it, its colours are inverted. Next, if you click in it, the reaction of the system will depend on the type of MBB clicked:

- If the MBB is not a SuperMBB, a dialogue window will open. This dialogue window is the user interface of the MBB. It has the double function to receive the user inputs and to display the model outputs.
- If the MBB is a SuperMBB, then a more detailed diagram of the underlying sub-model is shown, in which each MBB functions as has been explained previously.

A non-active MBB is represented in grey in the diagrams. They are copies of MBBs that are located elsewhere in the *Lobster Fishery Model* window (usually on a different hierarchical level). Their only function is to improve the readability of a diagram.

7.5.2 Connectors and Connections

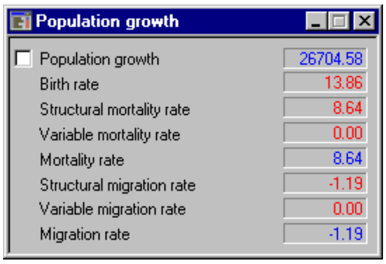
Variables and parameters can be passed from one MBB to the other via Connections, or Pipes. MBBs will dispense variables or parameters with the rest of the model via *Out-connectors*, and will take-in information from other MBBs via *In-connectors*.



The actual exchange between MBBs is possible via a *Connection* (or Pipe) between an Out-connector of the issuing block and the In-connector of the receiving block. For each variable or parameter exchanged, a connection is drawn (except if one of the MBBs is a SuperMBB) in the diagram.

All In-connectors are grouped on the *In-side* of the MBB and all the Out-connectors are grouped on the *Out-side*. The In-side and Out-side of a MBB can never coincide.

7.5.3 Dialogue windows



Each MBB has a dialogue window associated with it. It is the vehicle that permits the interactive exchange of information between the user and the Model Building Block: the MBB will communicate the results (output) of its numerical operations to the user, and it will take in the data entered (input) by the user that are required for the execution of the MBB. It concerns data that are internal to the MBB and that it does not get from other MBBs via its In-Connectors.

In the dialogue window, model results and model inputs are presented in Edit Fields. The difference between input and output fields is indicated by the colour of the text: green for state variables (to be entered as initial values and for the remainder output of the MBB); purple and blue for intermediate variables; black for variables or parameters copied from other MBBs (output of the MBB); and finally red for internal parameters of the MBB (input for the MBB).

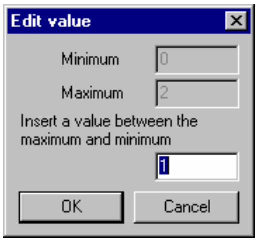
Colour of the text in the Edit Fields of the MBB-dialogue window	Description
RED	Input. Parameter value (can be changed during the entire simulation) or State variable at the initial state.
PURPLE	Input. Variable value. If the edit field is clicked, an extra dialogue will pop-up requesting input necessary for the calculation of the variable
GREEN	Input / Output. State variable that can be changed at the start of the simulation, and that displays output for the remainder of a simulation.
BLUE	Output. Variable value. (read-only)
BLACK	Output. Values copied from other MBBs. (read-only)

A simple way of finding out whether a field will accept input or not is by simply moving the mouse over it. If the field can be edited, the mouse pointer will change shape in function of the type of data that are to be entered. If the mouse is clicked, the appropriate editor is opened. In Annex B, the meaning of the different pointers is explained.

One of four dedicated editors will open:

Single value editor

This editor enables the user to enter a single numerical value. More precisely it will display a default setting --the result of the calibration of the model-- with the request to replace it by a value within the range specified:



Minimum ≤ value entered ≤ Maximum

If values outside this range are entered, the user is warned.

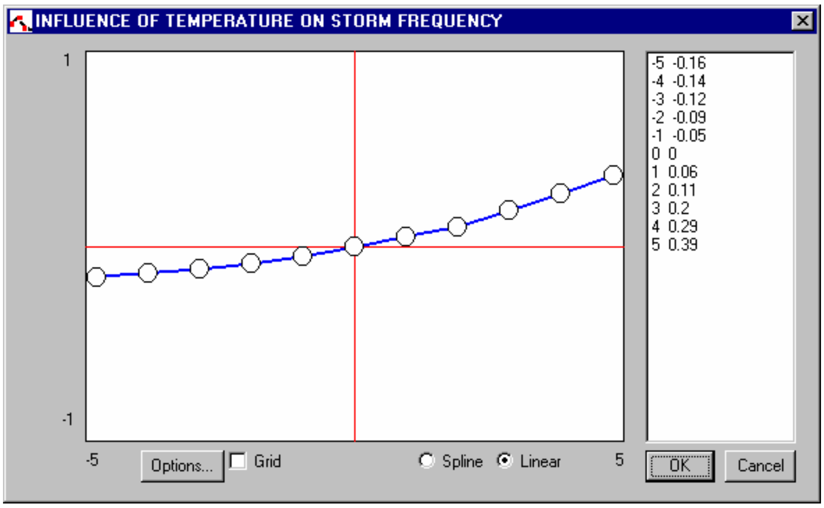
Table (of values) editor

This editor enables the user to enter a series of numerical values. More precisely it will request the user to adjust default values. The values entered will all have to be inside a range specified:

Minimum ≤ value entered ≤ Maximum

If values outside this range are entered, the user is warned.

Graph editor



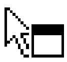
This editor is used very extensively in the LOBSTER IBM-MODEL to define two-dimensional relations: time series, distance decay functions, etc. It has two main parts: to the left is a graphical representation of the two-dimensional relation, and to the right a list box with the co-ordinate pairs defining the relation. The relation can be edited in the graphical part only. Changes made are immediately visible in both parts.

A relation is entered or changed by entering points in the graphical representation. Move the cursor to the abscissa position for which you want to enter a new (ordinate) value. Double-click with the left mouse pointer to add a point to the relation. As a result a little circle will be drawn and line segments will connect the new point to the nearest points left and right in the graph created thus far. To reposition a point, click it with the left mouse pointer and move it to its new position with the left mouse button held down. Or, in order to position it to a particular x-y location, click it with the right mouse button and enter the abscissa (x) and ordinate (y) values in the *Edit point* dialogue window which opens. To remove a point from the relation, double-click it with the left mouse button.

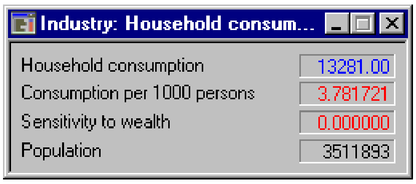
The ranges of the x-axis and the y-axis can be changed by clicking the Options button. Next, the *Options* dialogue window will open which enables to enter the lower and the upper bounds of the x and y axis.

When the Grid check box is clicked, a grid is drawn in the graph window.

Finally, it is possible to decide on the type of relation connecting the points in the graph. Click the Linear radio button if you want to connect them by means of linear line segments, or click the Spline radio button if you want to connect them by means of a spline curve. Mind you that it is not possible to calculate a spline curve for just any set of points in the graph.



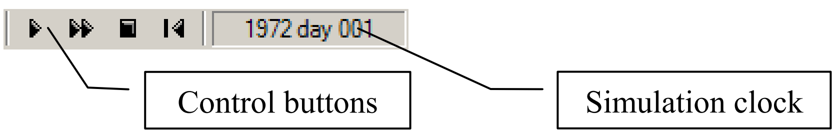
Dialogue editor



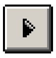
The dialogue editor is opened to enter the data for a dedicated sub-model required to calculate the value of an intermediate variable (printed in purple). This editor is used frequently to edit data that are subject to stochastic perturbations.

7.5.4 Running a simulation

Once a simulation file has been opened and the *Lobster Fishery Model* window is displayed in the application window, the program has read the default values for all the parameters as well as the initial values for all the state variables of the model. The program is ready to run a simulation. You can run a simulation with the simulation control buttons in the *Toolbar* or with the commands in the Simulation menu.



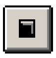
The outer right box displays the Simulation clock, which indicates the progress of the simulation: it displays the year and the day in the season until which the simulation has run. Initially it displays 1972 day 001, which is the first day of the simulation.



To verify that the program is ready to run, press the Step button. Once pressed, the **LOBSTER IBM-MODEL** will go through a number of essential phases (such as the initialisation and testing of its inputs) that are of no direct interest to the user before it will make 1 simulation step. This will take a while. You will notice that the action is finished when the simulation time changes from '1972 day 001' to '1972 day 002' and all the opened maps and dialogue windows are updated.

You can undo the simulation step by pressing the Reset button from the *Toolbar*. This action will reset all the state variables of the model to their initial value, but will not do so for the values of the parameters. These are not influenced by Reset and remain unchanged. In the sequence of events explained in this paragraph we have not changed parameter values, hence, the latter is of no importance now, but it will be an important issue once you will be changing parameter settings while running and testing different scenarios with the system.

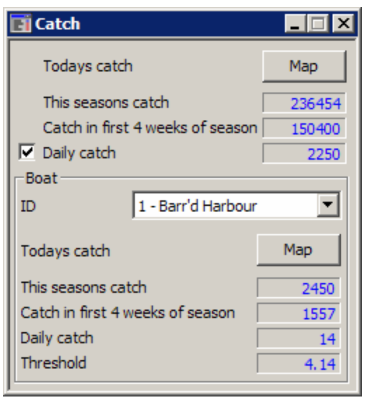
To perform the simulations for the whole simulation period, press the Run button. The simulation will start running and the progress can be followed as the simulation clock and all opened maps or dialogue windows are updated.



The simulation can be paused by pressing the Stop button. Pressing the Run button will resume the simulation. You can also pause the simulation at predefined instances, by means of the Pauses… command in the Simulation menu.

7.6 Viewing simulation output

7.6.1 Viewing output in dialogue windows



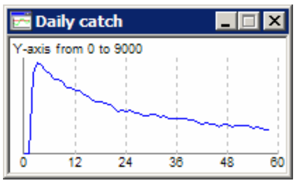
The outputs of the **LOBSTER IBM-MODEL** are not visible unless you open the parts of the model that you want to consult. To that effect, it suffices to click an MBB and to open its associated dialogue window. The state variables in the dialogue window will be updated as the simulation progresses through time.

The most important state variables are preceded by a white check box. When you click the check box one of two things will happen:

Choroplete maps *(not applicable in the LOBSTER IBM-MODEL* yet)
If it concerns a spatial variable, a small choroplete map will open, which will show the geographical distribution of the specific state variable

aggregated at the levels of the fishing areas. This map is dynamically updated: if you keep it open during the simulation, you will be able to see how the distribution of the variable changes in time and space. This mapping tool is equipped with a legend that automatically sets the lower limit of the lowest class and the upper limit of the highest class so that all the values for all regions can be shown. The number of classes has been set to 8 and cannot be changed.

Time graphs



If it is not a spatial variable, a small window is opened showing the variable in a time graph. The value of the graph is drawn from year to year. The Y-axis of the graph is set automatically so that the minimum and the maximum value of the time scale is shown.

To close the choroplete maps or time graphs, you should click the Close-window button in the *Caption bar* of the respective windows.

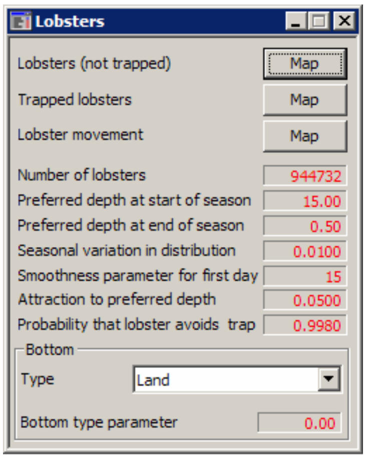
The **LOBSTER IBM-MODEL** is a very open and flexible instrument as far as viewing and displaying its output is concerned. Indeed, you have complete freedom in selecting and organising the input and output windows. The **LOBSTER IBM-MODEL** has tens of windows available with results and information that relate to the model. All these windows are accessible via the earlier explained structures and logic, and can be selected via the *Lobster Fishery Model* window or the main menu. When you open a window, dialogue window, map or graph, then this object stays open until you close or minimise it. While the simulation is running all the opened objects are updated. This offers the important advantage that information about different parts of the model can be seen simultaneously and enables analysis aimed at finding out how different parts of the model are linked and change accordingly.

If you place the pointer on the caption bar of the window, you can then move the window to another position by keeping the left mouse button pressed. During a simulation exercise it is advisable to order the dialogue windows so that they are all grouped on the one side of the application window and the maps on the other side.

Tip

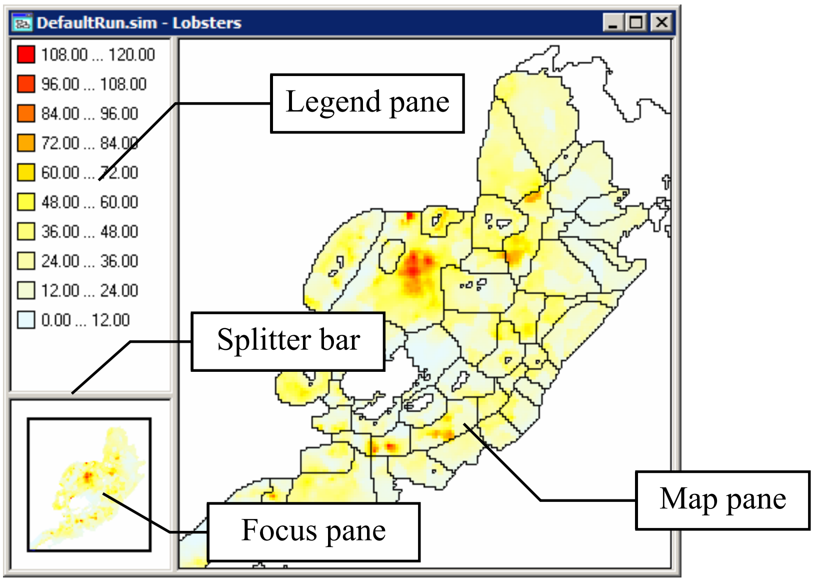
Important ! You should learn to manage output windows carefully and to close or minimise those windows that you are not directly interested in. If not, your screen will get overloaded with ‘left-over’ information. Moreover, every opened window is updated by the **LOBSTER IBM-MODEL** and updating information takes execution time and memory space. Hence the program may slow down considerably. It is certainly advisable to close as much as possible the maps that are not of immediate interest.

7.6.2 Viewing detailed Map output



Many state variables of the model are presented on detailed cellular map representations of the modelled region. These can be opened by pressing the Map buttons in the dialogue windows. Typically, the Map buttons are preceded by the name of the variables mapped.

Windows presenting map results are split into 3 viewing areas, called *Panes*. Panes are separated from one another by means of *Splitter bars*. You can displace the splitter bars to change the size of the panes. To this effect, position the cursor over the splitter bar. Move the mouse with the left mouse button hold down to move the splitter bar and change the size of the panes accordingly.



- The pane to the left of a map window --called the *Map pane*-- contains a geographical representation of the variable modelled in the study region.
- The legend of the map is displayed in the upper left pane --called *Legend pane*.
- The lower left pane --called the *Focus pane*-- of the map window shows the mapped region in small. The wire-frame (an inverted rectangle) shown in this small map localizes the region that is displayed in the map pane in more detail. If the mouse is placed inside this wire-frame, the frame can be moved while holding the left mouse button clicked down. While the wire-frame is moved, you will notice that the focus of the map in the Map pane is changing accordingly. The focus of the map can also be changed by means of the scroll-bars of the Map pane. The Legend pane too is equipped with a vertical scroll bar.

When the simulation is running, the map window will be updated and the changed spatial distribution will be displayed in the Map pane.

The commands of the View menu permit to change the manner in which spatial results of the model are displayed in the Map pane. Display options include amongst others: Zoom in, Zoom out and Show elevation. For a complete overview, consult the View menu in Chapter 8.

The legend of a map can be changed interactively. This is explained in the next paragraph.

7.7 Customizing map output

Each map in the **LOBSTER IBM-MODEL** is represented with its dedicated legend. These legends are completely customisable. The legends may contain the colour information for the different legend items or they may apply colours from a palette file. That is why, this chapter contains a section about the **LEGEND EDITOR** and a section about the **PALETTE EDITOR**. For most users it suffices to only use the **LEGEND EDITOR**.

7.7.1 The Legend editor

To customize the legend of a map it suffices to double-click in the Legend pane of the map window. As a result the *Legend editor* dialogue window will open.

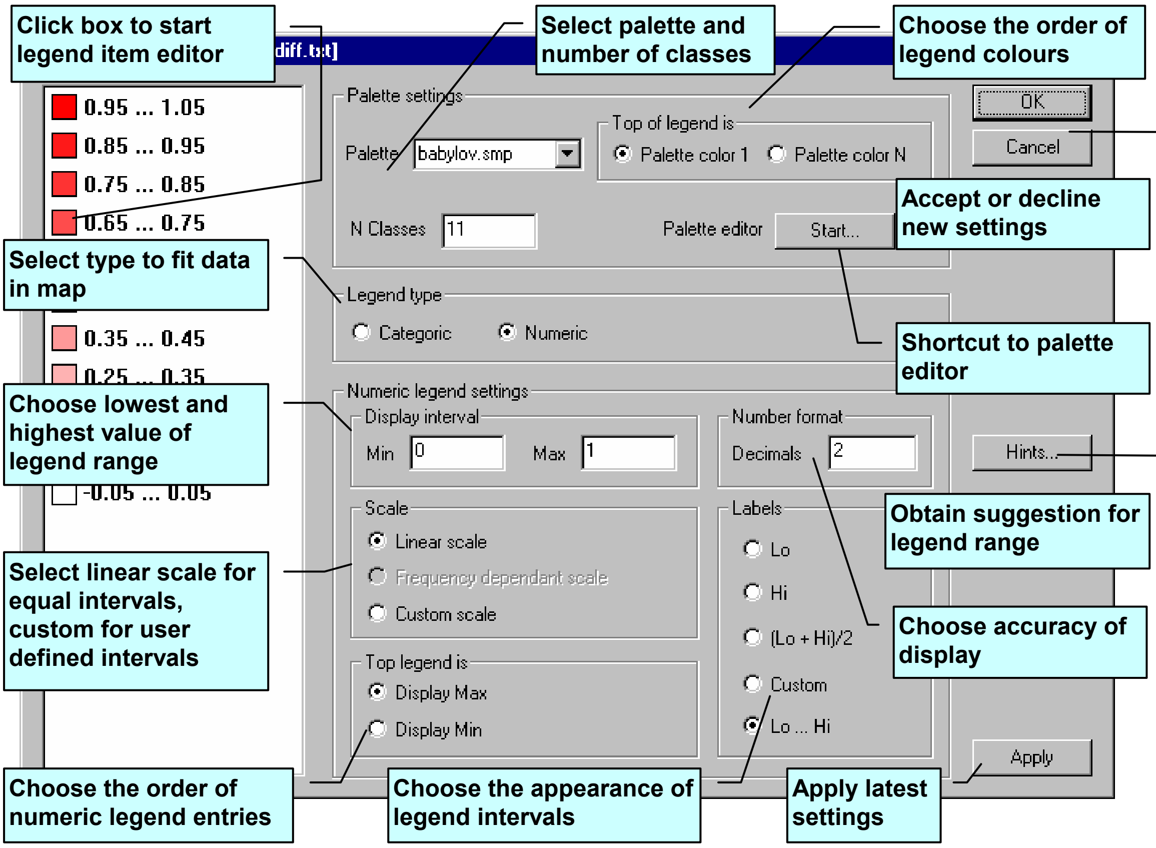
In the figure below, the dialogue window is shown and the different options are explained.

It is possible to apply the colours from a ready-made palette from the palette sub-directory. In the illustration below the 'babylov.SMP' palette is used. If you wish to customize the colours used there are two options (besides selecting an alternative palette):

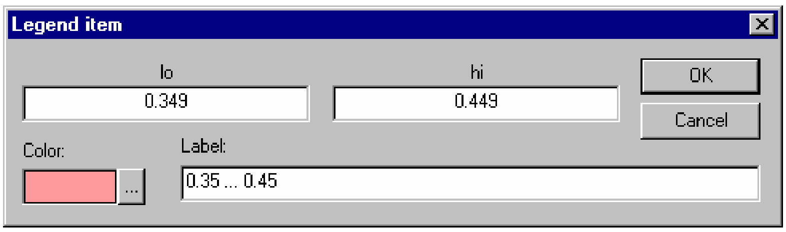
(1) define the colour of legend items by means of the **LEGEND ITEM EDITOR**, see Section 7.7.2;
(2) modify the palette using the **PALETTE EDITOR**, see Section 7.7.3.

Tip

Important ! Modifying the palette will affect all legends using this palette. Modifying the colours in the legend itself will only affect the legend of the particular map. If you are not certain whether a palette is used elsewhere in the application it is better not to use the **PALETTE EDITOR**, but use the **LEGEND ITEM EDITOR** instead.



7.7.2 The Legend item editor



When you click in a colour box of a category in the *Legend editor* dialogue window, the **LEGEND ITEM EDITOR** is invoked and the *Legend item* dialogue opens. In this dialogue you can define the names of the labels of the legend classes and set the lower (lo) and upper (hi) limits of the class range. Also you can select a new colour to represent the cells belonging to the class. If you have configured the **LEGEND EDITOR** to create a linear scale, then you can only set a new colour with the **LEGEND EDITOR** itself.

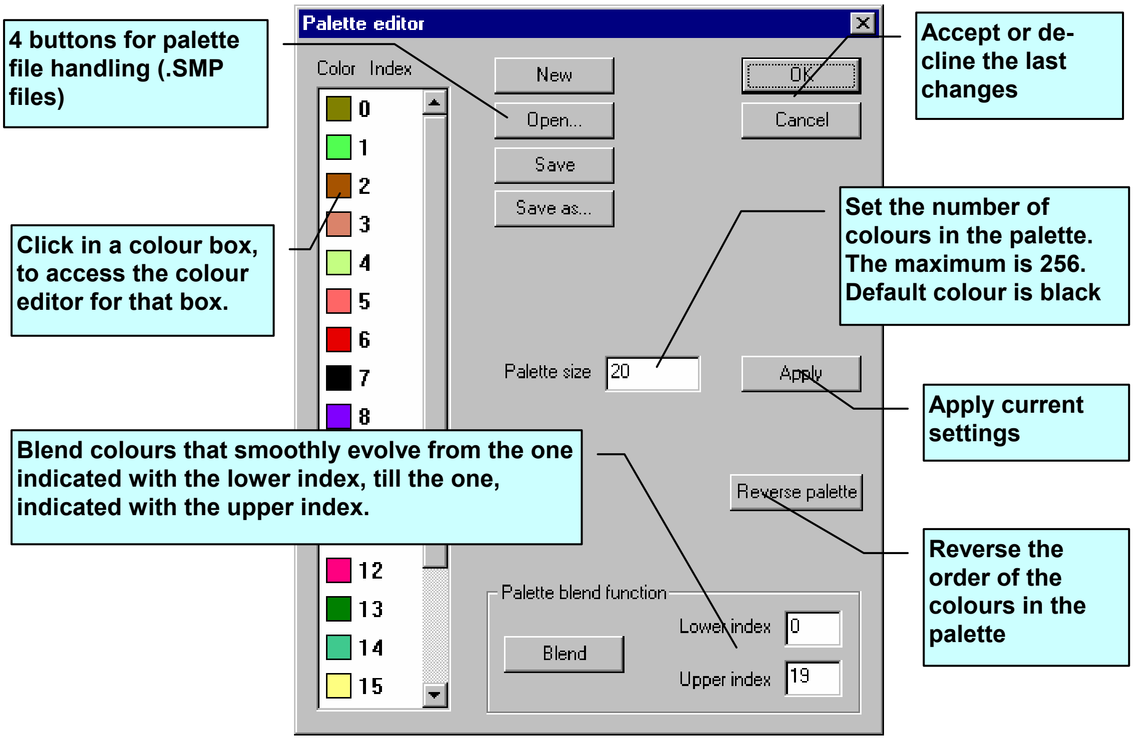
Your configurations are saved as part of a legend file associated with the map that you are defining the new legend for. This file is saved as soon as you press the OK button in the *Legend editor* dialogue window.

7.7.3 The Palette editor

With the **PALETTE EDITOR** you can edit the colour palettes used in the legends of the **LOBSTER IBM-MODEL**. It is also possible to create new colour palette files (.SMP extension).

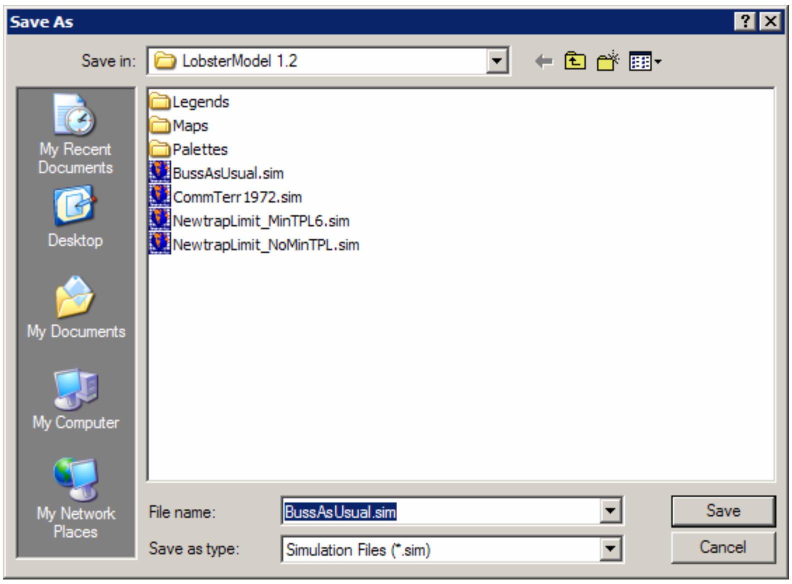
When the PALETTE EDITOR Start… button is selected in the *Legend editor* dialogue window, the *Palette editor* dialogue window opens. In the figure below, this window is shown and the relevant settings are explained.

The changes that you make to a palette can be saved by clicking the Save or Save as… button. The changes are then saved in a palette file (.SMP). The PALETTE EDITOR is closely associated with the LEGEND EDITOR. The latter enables to define the way in which data are presented on a map.



7.8 Saving simulation results

Working with a simulation model is to a large extent an iterative process: the user explores different combinations of interventions and alternatives to learn interactively which are likely to satisfy the preset criteria. These explorative exercises require a capability of storing simulation results in an efficient manner.



The LOBSTER IBM-MODEL saves its output in exactly the same format as its inputs. Hence, you can always use saved simulations as the input for a new run. Keep in mind however that the simulation clock will always be set back to 1972 day 001 when you continue a simulation in this way.

The Save as… command from the File menu allows you to store simulation results in a straightforward manner. When the command is selected, the *Save As* dialogue window opens, requesting you to enter the name of the file. If the name of the current simulation file is entered, then it will be overwritten, and the original information will be lost. Overwriting files can be avoided simply by choosing another file name than the current one.

The Save as… command will only save final results of the model (current values of parameters) but will not store intermediate results. To save the latter, you should make use of the File simulation output command from the Options menu. See for more information on this Paragraph 8.5.2.

It is also possible to make movies of dynamic maps during a simulation and store them for later use. To that effect us the Animations… command from the Options menu described in Paragraph 8.5.1.

7.9 Printing simulation results

Printing simulation results can be done in different manners. The **LOBSTER IBM-MODEL** supports very directly the printing of maps. To this end, you can invoke from the File menu the commands Page Setup…, Print Preview, Print Setup… and Print…. The **LOBSTER IBM-MODEL** prints the map and legend on separate pages.

The maps of the **LOBSTER IBM-MODEL** are generally raster maps with a resolution of 200 meters per cell. You can print them on your preferred scale. First select Page Setup… to decide on the scale of the map, expressed as the number of cells that you want to print per unit of measurement (for example 10 cells per cm for a **LOBSTER IBM-MODEL**-map of 180 by 140 cells, designates a picture of 18 by 14 cm. Given a cell size of 200 meter, this represents a map at a scale of 1/200.000). Optionally select Print Preview to get an idea of how the printed maps will look like. Press the Print… button in the *Print Preview* window or from the *Toolbar* to print the map. In the *Print* dialogue window that will appear press OK to print both map and legend, alternatively select page 1 or 2 to print respectively the map or the legend.

Maps from the model can also be printed by clicking the right mouse button in the Map pane. As a result a Copy button is drawn on top of the map. Next click this button to put the map on the Clipboard. From the Clipboard the map can be copied in other applications and printed. The same procedure applies for the legends of maps.

If you desire printing output from a particular MBB, then click it to open its dialogue window. Once the dialogue window is open, key-in Alt + Prt Sc. This will put the dialogue window on the Windows Clipboard. Now open your Paint or Word Processing package and select Paste from its Edit menu. Once the dialogue window has been pasted, print the dialogue by using the print facilities of the receiving package. This procedure works not only for dialogs, but also for all the windows in the **LOBSTER IBM-MODEL**.

7.10 Exiting the Lobster IBM-Model

Closing and exiting the **LOBSTER IBM-MODEL** is simply done by selecting the Exit command from the File menu. If you have simulation files open and you have made any changes, the application will ask whether you want to save simulation results, before exiting. If you confirm to be interested in saving results then you will get the *Save As* dialogue window. More information on saving simulation results is given in Paragraph 7.8 of this manual.

7.11 If you experience problems

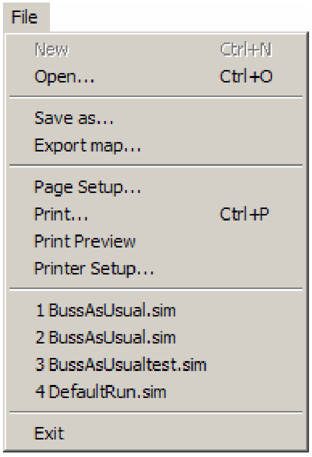
If you experience problems installing or running the **LOBSTER IBM-MODEL**, contact:

Research Institute for Knowledge Systems bv
To the attention of **Inge Uljee** or **Guy Engelen**
P.O. Box 463
6200 AL Maastricht
The Netherlands
Tel: +31 (43) 388.33.22
Fax: +31 (43) 325.31.55
E-mail: info@riks.nl
Http: www.riks.nl
Ftp: ftp.riks.nl

8 The Menu System

This chapter explains the different functions that are available from the menus of the **LOBSTER IBM-MODEL**. The menus are treated as they appear in the *Menu bar* from left to right and per menu from the top to the bottom.

8.1 File Menu



Use the File menu to open or save a simulation file, to export and print maps, and to exit the **LOBSTER IBM-MODEL**.

8.1.1 New Command

Use the **GEONAMICA®**-command New to create a new simulation file. When New is selected, a new simulation file can be created, for which you can enter the relevant values for variables, parameters, maps, etc.

Tip

Important ! The **GEONAMICA®**-command New is not available in this version of the **LOBSTER IBM-MODEL** and is therefore greyed out.

New simulation files can be created with the **LOBSTER IBM-MODEL** in a relatively easy way, by adjusting the values of parameters and variables via the user interface and by then saving the simulation under a different name by means of the Save simulation as... command from the File menu.

8.1.2 Open... Command

Use the Open... command to open a simulation file stored on a disk. You can open only one simulation file at the time.

When you select Open ..., the *Open* dialogue window appears. It shows all the files that are in the work directory and have the right extension (.SIM). If the name of the file of your choice is not visible in the list box, use the scroll bars to move through the list of filenames in the directory or disk you are working in. If the file you want to open is not in the current directory or on the current disk, use the scroll list or browse symbols in the section named *Look in* to change directories, drives or network sites.

Double-click the name of the file you want to open. You can also type the name and path of the file in the *Filename* edit field. Press the Open button after you have typed the name of the file.

This command is identical to pressing the Open button in the *Toolbar*.

8.1.3 Save as... Command

Use the Save as... command to save a simulation to a disk. When you save a simulation, the *Save As* dialogue window will appear. This window allows you to specify the name of the file and the location where you want to store it. The **Lobster IBM-Model** automatically adds the extension .SIM to the name of a simulation file.

Tip

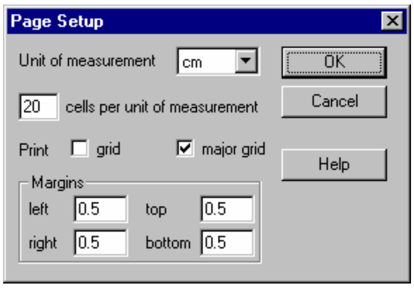
Important ! Be careful not to change the extension of the files.

This command is identical to pressing the Save button in the *Toolbar*.

8.1.4 Export map... Command

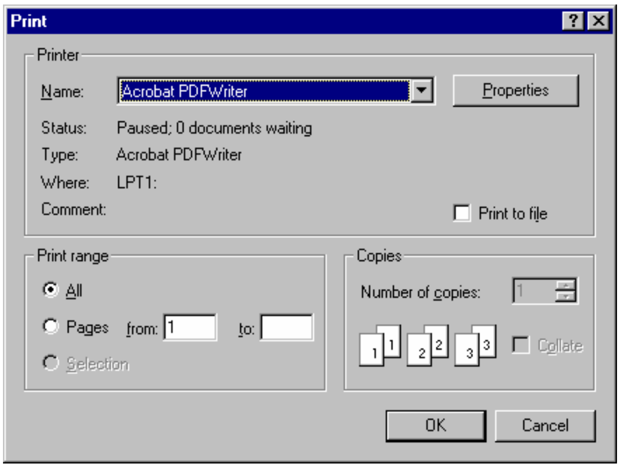
Use the Export map... command to export a map from the simulation to a file on the disk. Exporting maps is possible for all maps. The map to be exported has to be in the active window before the Export map... command will be active. When selected, the Export map... command will open a *Save As* dialogue window displaying all the files in the working directory with the correct extension. If the user changes the type of file to export in the appropriate scroll list, he will be able to export files in IDRISI image (.IMG extension) or ArcInfo ASCII (.ASC extension) format.

8.1.5 Page Setup... Command



Use the Page Setup... command to decide on the size and scale at which you want the **Lobster IBM-Model** to print the active map. When the command is selected, the *Page Setup* dialogue window is opened, enabling you to enter how many cells (of 200 meters) you wish to print per measurement unit (cm or inch). You can also switch on the options to print the 250m grid or a coarser 2500m major grid on the map. Lastly it is also possible to adjust the margins of the page.

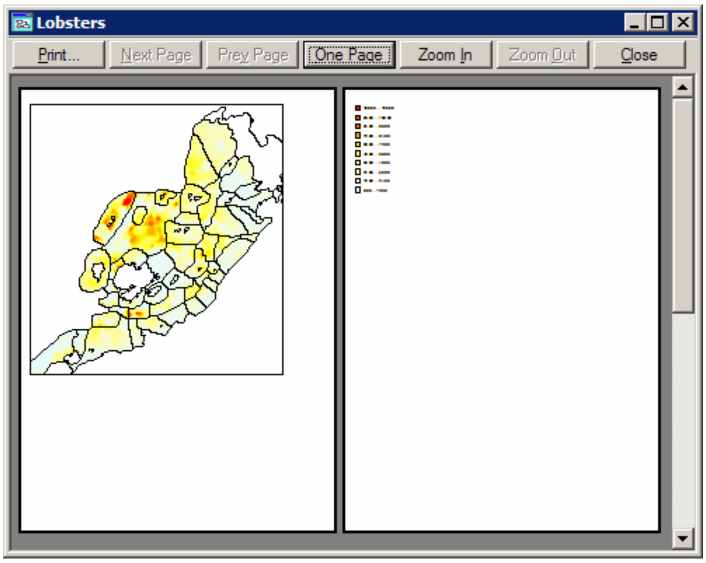
8.1.6 Print... Command



Use the Print... command to print the map in the active map window.

This command is identical to pressing the Print button in the *Toolbar*.

8.1.7 Print Preview Command



Use the Print Preview command to get a preview of the printed map on the screen.

8.1.8 Print Setup... Command

Use the Print Setup... command to prepare your printer so that the **LOBSTER IBM-MODEL** will correctly print the simulation results.

8.1.9 List of Recent Files (1, 2, 3, 4)

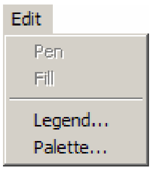
The **LOBSTER IBM-MODEL** keeps track of the 4 most recently opened simulation files. It will display their names and path in the File menu. If you select one of the 4 files, it will be opened.

8.1.10 Exit Command

Use the Exit command to quit the **LOBSTER IBM-MODEL** and return to Windows.

If you have made changes to simulation files since the last time that you saved the simulation, the *GEONAMICA* dialogue window appears asking whether you want to save the simulation file before exiting the **LOBSTER IBM-MODEL**. If you answer yes to this question, you are given the opportunity to save the results as explained in the paragraph on the Save as… command.

8.2 Edit Menu



The Edit menu offers access to the editors for map files, legend files and palette files.

Important ! Currently there are no editable maps in the **LOBSTER IBM-MODEL**. As a result, the **Pen** and **Fill** menu commands are greyed out.

8.2.1 Pen Command

Use the Pen command to pick a value from the legend of the map (by clicking the radio button next to a reading) and entering it in the map by clicking the cell of your choice.

This command will change the pointer into a pen.

While the function is selected, the menu option is preceded with a mark.

8.2.2 Fill Command

Use the Fill command to pick a value from the legend of the map (by clicking the radio button next to a reading) and entering it in larger, contiguous areas of the map. The Fill command will change the values of the all joined (directly adjacent, not diagonally) cells that are in the same state (or have exactly the same value) as the cell that is clicked. All the adjacent cells will get the value that is selected from the legend.

This command will change the pointer into a bucket.

While the function is selected, the menu option is preceded with a mark.

Tip

Important ! When using the Fill command ensure that all cells bordering and outside the area that you wish to fill have a different value from the cells inside. If you ignore this, the fill algorithm will try to find the border and might end up changing the whole map to the new value.

8.2.3 Legend… Command

Use the Legend… command to open the **LEGEND-EDITOR**.

The **LEGEND-EDITOR** enables you to adjust the legends of all the maps in the **LOBSTER IBM-MODEL** and to create new legends. See also Paragraph 7.7.1 of this manual.

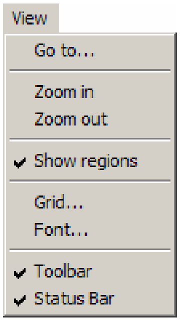
8.2.4 Palette... Command

Use the Palette... command to open the **PALETTE-EDITOR**. The **PALETTE-EDITOR** enables you to adjust the palettes of all the maps in the **LOBSTER IBM-MODEL**. See also Paragraph 7.7.3 of this manual.

Tip

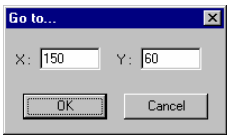
Important ! Modifying a palette will affect all legends that use this palette. Modifying the colours in the legend will only affect the legend belonging to the particular map. If in doubt do not use the **PALETTE-EDITOR**, but use the **LEGEND ITEM EDITOR** instead. See also Paragraph 7.7.2.

8.3 View Menu



Use the View menu to change the manner in which the maps are presented in the active map window and to show or hide the *Toolbar* and *Status Bar*.

8.3.1 Go to... Command



Use the Go to... command to move the cursor to a specific cell on the map. Selecting this command from the View menu opens the *Go to...* dialogue requesting to enter the co-ordinates of the desired cell. When you have entered the co-ordinates and clicked OK the pointer will move to the desired cell in the active map.

8.3.2 Zoom in Command

Use the Zoom in command to increase the size of the map in the active map window by a factor 2.

This command is identical to pressing the Zoom in button from the *Toolbar*.

8.3.3 Zoom out Command

Use the Zoom out command to increase the size of the map in the active map window by a factor 2.

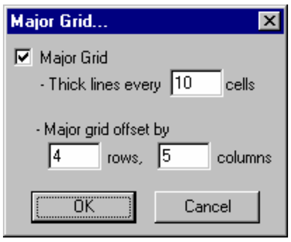
This command is identical to pressing the Zoom out button from the *Toolbar*.

8.3.4 Show Regions Command

Use the Show regions command to draw (or remove) the boundaries of the fishing areas on top of all the map of the application. The boundaries drawn are those defined in the *Regions map*.

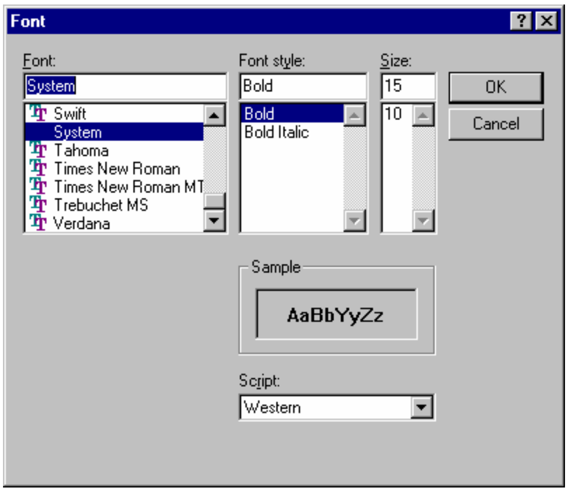
While the function is selected, the menu option is preceded with a tick mark.

8.3.5 Grid... Command



Use the Grid... command to draw a (Major) grid on top of all the maps opened in the application window. When Grid... is selected, the *Major Grid...* dialogue window opens requesting you to check the *Major Grid* check box. Next you are to enter the size of the major grid expressed in number of cells. Finally you can offset the origin of the grid by a certain amount of cells (in order to coincide with another reference system).

8.3.6 Font... Command



Use the Font... command to change the Font, Font style and Size of the character set used to print the legends of maps and the informative windows. When Font... is selected, the *Font* dialogue window opens requesting you to specify the characteristics of the font used.

8.3.7 Toolbar Command

Use the Toolbar command to view or hide the *Toolbar* in the application window.

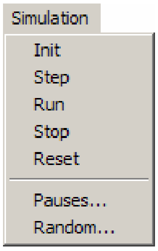
While the function is selected, the menu option is preceded with a mark.

8.3.8 Status Bar Command

Use the Status Bar command to view or hide the *Status bar* in the application window.

While the function is selected, the menu option is preceded with a tick mark.

8.4 Simulation Menu



Use the Simulation menu to control the simulation.

The commands Step, Run, Stop and Reset can also be invoked when pressing the respective buttons from the *Toolbar*.

8.4.1 Init Command

Use the Init command to initialise the simulation. After the model has been initialised all the variables are displaying the correct initial condition (values) of the model. If the Step or Run command is selected the simulation will be automatically initialised if this has not been done manually.

The command Init can also be used to have the model perform a step without advancing the simulation clock. This is especially handy to test the immediate effects of a newly entered (set of) parameter(s) before running the model.

8.4.2 Step Command

Use the Step command to advance the simulation with one time step. If the Step command is selected the simulation will be automatically initialised if this has not been done manually.

This command is identical to pressing the Step button in the *Toolbar*.

8.4.3 Run Command

Use the Run command to advance the simulation till the next pause tab has been reached. The model is updated on a yearly basis. Unless other pause tabs have been set by means of the Pauses... command, the simulation will halt at the end of the simulation period. If the Run command is selected the simulation will be automatically initialised if this has not been done manually.

Once the simulation is running, the user can halt it by selecting the Stop command from the Simulation menu or by pressing the Stop button from the *Toolbar*.

This command is identical to pressing the Run button in the *Toolbar*.

8.4.4 Stop Command

Use the Stop command to interrupt the simulation. Once interrupted, the simulation will halt till the user selects the Step or Run command from the Simulation menu or till the Step or Run button from the *Toolbar* is pressed.

This command is identical to pressing the Stop button in the *Toolbar*.

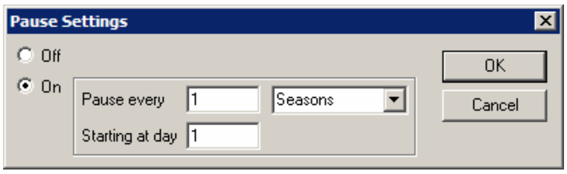
8.4.5 Reset Command

Use the Reset command to switch the simulation clock back to the initial year of the simulation.

The input information of the model is not affected by resetting the simulation. Hence, corrections made by the user to parameters and maps are not affected. However, after resetting the simulation, the state variables, including the dynamic maps are calculated again for the initial year and on the basis of the available input information.

This command is identical to pressing the Reset button in the *Toolbar*.

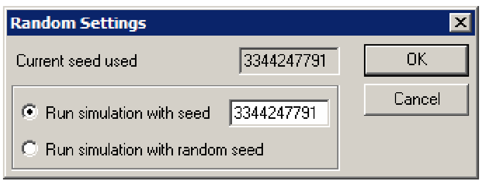
8.4.6 Pauses... Command



Use the Pauses... command to set the pause tabs of the simulation. When Pauses... is selected, the *Pause Settings* dialogue window opens requesting you to switch On or Off the Pause function, to enter the first year that you want to halt the simulation as well as the time interval (in years) for which you want to halt the simulation.

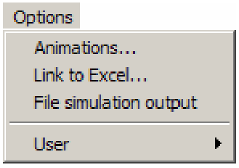
While the function is selected, the menu option is preceded with a mark.

8.4.7 Random... Command



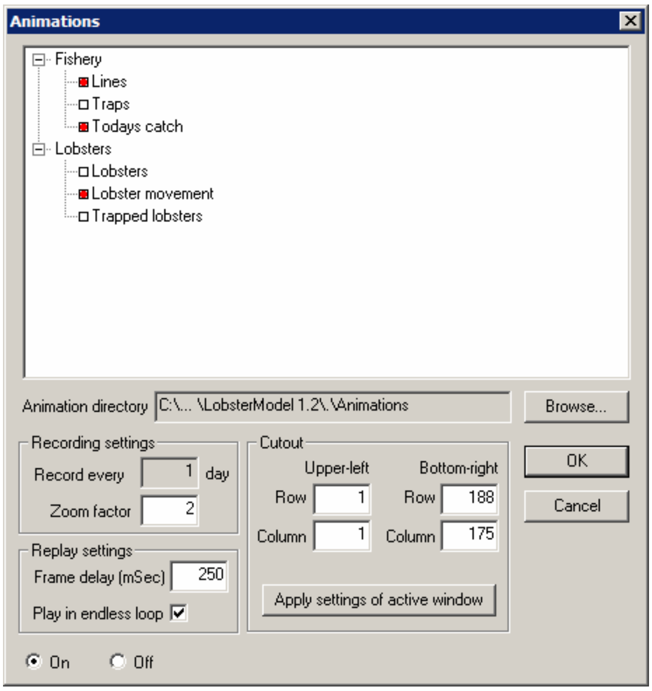
Use the Random... command to specify the stochastic mode of the simulation. When Random... is selected, the *Random Settings* dialogue window opens requesting you to click the radio button for running the simulation in full random mode or in pseudo-random mode. In the latter case you need to specify a random seed number.

8.5 Options Menu



Use the Options Menu to store model results on animated maps or file and to personalise your workspace.

8.5.1 Animations... Command



Use the Animations... command to store dynamic maps produced by the model in the form of .GIF animations. When this command is selected the *Animations* dialogue window opens. To activate the animation function select the radio button On, and to switch it off, the radio button Off. Then the path and name of the file where the animations are stored will appear in the *Animation directory* text field. The program automatically sets the filename and location where the information will be stored. You cannot change it or enter it yourself.

As soon as you have clicked the radio button On the selection tree in the window will become active. In this tree you can indicate which maps you want to store in the .GIF file. To this effect click on the small square situated just left of the name of the map of your choice or double click on the name of the map. The map will be animated if the interior of the square is coloured red.

In the Recording settings section the interval for which maps are animated can be specified. The zoom factor specifies the number of pixels in the animation used to represent one cell on the map. For example, if the zoom factor is 5 an area of 100*50 cells will be displayed by a gif animation of 500 by 250 pixels.

In the Replay settings section you can specify the time that one image (one map) will be shown when animated. Also, you can specify whether the animation is presented as an endless loop or not. In the former case, the first map will be viewed again after the last map has been shown. In the latter, the animation will stop when the last map has been viewed.
By default the whole map is animated. But it is possible to clip from the map a smaller area by specifying the upper-left and bottom-right column and row numbers in the Cutout section of the dialogue. It is also

possible to cut out the part of the map that is visible in the map pane of the active map window by pressing the Apply settings of active window button. This is a particularly useful function because it enables you to home into the area of interest by means of the scroll bars and the zoom functions without a need for finding and entering the coordinates manually.

You can view the animations in most state of the art Internet browsers as well as some graphics packages equipped with an animation facility (for example: Paint Shop Pro) and you can import them in Microsoft PowerPoint.

While the function is selected, the menu option is preceded with a mark.

Tip

Important ! Mind you that the **LOBSTER IBM-MODEL** enables you to generate large animations. For example simulations covering the whole 40 seasons consisting of one map per day will entail some 2800 images. These large animations will be of limited use to the analyst, but will also use vast amounts of disk space. It is therefore recommended to carefully consider the period for which the animation is needed, to advance the simulation manually or by means of the command Pauses… to the beginning of this period, to switch the animation function 'On' for the length of the period and turn it 'Off' again when the period is over.

Link to Excel… Command

Use the Link to Excel… command to establish (or interrupt) a link between **LOBSTER IBM-MODEL** and the Microsoft Excel Workbook LOBSTERS.XLS. This link is a DDE (Dynamic Data Exchange) link. Via this link the **LOBSTER IBM-MODEL** is sending model output to the Workbook while the simulation is advancing. The data transferred to the Workbook consist of a predefined set of state variables of the Macro-model, calculated at both the national and the regional level, as well as summed information relative to the Micro-model. The Workbook LOBSTERS.XLS is available to the user once the simulation is finished. It can be used for further analysis of the simulation data.

It is strongly recommended to save the workbook data under another name than LOBSTERS.XLS, because the file LOBSTERS.XLS is overwritten every time it is linked to a simulation.

While the function is selected, the menu option is preceded with a mark.

Tip

Important ! Note that in order to establish a successful link, it is required that Excel is installed on your machine and that the Workbook LOBSTERS.XLS is in the working directory. If the **LOBSTER IBM-MODEL** cannot find Excel or the workbook LOBSTERS.XLS, the menu option will be greyed out.

Important ! Do not Quit Excel or Close the workbook LOBSTERS.XLS manually while the link with the **LOBSTER IBM-MODEL** is established, rather interrupt the link by selecting the command again. If you ignore this, the **LOBSTER IBM-MODEL** will produce error messages caused by the ill-interrupted connection.

8.5.2 File simulation output Command

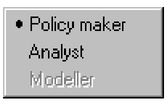
Use the File simulation output command to start (or interrupt) writing the model output to a file. The model will write to file catch results for every boat in every fishing area and for every day in every season modelled. After the simulation, the file is available for further analysis with Microsoft Excel among others.

The model will name the file automatically. A file name will consist of the concatenation of (1) the name of the simulation file, (2) the extension '_out', (3) the sequential number starting with a blank for the first file, and finally (4) the file extension '.TXT'. For example, the second file generated by a simulation on the basis of the simulation file BussAsUsual.SIM will be named: BussAsUsual_out1.TXT.

While the function is selected, the menu option is preceded with a mark.

Important ! Mind you that the **LOBSTER IBM-MODEL** writes a lot of information to the file. For example, simulations covering the whole 40 seasons will generate a file as big as 160Mb. These large files can no longer be opened in Microsoft Excel. Hence they may be of limited use to the analyst, but may also use vast amounts of disk space. It is therefore recommended to carefully consider the period for which data are needed on file, to advance the simulation manually or by means of the command Pauses... to the beginning of this period, to select the function and deselect it when the period is over.

8.5.3 User Command



Use the User command to select one of the three user types supported by the **LOBSTER IBM-MODEL**: Policy maker, Analyst and Modeller.

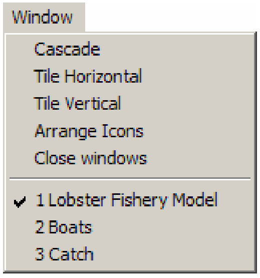
- User type Policy Maker can change all parameters, variables and maps within the range defined by the minimum and maximum values.
- User type Analyst can change all parameters, variables and maps and can also change the minimum and maximum values of their ranges.
- User type Modeller has the same control over parameters, variables and maps as the user type Analyst. In future versions of the **LOBSTER IBM-MODEL** he can also change the configuration of the model.

For the user type selected, the menu option is preceded with a mark.

Tip

Important! In this version of the **LOBSTER IBM-MODEL** user type modeller is not supported, therefore this function has been greyed out in the menu.

8.6 Window Menu



Use the Window menu to arrange the contents of the screen and to activate one of the opened windows.

8.6.1 Cascade Command

Use the Cascade command to arrange multiple opened windows in an overlapped fashion so that the *Caption bar* of each window is visible.

8.6.2 Tile Horizontal Command

Use the Tile Horizontal command to arrange multiple opened windows one above another in a non-overlapped fashion so that all windows are visible.

8.6.3 Tile Vertical Command

Use the Tile Vertical command to arrange multiple opened windows side by side in a non-overlapped fashion so that all windows are visible.

8.6.4 Arrange Icons Command

Use the Arrange Icons command to arrange the icons of minimized windows at the bottom of the screen.

Tip

Attention ! if windows are positioned at the bottom of the screen, they may hide some or all of the icons.

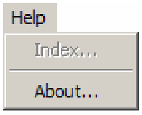
8.6.5 Close windows Command

Use the Close windows command to close all opened windows except for the *Lobster Fishery Model* window which cannot be closed.

8.6.6 List of Windows (1, 2, 3, 4, ... , 9)

A list of open windows is presented at the bottom of the Window menu. A check mark marks the name of the active window. Select a window from this list to make it the active one.

8.7 Help Menu



Use the Help menu to select the type of help that you want the **LOBSTER IBM-MODEL** to display on the screen. The different commands in this menu will permit to look up information about the **LOBSTER IBM-MODEL**, its commands, options, and tools.

8.7.1 Index Command

Use the Index command to get the opening screen of the *Help* file of the **LOBSTER IBM-MODEL**. From the opening screen, you can jump to step-by-step instructions for using the **LOBSTER IBM-MODEL**. Double click the topic that you want help on. A help screen will appear. Once you open help, you can click the Contents button whenever you want to return to the opening screen.

Important ! In this version of the **LOBSTER IBM-MODEL** the on-line help is not operational.

8.7.2 About... Command

Use the About... command to get the copyright notice and version number of the **LOBSTER IBM-MODEL** that you are using. The latter is important if you need assistance with the software from the developers or when you request an update of the software.

ANNEX A: Geonamica® DSS Generator

For the technical implementation, or software coding, of the **LOBSTER IBM-MODEL**, the DSS-Generators **GEONAMICA®** has been used. A DSS Generator is a term introduced by Sprague and Carlson (1982) refering to '*a package of hardware/software which provides a set of capabilities to build specific DSS[s] quickly and easily*'. Hence, this refers mostly to a special purpose software environment for the creation of new DSS applications in a more or less narrowly defined domain.

GEONAMICA® is an *object-oriented application framework*, developed by RIKS bv for use by DSS developers. It is specially tailored for developing Spatial Decision Support Systems featuring models that run at multiple spatial and temporal resolutions. Typically it will combine *system dynamics, cellular,* and/or *agent based models* for this purpose. In particular use is made of *spatial interaction based models*, different kinds of *cellular automata models*, *individual based* or other kinds of *rule-based models*. It is equipped with highly efficient computational techniques and algorithms for addressing spatial problems, but also with additional analytical tools, visualization tools, and input, import, export and output tools. It is equipped with a number of tools for interactive map manipulations, in particular: map editors and display tools for 1-D network and 2-D map objects, map comparison, and overlay-analysis.

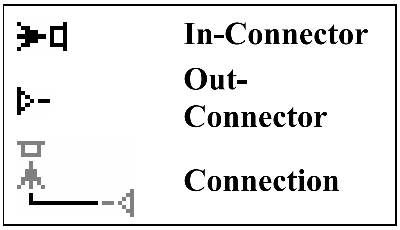
The cornerstone of the **GEONAMICA®** application framework is the way in which it enables the DSS-developer to set up a new modelbase consisting of a set of exchangeable and interchangeable *Model Building Blocks* (MBB) that can be entered, exchanged, re-arranged and re-used in the modelbase of the DSS nearly as easily as Lego building blocks.

A Model Building Block represents a part of a model: an action or process. Hence, it is a more or less complete model varying from a simple mathematical operator to a complete model consisting of coupled mathematical equations performing large numbers of sophisticated calculations. MBBs may simply represent sources of information (i.e. entered from file), while others will transform information as it passes through them, and still others will simply communicate, in a synthetic manner, the outputs of the model to the user. Despite the fact that all these MBBs play different roles in the model, in Object-oriented jargon, they are all *'specializations'* of the same *'abstraction'*, which is essentially a MBB capable of exchanging and transforming information.

Each MBB has two graphical representations:

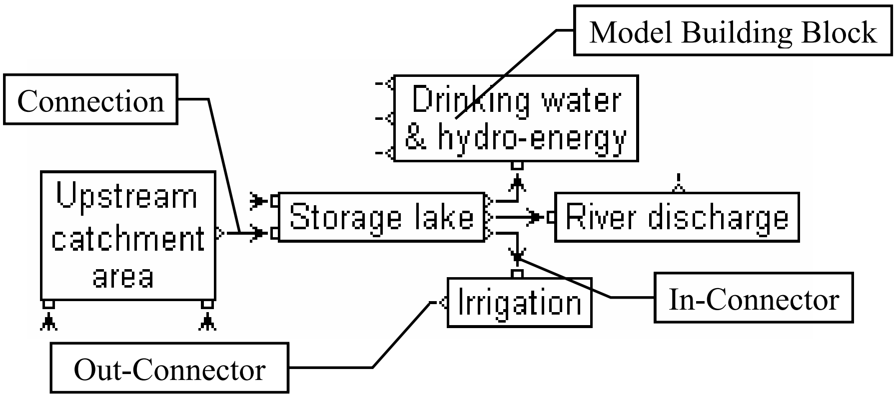
1. A ***rectangle*** (or *box*):

A unique graphical object in the user-interface of the integrated model that shows how the MBB relates and is connected to other MBBs in the integrated model. A user can know from this connection scheme where the MBB gets its input(s) from, and where it sends its output(s) to;



Values of variables and parameters can be exchanged between MBBs via *Connections*. MBBs send information to the rest of the model via *Out-Connectors* and receive information from other MBBs via *In-Connectors*.

Real exchange between MBBs is possible if a Connection exists between the Out-Connector of the broadcasting MBB and the In-Connector of the receiving MBB. For each variable or parameter a connection is made (except if one of the MBBs is a SuperMBB).



A Model Building Block has a so-called In-Side and an Out-Side. The In-Side is where the In-Connectors are placed, the Out-Side is where Out-connectors are placed.

2. A ***user interface*** (UI):

The user interface of the MBB presents itself as a *dialogue window*. The user interface gives the user read/write access to all the MBB specific *parameters* as well as the initial *(input)* values of its state variables. While a simulation is running, it enables read-only access to all the updated values of *output* variables. Entering data in dialogs is done in a manner which will support and protect the user to some extent, because each edit box in a dialogue window knows what type of data it should get from the user: a single number, a vector of numbers, a matrix, or a table (i.e. time series). For each type the appropriate editor is opened when the user clicks in the edit box.

Model results and model inputs are presented in Edit Fields. The difference between input and output fields is indicated by the colour of the text in the edit field: green for state variables (to be entered as initial values and for the remainder output of the MBB); purple and blue for intermediate variables; black for variables or parameters copied from other MBBs (output of the MBB); and finally red for internal parameters of the MBB (input for the MBB).

Colour of text in the Edit Fields of MBB-Dialogue Boxes	Description
RED	Input. Parameter value (can be changed during the entire simulation) or State variable (can be changed at the initial state only).
PURPLE	Input. Variable value. If the edit field is clicked, a dialogue will pop-up requesting input necessary for the calculation of the variable
GREEN	Input / Output. State variable which can be changed at the start of the simulation, and which display output for the remainder.
BLUE	Output. Variable value. (read-only)
BLACK	Output. Value copied from another MBB. (read-only)

Each MBB has its *Documentation page* in the *Documentation system*. It is accessible when the dialogue window of the MBB is opened (by pressing the F1-key or clicking in the dialogue window by means of the Context Sensitive Help cursor). This Documentation page gives technical information about the MBB and may include the mathematical expression, scientific references, the specification of the input and outputs, etc.

The MBB manages the memory for its *parameters* and its *outputs*. An advantage of this design is that it makes the MBBs self-contained and independent. The *inputs* of a MBB are pointers to the memory location where the required output is residing. As an output *X* is always managed by the MBB producing that output *X*, the input is pointing to a memory location managed by the MBB producing the output *X*. A MBB does not know what MBB it receives input from. It is the responsibility of the simulation engine to connect the inputs of the receiving MBB to the outputs of the producing MBB while executing an integrated model.

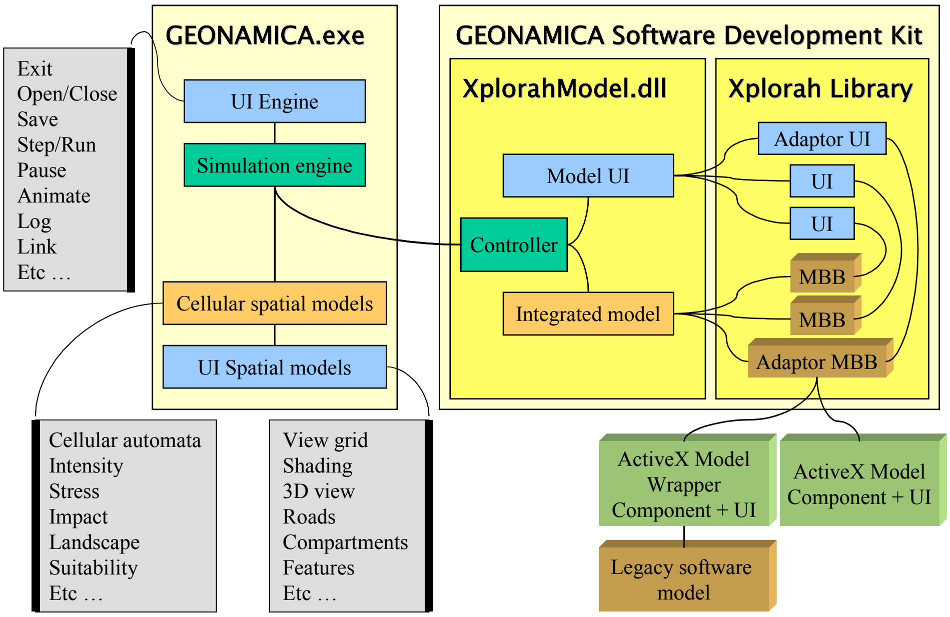
The *Step* function of the MBB contains the software code that implements the mathematical model of the MBB. It specifies how each of the outputs of the MBB changes depending on the time, the current input values, and the current parameter values. Each MBB runs at its own pace. The *Step* function of the MBB is called by the *simulation engine*, it is executed, and the MBB tells the simulation engine when it should be called again. The MBB does not know about other MBBs as they are kept as independent of one another as possible. It is the responsibility of the simulation engine to keep all the MBBs synchronized in time.

Libraries are repositories of MBBs. The entire definition of the MBB (its code, its graphical representation, its dialogue, its connectors) is stored in the MBB Library. When a MBB is included in a model, the block itself is not copied to the model; rather a reference to the block in the library is made. MBBs can be reused in the same model more than once. But, MBBs and Libraries can be re-used in other applications equally well. The factual re-usability will depend largely on the process modelled and the level of generic applicability attained in the implementation.

The **GEONAMICA®** **SDK** *(Software Development Kit*) provides all the templates required to start a new application and a new library. Building an application and connecting the MBBs into an integrated model, is enabled by means of a piece of application specific software specifying which MBBs are part of the application, and how they are interconnected (which inputs are connected to which outputs). This application is the so-called *model.dll*. Not all models have to be available as MBBs within the library in order to be integrated into an application. An external (existing) software model can become part of a **GEONAMICA®** application via an *adaptor MBB*. This is done by means of an *ActiveX Model Wrapper Component*, which wraps the external model into a piece of software so that it looks from the outside like a **GEONAMICA®** MBB, and thus can function within a model like all the other MBBs. The ActiveX Model Wrapper Component delegates most of the work to the actual external model, but performs some missing functionality, such as displaying and effecting the user interface or the conversion of data between the **GEONAMICA®** framework and the external model. If the external model is developed according to the specifications of the COM/ActiveX component technology, hence is an *ActiveX Model Component* equipped with all the necessary interfaces, then, a specific Adaptor MBB can integrate it directly into the application.

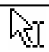
The *user interface of an application* consists of a number of system diagrams with sensitive areas. The diagrams are graphical representations of the application. The MBBs, represented by rectangles are the sensitive areas. They are connected to either more specific diagrams, representing the MBB at a deeper level of detail (when the sensitive area is connected to a *SuperMBB*), or to the user interface (the dialogue window) of the MBB (when the sensitive area is directly connected to a single MBB). The user can navigate through the system diagram hierarchy by clicking the sensitive areas.

GEONAMICA.exe is the piece of software capable of loading a specific application and thus launching the Decision Support System. GEONAMICA.exe is a generic executable, and the integrated model is a project-specific model.dll. GEONAMICA.exe features also a set of cellular spatial models. They perform operations on a grid representation of the region modelled. In the **LOBSTER IBM-MODEL** for example they are invoked to calculate the Cellular Automata Lobster dynamics.



GEONAMICA.exe will also launch the *toolbase* and the *databases* of the application. While an application is running, the tools are automatically invoked as the result of user actions, or they can be purposely selected from the menu system of the DSS. Pre-processing and post-processing tools such as the **OVERLAY-TOOL** and the **MAP COMPARISON KIT** are available as separate applications that are launched independently of GEONAMICA.exe.

ANNEX B: Pointer Shapes

Pointer Shape	Description
Standard MS Windows Pointer Shapes	
	Standard Windows© cursor.
	Standard I-beam cursor for editing windows.
	Sizing cursor to change the width of the window. Displayed when the pointer is on top of a vertical window border.
	Sizing cursor to change the height of the window. Displayed when the pointer is on top of a horizontal window border.
	Sizing cursor to change the width and/or the height of the window. Displayed when the pointer is on top of a window corner.
	Sizing cursor to change the width and/or the height of the window. Displayed when the pointer is on top of a window corner.
	Context Sensitive pointer to get on-line help on the command that you invoke.
Application specific Pointer Shapes	
	Insert cursor to insert values in an MBB-dialogue edit field by means of the text value and table editor.
	Insert cursor to insert values in an MBB-dialogue edit field by means of the text value and graph editor.
	Pen cursor to pick a value from the legend of a map and to enter it in a cell; or to enter ordinal values in the X-Y graph part of the graph editor.
	Bucket cursor to pick a value from the legend of a map and to enter it in a group of contiguous cells.