Bioeconomic Modelling: An Application to Fisheries

By
Gilles Reinhardt

Master's of Science thesis
Systems Science programme

University of Ottawa

ISBN 0-315-60056-X

Abstract

This thesis presents a bioeconomic model of a commercial fishery. Emphasis is on the decision making processes of the harvesting sector. A model of dynamic decision making by fishermen is developed to study the biological and economic impact of the commercial exploitation of the Georges Bank scallops fishery (*Placopecten Magellanicus*). Stock biomass abundance dynamics are modelled using Deriso's age structured population model. Growth rate, mortality rates and recruitment are included as part of the biological component of the model.

Decision making by fishermen is modelled using two discrete decision algorithms, *myopic* and *adaptive*. Fishermen's decision alternatives before each trip to the fishing grounds consist of deciding on which particular fishing areas to search for fish. Each area has an associated fishing cost and an associated catch level conditional on the level of abundance of the resource in that area.

Before each fishing trip, the myopic model provides the fisherman with the area yielding the highest immediate expected return on the basis of cost, expected catch and the current measure of biomass abundance. The selected area is found using a decision tree. The objective of the adaptive model is to maximize the expected return over all trips to the fishing grounds for the entire season. A trip by trip fishing policy is developed for every season. This policy consists, for each fishing trip, of a subset of the accessible fishing areas. At each stage (fishing trip), the fisherman updates his information about the abundance of the resource and selects an area from this subset.

A computer model simulates possible outcomes over a finite number of fishing seasons. The program simulates the application of both the myopic and adaptive algorithms by a single fisherman as well as by a fleet of fishermen. In the later case, each fisherman is assumed to be initially identical and independent of the others in the fleet with respect to landings and catch information compiled. Vessel performance is measured in terms of total catch, total costs, landed values and net incomes.

The results provide insight into the dynamic evolution of the commercial fishery with respect to stock abundance levels, fishermen's incomes and investment potential.

Acknowledgements

Special thanks to you, readers. Please be aware that comments and questions about this thesis are always welcome.

I wish to direct a generous amount of recognition towards Dr. Daniel E. Lane who introduced me to the challenging field of bioeconomics. The time and dedication he provided were quite essential and very appreciated.

I also want to underline the support of some very special friends: Catherine Lapointe, Pierre Falardeau, Johane Desjardins, Kilani Ghoudi, Hélène St-Jean, Marc Lachance, Guy Beaulieu, Sylvain Bacon, Tanis and Douglas, Pauline et Alain, Lucie et Robert and the DuPont family.

Enfin, je dédie fièrement cette thèse à ceux qui m'ont enseigné courage et volonté requis pour la la réalisation de tels projets. Ce travail vous appartient, Carmen et Julien. À mon tour maintenant de vous exprimer ma reconnaissance.

Merci, et bonne lecture...

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background to the thesis

Canada's ocean fisheries resources have historically been significant to this country's economic development. The extensive coastal shore lines as well as high abundance and variety of harvestable species contribute to realizing the significant economic potential of this resource. The fisheries now support major industries and several thousand jobs ranging from biological research and development to harvesting, processing and manufacturing.

Initially, marine resources were freely exploited by fishermen as 'common property' resources. However, as the exploitation potential grew so did the need for regulations on season length, gear size, capacity restrictions, area restrictions, allowable catch quotas and so on. Strategies for both biological and economical exploitation are being developed and implemented on an on-going basis. Stock protection and harvest limitation policies constitute the main focus of fisheries management.

Stock control involves biological considerations such as survival, mortality, recruitment, size and many other components. For example, numerous stock-recruitment models have been developed to describe the wide range of possible dynamic stock biomass behaviour. Stock management involves strategic management decisions. Harvest rates are one of many various elements in a biologically-based system.

With respect to harvest management, the economic and social impacts of management policies on commercial fishermen are important. Harvest management deals with the implementation and determination of harvest rates and seasonal catch quotas called TAC's (Total Allowable Catches) for individual commercial fisheries. Whereas stock management deals more with strategic implications, harvest management deals more with the *operational* aspects of fisheries including policy implications on fishermen's earnings, fleet management, capital investment, quota distribution within different fleets of vessels and fishing season regulations with respect to the season's length and available grounds for exploitation.

Over the past twenty years, concepts such as biological or economical sustainable yields derived from resource growth dynamics and population level equilibria contributed to the dual modelling of biology and economics now referred to as **bioeconomics**. It deals with the key issue of maximizing returns to fishermen while respecting seasonal and global sustainable yields for required survival levels of future biomasses.

Bioeconomics combines the main components of stock-recruitment models and economic decision making models. This includes survival, natural and fishing mortality rates, biomass growth measures, previous stock and recruitment abundance, stock distribution by age, costs of fishing, market price behaviour, relationships between stock abundance and probable catch results, and so on. The purpose of bioeconomic modelling is to examine strategic and operational policy implications while respecting biological objectives, towards improved returns from the fisheries.

The model developed in this thesis is applied to the scallop fishery in the Canadian portion of Georges Bank. The scallop biomass is spatially distributed over numerous distinct beds within the Bank. The beds have more or less independent growth characteristics. The migration between the beds is negligible. The scallop fishery is regulated by the $F_{0.1}$ fishing mortality rate that determines the seasonal TAC. ($F_{0.1}$ is a Department of Fisheries and Oceans target harvest rate based on a model of stock yield by weight per recruit to the fishery.) An average of fifty dredges (scallop fishing boats) are active during any one season. They make a maximum of fifteen fishing trips over the fishing season lasting from April to October. The season may be shortened if the TAC is exceeded. Presently the TAC is divided among the seven fishing companies who operate dredges on the Bank. The company quotas are known as Entreprise Allocations (EAs).

1.2 Focus of the thesis

This thesis presents a bioeconomic model that encompasses the economic decision making dynamics of harvestors and the biological behaviour of a single biomass. The decision making dynamics component consists of the construction of decision policies that depend on the unobservable state of the system (scallop biomass abundance). The stock dynamics component consists of a biological growth model and a stock-recruitment model.

The model output includes a set of economic and biological performance indicators computed through simulations of actual fishing seasons. The model reveals reactions and trends of both biomass and fishermen's decisions to different model parameters.

The results produced by the model include intra-seasonal fleet dynamics (simulated movement and distribution of the fleet during a season over the fishing areas), seasonal economic performance dynamics (net income, total catch, total cost and landed value), impact of various regulation schemes including quota adjustments, changes in the size of the active fleet and intra-seasonal price fluctuations. The performance indicators are also studied on a 'per trip' basis to allow for an enhanced look at intra-seasonal dynamics.

The stock dynamics of the biomass system is described by a model that includes aspects of aggregate surplus production and time-metered stock recruitment methods. Deriso's model represents the biomass dynamics of populations by a delay-difference equation that closely resembles a surplus production model with parameters which are usually referred to in dynamic pool models. The model assumes that all stock above a fixed age are equally vulnerable to harvest at a given proportional rate h ('knife-edge' selection) and that given the survival rate, the recruitment equation can be written in terms of the previous years' stock size.

The regulatory control of the stock abundance is imposed in the model by setting the maximum seasonal fishing mortality. As mentioned, most commercial TAC quota regulations are based on the $F_{0.1}$ fishing mortality. Once computed, this rate is transformed into a seasonal harvesting quota for the harvesting sector. It can be viewed as a *stopping* condition since the fishing season ends when the harvesting quota is reached or surpassed.

Two different approaches were designed to construct harvesting decision models. The first technique consists in a 'per trip' decision tree model that incorporates fishing costs, discretized catch levels dependent on biomass abundance, fixed number of fishing areas and intra-seasonal price behaviour function. Because each decision is considered independently of all future decisions, the technique will be referred to as the 'myopic' algorithm. The second method, the 'adaptive' algorithm, consists in a dynamic programming algorithm that incorporates all trips in a season via an intra-seasonal Bayesian updating scheme. This model creates a decision policy with expected reward distributions for each fishing trip. Given a probability measure on the discretized state of abundance, expected returns from fishing by area are computed and the area yielding the highest expected return is selected. Both models use the same biological and harvesting elements (actual catch conditional on the actual state of abundance, growth and survival rates, costs of fishing, size of catches per area).

The model can accommodate various regulation scenarios such as a variable harvesting quota, periodical closure of some fishing areas, differences in the landed value functions and a fleet size dependent on the current stock. In this way, experimenting with the model provides a strategic view toward regulatory impacts on the biological and economic performance of a fishery.

1.3 Plan of the thesis

The remainder of the thesis is comprised of the following chapters:

Chapter 2 presents a survey of the literature relevant to the thesis. The review is concentrated in three specific areas, namely bioeconomic modelling, decision making dynamics, and the application to Georges Bank scallop fishery.

Chapter 3 presents the methodology for the bioeconomic model, including model definition, development and implementation. The bioeconomic components are described and their representative values relevant to the application computed and justified. The two decision models (myopic and adaptive) are formally presented.

Detailled results of the model simulations and analysis are presented in Chapter 4. This includes a sample experimental design and associated model results.

Chapter 5 closes out the thesis with a summary of results and a discussion of possible extensions to the models and experiments derived from them.

Chapter 2

Literature Review

Since fisheries management requires a singular mixture of zoological, administration, and mathematical ideas, there are many sources of relevant published literature. The following survey groups published works relevant to this thesis in three categories, namely bioeconomic modelling, decision dynamics and applications to scallop fisheries.

2.1 Bioeconomics

A basic and essential textbook in bioeconomics is Clark's *Mathematical Bioeconomics* [16]. It presents the elements of deterministic bioeconomic modelling by dealing with elementary dynamics, economic models and optimal control theory. Also included are analyses of discrete time models, growth and aging and multispecies fisheries problems.

Clark's *Bioeconomic Modelling and Fisheries Management* [15] is a follow-up work that deals with stochastic models for fisheries. It presents concepts of fishing effort, fishing in a stochastic environment, discrete-time and age-structured models. Several types of models ranging from fishery regulations to fluctuations and uncertainty are presented in the subsequent chapters.

Walters [84] proposes a new perspective with the study of bioeconomic models for actively adaptive management methods. They consist of constructing decision policies given the decision maker's limited awareness of the uncertainty in resource stocks. Once a policy is finalized (through extensive backward recursion as in the dynamic programming algorithm), it is then applied in a 'forward' motion with decisions adapted from all past information by Bayesian updating.

A widely used model for describing the biological growth of a single stock faced with commercial harvesting is that of Deriso [20]. The model combines the aggregate surplus production models with components of age-structured analysis and stock-recruitment. The biomass is determined by the product of the number of stock units and their average body weight at a given age. New recruitment to the fishery may be included using a variety of functions.

An important work in bioeconomic modelling is Schnute's [79] work on the analysis of catch and effort data. It presents parameters present in many well-known stock-recruitment models (Ricker, Beverton-Holt, Schaefer-logistic or Deriso) and explains these variables in their most appropriate context. A valuable generalization of the above models is also provided.

Gulland [28] cites that 'the science of stock assessment is concerned with the provision of this [state of fish stock] advice'. His book describes the basic concepts such as the unit stock (distribution of fishing, spawning areas, general population parameters) and the catch per unit effort (catchability, standardization, gear specifications). The elementary production models and methods of parameter estimation are also presented. These parameters include growth rates, mortality rates, selection age and recruitment numbers. The work is complemented by an extensive description of the process for computing yield per recruit functions. Finally, a discussion on sensitivity analysis of changes to values and estimates on policies results is provided.

With respect to uncertainty in harvesting, Mangel and Clark [56] give an excellent introduction by considering 'the problem of modelling uncertainty regarding the location of fish concentration, and the effect of search by fishing vessels in reducing such uncertainty'. This paper examined uncertainty in bed locations, a most important problem for fisherman. Prior probabilities are assumed with respect to abundance in the fishing grounds and Bayesian updating of these priors takes place following each fishing period. The authors find that 'sampling' the stock increases the expected catches. By using the Bayesian updating method, expected revenues were also found to be higher than those not using search information.

In a recent fisheries economics paper Lane [46] deals with the $F_{0.1}$ fishing mortality regulation policies. This paper defines the $F_{0.1}$ figure by first giving its historical context and then justifying the computation specifications. An alternative model framework is provided in which three key components are detailled: stock mortality, growth and recruitment; decision making behaviour of fisherman; economic performance measures. It is noted that more conservative estimates ($F_{0.1}$ vs F_{max}) are not necessarily the optimal choice over all fisheries since stock-recruitment relationships and 'economic underpinnings of the level of fishing mortality' should also be given major considerations in determining the TAC levels.

Hannesson and Steinshamn [30] compare two strategies for the establishment of catch quotas. These are (1) fixed catch quotas and (2) the resulting catches from fixing the fishing effort at a constant level. The paper gives a general theory on each strategy's profitability and then simulates a multi-year class fish stock focusing on fluctuations in the annual recruitment. Some remarks on validation and applicability of the model to current issues in fisheries management conclude the paper.

Hutchinson and Fischer [39] present already existing (and popular) stock growth models such as Ricker, Beverton-Holt and Schaefer-logistic. Their article focuses on recognizing the perturbation of the biomass systems (and outlining their importance). Control methods (deterministic and stochastic) are also displayed. A generalized stochastic fisheries model is presented and a specific example to Atlantic sea scallop is simulated. The major conclusion to be drawn is the fact that stochastic control leads to policies that produce 'a much larger increase in expected profit when compared to the open-loop deterministic control approach'.

2.2 Decision Dynamics

Hilborn and Walters [34] introduce dynamics in spatially heterogeneous fisheries using the Deriso model. Their paper presents a simulation model of multispecies or multistock fisheries that are spatially distributed and includes the dynamics of fishing fleets.

Lane [50] presents an intra-seasonal model of fishermen's decision making dynamics. It consists of a partially observable Markov decision process (POMDP) for the decisions of fishing vessel operators. The abundance as well as the catch levels are discretized in order to allow for dynamic probability measures. Before each trip within a season, fishermen decide on the area which will yield the highest expected return. The catch result depends on the actual state of abundance level and the state probability vector π is updated according to observed catch information.

On a more theoretical basis, Smallwood and Sondik [83] examine 'The Optimal Control of Partially Observable Markov Processes over a Finite Horizon'. This paper underlines the inherent shortcomings of deterministic observation methods (infeasible for underwater biomasses). Instead, a discretization procedure is applied to the unknown state and a probability measure on the system is introduced over N discrete states of the system. This measure is described by a $\pi = (\pi_1, \pi_2, \ldots, \pi_N)$ probability distribution vector. A probability transition matrix is constructed with the p_{ij} elements defining the probability that the system will transit to state j from state i over the next period. The paper also highlights some shortcomings of 'discretizing' (conversion of a continuous state Markov process into a finite state Markov process) including the 'curse of dimensionality' and the long processing time for convergence of the algorithm on a mainframe computer.

2.3 Application to Scallops

Caddy [6] explores the spatial distribution of biomass and effort over a fishing ground for Georges Bank sea scallops. This study quantifies and differentiates the accessible scallop fishing areas within Georges Bank. It gives estimates of 'virgin biomass per 10 minute unit area of latitude and longitude in metric tons'. The paper also provides information on the Georges Bank scallop resource such as dynamic pool yield, gear selection, unexploited biomass, and total and fishing mortality estimates.

In a 1987 manuscript, Mohn *et al.* [64] look at yield per recruit analysis as well as stock projections in order to set the TAC for the Georges Bank scallop stocks. It focuses biological information on the problem of defining fishing strategies and target exploitation rates. The $F_{0.1}$ fishing mortality level is found to be a conservative basis for determining the TAC, even though when noting that 'an F based on a yield per recruit model is being used to address failing recruitment'. The paper is critical of policies to treat the oldest age group as a 'plus group'. This means that the resource holds an infinite potential lifespan. With respect to scallop catches, the proportion of animals over age 11 (where the plus group begins) is very low ($< 1\%$).

The '1988 Offshore scallop Fishery Mangement Plan' [9] provides a good source of the major objectives and advice features of actual annual management policy. Developed by the Offshore Scallop Advisory Committee (OSAC), the plan underlines the dual goal of conserving a limited and valuable resource while providing 'reasonable access' to Georges Bank for the licensed (enterprise allocation) scallop fishery vessel operators. The major objectives call for increased stability of the biomass (from the recent instabilities in the annual landings). It also requires increased biological input in order to coordinate better the TAC and EA quotas. Finally, as a major objective, the committee calls for 'reduction in harvesting capacity to improve economic viability' by continued encouragement for the use of the EA program.

Chapter 3

Methodology

The major aspects of the bioeconomic model include the resource's behaviour through stock-recruitment dynamics and the exploitation of the resource through harvesting decision dynamics. This chapter presents the design and development of appropriate models to address these two aspects.

3.1 Stock Dynamics

A stock dynamics model that provides a suitable, parsimonious biomass representation would be one that incorporates age structure, growth effects and stock-recruitment. In a detailled presentation of the Deriso model [20], Hilborn and Walters [34] highlight its simplicity and efficiency by comparing it to a surplus production model that includes current and past biomasses and recruitment as well as considerations for fishing and natural mortality rates. To briefly introduce stock abundance dynamics, we begin by describing the necessary elements of the actual process.

3.1.1 Stock-recruitment Process

Consider a stock-recruitment model in which the response of the population to certain external forces (e.g. harvesting) is not instantaneous, and a measurable amount of 'time of reaction' is to be considered. When this occurs, the population may be modelled using discrete-time. Clark [16] provides a 'general, time-metered stock-recruitment model' which illustrates the sensitivity of a discretely responsive system by initially defining the current (tth) generation's parent stock, X_t, and the resulting recruits, R_t. These recruits will either be harvested, H_t, or become part of the next generation's parent stock X_{t+1}. Figure 3.1 gives a graphic overview of the above process. It is assumed that the estimated stock at the beginning of the season will vary only within its initial class-mark, and on-going

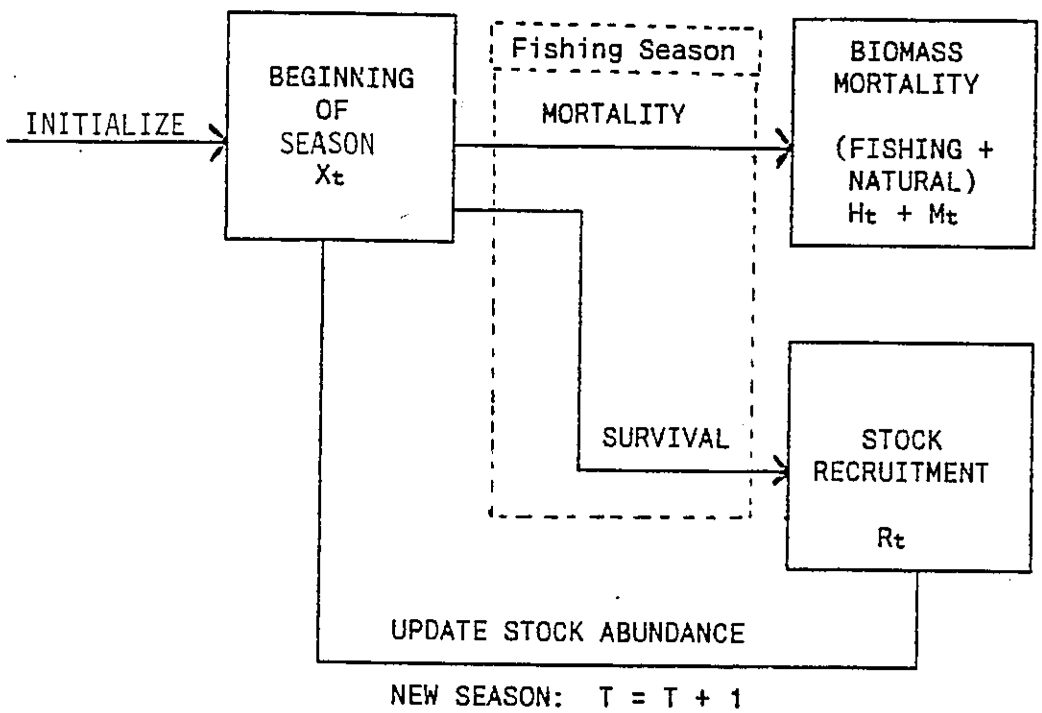


Figure 3.1: Harvestable stock reproduction process

adjustments in stock for natural mortality, harvesting mortality and recruitment will be compiled and included in the stock estimates at the end of the current season. Moreover, the beginning of season aggregate stock level is assumed to be distributed in an unchanging manner throughout all relevant fishing areas. Thus each area provides information about the overall state of scallop abundance.

3.1.2 Deriso's Recruitment Model

Let the current population biomass, X_t, be defined in terms of: N_{at} the number of animals at age a in time t and W_{at}, the unit weight at age a and in time t. The single stock population size by weight can be expressed as (Walters [84])

$$X_t = \sum_{a=k}^{\infty} N_{at} W_{at} \tag{3.1}$$

where k is the fixed age of recruitment to the fishery (an animal is harvestable when $a \geq k$). Assume that each animal is equally likely to be harvested at a rate h_t in time t ('knife-edge selection') and that the biomass is characterized by a natural survival rate of s. The total survival rate can be represented as $l_t = (1 - h_t)s$. Equation (3.1) can now be written as

$$X_t = l_{t-1} \sum_{a=k+1}^{\infty} N_{a-1,t-1} W_{at} + N_{kt} W_{kt} \tag{3.2}$$

Supposing now that the body growth rate ϱ is decelerating, i.e. $\varrho < 1$. The Brody equation (Schnute [79]) characterizes this rate by

$$W_{at} = W_k + \varrho W_{a-1,t-1} \qquad (a \geq k) \tag{3.3}$$

where W_k and ϱ are given growth parameters. Substituting into (3.2) yields

$$X_t = l_{t-1} \varrho X_{t-1} + l_{t-1} W_k \sum_{a=k+1}^{\infty} N_{a-1,t-1} + N_{kt} W_{kt} \tag{3.4}$$

since from (3.1),

$$X_{t-1} = \sum_{a=k+1}^{\infty} N_{a-1,t-1} W_{a-1,t-1} \tag{3.5}$$

By rewriting the $t-1$ terms and by shifting back by 1 time unit we get an intermediate result

$$l_{t-2} W_k N_{t-2} = X_{t-1} - l_{t-2} \varrho X_{t-2} - N_{k,t-1} W_k \tag{3.6}$$

Using this result in (3.4), the basic Deriso stock-recruitment equation is obtained:

$$X_{t+1} = (1+\varrho)l_t X_t - \varrho l_t l_{t-1} X_{t-1} + W_k N_{k,t+1} \tag{3.7}$$

where the first term is the growth and survival of the current biomass, the second term is the correction applied for age-structure and growth fluctuations and the third term is the biomass of new recruits ($R_{t+1} = W_k N_{k,t+1}$)

Now assuming a fixed harvest rate ($F_t = F$ for all t), and a Brody growth coefficient ϱ computed through a *Walford* plot, i.e., X_t vs X_{t+1} (Schnute [79], see also Appendix A) and a model for recruitment (Ricker [73], Walters [84]), the Deriso model can be simplified in the following way:

$$X_{t+1} = (1+\varrho)SX_t - rS^2 X_{t-1} + R_t - rSR_{t-1} \tag{3.8}$$
$$R_t = \alpha X_t e^{\nu\epsilon} \tag{3.9}$$

Where:

X_t .... Biomass (metric tons) at time period t

R_t .... Recruitment (metric tons) observed at time period t

$S = (1 - \frac{F}{F+M} \times (1 - e^{-F-M}))$ .... Total survival level

F .... Fishing mortality rate

M .... Natural mortality rate

ϱ .... Brody growth coefficient

α .... Constant recruitment factor of current biomass

ϵ .... Standard Normal Error ($\epsilon \sim N(0,1)$)

ν .... Error variability

Figure 3.2 represents a graph of X_t vs t for the Deriso stock dynamics model given values of F(=0.3) and M(=0.1), ϱ(=0.958), a constant annual recruitment R_t(=40) and two distinct initial values for the biomass (X_0=100 and 300). Figure 3.3 shows the function of R vs X for equation 3.9 for a given α(=0.4) and different variability values (ν=0.0,0.3).

The Fishing Mortality Rate $F_{0.1}$

Most commercial fisheries stocks on the Atlantic coast are regulated individually using the $F = F_{0.1}$ harvesting policy (Gulland and Boerema [29], Gulland [28], Lane [46]) as a conservative target fishing mortality. The $F_{0.1}$ fishing mortality is obtained from the function yield per recruit versus fishing mortality rate. Yield per recruit is a function of the growth rate and natural mortality of the stock.

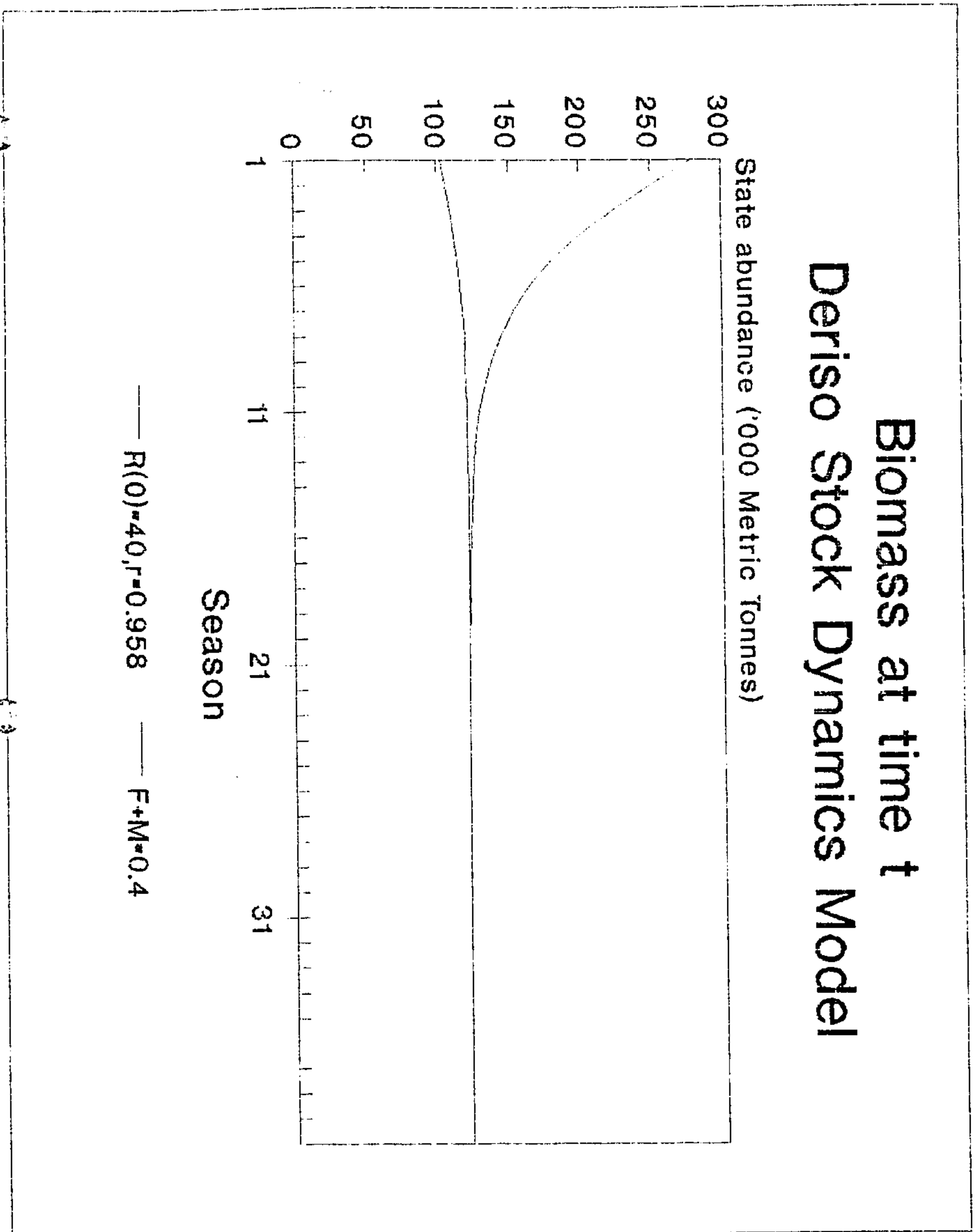


Figure 3.2: X_t vs t

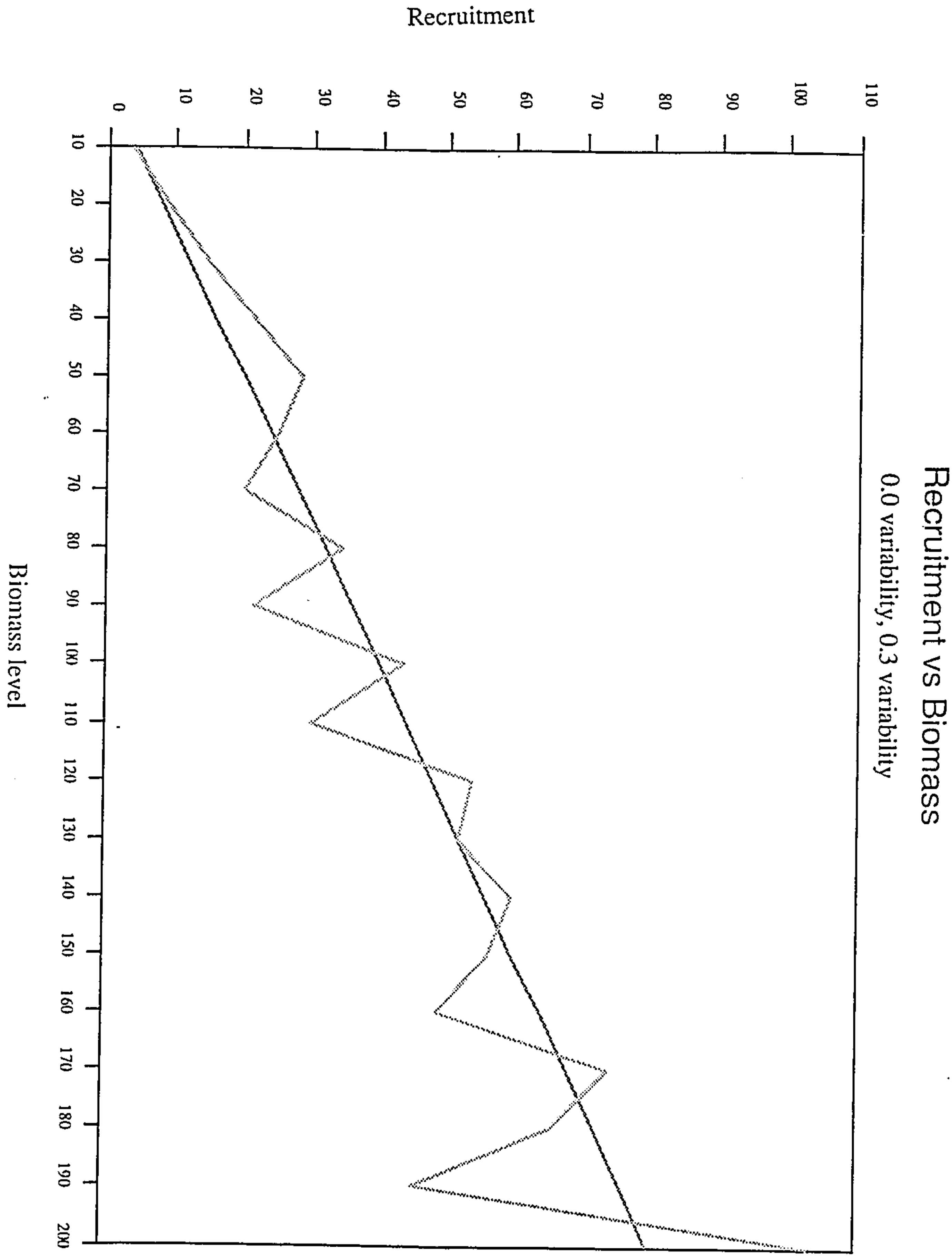


Figure 3.3: R vs X with $\alpha = 0.4$, $\nu = 0.0, 0.3$

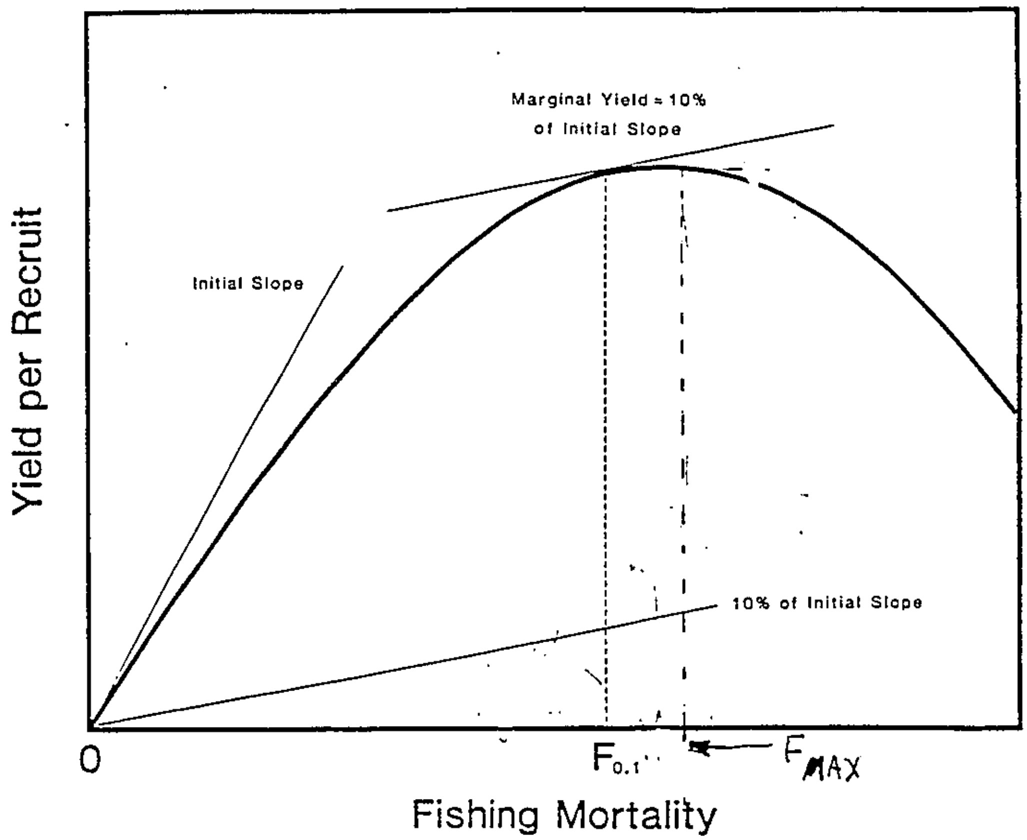


Figure 3.4: Determination of $F_{0.1}$ fishing mortality

$F_{0.1}$ is found by determining the slope of the yield per recruit curve at the origin (derivative at $F = 0$). The curve from the origin of a slope of 10% of the initial slope is drawn and 'pushed' up until tangent to the yield per recruit curve. The first coordinate of this point of tangency is the $F_{0.1}$ fishing mortality rate. This policy, gives a 10% marginal increase in yield per recruit due to a small increase in F in a lightly exploited fishery. Doubleday [22] mentions that 'Most of the fish stocks within the Canadian Atlantic are currently managed at the level of fishing corresponding to $F_{0.1}$...'. Figure 3.4 displays the graphical determination of the $F_{0.1}$ fishing mortality level.

An alternative numerical method to calculating $F_{0.1}$ is to use a third order approximation of the yield per recruit curve. Deriso [21] provides a set of parameters corresponding to coefficients of the equation that gives close approximation to $F_{0.1}$:

$$F_{0.1} = M \exp\left(A_0 + A_1 m + A_2 m^2 + A_3 m^3\right) \tag{3.10}$$

The parameter m is the ratio M/K, where M is the natural mortality rate and K is the growth rate in the Von Bertalanffy length equation with $K = -\log \varrho$ (Schnute[79]). The A_i coefficients are provided by Deriso as in Table 3.1.

A_0	A_1	A_2	A_3
0.699	-0.698	0.236	-0.029

Table 3.1: Parameter values for approximation of $F_{0.1}$

3.2 Harvest Model Dynamics

For a fisherman, the harvesting season is divided into a fixed number of observation periods; i.e., fishing trips. Before each trip (there is a maximum number of trips per season), the fisherman must decide where to fish in order to maximize his expected return. Fishing activities occur successively over a finite maximum number of trips per fishing season. The number of trips is constrained by the Total Allowable Catch (TAC) based on the $F_{0.1}$ fishing mortality target in a season.

Several models can be conceptualized for structuring fish harvesting decision making models. The two algorithms are developed here. The myopic algorithm reacts to immediate information, computes conditional probabilities about the overall state of stock abundance and expected returns, and provides its user with an optimal decision (area) based on the expected return criteria. This algorithm is updated before each fishing trip.

The adaptive algorithm also searches for optimality with respect to the expected returns per fishing area. However, in this algorithm, the user is given a seasonal policy. This means that the current trip's decision will depend on the current measure of the abundance level to date only, all the other decision variables having been accounted in the policy development procedure. Figure 3.5 shows the dynamics of the harvesting sector. Details on the development of both the myopic and adaptive algorithm are provided further in this chapter.

For each algorithm, two basic scenarios are modelled and simulated. These are, first, that all fishermen are alike, i.e., behave in the same 'average' way. This scenario examines the decision making of a single 'average' fisherman independent from the fleet. This fisherman is assumed to make a fixed number of trips with information on stock abundance and catch data being transfered or updated from one trip (or one complete season) to another.

The second scenario is designed to model a fishing fleet with no intra-seasonal information sharing between the fishing boats. Each of the boats in the fleet is treated as a single fisherman. Each have their unique measure of information on abundance level and catch level. In this scenario the number of trips each fisherman makes is also bounded by the total allowable harvest level computed from the fishing mortality rate F applied at that moment.

The π-vector

As previously noted, two decision models for fishermen are applied to the bioeconomic modelling framework. The first is a decision tree formulation with trip by trip analysis (myopic algorithm). The second is a seasonal policy development procedure (adaptive algorithm) using dynamic programming. The following bioeconomic model variables are common to both decision models. Elements peculiar

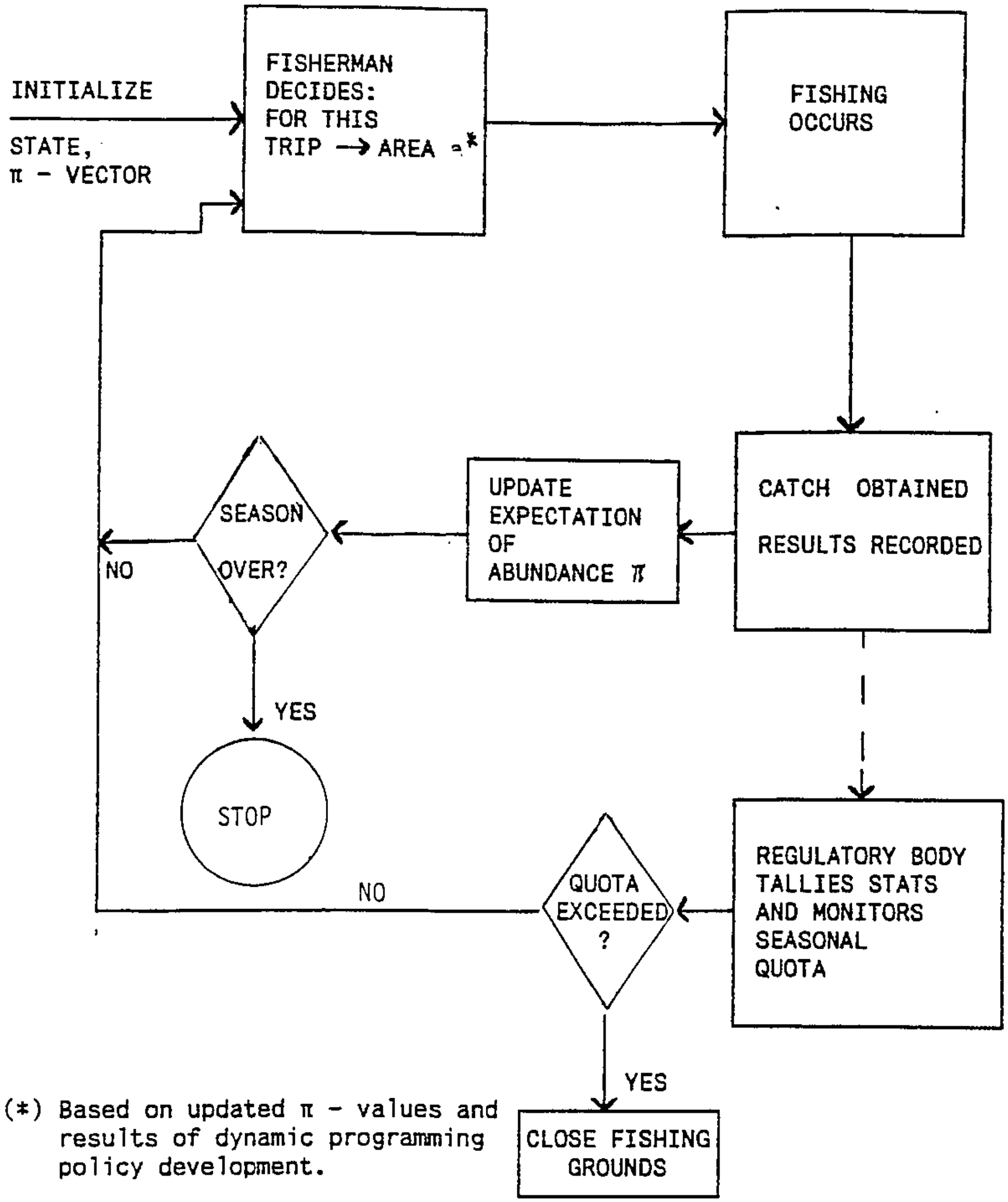


Figure 3.5: Seasonal dynamics of the harvesting sector

to each individual model will be defined subsequently.

3.2.1 Variable Definition

For this application, the state of stock abundance over all fishing areas during season t, X_t, is discretized into three 'fuzzy' categories: Good (G), Average (A), and Poor (P) abundance. Fishermen, never fully aware of the actual state abundance, estimate the state of the fishery with a probability distribution vector over the states, denoted by $\pi = (\pi_G, \pi_A, \pi_P)$. Pre-season abundance estimates and the initial π-vector elements are estimated from historical data. Intra-seasonally, the π-vector is updated through imperfect observations from each fishing trip's catch result. The catch, denoted by Y, is also discretized in G , A , and P categories with numerical ranges and class-marks (boundaries defining the discrete levels) specific to each fishing area.

The functional relationship between Y, the observation of abundance, and X, the actual state of abundance, is characterized by the conditional probability measures:

$$q_{jl}(a) = \Pr(Y = l \mid X = j, \text{area} = a) \tag{3.11}$$

where $q_{jl}(a)$ represents the probability of observing in area a, a catch level $Y = l$, given the actual state level $X = j$.

The controllability of the system is restricted to the one action the fishermen can take per period, that is, which fishing area to select for the next trip? This action is denoted by a_k, the decision to fish in area a on trip $k+1$ during the season.

The reward function for each decision is made up of three basic components:

1. p_k - the price per unit weight of landed value at trip k.
2. $f(l, a)$ - the class mark of landings by weight of catch level (constant over the simulation horizon)
3. $cf(a)$ - the cost of fishing per trip in area a (constant over the simulation horizon)

3.2.2 The Myopic Algorithm

Optimizing the profits over one trip and using this and past collected information to make a decision for the next trip only are evidence of 'nearsightedness' or myopic decision making.

Consider a typical fisherman who is faced with the initial decision of 'which fishing area to choose for the next trip?' It is assumed that the chosen area should be the one which maximizes the immediate expected return from fishing. The expected returns depend on the stock abundance level and the catch result. A model that adequately replicates this decision process may be described by decision tree methodology (Raiffa [71]). The decision tree is used to sketch out the possible options and outcomes facing the fisherman each trip. A 'quantified preference' on fishing areas can then be established since in decision tree analysis the optimal fishing area will be the one that maximizes the immediate *expected* return. Figure 3.6 represents the possible options and outcomes facing the decision maker in a decision tree diagram. The myopic model includes alternatives of test dragging in particular areas. Feedback from the test drag is used to determine continued effort in the area as occurs in practice.

Consider now the option of 'fishing in area a' for a given trip. Under the myopic model assumption this option's expected return is obtained in the following way:

$$\Pr(Y = l \mid \text{area } a) = \sum_{j=G,A,P} q_{jl}(a) \cdot \pi_j(k) \tag{3.12}$$

This yields the marginal probability that the fisherman observes a catch of level l in area a where $\pi_i(k)$ represents the prior probability on the state abundance ('probability that the state abundance level equals j, j=G,A,P') for fishing trip k. From $\Pr(Y \mid a)$ and the catch class-marks specific to each area, $f(l, a)$, the return $R_{k+1}(l, a)$ is computed by calculating the current reward function

$$R_{k+1}(l, a) = p_{k+1} \cdot f(l, a) - cf(a) \tag{3.13}$$

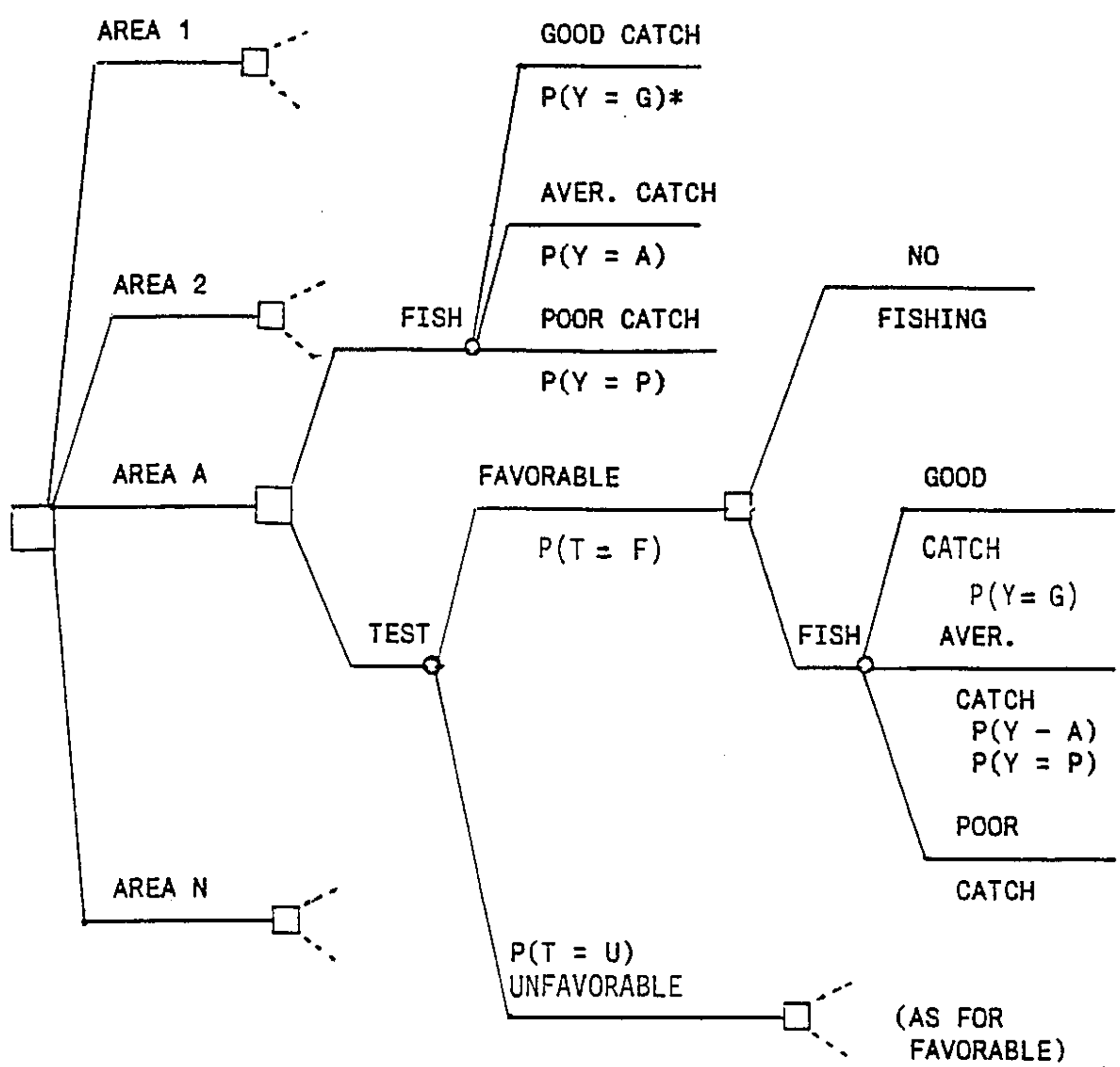
for fishing in area a during trip $k + 1$ of the season. The total expected return from fishing in area a is:

$$g_{k+1}(a) = \sum_{l=G,A,P} \Pr(Y = l \mid \text{area } a) R_{k+1}(l, a) \tag{3.14}$$

Similarly, the 'testing' option's expected return is computed. Here, the 'no fishing' option includes only the testing cost of area a. Only the $\Pr(T)$ probabilities need to be computed ($\Pr$(Test is *favorable* $(T = f)$ or *unfavorable* $(T = u)$). $\Pr(Y \mid a)$ was obtained above and $\Pr(T = u \mid Y = j) = 1 - \Pr(T = f \mid Y = j)$, the conditional probability of the test outcome for a given catch level is a constant data input matrix. Hence $\Pr(T) = \Pr(T \mid Y) \cdot \Pr(Y)$. Finally, to compute $g_{k+1}(a)$, $\Pr(Y \mid T)$ is used instead of $\Pr(Y)$. $P(Y \mid T)$ is obtained by applying Bayes' theorem:

$$\Pr(Y \mid T) = \frac{\Pr(T \mid Y) \cdot \Pr(Y)}{\sum_{Y=G,A,P} \Pr(T \mid Y) \cdot \Pr(Y)} \tag{3.15}$$

MYOPIC ALGORITHM: TRIP DECISION TREE



* All probability values are derived from (1) a prior probability distribution of stock abundance, π; (2) the probability of catch given stock abundance; and (3) the 'efficiency' of test drag procedure.

Figure 3.6: Decision tree representation of the myopic model

This completes the calculations required to determine the current area decision for the myopic model.

Once the area a yielding highest expected return is selected, fishing takes place, a catch level classmark is observed (l) according to the actual stock level and $q_{jl}(a)$ probabilities for area a. The catch is recorded and the observed state abundance vector, π, is updated for the next trip $(k+1)$in the following way:

$$\pi_i(k+1) = \frac{q_{il}(a) \cdot \pi_i(k)}{\sum_j q_{jl}(a) \cdot \pi_j(k)}, \qquad i = G, A, P \tag{3.16}$$

This updated π-vector is carried through to the next fishing trip. The same decision tree is analysed, the expected values and probabilities modified accordingly and a fishing ground is selected again. This procedure continues until either the total allowable catch has been filled or the fixed maximum number of trips per season has been reached. At the end of the season, stock-recruitment occurs. At the beginning of the next season the fisherman selects a fishing area from the decision tree using the ending π-vector from previous seasons' information as the new season's initial π-vector.

3.2.3 The Adaptive Algorithm

The myopic algorithm requires current period computations at the beginning of the fishing trip only. The adaptive algorithm however develops a forward looking decision policy over all trips during the season prior to the first fishing trip. That is, a seasonal policy is developed before the fishing season. This policy consists of identifying subsets of the accessible fishing areas as best candidates for fishing in each trip. Figure 3.7 charts the method for determining the fisherman's seasonal policy. Adaptive decision making models are derived from stochastic dynamic programming. We begin with the last and simplest stage of the decision problem by computing the returns when there is only one fishing trip to go. The expected returns from fishing in each area can be computed for all possible discrete state levels of the system, making the final decision strictly dependent on the probabilistic measure of the state, the π-vector.

Elimination of non-optimal yielding areas

Fishing areas that are strictly dominated relative to their contribution to expected returns over the π-vector may be eliminated as decision alternatives. Consider the option of fishing in area a. The elements of the expected return vector over all actual states of the system are denoted here by a distribution of corresponding reward values for every possible state level $(\alpha_{aG}, \alpha_{aA}, \alpha_{aP})$. The expected reward from fishing area a is now given by

ADAPTIVE ALGORITHM DECISION POLICY

(Backward recursion with n fishing trips per season)

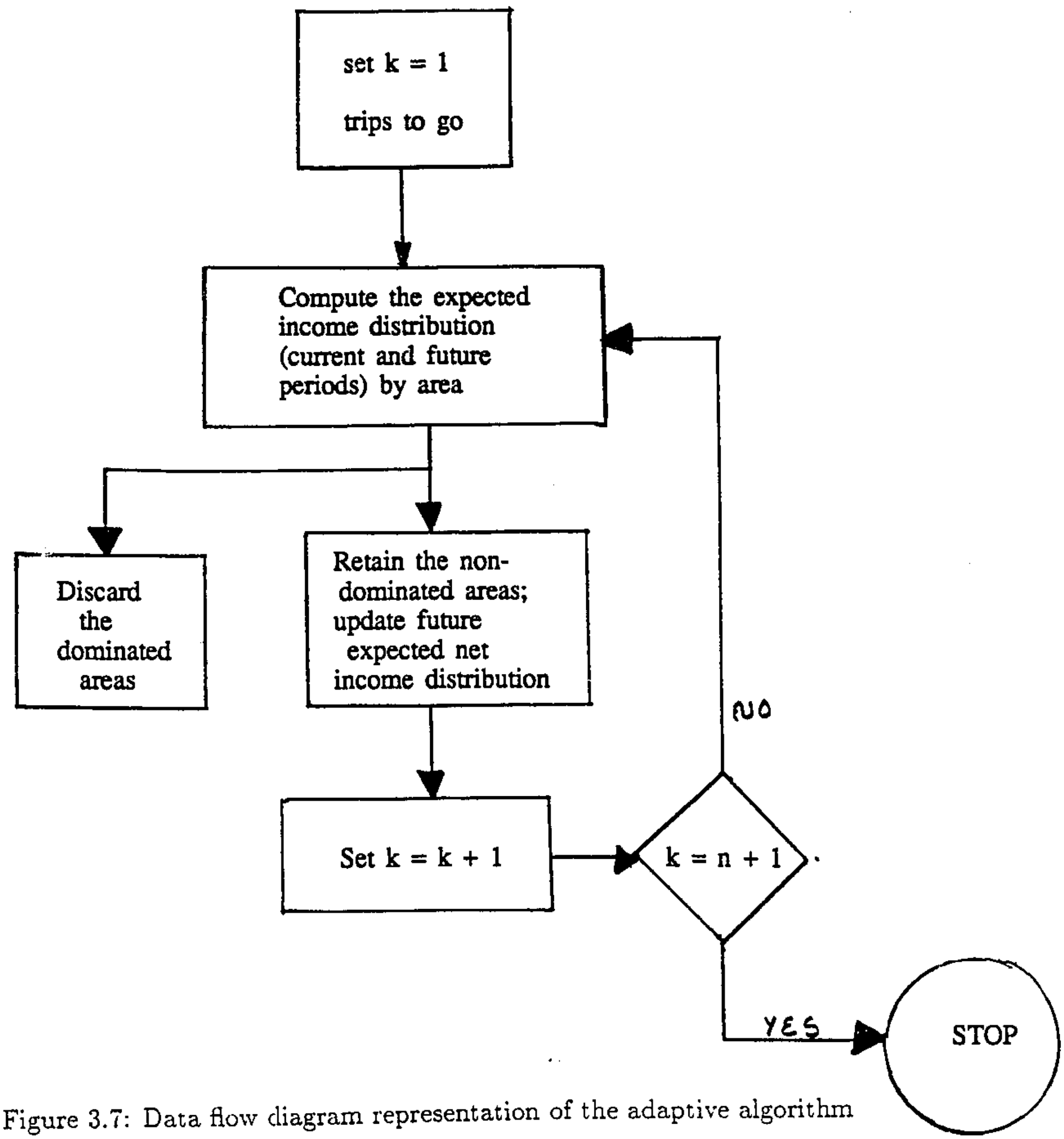


Figure 3.7: Data flow diagram representation of the adaptive algorithm

$$\sum_{i=G,A,P} \alpha_{ai}\ \pi_i \tag{3.17}$$

When matching this set against the α-values for some other area, it may be observed that all α_{ai} values are strictly larger or smaller than those of the other area. ($\alpha_{aj} > \alpha_{a'j}, j = G, A, P$). When area a dominates area a' then a' can be discarded immediately, regardless of the values of the π-vector. Therefore, area a' would be excluded from this trip's policy subset.

The backward recursion is modelled in the α-update procedure when proceeding from i trips to go to $i+1$ trips to go. All α-values are then updated by adding the expected return of trip $k+1$ and averaging on the α-values with k trips to go.

Total enumeration of the decisions from trip k to the last trip (we are moving backwards) would of course yield full accuracy for the α_{k-1}-value vector. The large number of combinations (number of areas $\times$ number of states)^k is symptomatic of the 'curse of dimensionality' in dynamic programming. To circumvent this problem the heuristic procedure used here is to average the α_k values obtained at every stage or fishing trip of the problem. The resulting values in the vector α represent the future expected reward attributable to all dominant strategies.

Mathematical model of the adaptive algorithm

The expected net operating income from fishing in area a at trip k (action a_{k-1}) if the actual state abundance level is $X_t = j$ is defined as a function of the expected catch, the resource's unit price at the current period and the area's fishing cost. As in the myopic algorithm and from 3.14 we have:

$$g_k(a) = \sum_{j,l} p_k \cdot f(l,a) q_{jl}(a) \pi_j(k) - cf(a) \tag{3.18}$$

The objective function maximizing the total seasonal expected net operating income can now be formally stated:

$$J(\pi) = E\left\{\sum_{k=1}^{N} g_k(a_k)\right\} \tag{3.19}$$

where N is the number of trips a fisherman makes in a season and a_k is the action (decision) taken at trip k.

Above, we mentioned the π-updating procedure resulting from catch levels, $Y_1,\ Y_2, \ldots,\ Y_{k-1}$ is

$$\pi_j(k) = \Pr(X_t = j \mid Y_1,\ Y_2, \ldots,\ Y_{k-1},\ a_1,\ a_2, \ldots,\ a_{k-1}) \tag{3.20}$$

Once the decision $a_k = a$ has been determined and $Y_k = l$ is taken, then we find $\pi_j(k+1)$ through a Bayesian updating transfer function T_k:

$$T_k(\pi \mid l, a)_j = \frac{q_{jl}(a)\pi_j}{\sum_i q_{il}(a)\pi_i} = \pi_j(k+1) \tag{3.21}$$

We define the last trip's (N) (or 1 trip to go) reward function as:

$$J_N(\pi) = \max_a [\sum_{i=G,A,P} g_N(a)\pi_i(N)] \tag{3.22}$$

Applying backward recursion for one period, the current period is now the season's next to last period ($N-1$):

$$J_{N-1}(\pi) = \max_a [\sum_{i=G,A,P} g_{N-1}(a)\pi_i(N-1) + \sum_{i,l} J_N(T_{N-1}(\pi \mid l, a))q_{jl}(a)\pi_i(N-1)] \tag{3.23}$$

Finally we may write the dynamic programming equation:

$$J_k(\pi) = \max_a [\sum_{i=G,A,P} g_k(a)\pi_i(k) + \sum_{i,l} J_{k+1}(T_k(\pi \mid l, a))q_{jl}(a)\pi_i(k)] \tag{3.24}$$

or in expectation terms:

$$J_k(\pi) = \max_a [E\{g_k(a) + J_{k+1}(T_k(\pi \mid l, a)) \mid \pi(k)\}] \ , \ k = \ 1, \ 2, \ldots, \ N \tag{3.25}$$

where $J_{N+1} = 0$. The dynamic program produces a decision *policy* for each fishing trip consisting of α-vectors for each of the non-dominated decisions (areas). The α-vector elements represent the expected returns from the three possible states of the system. Figure 3.8 and Figure 3.9 show respectively the mathematical decomposition of the α-values and a numerical example from the application of the model when there are $k = 1$ trip to go and $k = 2$ trips to go.

The vector product of a realization of the π-vector with the α-vectors yield the expected return per decision alternative α_i^0 (equation 3.26).

$$\begin{pmatrix} \alpha_{1G} & \alpha_{1A} & \alpha_{1P} \\ \alpha_{2G} & \alpha_{2A} & \alpha_{2P} \\ \vdots & \vdots & \vdots \\ \alpha_{n^*G} & \alpha_{n^*A} & \alpha_{n^*P} \end{pmatrix} \begin{pmatrix} \pi_G \\ \pi_A \\ \pi_P \end{pmatrix} = \begin{pmatrix} \alpha_1^0 \\ \alpha_2^0 \\ \vdots \\ \alpha_{n^*}^0 \end{pmatrix} \tag{3.26}$$

The alternative area with the maximum value obtained (highest α^0 value) is chosen as the area to fish in the forward pass through the dynamic programme policy. The set of n^* decision alternatives (areas) is a subset of the total number of fishing areas which include the non-dominated areas at the current fishing trip. As in the myopic algorithm, the π-vector is updated according to catch, and the process moves forward in time to the next fishing trip until the season ends.

The heuristic policy results are used in the simulation model to replicate the fishing activity and associated trip decisions. Simulated fishing profits are tallied by multiplying the class mark of the observed catch level by the current unit price and subtracting the cost of fishing.

DP Policy Development

Computation of the α values for area a and states G, A and P:(α_G, α_A, α_P) with k=1 period to go

$$\alpha_G = (p_k \times E(catch | G, a)) - c_f(a)$$

p_k is the market price in period k

E(catch G, a) =

Pr (Good | G, a)	x	Good catch
+Pr (Average | G, a)	x	Average catch
+Pr (Poor | G, a)	x	Poor catch

c_f (a) is the cost of fishing in area a.

With k = 2 periods to go:

$$\alpha_G = p_k \times E(catch\ \ G, a) - c_f(a) + \frac{\Sigma_i \alpha_{iG}(k-1)}{\text{Number of options}_i}$$

{ Immediate Reward } + {Future Reward}

Future reward is the average value over all non dominated areas i with k - 1 periods to go.

Figure 3.8: Computation of the adaptive algorithm decision policy

α - computation example

k = 1 period to go, the immediate reward values are:

STATE

AREA		G	A	P	
	1	28.05	27.00	23.85	
	2	31.55	25.70	21.35	Non dominated areas
	3	30.85	27.40	24.10	Non dominated areas
	4	29.55	26.85	24.60	Non dominated areas
	5	30.35	26.90	23.60	
	6	30.35	26.90	23.60	

(*) **Figure D.2 in thesis**

Future averaged net income distribution:

	G	A	P
2	31.55	25.70	21.35
3	30.55	27.40	24.10
4	29.55	26.85	24.60

$$\frac{91.95}{3} \quad \frac{79.95}{3} \quad \frac{70.05}{3}$$

= 30.65 26.65 23.35

When k = 2 periods to go, the reward matrix is updated

STATE

AREA		G	A	P
	1	58.70	53.65	47.20
	2	62.20	52.35	44.70
	3	61.50	54.05	47.45
	4	60.20	53.50	47.95
	5	60.00	53.55	46.95
	6	60.00	53.55	46.95

	G	A	P
3	30.85 +30.65	27.40 +26.65	24.10 +23.35

Figure 3.9: Numerical example of the adaptive algorithm policy development

3.3 Other Model Specifications

Total allowable catch considerations

When simulating the fishing fleet scenario (i.e. multiple independent vessels), monitoring of the cummulative actual catch is necessary to determine when the preset TAC has been harvested. A **stopping rule** is included in order to model the enforcement of TAC regulations. This rule is implemented by first computing the season's harvest fraction h of the total stock for the given value of management policy F:

$$h = \frac{F}{F+M}(1 - \exp(-F - M)) \tag{3.27}$$

Following every fishing trip, the ratio of total simulated catch and estimated state abundance is computed and compared to the h value. If the computed ratio is higher than h, then the current fishing season is terminated and the statistics for the season are tallied.

Investment Factor

At the end of each season, fishermen are assumed to reinvest a share of the season's net income back into fishing capital. To model this occurrence, an investment factor for each fisherman i is computed. High relative net income and associated positive investment is assumed to improve the relative cost position of an individual fisherman. The investment factor, x_i, determines the adjustment in the next season's cost by the percent deviation from the mean of the current year's net income where

$$x_i = \frac{\overline{NI} - NI_i}{\overline{NI}} \tag{3.28}$$

and

$$\overline{NI} = \sum_{i=1}^{N_b} NI_i / N_b \tag{3.29}$$

where N_b is the number of fishermen (boats). And, the resulting cost adjustment for next season is $1 - x_i$ on total costs for fisherman i.

Required Model Parameters

Other required model parameter include the number of fishing areas, the maximum number of fishing trips per season, the simulated horizon, i.e. the number of fishing seasons, and the number of fishing boats in the fleet.

Chapter 4

Model Outcomes

This chapter presents the results of the models for data fit to a given commercial fishery. For reasons of data accessibility, modelling procedures, stationarity of the resource and current stock management strategies, the scallop fisheries on Georges Bank (North Atlantic Fisheries Organization Zone 5Ze - Canadian portion) has historically been observed to be an appropriate activity on which to apply these modelling procedures.

4.1 Georges Bank Scallop Fisheries Specifications and Regulations

With respect to current activities of the Georges Bank commercial fishery, a brief summary of the institutional context of this fishery is provided here. (See also Canada [8,9], Mohn *et al.* [61,62,64], and Caddy [6]).

The scallop fishery is a major industry on which many smaller businesses depend, especially in southwest Nova Scotia. Scallop fishermen are clustered under a fixed number of operating firms that dock in Lunenburg, Riverport or Yarmouth. Gear regulations include specifications on the size of the dredge: no shorter than 20m and no longer than 60m with no restriction as to width (most boats vary from 29m to 41m in length). There is normally a 16 member crew per fishing boat. The crew receives approximately 40% of the catch's landed value.

With respect to licensing, an average of 75 to 80 scallop fishing licences are issued every year. The scallop fishery operates under an *Enterprise Allocation* scheme, that is each fishing company which receives a dragging license also receives a Total Allowable Catch (TAC) quota for scallops effective for the season. Other key regulations include the trip length adjusted every year (for 1988 it was either a 13,700 kg catch or 12 days dock-to-dock) and, for some fishing zones, the *meat count regulation.* Meat count is a measure of average scallop age/size and is used to ensure that the dredge's fishing rings are not capturing nor removing too many younger, smaller recruits. In 1988, the meat count regulation for some areas in Georges Bank was set at either 33 meats/500g or 44 meats/500g. Finally, the fishing mortality rate for the 1988 season decreased towards $F_{0.1}$ from F_{max}. This decrease was justified by a reduction in fishing effort. The major objective of this rate is the long-term stability of the harvest.

Scallop prices will remain sluggish until a significant reduction in the high inventories is observed, the prices having suffered a severe 45% real drop since 1987 (See also Appendix B).

The United States represents, by far, the biggest market for Georges Bank scallop exports. Among scallop export sales to other countries, the largest are negligible when compared to exports to the U.S. In 1988 Canada held a 38% share of the U.S. scallop market. Details (DFO [9]) with respect to 'Q'uantity (Metric Tonnes) and 'V'alue ('000$) are shown in Table 4.1.

Country	*1985*		*1986*		*1987*	
	Q	V	Q	V	Q	V
France	20	129	0	0	177	2704
West Germany	29	290	0	0	0	0
Switzerland	0	0	10	68	5	86
Japan	7	54	8	44	35	486
Bermuda	0	1	0	0	2	31
U.S.A.	4068	59153	4213	64837	4251	63844
Others	3	53	16	271	189	1149
Total	4127	59677	4247	65220	4659	68300

Table 4.1: Canadian Exports of Frozen Scallops

4.2 Model Parameters

A horizon of five fishing seasons is simulated, with each season's length bounded by the TAC quota or a maximum number of fifteen fishing trips. With respect to the fishing grounds, Caddy [6] justifies the split of fishing zone 5Ze of the North Atlantic Fisheries Organization (Canadian portion of Georges Bank) into a fixed number of areas, this from underwater contour reproductions and statistical analysis on historical catches and age data. The models' results have been obtained using 6 distinct areas.

Two simulation scenarios are modelled. On the one hand, the *single fisherman* scenario models a single fishing boat making 15 trips per season for 5 seasons. On the other hand, the *fleet simulation* scenario models a fleet of 50 fishermen making 15 or less trips per season (the fishing grounds close when the TAC for the season has been reached) for 5 seasons. As previously mentioned, the Georges Bank scallop fishery currently operates under an Entreprise Allocation scheme, i.e. the quota is proportionally divided into the number of companies operating that fishery. This situation is not modelled. However, since it is a compromise between the single and fleet scenarios modelled here, the performances of these scenarios will provide an idea on how the actual costs and earnings currently behave, since each company exploit the fishing ground (up to their EA) according to their own set of policies.

Currently, around 77 scallop fishing licences are issued for the fishing zone 5Ze of the North Atlantic Fisheries Organization and historically, under 60 fishermen honour their licence in a given season (Offshore Scallop Fishery Management Plan [9]).

No precise specification was input for catch or economic return units (the lack of public information limits the collection of actual cost and earning data). Rather, catch values were scaled by area. For example, when simulating the adaptive algorithm, a *good* catch in area 1 will yield a value of 0.10 standard weight units. The complete tables of model input data for both algorithms are provided in Appendix E.

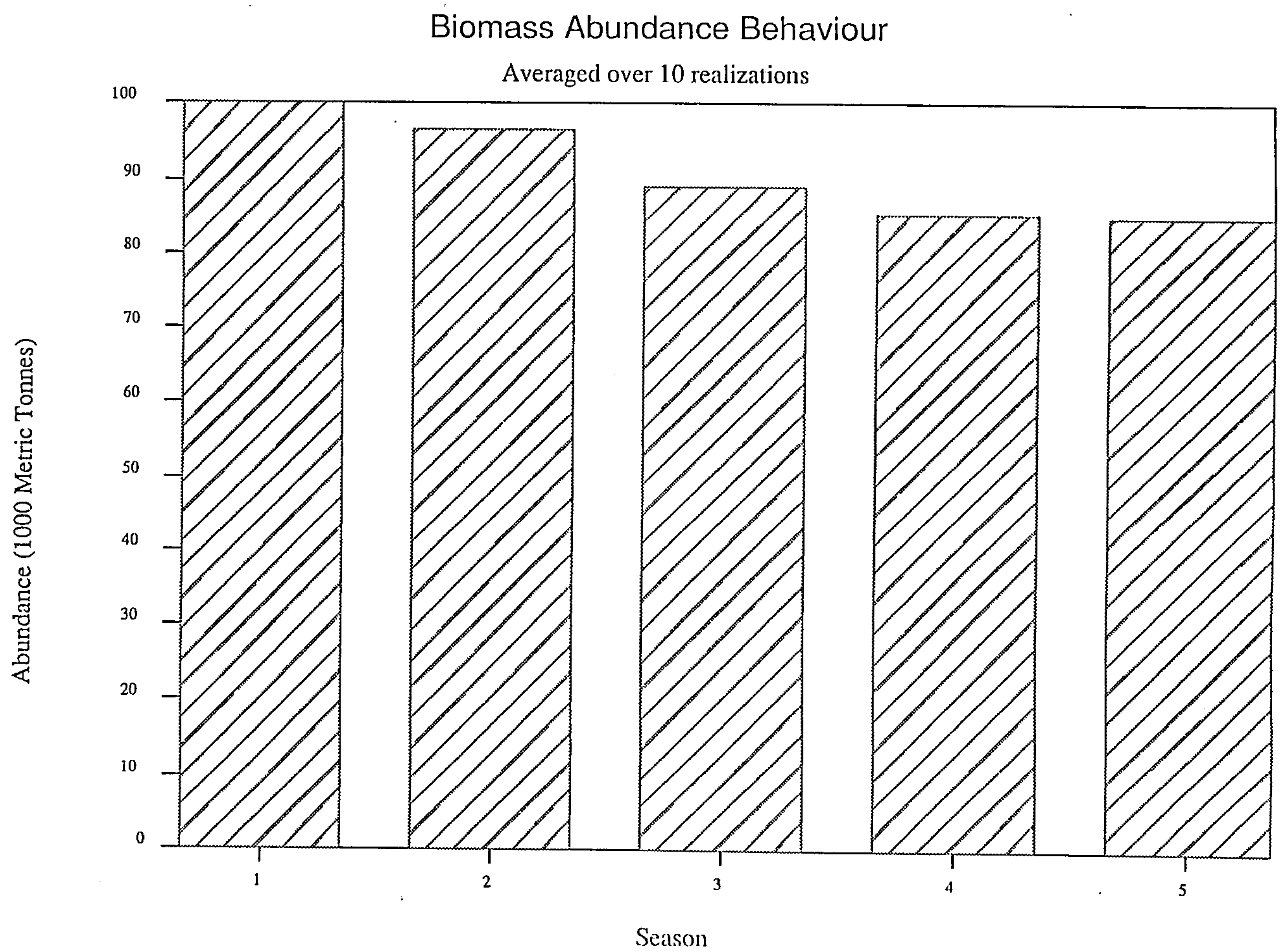
Little is known about the actual stock-recruitment behaviour of scallops (Sinclair *et al.* [82], Mohn *et al.* [61,62,63,64]). We can however provide reasonable assumptions about recruitment: the error term in the recruitment equation $R = \alpha X e^{-\nu\epsilon}$ follows a standard normal distribution ($\mu = 0, \sigma = \nu$). The stock biomass is therefore randomized over the simulated seasons in the model. Figure 4.1 shows how the biomass behaves over a five season simulation under the $F_{0.1}$ fishing mortality regulation for scallops. The level of biomass is the key biological performance measure of the system.

4.3 Results

Different economic performance measures are computed in the model to provide feedback about the system. These indicators are calculated for each boat and fishing trip. For the fleet scenario, the 'length' of a season i.e., number of fishing trips, is equal to the number of trips required to reach the Total Allowable Catch quota up to a maximum of 15 trips. For the single fisherman scenario, the number of trips within a season is set at the maximum, 15.

Definition of Performance Indicators

Define Y_{ik}, C_{ik} and p_k to be the class-mark catch level of boat number i during trip k, the cost of fishing for boat number i during trip k, $k = 1, \ldots, N$ and the



Biomass Abundance Behaviour
Averaged over 10 realizations
Abundance (1000 Metric Tonnes)
0
10
20
30
40
50
60
70
80
90
100
1
2
3
4
5
Season

price per unit weight for catch in trip k. Four seasonal economic performance indicators are defined as follows:

Total Seasonal Catch by weight for fisherman i (TC_i)

$$TC_i = \sum_{k=1}^{N} Y_{ik} \tag{4.1}$$

Seasonal Landed Value per fisherman i (LV_i)

$$LV_i = \sum_{k=1}^{N} p_k Y_{ik} \tag{4.2}$$

Seasonal Total Cost per fisherman i (TS_i)

$$TS_i = \sum_{k=1}^{N} (1 + x_i) C_{ik} \tag{4.3}$$

Seasonal Net Income per fisherman i (NI_i)

$$NI_i = LV_i - TS_i \tag{4.4}$$

where x_i is the cost saving investment factor computed by subtracting the average of the boats net income during the previous season from the i th boat's previous season income and then dividing by the aforementioned average:

$$x_i = \frac{\overline{NI} - NI_i}{\overline{NI}} \tag{4.5}$$

A set of ten realizations of the simulation model was run and the average results computed. Each realization consisted of a set of five seasons. The reported performance indicators were averaged over the results of these realizations.

4.3.1 Results: Myopic algorithm

Sample outputs of the myopic algorithm's code are provided in Appendix C based on the input data from Appendix E. Table 4.2 gives a summary of the season by season economical performance of both the fishing fleet scenario and the single fisherman scenario for the myopic model.

50 boats		Season 1	2	3	4	5
Total Catch	Max	1.49	1.39	1.37	1.26	1.02
	Min	0.87	0.80	0.83	0.61	0.45
	Avg	1.17	1.13	1.02	0.90	0.74
	Stdev	0.14	0.14	0.12	0.14	0.12
Landed Value	Max	1587	1487	1462	1324	1073
	Min	923	842	881	640	472
	Avg	1251	1208	1090	947	775
	Stdev	153	150	125	150	128
Total Cost	Max	198	248	234	204	190
	Min	198	136	110	109	75
	Avg	198	189	173	147	125
	Stdev	0	27	26	20	24
Net Income	Max	1388	1317	1286	1158	947
	Min	725	623	684	503	387
	Avg	1053	1019	917	799	650
	Stdev	153	154	127	149	126
Number of trips		7	4	9	11	9
One boat	Tot. Catch	2.76	2.19	2.81	2.37	2.70
	Lan.Value	3571	2913	3615	2976	3492
	Tot. Cost	480	480	480	480	480
	Net Income	3091	2433	3135	2496	3012
Number of trips		15	15	15	15	15

Table 4.2: Seasonal Results for the Myopic Algorithm

Analysis

For the single boat scenario results in Table 4.2, fluctuations occur in each indicator's results. Intuitively, this makes sense since the catch quota is never reached with one single fisherman and the impact on the stock level is minimal from season to season.

For the fleet scenario, it is interesting to note the downward trend over all seasons for each of the four performance indicators also highlighted in the graph of these indicators (Appendix C). This corresponds to the downward trend observed in the stock biomass behaviour. The stock behaviour is the major contributor to this trend since stock abundance governs the catch result. In addition, the fishing season ends when the TAC has been reached since the number of fishing trips per season never exceeds 11. This number varies from 4 (season 2) to 11 (season 4). Included in Appendix C are the average economic performance by trip for all vessels in the fleet for each of the four indicators.

As indicated in Figure 4.2, most fishermen are situated in the higher levels of the net income range for the initial trips of the season. When more trips are required to reach the TAC, a shift toward the lower levels of the net income range in the fleet distribution can be observed since seasons are generally completed (TAC reached) with 7 or 8 fishing trips.

4.3.2 Results: Adaptive Algorithm

As previously detailed, the adaptive algorithm yield a decision policy derived from dynamic programming theory. This policy consists of a series of non-dominated decisions at each fishing trip. A copy of the computed policy for the algorithm is provided in Appendix D. Similarly to the myopic algorithm, the fishing activity is simulated and the results and associated performance indicators recorded based on input data in Appendix E. An excerpt of the fishing simulation model is also provided in Appendix D. Table 4.3 provides the summary of the average season by season performance indicators for the single fisherman and the fishing fleet scenarios.

Analysis

The adaptive algorithm's performance indicators do not exhibit a downward trend throughout the five season simulation. Rather, a correction is implicitly imposed in the policy in order to forecast the biomass' behaviour and *adapt* to it. This is an example of the major difference between the myopic and adaptive algorithm. The myopic policy will not 'see' this trend and react to it in the same way the adaptive policy will. The adaptive policy is more sensitive to state fluctuations.

```
STATISTICS ON ACCUMULATED      NET INCOME
              MAXIMUM VALUE =      1337.53
              MINIMUM VALUE =         0.00
```

HISTOGRAM

CLASS MARK															
133.75	0	0	0	0	0	0	32	50	50	50	50	50	50	50	50
267.51	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0
401.26	0	0	0	0	2	7	3	0	0	0	0	0	0	0	0
535.01	1	3	1	5	12	17	0	0	0	0	0	0	0	0	0
668.76	6	6	12	16	15	17	0	0	0	0	0	0	0	0	0
802.52	18	17	15	16	15	8	0	0	0	0	0	0	0	0	0
936.27	16	11	16	11	5	1	0	0	0	0	0	0	0	0	0
1070.02	8	11	6	1	1	0	0	0	0	0	0	0	0	0	0
1203.77	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0
1337.53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Figure 4.2: Myopic algorithm; Fleet distribution; Net Income

		Season				
50 boats		1	2	3	4	5
Total Catch	Max	0.97	0.91	0.84	0.80	0.82
	Min	0.87	0.80	0.72	0.66	0.69
	Avg	0.91	0.86	0.79	0.74	0.75
	Stdev	0.02	0.03	0.02	0.03	0.02
Landed Value	Max	1166	1104	1017	962	990
	Min	1040	960	878	799	827
	Avg	1098	1036	955	879	906
	Stdev	28	33	26	34	30
Total Cost	Max	273	276	257	242	255
	Min	271	233	214	189	196
	Avg	272	253	238	220	224
	Stdev	0.49	9	11	9	12
Net Income	Max	893	846	785	744	769
	Min	768	709	621	579	619
	Avg	826	783	717	659	681
	Stdev	28	34	29	34	31
Number of trips		12	13	9	9	15
One boat	Tot. Catch	1.11	1.09	1.10	1.06	1.10
	Lan. Value	1414	1387	1402	1345	1402
	Tot. Cost	322	321	322	322	322
	Net Income	1092	1066	1080	1023	1080
Number of trips		15	15	15	15	15

Table 4.3: Seasonal Results for the Adaptive Algorithm

Figure 4.3 provides the seasonal net income distribution for the adaptive model fleet scenario. Appendix D plots the performance indicators over the five seasons for the adaptive algorithm in addition to the myopic algorithm's. The 'correction' referred to above is clearly seen when moving from season 4 to season 5 when the trend reverses into an increase for this seasonal update.

An interesting feature to the adaptive algorithm is its practicality. This algorithm begins by characterizing an optimal action *policy*. This policy summarizes future decision options throughout the season and is indicative of empirical fleet dynamics patterns actually observed. The myopic model is somewhat more artificial especially for the many experienced vessel operators in this closed fishing sector. It may however be representative of decision making by new or learning vessel operators as they establish a more expansive database of knowledge about this particular fishery.

Sensitivity

Throughout the testing phase when developing either algorithm, significant domination by some fishing areas was found for many different datasets. For example, if all areas have the same q_{jl} probability matrix and one has a favorable difference of only 10% in the cost of fishing, everything else being the same, that area would generally be dominant throughout a season. If fact, this is what is observed in practice. Fishermen identify areas that are more attractive than others, and those areas are exploited over a season. Only after exhausting rewards from an area are forays made into new areas as a search for higher returns.

STATISTICS ON ACCUMULATED NET INCOME

MAXIMUM VALUE = 323.99

MINIMUM VALUE = 78.54

HISTOGRAM

CLASS MARK															
103.09	0	0	0	0	0	0	0	0	0	0	0	0	0	0	29
127.63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21
152.18	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0
176.72	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0
201.27	0	0	0	0	0	0	0	0	0	0	2	0	2	12	0
225.81	1	3	0	1	0	0	0	0	0	1	16	1	19	1	0
250.36	17	12	8	9	2	2	2	1	12	7	25	25	20	0	0
274.90	29	26	30	27	25	28	24	13	21	31	7	18	9	0	0
299.45	2	9	12	13	17	14	20	25	14	8	0	6	0	0	0
323.99	1	0	0	0	6	6	4	10	3	3	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Figure 4.3: Adaptive algorithm; Fleet distribution; Net Income

Chapter 5

Conclusion

5.1 Discussion

Stock-Recruitment

The biological performance of Deriso's stock dynamics model governs the behaviour of the economic performance indicators. Total catch and total costs (net income and landed value being a linear combination of those and of the market unit price) react to the behaviour of the abundance's.

The recruitment function assumed a linear growth (log-normal error) type recruitment (Equation 3.9). In the abscence of a density dependent parameter this function characterizes a monotonically increasing stock abundance curve that does not converge to a stable equilibrium point (carrying capacity). Figures 3.2 and 3.3 present the stock and recruitment interactions without a density-dependent term. However in the results of the experiments carried out using this linear recruitment function with variability and a conservative harvesting component, the stock was not observed to increase. Rather, over the admittedly short horizon of 5 seasons, the stock was observed to fall relative to initial values. It would be anticipated that density dependence, bounded stock growth, would accentuate this stock decline, everything else being equal.

Harvest

Several models could be developed to describe the fishermen's decision-making process. The two presented here (decision tree, dynamic programming) require a few basic components fundamental to the mechanics of the actual process.

The fact that they require easily accessible data adds to their robustness. Further development within this methodology will yield improved algorithms incorporating better data on probability measures, costs levels and other model parameters.

Fleet Dynamics

An interesting phenomenom was observed on the evolution of the fleet over the 5 seasons. Consider fishermen whose total net income after the initial seasons are higher than the fleet's average. The difference is explained by the effect of probability since everyone starts with the same knowledge and information. As these fishermen proceed to later seasons, their lead increases since they reinvest their earnings in increasing capital efficiency. They now have improved travelling and exploring capabilities and can therefore reach faster their preset TAC.

For fishermen with earnings below average, their lack of investment resources will cause the gap between them and the other fishermen to increase also. The capital resources do not profit from seasonal reinvestment. The need to do more and more surveying to find beds which have not been fished by highliners will increase hence causing and increase in their fishing costs.

5.2 Extensions

One interesting scenario not presented in this thesis but of interest to many is the fleet scenario 'with information sharing'. This means that each boats' π-vector would also reflect the other boats trip and catch information. This would increase the accuracy in the model versus real life situation of the fishery. As previously mentionned, since between 7 and 10 companies operate scallop fishing vessels under Entreprise Allocations on Georges Bank, 'information sharing' could be restricted for boats within each company. This would also allow for an enhanced look at the competitive market in which Entreprise Allocation fishing quota system is currently implemented.

Several simulation model parameters are actually data input. It is therefore possible to observe what happens with different fleet sizes. If some indicators are fixed (total cost or total catch), it could be interesting to study what 'optimal' fleet size should exploit the resource. The optimality criterion could be the season's length before reaching quota, the number of areas visited in a fixed number of trips within the season or some other bioeconomic measure.

A fishing area restriction scenario can also be studied by inputing an infinite fishing cost for specific area to be closed over a season. Results could provide insights into possible alternative control measures than the current TAC quota tactics.

The resource's unit price is also modelled under a given function (negative quadratic) in this case corresponding roughly to actual price fluctuations during a season (see also Appendix B). It could be interesting to reflect on the effects of a seasonal linear growth of the price, or looking at the indicators resulting from the application of a constant price.

It is difficult to model a fisherman's own peculiar way of decision making. We have tried in this thesis to represent it by researching the quantifiable elements of the decision-making process. These elements were analyzed. the relevant one transformed (discretization, measurement, or else) and quantified. Most of the values in the initial tableau (Appendix E) consisted of logical input, not necessarily exact in real life, but justifiable by their availability and use elsewhere. Another possible extension to this work would be to precisely quantify those variables (area cost of fishing, q_{jl} matrix) with an in-depth statistical study of current fishing fleet operations.

The fishery's actual costs and earnings are also measured to a certain degree. These actuals are not published or available from public sources. Given the estimates on the price and the costs, a cost function from the resulting data can be evaluated. A supplementary decision tool is readily available from minimal transformation of the model results.

With respect to algorithm design, the models exposed here use quantifiable components of the economical and biological environments. These are also the main constituents of the harvestors' decision making process. However, within a structural framework, we have tried to show that the actual components provide rational, easily applied decision tools for fishermen.

Appendix A

Walford plot

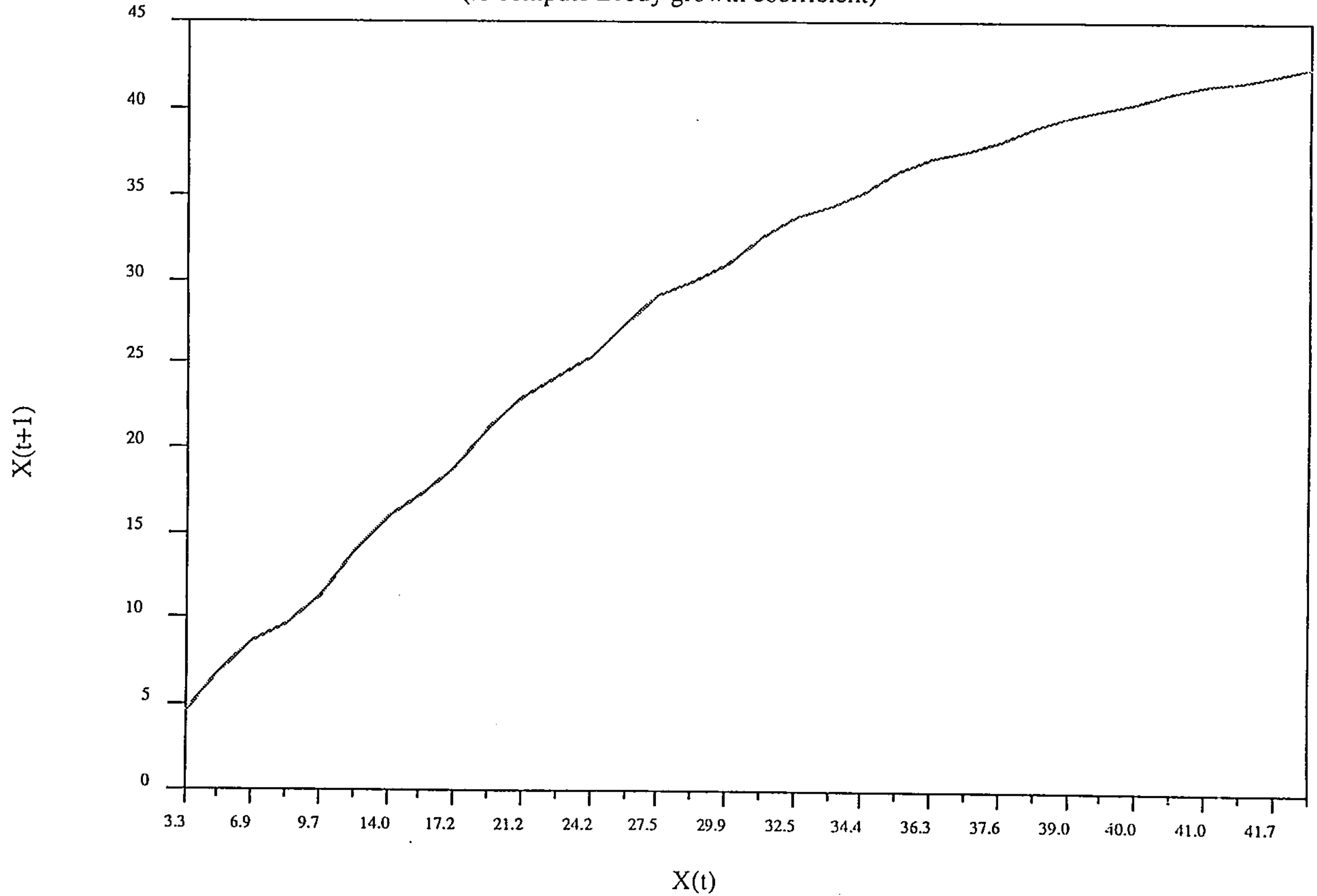
Gulland [28] estimates the rate of body growth by using the Brody equation. The parameters of that equation are measured by applying a linear regression model to W_t, the unit weight at age t, against W_{t+1}. The data was taken from the latest assessment report (Mohn [64]).
The following linear equation was obtained from data for scallops from 3 to 12 years old.

$$Y = 2.29 + 0.958X \tag{A.1}$$

($R^2 = 99\%$)
Therefore, the Brody growth coefficient (ϱ) equals 0.958.

Walford plot X(t) vs X(t+1)

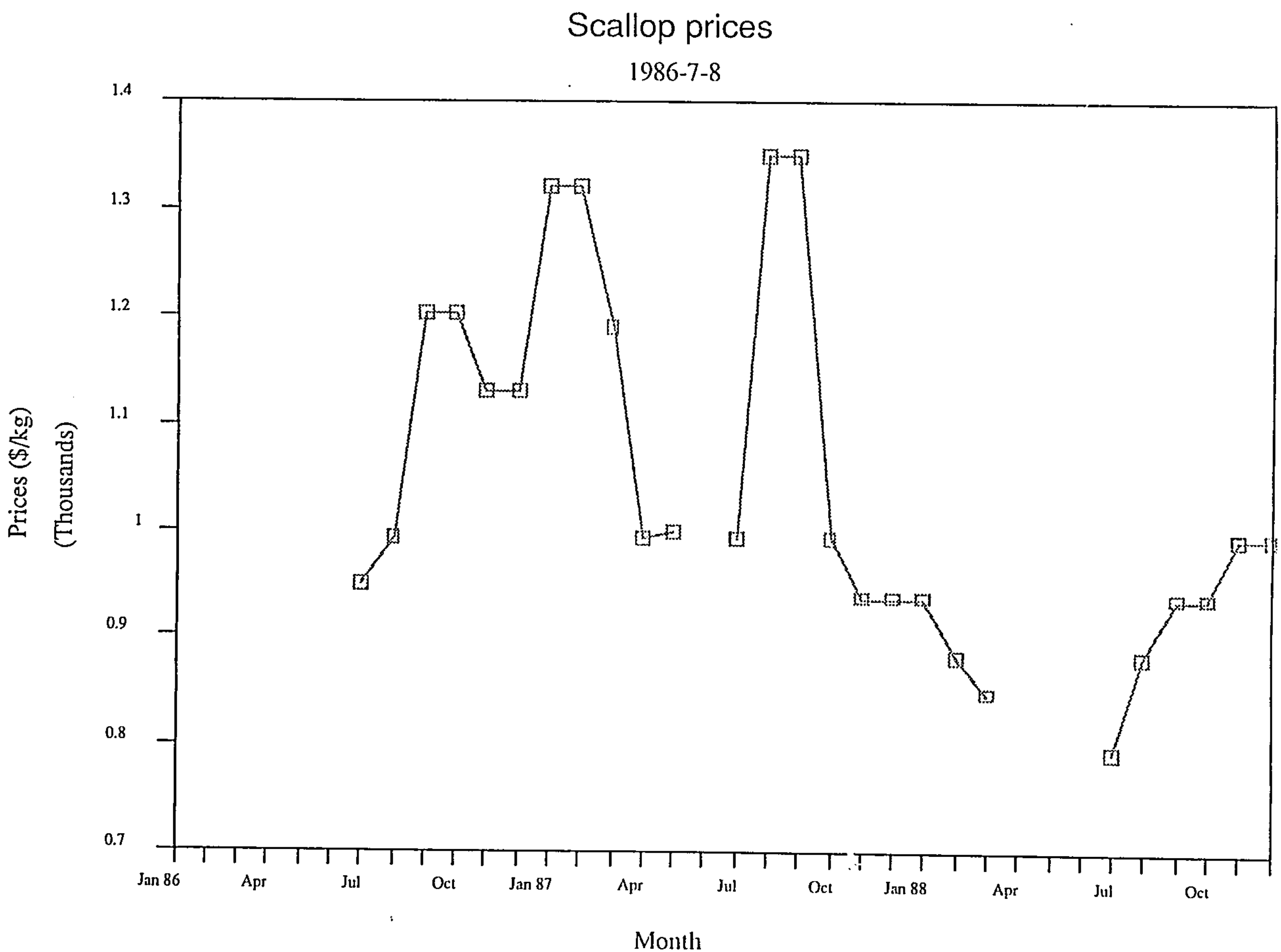
(to compute Brody growth coefficient)



Appendix B

Scallop prices

The following graph show the Georges' Bank scallop unit prices (unadjusted) for three years of available data 1986-1988 (Canada [8]).


Scallop prices
1986-7-8
Prices ($/kg)
(Thousands)
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
Jan 86
Apr
Jul
Oct
Jan 87
Apr
Jul
Oct
Jan 88
Apr
Jul
Oct
Month

Appendix C

Myopic Algorithm

This appendix presents the key computer code for the myopic algorithm (Figure C.1). The code is written in FORTRAN-VS.
Also included is a sample of the fishing simulation output for the fleet scenario (Figure C.2) from the myopic algorithm and compiled statistics for some performance indicators. A graphical overview of these indicators (seasonal) is provided in Figure C.3.

Figure C.1: Myopic Algorithm: Main program

```
      PROGRAM MYOPC
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER AREA,CATCHN,IREAL,ISEED,NAREAS,NBOATS,NS,NSEASN,NTRIPS
      INTEGER REFER,NT(10)
      CHARACTER*8 CATCH(3)
      CHARACTER*8 ACTUAL
      DIMENSION PD(2,6),PYD(3,2,6),PDY(2,3),PY(3,6),PYX(3,3,6)
      DIMENSION P(3,3),PI(3),PIB(50,3),CD(6),CF(6),QCATCH(3,6)
      DIMENSION TCATCH(5,15,50),TCOST(5,15,50),NINC(5,15,50)
      DIMENSION VALAND(5,15,50),PRCDEV(50),TTCATC(5,15,50)
      DIMENSION TTCOST(5,15,50),NNINC(5,15,50),VVALAN(5,15,50)
      COMMON/BLOC1/PYX,PDY,QCATCH
      COMMON/BLOC3/TCATCH,TCOST,VALAND,NINC
      COMMON/BLOC4/TTCATC,TTCOST,VVALAN,NNINC
      DATA XTM1/100.0/,RI/40.0/,RTM1/40.0/
      DATA ISEED/6325791/
      CATCH(1)='GOOD    '
      CATCH(2)='AVERAGE '
      CATCH(3)='POOR    '
C-----------------------------------------------------------------------
C  -- MAIN PROGRAM- READ AND WRITE CONSTANT/INITIAL DATA
C-----------------------------------------------------------------------
      CALL LIRE(IREAL,NAREAS,NBOATS,NTRIPS,NSEASN,CMF,CD,CF)
      DO 300 I=1,NSEASN
       DO 300 II=1,NTRIPS
        DO 300 III=1,NBOATS
         TTCATC(I,II,III)=0.0
         TTCCST(I,II,III)=0.0
         NNINC(I,II,III)=0.0
         VVALAN(I,II,III)=0.0
300   CONTINUE
C
C     CALL ECRIR1(NAREAS,NBOATS,NTRIPS,CD,CMF,PRICE,XT,PI,PY,
C    *            PD,P,NS,CATCH,CF)
C
      DO 10 IR=1,IREAL
       DO 100 I=1,NBOATS
100     PRCDEV(I)=1.0
       DO 200 I=1,3
200     PI(I)=1.0/3.0
       DO 400 J=1,NBOATS
        DO 400 I=1,3
400      PIB(J,I)=PI(I)
       XT=100.0
       PRICE=600.0
       CALL HCOMP(H)
       DO 20 NS=1,NSEASN
        ISEED=ISEED+2
        ISTOP=0
        WRITE(6,*) ' DOING SEASON NUMBER....',NS
C  LOCATE MATRICES' ROW ASSOCIATED WITH CURRENT STATE
        CALL REFERE(XT,REFER,ACTUAL)
        DO 30 IINT=1,NTRIPS
         NT(NS)=IINT
C        WRITE(6,*) '          TRIP ',NT(NS)
```

```
C     COMPUTE CATCH, TESTING AND ASSOCIATED CONDITIONAL DISTRIBUTIONS
        DO 40 IBOAT=1,NBOATS
C         WRITE(6,*) '                BOAT:',IBOAT
          DO 500 I=1,3
500         PI(I)=PIB(IBOAT,I)
          CALL PYCALC(PI,PY)
          CALL PDCALC(PY,PD)
          CALL PYDCAL(PYD,PY)
          CALL DECISI(AREA,II,ISEED,NAREAS,QBEST,NT,NS,CMF,PRICE,
     *                CD,CF,PD,PY,PYD)
          CALL FISHIN(CATCH,AREA,CATCHN,IBOAT,IR,IREAL,ISEED,NS,
     *                QBEST,REFER,NT,CMF,PRICE,CD,CF,PRCDEV)
          CALL UPDPI(PI,CATCHN,AREA)
          DO 600 I=1,3
600         PIB(IBOAT,I)=PI(I)
40      CONTINUE
        CALL CHECK(IR,ISTOP,NBOATS,NS,NT,H,TCATCH,XT)
        IF(ISTOP.EQ.1) GO TO 70
30     CONTINUE
70     IF(NT(NS).EQ.16) NT(NS)=15
       WRITE(4,1)'SEASONAL TOTAL CATCH',NS,XT,ACTUAL,IR
       CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,TCATCH)
       WRITE(4,1)'SEASONAL LANDED VALUE',NS,XT,ACTUAL,IR
       CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,VALAND)
       WRITE(4,1)'SEASONAL TOTAL COSTS',NS,XT,ACTUAL,IR
       CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,TCOST)
       WRITE(4,1)'SEASONAL NET INCOME',NS,XT,ACTUAL,IR
       CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,NINC)
       CALL DEVIA(NBOATS,NS,NT,PRCDEV,TCOST,VALAND,NINC)
       CALL UPDST(H,P,PIB,RT,RTM1,XT,XTM1,ACTUAL,ISEED,NBOATS,REFER)
20    CONTINUE
10    CONTINUE
C     ****************
      IFLAG=1
      DO 80 NS=1,NSEASN
        WRITE(4,2)'OVERALL TOTAL CATCH',NS
        CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,TTCATC)
        WRITE(4,2)'OVERALL LANDED VALUE',NS
        CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,VVALAN)
        WRITE(4,2)'OVERALL TOTAL COSTS',NS
        CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,TTCOST)
        WRITE(4,2)'OVERALL NET INCOME',NS
        CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,NNINC)
80    CONTINUE
C     ****************
      CALL STAT2(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,VVALAN)
1     FORMAT('1','STATISTICS ON **** ',A25,' **** SEASON NO.  ',I2,
     *       ', CURRENT STATE: X= ',F6.1,' -----> ',A8,/,46X,'****',
     *       'REALIZATION:',I2)
2     FORMAT('1','STATISTICS ON **** ',A25,' **** SEASON NO.  ',I2)
      CALL STAT3(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,VVALAN)
      STOP
      END
C
C     WRITE(8,300) NT,NS
```

```
C        WRITE(8,305)((PY(I,J),I=1,3),J=1,6)
C        WRITE(8,310)((PD(I,J),I=1,2),J=1,6)
C300     FORMAT('1',30X,'TRIP NO. ',I3,' OF SEASON ',I2,/,25X,37('-'),//)
C305     FORMAT('0','CATCH DISTRIBUTION -----> P(Y)',/,1X,18('-'),//,
C       *14X,'AREA 1',10X,'AREA 2',10X,'AREA 3',10X,'AREA 4',
C       *10X,'AREA 5',10X,'AREA 6',/,12X,94('-'),/,13X,
C       *6('G',4X,'A',4X,'P',5X),/,12X,94('-'),/,12X,6(3(F4.2,1X),1X),/)
C310     FORMAT('0','DRAG DISTRIBUTION -----> P(D)',/,1X,18('-'),//,
C       *14X,'AREA 1',10X,'AREA 2',10X,'AREA 3',10X,'AREA 4',
C       *10X,'AREA 5',10X,'AREA 6',/,12X,94('-'),/,12X,
C       *6(' FAVOR.   UNFAV. '),/,12X,94('-'),/,13X,6(2(F5.3,3X)),/)
         STOP
         END
C
C
         SUBROUTINE TRANS(P,XTM1,RTM1)
         IMPLICIT REAL *8(A-H,M-Z)
         DIMENSION P(3,3)
         DOUBLE PRECISION XX,Y
         DATA GL/125.0/,AL/75.0/,X/50.0/
         DATA R/0.95776147/,M/0.1/,ALPHA/0.4/
         DISC=SQRT((R*(ALPHA-1.0)-1.0)**2-4.0*R*(1.0-ALPHA))
         F1=-LOG((1.0-R*(ALPHA-1.0)+DISC)/(2.0*R))-M
         F2=-LOG((1.0-R*(ALPHA-1.0)-DISC)/(2.0*R))-M
         IF(F1.GT.0.0.AND.F1.LT.1.0) THEN
           F=F1+0.05
         ELSE
           F=F2+0.05
         ENDIF
         PI=4.0*ATAN(1.0)
         S=EXP(-F-M)
         H=F*(1.0-S)/(F+M)
         I=0
99       I=I+1
         R=0.0
C        THE FOLLOWING BOUNDS ARE FOR THE DERISO MODEL
C        AA=((GL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/(ALPHA*X))
C        WRITE(6,*) ' GL:',AA
         IF(AA.LE.0.0) WRITE(6,1) X,XTM1,RTM1,AL,R,S,ALPHA,F
1        FORMAT(1X,4(F5.0,1X),4(F5.3,1X))
         AA=((AL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/(ALPHA*X))
C        WRITE(6,*) ' AL:',AA
         IF(AA.LE.0.0) WRITE(6,1) X,XTM1,RTM1,AL,R,S,ALPHA,F
         RHSL=-LOG((GL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/
        *(ALPHA*X))
         RHSR=-LOG((AL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/
        *(ALPHA*X))
         XX=RHSL
         CALL MDNORD(XX,Y)
C        P(I,1)=L'INTEGRALE DE -L'INFINI A RHSL DE F(X) - DENSITE NORMALE
         P(I,1)=Y
         XX=RHSR
C        P(I,2)=L'INTEGRALE DE RHSL A RHSR DE F(X) - DENSITE NORMALE
         CALL MDNORD(XX,Y)
         P(I,2)=Y
```

```
      P(I,2)=P(I,2)-P(I,1)
C     P(I,3)=L'INTEGRALE DE RHSR A L'INFINI DE F(X) - DENSITE NORMALE
      P(I,3)=1.0-P(I,1)-P(I,2)
      IF(I.LT.3) THEN
C      WRITE(6,*) ' I= ',I
       X=X+50.0
       GOTO 99
      ENDIF
      RETURN
      END
C
C
      SUBROUTINE LIRE(IREAL,NAREAS,NBOATS,NTRIPS,NSEASN,CMF,CD,CF)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NAREAS,NTRIPS,NSEASN,NBOATS
      DIMENSION P(3,3),PYX(3,3,6),QCATCH(3,6),PDY(2,3)
      DIMENSION CF(6),CD(6)
      COMMON/BLOC1/PYX,PDY,QCATCH
C  READ IN :
C     PYX(I,J) CATCH GIVEN STATE DISTRIBUTION
      DO 15 I=1,6
       READ(5,81)
       DO 20 J=1,3
        READ(5,*)(PYX(J,K,I),K=1,3)
20    CONTINUE
15    CONTINUE
C
C      READ PDY VECTOR (TEST RESULTS CONDITIONAL ON CATCH)
       READ(5,81)
       DO 40 I=1,2
40      READ(5,*)(PDY(I,J),J=1,3)
C
C      READ NUMBER OF AREAS, NO OF TRIPS/SEASON
C       AND NUMBER OF SEASONS FOR SIMULATION.
C
       READ(5,81)
       READ(5,*) NAREAS,NBOATS,NTRIPS,NSEASN,IREAL
C
C      READ THE COSTS OF FISHING FOR EACH AREA
C
       READ(5,81)
       READ(5,*)(CF(I),I=1,NAREAS)
C
C      READ THE COSTS OF TEST-FISHING FOR EACH AREA
       READ(5,81)
       READ(5,*)(CD(I),I=1,NAREAS)
C
C      READ THE COSTS OF MOVING
       READ(5,81)
       READ(5,*) CMF
C     READ AREA'S QUANTITATIVE CATCHES ASSOCIATED WITH GOOD,AVG AND POOR
C
       READ(5,81)
       DO 30 I=1,3
30      READ(5,*)(QCATCH(I,J),J=1,6)
```

```
81    FORMAT(A50)
      RETURN
      END
C
C
      SUBROUTINE RAND(ISEED,U)
C
C MODIFIED RANDOM NUMBER GENERATOR (V.DYCK,J.LAWSON,J.SMITH
C  "INTRODUCTION TO COMPUTING", RESTON,VA.(1979))
C ==============================================================
C LINEAR CONGRUENTIAL GENERATOR FOR 32-BIT WORD COMPUTER.
C CYCLE IS 2**29 IN LENGTH: VARIABLES ARE U(0,1)
C
C
      INTEGER ISEED
      ISEED=ISEED*843314861+453816693
      IF(ISEED.LT.0)ISEED=ISEED+2147483647+1
      U=ISEED*0.4656612E-9
      RETURN
      END
C
C
      SUBROUTINE REFERE(XT,REFER,ACTUAL)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER REFER
      CHARACTER*8 ACTUAL
      IF(XT.LT.75.0) THEN
       REFER=3
       ACTUAL= 'POOR'
C      XT=50.0
      ELSE
       IF(XT.LT.125.0) THEN
        REFER=2
        ACTUAL='AVERAGE'
C       XT=100.0
       ELSE
        REFER=1
        ACTUAL='GOOD'
C       XT=150.0
       ENDIF
      ENDIF
      RETURN
      END
C
C
      SUBROUTINE ECRIR1(NAREAS,NBOATS,NTRIPS,CD,CMF,PRICE,X,PI,PY,PD,
     *P,NS,CATCH,CF)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NAREAS,NBOATS,NTRIPS,NS
      CHARACTER*8 CATCH(3)
      DIMENSION CF(6),CD(6),PI(3),PY(3,6),PYX(3,3,6),PD(2,6)
      DIMENSION PDY(2,3),P(3,3),QCATCH(3,6)
      COMMON/BLOC1/PYX,PDY,QCATCH
      WRITE(8,91) NAREAS,NTRIPS
      WRITE(8,92) (CF(I),I=1,6),
```

```
     *              (CD(I),I=1,6),CMF
      WRITE(8,93)
      WRITE(8,94)
      WRITE(8,95) X,NS
      WRITE(8,96) (P(1,J),J=1,3),
     *            (P(2,J),J=1,3),
     *            (P(3,J),J=1,3)
      WRITE(8,97) (PI(I),I=1,3)
      WRITE(8,980)
      DO 10  I=1,3
10     WRITE(8,98) CATCH(I),((PYX(I,J,K),J=1,3),K=1,6)
      WRITE(8,99) (PDY(1,J),J=1,3),
     *            (PDY(2,J),J=1,3)
      WRITE(8,990)(CF(I),I=1,6),
     *            (CD(I),I=1,6),CMF,PRICE
      WRITE(8,995) (CATCH(I),(QCATCH(I,J),J=1,6),I=1,3)
C
91    FORMAT('1','         ADAPTIVE DECISION MAKING BY A SINGLE FISHERMAN
     *WITH: ',//,15X,I3,3X,'FISHING AREAS,',/,15X,I3,3X,
     *'TRIPS PER SEASON',//)
92    FORMAT(' THE COSTS ARE DISTRIBUTED AS FOLLOWS (*)',//,25X,
     *'A R E A',/,15X,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,'6',/,
     *14X,35('-'),/,1X,'   FISHING:     ',6(F5.1,1X),/,1X,
     *'    TESTING:     ',6(F5.1,1X),//,1X,
     * '(*)  THE COST OF MOVING AND THEN FISHING IS : ',F7.2,///)
93    FORMAT(1X,'STATE DEFINITION:',/,10X,'GOOD ---------->  OVER 125',
     */,10X,'AVERAGE ------->  BETWEEN 75 AND 125',/,10X,'POOR --------',
     *'-->  BELOW 75',//,1X,20('-'),//)
94    FORMAT(' INITIAL STATE: X(0) = 100   ------> AVERAGE',//,1X,
     *20('-'),//)
95    FORMAT(1X,'ACTUAL STATE LEVEL ---------> ',F5.1,5X,'SEASON NUMBER'
     *,1X,I2,//,1X,56('-'),//)
96    FORMAT(' STATE TRANSITION MATRIX',/,1X,23('-'),//,13X,'TO',/,
     *' FROM :    G        A        P',/,1X,26('-'),/,3X,'G   :',
     *3(F6.2,1X),/,3X,'A   :',3(F6.2,1X),/,3X,'P   :',3(F6.2,1X),//,
     *1X,26('-'),//)
97    FORMAT(1X,'PI VECTOR',/,1X,9('-'),/,3X,'G       A       P',/,1X,
     *20('-'),/,2X,3(F5.3,1X),//,1X,20('-'),//)
98    FORMAT(3(1X,A8,':',6(3(F4.2,1X),1X),/),/)
980   FORMAT('1','CONDITIONAL DISTRIBUTION OF CATCH GIVEN STATE',/,1X,
     *45('-'),//,14X,'AREA 1',10X,'AREA 2',10X,'AREA 3',10X,'AREA 4',
     *10X,'AREA 5',10X,'AREA 6',/,1X,105('-'),/,4X,'Y',4X,
     *':  ',6('G',4X,'A',4X,'P',5X),/,1X,105('-'),/)
99    FORMAT( //,' CONDITIONAL DISTRIBUTION OF DRAG GIVEN CATCH',/,1X,
     *45('-'),//,16X,'Y',/,5X,21('-'),/,2X,'D',2X,
     *':    G       A       B',/,1X,
     *25('-'),/,2X,'F   :',3(F6.3,1X),/,2X,'U   :',3(F6.3,1X),//)
C
C
990   FORMAT(//,' COSTS OF FISHING AND TESTING PER AREA',/,1X,
     *37('-'),//,32X,'A R E A',/,11X,52('-'),/,11X,':       1',8X,'2',8X,
     *'3',8X,'4',8X,'5',8X,'6',/,11X,52('-'),/,1X,'FISHING   :',
     *6(F6.1,3X),/,1X,'TESTING   :',6(F6.1,3X),//,1X,
     *'THE COST OF MOVING AND THEN FISHING IS ',F7.2,/,1X,
     *' AND THE UNIT PRICE IS FIXED AT ', F7.2,//)
```

```
995   FORMAT(//,' QUANTITATIVE VALUES FOR CATCHES PER AREA',/,1X,       NBM03310
     *40('-'),//,32X,'A R E A',/,11X,46('-'),/,11X,'‡    1',7X,'2',7X,  NBM03320
     *'3',7X,'4',7X,'5',7X,'6',/,1X,56('-'),/,3(1X,A10,'‡',6(F5.2,3X),  NBM03330
     */),//)                                                              NBM03340
      RETURN                                                              NBM03350
      END                                                                 NBM03360
C                                                                         NBM03370
C                                                                         NBM03380
      SUBROUTINE PYCALC(PI,PY)                                            NBM03390
      IMPLICIT REAL *8(A-H,M-Z)                                           NBM03400
      DIMENSION PYX(3,3,6),PI(3),PY(3,6),PDY(2,3),QCATCH(3,6)             NBM03410
      COMMON/BLOC1/PYX,PDY,QCATCH                                         NBM03420
      DO 25 K=1,6                                                         NBM03430
       DO 30 I=1,3                                                        NBM03440
        SUM=0.0                                                           NBM03450
        DO 35 J=1,3                                                       NBM03460
35        SUM = SUM + (PYX(I,J,K) * PI(J))                                NBM03470
        PY(I,K)=SUM                                                       NBM03480
30     CONTINUE                                                           NBM03490
25    CONTINUE                                                            NBM03500
      RETURN                                                              NBM03510
      END                                                                 NBM03520
C                                                                         NBM03530
C                                                                         NBM03540
      SUBROUTINE PDCALC(PY,PD)                                            NBM03550
      IMPLICIT REAL *8(A-H,M-Z)                                           NBM03560
      DIMENSION PDY(2,3),PY(3,6),PD(2,6),PYX(3,3,6),QCATCH(3,6)           NBM03570
      COMMON/BLOC1/PYX,PDY,QCATCH                                         NBM03580
      DO 40 K=1,6                                                         NBM03590
       DO 40 I=1,2                                                        NBM03600
       SUM=0.0                                                            NBM03610
       DO 45 J=1,3                                                        NBM03620
45      SUM = SUM + (PDY(I,J) * PY(J,K))                                  NBM03630
       PD(I,K)=SUM                                                        NBM03640
40    CONTINUE                                                            NBM03650
      RETURN                                                              NBM03660
      END                                                                 NBM03670
C                                                                         NBM03680
C                                                                         NBM03690
      SUBROUTINE PYDCAL(PYD,PY)                                           NBM03700
      IMPLICIT REAL *8(A-H,M-Z)                                           NBM03710
      DIMENSION PDY(2,3),PY(3,6),PYD(3,2,6),PYX(3,3,6),QCATCH(3,6)        NBM03720
      COMMON/BLOC1/PYX,PDY,QCATCH                                         NBM03730
      DO 50 L=1,6                                                         NBM03740
       DO 50 I=1,3                                                        NBM03750
        DO 50 J=1,2                                                       NBM03760
         TOP = PDY(J,I)*PY(I,L)                                           NBM03770
         BOTTOM = 0.0                                                     NBM03780
         DO 60 K=1,3                                                      NBM03790
60        BOTTOM=BOTTOM+(PDY(J,K)*PY(K,L))                                NBM03800
         PYD(I,J,L) = TOP/BOTTOM                                          NBM03810
50    CONTINUE                                                            NBM03820
      RETURN                                                              NBM03830
      END                                                                 NBM03840
C                                                                         NBM03850
```

```
C
      SUBROUTINE DECISI(AREA,II,ISEED,NAREAS,QBEST,NT,NS,CMF,PRICE,
     *                CD,CF,PD,PY,PYD)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER AREA,NAREAS,NTRIPS,NS,QBEST,BEST(20),NT(10)
      DIMENSION PY(3,6),PD(2,6),PYD(3,2,6),CF(6),CD(6),VDEC(20)
      DIMENSION QCATCH(3,6),EVALD(20),TOT(10,10),TAB(30),TOTAL(10)
      DIMENSION PDY(2,3),PYX(3,3,6)
      COMMON/BLOC1/PYX,PDY,QCATCH
      PK=1.0*NT(NS)
      PPRICE=15.0*(10.0*(PK**2.0)-160.0*K+3335.0)/49.0
      NDEC=(NAREAS*2)+1
      VDEC(1)=0.0
      DO 405 L=2,NDEC-1,2
       SUM=0.0
       CO 410 I=1,3
410     SUM = SUM+(PY(I,(L/2))*QCATCH(I,(L/2)))
       VDEC(L)=PRICE*SUM-CF(L/2)
       DO 415 I=1,2
        EVALD(I)=0.0
        DO 420 J=1,3
420      EVALD(I)=EVALD(I)+(PYD(J,I,L/2)*QCATCH(J,L/2))
415    CONTINUE
       DO 425 I=1,2
        TOT(I,1)=PRICE*EVALD(I)-CF(L/2)
        TOT(I,2)=PRICE*EVALD(I)-CMF
425    CONTINUE
       DO 430 I=1,2
        IF(TOT(I,1) .GE. TOT(I,2)) THEN
         TOTAL(I)=TOT(I,1)
        ELSE
         TOTAL(I)=TOT(I,2)
        ENDIF
430    CONTINUE
       SUM1=0.0
       DO 435 I=1,2
435     SUM1= SUM1+(TOTAL(I)*PD(I,(L/2)))
       VDEC(L+1)=SUM1- CD((L+1)/2)
405   CONTINUE
C
      DO 440 I=1,NDEC
440    TAB(I)=VDEC(I)
      IF (TAB(1).LT.TAB(2)) THEN
       TEMP=TAB(1)
       TAB(1)=TAB(2)
       TAB(2)=TEMP
      ENDIF
C
      DO 445 I=3,NDEC
       DO 450 J=1,I-1
        IF (TAB(I) .GT. TAB(J)) THEN
         TEMP = TAB(I)
         DO 455 K=J+1, I
455       TAB(I-K+J+1) = TAB(I-K+J)
         TAB(J)=TEMP
```

```
         ENDIF
 450    CONTINUE
 445    CONTINUE
C
        II=0
        DO 460 J=1,NDEC
         IF (TAB(1).EQ.VDEC(J)) THEN
          II = II + 1
C         WRITE(8,480) J,VDEC(J)
          BEST(II) = J
         ENDIF
 460    CONTINUE
        IF (II.GT.1) THEN
         CALL RAND(ISEED,RNI)
         CO 465 I=1,II
          IF (RNI.LE. (1.0*I/II)) THEN
           AREA = BEST(I)
           GOTO 470
          ENDIF
 465     CONTINUE
        ELSE
         AREA = BEST(II)
 470    ENDIF
C       WRITE(8,485) AREA
        QBEST=MOD(AREA,2)
        IF (AREA.EQ.1) THEN
         AREA=0
C        WRITE(8,490)
        ELSE
         IF ( QBEST.EQ.1) THEN
          AREA=(AREA-1)/2
C         WRITE(8,493) AREA
         ELSE
          AREA=AREA/2
C         WRITE(8,496) AREA
C         WRITE(6,*)' AREA=',AREA
         ENDIF
        ENDIF
C480    FORMAT(1X,'OPTIMAL DECISION NO.',I3,' WILL YIELD AN EXPECTED',
C      *' REVENUE OF ',F8.2,' DOLLARS',//)
C485    FORMAT(1X,'OPTIMAL DECISION NO.',I3,' IS APPLIED',//)
C490    FORMAT(1X,'WHICH MEANS --------->  STAY AT PORT',//)
C493    FORMAT(1X,'WHICH MEANS ---------> TEST DRAG AREA ',I2,//)
C496    FORMAT(1X,'WHICH MEANS ---------> FISH IN AREA ',I2,//)
        RETURN
        END
C
C
        SUBROUTINE FISHIN(CATCH,AREA,CATCHN,IBOAT,IR,IREAL,ISEED,NS,
     *                    QBEST,REFER,NT,CMF,PRICE,CD,CF,PRCDEV)
        IMPLICIT REAL *8(A-H,M-Z)
        INTEGER AREA,CATCHN,ISEED,NAREAS,NS,NT(10),NTRIPS,NSEASN
        INTEGER QBEST,REFER
        CHARACTER*8 CATCH(3)
        DIMENSION CD(6),CF(6),PYX(3,3,6),QCATCH(3,6),PRCDEV(50)
```

```
      DIMENSION TCATCH(5,15,50),TCOST(5,15,50),VALAND(5,15,50)
      DIMENSION TTCATC(5,15,50),TTCOST(5,15,50),VVALAN(5,15,50)
      DIMENSION NINC(5,15,50),NNINC(5,15,50)
      COMMON/BLOC1/PYX,PDY,QCATCH
      COMMON/BLOC3/TCATCH,TCOST,VALAND,NINC
      COMMON/BLOC4/TTCATC,TTCOST,VVALAN,NNINC
      CDRAG=0.0
      PK=1.0*NT(NS)
      PPRICE=15.0*(10.0*(PK**2.0)-160.0*K+3335.0)/49.0
      CALL RAND(ISEED,Y)
      IF(Y.LE.PYX(1,REFER,AREA)) THEN
       CATCHN=1
      ELSE
       IF(Y.LE.(PYX(1,REFER,AREA)+PYX(2,REFER,AREA))) THEN
        CATCHN=2
       ELSE
        CATCHN=3
       ENDIF
      ENDIF
      IF (AREA .GT. 0) THEN
       YY=QCATCH(CATCHN,AREA)
      ELSE
       CALL RAND(ISEED,A)
       AREA=INT(1.0+A*6.0)
       WRITE(6,*)' AREA=0 ===> AREA= ',I2
C      WRITE(7,585)
       GOTO 590
      ENDIF
      TCATCH(NS,NT(NS),IBOAT)=YY
      TTCATC(NS,NT(NS),IBOAT)=TTCATC(NS,NT(NS),IBOAT)+
     *                        (TCATCH(NS,NT(NS),IBOAT)/IREAL)
      IF(QBEST.EQ.1) THEN
       CDRAG=CD(AREA)
      ENDIF
      TCOST(NS,NT(NS),IBOAT)=(CF(AREA)+CDRAG)*PRCDEV(IBOAT)
      TTCOST(NS,NT(NS),IBOAT)=TTCOST(NS,NT(NS),IBOAT)+
     *                        (TCOST(NS,NT(NS),IBOAT)/IREAL)
      NINC(NS,NT(NS),IBOAT)=YY*PPRICE-(CF(AREA)+CDRAG)*PRCDEV(IBOAT)
      NNINC(NS,NT(NS),IBOAT)=NNINC(NS,NT(NS),IBOAT)+
     *                        (NINC(NS,NT(NS),IBOAT)/IREAL)
      VALAND(NS,NT(NS),IBOAT)=YY*PPRICE
      VVALAN(NS,NT(NS),IBOAT)=VVALAN(NS,NT(NS),IBOAT)+
     *                        (VALAND(NS,NT(NS),IBOAT)/IREAL)
      APRFT=(YY*PPRICE)-CF(AREA)-CD(AREA)
C     WRITE(7,580) IBOAT,AREA,YY,CATCH(CATCHN)
580   FORMAT(1X,'BOAT: ',I2,1X,'AREA: ',I2,1X,'CATCH: ',F6.3,1X,
     *      '---->',A8)
585   FORMAT(1X,'STAYING AT PORT -----> REVENUE = 0.00'/)
590   RETURN
      END
C
C
      SUBROUTINE UPDPI(PI,CATCHN,AREA)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER CATCHN,AREA
```

```
      DIMENSION PI(3),PDY(2,3),QCATCH(3,6),PYX(3,3,6)
      COMMON/BLOC1/PYX,PDY,QCATCH
      BOTTOM=0.0
      DO 600 I=1,3
600    BOTTOM=BOTTOM+(PI(I)*PYX(CATCHN,I,AREA))
      DO 610 I=1,3
610    PI(I)=PI(I)*PYX(CATCHN,I,AREA)/BOTTOM
C     WRITE(7,680) ' NEW PI: ',(PI(I),I=1,3)
680   FORMAT('+',48X,A10,3(F6.3,1X))
      RETURN
      END
C
C
       SUBROUTINE UPDST(H,P,PIB,RT,RTM1,XT,XTM1,ACTUAL,ISEED,NBOATS,
     *              REFER)
       IMPLICIT REAL *8(A-H,M-Z)
       INTEGER ISEED,NBOATS,REFER
       DOUBLE PRECISION X,Y
       CHARACTER*8 ACTUAL
       DIMENSION P(3,3),PIB(50,3)
       DATA GL/125.0/,AL/75.0/,R/0.95776147/,M/0.1/,ALPHA/0.4/
       DATA A0/0.699/,A1/-0.698/,A2/0.236/,A3/-0.029/
       DISC=SQRT((R*(ALPHA-1.0)-1.0)**2-4.0*R*(1.0-ALPHA))
C      COMPUTATION OF "F ZERO POINT ONE"
       MM=M/(-LOG(R))
       F2SM=EXP(A0+A1*MM+A2*(MM**2.0)+A3*(MM**3.0))
       F=M*F2SM
C      END OF COMPUTATION OF F0.1 (NATURAL MORT. SET AT 0.1... COULD
C      BE SET HIGHER.
       F=F+0.7
       I=1
99     WRITE(8,2)
2      FORMAT('1','SEASON UPDATING ( DERISO"S MODEL)',//,1X,
     *'POOR -------> (0,75) ...... XT =  50',/,1X,
     *'AVERAGE ----> (75,125) .... XT = 100',/,1X,
     *'GOOD -------> (125, +) .... XT = 150',//)
       CALL REFERE(XT,REFER,ACTUAL)
       WRITE(8,3) ACTUAL, XT
3      FORMAT(1X,'ACTUAL STATE : ', A8,'  ( XT = ',F5.1,' )',/)
       WRITE(8,4) ACTUAL
4      FORMAT(1X,'P(XTP1 = NEWSTATE : XT = ',A8,' ) = ',/,
     *20X,'- NEW STATE -',/,
     *10X,'  GOOD  ',2X,'AVERAGE',2X,'  POOR  ',/,
     *10X,8('-'),2X,8('-'),2X,8('-'),/)
       PII=4.0*ATAN(1.0)
       SS=EXP(-F-M)
       S=1.0-(F/(F+M))*(1.0-EXP(-F-M))
       H=F*(1.0-SS)/(F+M)
C      THE FOLLOWING BOUNDS ARE FOR THE DERISO MODEL
C      AA=((GL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1 )/(ALPHA*XT))
       IF(AA.LE.0.0) WRITE(6,*) I
       AA=((AL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1 )/(ALPHA*XT))
       IF(AA.LE.0.0) WRITE(6,*) I
       RHSL=-LOG((GL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1 )/
     *(ALPHA*XT))
```

```
      RHSR=-LOG((AL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/
     *(ALPHA*XT))
      X=RHSL
      CALL MDNORD(X,Y)
C     P(1)=L'INTEGRALE DE -L'INFINI A RHSL DE F(X) - DENSITE NORMALE
      P1=Y
      X=RHSR
C     P(2)=L'INTEGRALE DE RHSL A RHSR DE F(X) - DENSITE NORMALE
      CALL MDNORD(X,Y)
      P2=Y
      P2=P2-P1
C     P(3)=L'INTEGRALE DE RHSR A L'INFINI DE F(X) - DENSITE NORMALE
      P3=1.0-P1-P2
C     WRITE(8,5) P1,P2,P3
C5    FORMAT(11X,3(F6.4,4X),///)
C ****GENERATE RECRUITMENT ERROR (NORMAL, MU=0, SIGMA=1)
70    CALL RAND(ISEED,U1)
      U1=4.0*U1-2.0
      FU1=(1.0/SQRT(2.0*PII))*EXP(-(U1**2)/2.0)
      CALL RAND(ISEED,U2)
      U2=U2*(1.0/SQRT(2.0*PII))
      IF(U2.LE.FU1) THEN
C      VARIABILITY IS MODIFIED HERE ******************
       VAR=0.4
       ERROR=VAR*U1
      ELSE
       GOTO 70
      ENDIF
C     WRITE(8,7) ERROR
      RT=ALPHA*XT*EXP(-ERROR)
      XTP1=(1.0+R)*S*XT-R*S**2.0*XTM1+RT-R*S*RTM1
      RTM1=RT
      XTM1=XT
      XT=XTP1
      WRITE(6,6) XT
6     FORMAT(1X,'NEW STATE VALUE .....: ',F6.1)
C7    FORMAT(1X,'ASSOCIATED EPSILON ..: ',F9.6,/)
C     UPDATE TRANSITION MATRIX
      IF(ACTUAL.EQ.'GOOD')THEN
       P(1,1)=P1
       P(1,2)=P2
       P(1,3)=P3
      ELSE
       IF(ACTUAL.EQ.'AVERAGE')THEN
        P(2,1)=P1
        P(2,2)=P2
        P(2,3)=P3
       ELSE
        P(3,1)=P1
        P(3,2)=P2
        P(3,3)=P3
       ENDIF
      ENDIF
C
C     UPDATE PI-VECTOR FOR NEXT SEASON
```

```
      DO 10 IBOAT=1,NBOATS
       BOTTOM=0.0
       DO 20 I=1,3
        SUMP=0.0
        DO 30 J=1,3
30       SUMP=SUMP+P(I,J)
        BOTTOM=BOTTOM+(PIB(IBOAT,I)*SUMP)
20     CONTINUE
       DO 40 I=1,3
        TOP=0.0
        DO 50 J=1,3
50       TOP=TOP+(P(I,J)*PIB(IBOAT,I))
        PIB(IBOAT,I)=TOP/BOTTOM
40     CONTINUE
10    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,VECTOR)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10),NTRIPS,IMARK(10)
      DIMENSION VECTOR(5,15,50),CLASS(10)
      DO 50 I=1,10
50     IMARK(I)=0
      IINT=NT(NS)
      IF(IFLAG.EQ.1) THEN
       IINT=NTRIPS
      ENDIF
      CALL MAXMIN(VECTOR,IR,NBOATS,IINT,NS,MAX,MIN)
      CALL MEANC(VECTOR,IR,NBOATS,IINT,NS,MEAN)
      CALL VARNCE(VECTOR,IR,NBOATS,IINT,NS,MEAN,VAR)
      CALL STANDV(VECTOR,IR,NBOATS,IINT,NS,MEAN,STD)
      WRITE(4,2) MAX,MIN,MEAN,VAR,STD
2     FORMAT('0','STATISTICS ON THE ACCUMULATED ABOVE VECTOR:',/,
     *         15X,'MAXIMUM VALUE = ',F13.2,/,
     *         15X,'MINIMUM VALUE = ',F13.2,/,
     *         15X,'AVERAGE VALUE = ',F13.2,/,
     *         15X,'VARIANCE      = ',F13.2,/,
     *         15X,'STD DEVIATION = ',F13.2,/)
      DO 10 K=1,10
10     CLASS(K)=MIN+K*((MAX-MIN)/10.0)
      DO 20 I=1,NBOATS
       SUM=0.0
       DO 60 II=1,IINT
60      SUM=SUM+VECTOR(NS,II,I)
       DO 30 K=1,10
        IF(SUM.LT.CLASS(K)) THEN
         IMARK(K)=IMARK(K)+1
         GOTO 20
        ENDIF
30     CONTINUE
20    CONTINUE
      IMARK(10)=IMARK(10)+1
C ***** WRITE THE HISTOGRAM *******
```

```
      WRITE(4,*) '                                          HISTOGRAM'      NBM07160
      WRITE(4,*) ' CLASS MARK'                                              NBM07170
      DO 40 K=1,10                                                          NBM07180
       WRITE(4,1) K,CLASS(K),IMARK(K)                                       NBM07190
40    CONTINUE                                                              NBM07200
1     FORMAT(1X,I2,'( ',F8.2,' )',',...   ',I3)                             NBM07210
      RETURN                                                                NBM07220
      END                                                                   NBM07230
C                                                                           NBM07240
C                                                                           NBM07250
      SUBROUTINE STAT2(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,            NBM07260
     *                 VVALAN)                                              NBM07270
      IMPLICIT REAL *8(A-H,M-Z)                                             NBM07280
      INTEGER NBOATS,NSEASN,NTRIPS,NT(10)                                   NBM07290
      DIMENSION NNINC(5,15,50),TTCATC(5,15,50),TTCOST(5,15,50)              NBM07300
      DIMENSION VVALAN(5,15,50)                                             NBM07310
      CALL MAXM2(NNINC,NBOATS,NSEASN,NTRIPS,MAX,MIN)                        NBM07320
      WRITE(4,2) 'NET INCOME',MAX,MIN                                       NBM07330
      CALL ST2ANX(NNINC,NBOATS,NSEASN,NTRIPS,MAX,MIN)                       NBM07340
      CALL MAXM2(TTCATC,NBOATS,NSEASN,NTRIPS,MAX,MIN)                       NBM07350
      WRITE(4,2) 'TOTAL CATCH',MAX,MIN                                      NBM07360
      CALL ST2ANX(TTCATC,NBOATS,NSEASN,NTRIPS,MAX,MIN)                      NBM07370
      CALL MAXM2(TTCOST,NBOATS,NSEASN,NTRIPS,MAX,MIN)                       NBM07380
      WRITE(4,2) 'TOTAL COSTS',MAX,MIN                                      NBM07390
      CALL ST2ANX(TTCOST,NBOATS,NSEASN,NTRIPS,MAX,MIN)                      NBM07400
      CALL MAXM2(VVALAN,NBOATS,NSEASN,NTRIPS,MAX,MIN)                       NBM07410
      WRITE(4,2) 'LANDED VALUE',MAX,MIN                                     NBM07420
      CALL ST2ANX(VVALAN,NBOATS,NSEASN,NTRIPS,MAX,MIN)                      NBM07430
2     FORMAT('1','STATISTICS ON ACCUMULATED ',A15,/,                        NBM07440
     *         15X,'MAXIMUM VALUE = ',F13.2,/,                              NBM07450
     *         15X,'MINIMUM VALUE = ',F13.2,/)                              NBM07460
      RETURN                                                                NBM07470
      END                                                                   NBM07480
C                                                                           NBM07490
C                                                                           NBM07500
      SUBROUTINE ST2ANX(VECTOR,NBOATS,NSEASN,NTRIPS,MAX,MIN)                NBM07510
      IMPLICIT REAL *8(A-H,M-Z)                                             NBM07520
      INTEGER NBOATS,NSEASN,NTRIPS,NT(10),IIMARK(10,15),IAXIS(15)           NBM07530
      DIMENSION VECTOR(5,15,50),CLASS(10)                                   NBM07540
      DO 5 I=1,10                                                           NBM07550
       DO 5 II=1,15                                                         NBM07560
5       IIMARK(I,II)=0                                                      NBM07570
      DO 10 K=1,10                                                          NBM07580
10     CLASS(K)=MIN+K*((MAX-MIN)/10.0)                                      NBM07590
       DO 20 I=1,NTRIPS                                                     NBM07600
        DO 30 II=1,NBOATS                                                   NBM07610
         SUM=0.0                                                            NBM07620
         DO 40 III=1,NSEASN                                                 NBM07630
C         IF(I.GT.NT(III)) GO TO 40                                         NBM07640
          SUM=SUM+VECTOR(III,I,II)                                          NBM07650
40       CONTINUE                                                           NBM07660
         DO 50 K=1,10                                                       NBM07670
          IF(SUM.LT.CLASS(K)) THEN                                          NBM07680
           IIMARK(K,I)=IIMARK(K,I)+1                                        NBM07690
           GO TO 30                                                         NBM07700
```

```
                                                                          NBM07710
         ENDIF                                                            NBM07720
50        CONTINUE                                                        NBM07730
30       CONTINUE                                                         NBM07740
20      CONTINUE                                                          NBM07750
      DO 60  J=1,15                                                       NBM07760
C       IIMARK(10,J)=IIMARK(10,J)+1                                       NBM07770
        IAXIS(J)=J                                                        NBM07780
60    CONTINUE                                                            NBM07790
C ***** WRITE THE HISTOGRAM *******                                       NBM07800
      WRITE(4,*) '                                    HISTOGRAM'          NBM07810
      WRITE(4,*) ' CLASS MARK'                                            NBM07820
      WRITE(4,2)(CLASS(K),(IIMARK(K,I),I=1,15),K=1,10)                    NBM07830
      WRITE(4,3)'TRIPS' ,(IAXIS(I),I=1,15)                                NBM07840
2     FORMAT(/,10(F8.2,5X,15(I3,2X),//))                                  NBM07850
3     FORMAT(/,A8,4X,15(I3,2X))                                           NBM07860
      RETURN                                                              NBM07870
      END                                                                 NBM07880
C                                                                         NBM07890
C                                                                         NBM07900
      SUBROUTINE MAXMIN(VECTOR,IR,NBOATS,IINT,NS,MAX,MIN)                 NBM07910
      IMPLICIT REAL *8(A-H,M-Z)                                           NBM07920
      INTEGER NBOATS,NS,NT(10)                                            NBM07930
      DIMENSION SUM(50),VECTOR(5,15,50)                                   NBM07940
      MAX=0.0                                                             NBM07950
      MIN=10000000.0                                                      NBM07960
      DO 5 I=1,NBOATS                                                     NBM07970
5       SUM(I)=0.0                                                        NBM07980
      DO 10  I=1,NBOATS                                                   NBM07990
        DO 15  II=1,IINT                                                  NBM08000
15        SUM(I)=SUM(I)+VECTOR(NS,II,I)                                   NBM08010
        IF(SUM(I).GE.MAX) THEN                                            NBM08020
          MAX=SUM(I)                                                      NBM08030
        ENDIF                                                             NBM08040
10    CONTINUE                                                            NBM08050
      DO 20  I=1,NBOATS                                                   NBM08060
        IF(SUM(I).LE.MIN) THEN                                            NBM08070
          MIN=SUM(I)                                                      NBM08080
        ENDIF                                                             NBM08090
20    CONTINUE                                                            NBM08100
      RETURN                                                              NBM08110
      END                                                                 NBM08120
C                                                                         NBM08130
C                                                                         NBM08140
      SUBROUTINE MAXM2(VECTOR,NBOATS,NSEASN,NTRIPS,MAX,MIN)               NBM08150
      IMPLICIT REAL *8(A-H,M-Z)                                           NBM08160
      INTEGER NBOATS,NSEASN,NTRIPS                                        NBM08170
      DIMENSION SUM(15,50),VECTOR(5,15,50)                                NBM08180
      MAX=0.0                                                             NBM08190
      MIN=10000000.0                                                      NBM08200
      DO 5 I=1,NBOATS                                                     NBM08210
        DO 5  II=1,15                                                     NBM08220
5         SUM(II,I)=0.0                                                   NBM08230
      DO 10  III=1,NSEASN                                                 NBM08240
        DO 10  I=1,NBOATS                                                 NBM08250
          DO 10  II=1,NTRIPS
```

```
10        SUM(II,I)=SUM(II,I)+VECTOR(III,II,I)
      DO 20 III=1,NSEASN
       DO 20 I=1,NBOATS
        DO 20 II=1,NTRIPS
         IF(SUM(II,I).GE.MAX) THEN
          MAX=SUM(II,I)
         ENDIF
20    CONTINUE
      DO 30 III=1,NSEASN
       DO 30 I=1,NBOATS
        DO 30 II=1,NTRIPS
         IF(SUM(II,I).LE.MIN) THEN
          MIN=SUM(II,I)
         ENDIF
30    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE MEANC(VECTOR,IR,NBOATS,IINT,NS,MEAN)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10)
      DIMENSION VECTOR(5,15,50)
      SUM=0.0
      DO 10 I=1,NBOATS
       DO 10 II=1,IINT
10      SUM=SUM+VECTOR(NS,II,I)
      MEAN=SUM/NBOATS
      RETURN
      END
C
C
      SUBROUTINE VARNCE(VECTOR,IR,NBOATS,IINT,NS,MEAN,VAR)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10)
      DIMENSION SUM(50),VECTOR(5,15,50)
      DO 5 I=1,NBOATS
5      SUM(I)=0.0
      SUM1=0.0
      DO 10 I=1,NBOATS
       DO 10 II=1,IINT
10      SUM(I)=SUM(I)+VECTOR(NS,II,I)
      DO 20 I=1,NBOATS
20     SUM1=SUM1+(ABS(SUM(I)-MEAN)**2.0)
      VAR=SUM1/NBOATS
      RETURN
      END
C
C
      SUBROUTINE STANDV(VECTOR,IR,NBOATS,IINT,NS,MEAN,STD)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10)
      DIMENSION SUM(50),VECTOR(5,15,50)
      DO 5 I=1,NBOATS
5      SUM(I)=0.0
```

```
      SUM1=0.0
      DO 10  I=1,NBOATS
       DO 10  II=1,IINT
10      SUM(I)=SUM(I)+VECTOR(NS,II,I)
      DO 20  I=1,NBOATS
20     SUM1=SUM1+(ABS(SUM(I)-MEAN)**2.0)
      IF(NBOATS.EQ.1)THEN
       STD=0.0
       GO TO 30
      ENDIF
      STD=(SUM1/((NBOATS*1.0)-1.0))**0.5
30    RETURN
      END
C
C
      SUBROUTINE DEVIA(NBOATS,NS,NT,PRCDEV,TCOST,VALAND,NINC)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10)
      DIMENSION TCOST(5,15,50),VALAND(5,15,50),PRCDEV(50)
      DIMENSION NINC(5,15,50)
      SUM1=0.0
      DO 10  I=1,NBOATS
       DO 10  II=1,NT(NS)
10      SUM1=SUM1+NINC(NS,II,I)
      AVERA=SUM1/NBOATS
      DO 20  I=1,NBOATS
       SUM2=0.0
       DO 30  II=1,NT(NS)
30      SUM2=SUM2+NINC(NS,II,I)
       PRCDEV(I)=((SUM2-AVERA)/AVERA)+1.0
20    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE CHECK(IR,ISTOP,NBOATS,NS,NT,H,VECTOR,XT)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10)
      DIMENSION VECTOR(5,15,50)
      SUM=0.0
      DO 10  I=1,NBOATS
       DO 10  II=1,NT(NS)
10      SUM=SUM+VECTOR(NS,II,I)
      ACTHAR=SUM/XT
      IF(ACTHAR.GT.H) THEN
       ISTOP=1
       WRITE(6,2)  SUM,XT,ACTHAR,H,NT(NS)
C      WRITE(8,2)  SUM,XT,ACTHAR,H,NT(NS)
C     ENDIF
2     FORMAT(1X,60('*'),/,1X,'ACTUAL HARVEST EXCEEDS QUOTA:...',/,
     *        1X,'CATCH= ',F9.4,3X,'STATE= ',F9.4,/,
     *        1X,'CATCH/STATE ABUN. = ',F7.4,' H = ',F7.4,/,
     *        1X,'TOTAL NUMBER OF TRIPS THIS SEASON: ',I3,/,
     *        1X,60('*'))
      RETURN
```

```
      END
C
C
      SUBROUTINE HCCMP(H)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER ISEED,NBOATS,REFER
      DATA R/0.95776147/,M/0.1/
      DATA A0/0.699/,A1/-0.698/,A2/0.236/,A3/-0.029/
C     COMPUTATION OF "F ZERO POINT ONE"
      MM=M/(-LOG(R))
      FZSM=EXP(A0+A1*MM+A2*(MM**2.0)+A3*(MM**3.0))
      F=M*FZSM
      WRITE(6,*) ' F= ',F
C     END OF COMPUTATION OF F0.1 (NATURAL MORT. SET AT 0.1... COULD
C     BE SET HIGHER.
      F=F+0.7
      S=EXP(-F-M)
      H=F*(1.0-S)/(F+M)
      RETURN
      END
C
C
      SUBROUTINE STAT3(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,
     *                 VVALAN)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NSEASN,NTRIPS,NT(10)
      DIMENSION NNINC(5,15,50),TTCATC(5,15,50),TTCOST(5,15,50)
      DIMENSION VVALAN(5,15,50),AVER(4)
      DIMENSION SUM1(5),SUM2(5),SUM3(5),SUM4(5)
      AVR1=0.0
      AVR2=0.0
      AVR3=0.0
      AVR4=0.0
      DO 5 IS=1,5
        DO 10 IT=1,15
         DO 10 IB=1,50
          AVR1=AVR1+NNINC(IS,IT,IB)/5.0
          AVR2=AVR2+TTCATC(IS,IT,IB)/5.0
          AVR3=AVR3+TTCOST(IS,IT,IB)/5.0
          AVR4=AVR4+VVALAN(IS,IT,IB)/5.0
10      CONTINUE
        SUM1(IS)=0.0
        SUM2(IS)=0.0
        SUM3(IS)=0.0
        SUM4(IS)=0.0
5     CONTINUE
      WRITE(3,80) AVR1,AVR2,AVR3,AVR4
80    FORMAT('1','GLOBAL AVERAGE OF PERFORMANCE INDICATORS',/,
     *       1X,'NET INCOME',5X,'TOTAL CATCH',4X,'TOTAL COST',5X,
     *       'LANDED VALUE',/,1X,4(F13.5,2X),/,1X,60('='))
        WRITE(3,83) IS
83      FORMAT('1','PERFORMANCE INDICATORS',/,
     *         1X,'SN',3X,'NET INCOME',5X,'TOTAL CATCH',4X,
     *         'TOTAL COST',5X,'LANDED VALUE')
      DO 20 IS=1,5
```

```
      DO 30 IT=1,15
      DO 30 IB=1,50
       SUM1(IS)=SUM1(IS)+NNINC(IS,IT,IB)
       SUM2(IS)=SUM2(IS)+TTCATC(IS,IT,IB)
       SUM3(IS)=SUM3(IS)+TTCOST(IS,IT,IB)
       SUM4(IS)=SUM4(IS)+VVALAN(IS,IT,IB)
30    CONTINUE
C     SUM1(IS)=SUM1(IS)/NSEASN
C     SUM2(IS)=SUM2(IS)/NSEASN
C     SUM3(IS)=SUM3(IS)/NSEASN
C     SUM4(IS)=SUM4(IS)/NSEASN
      WRITE(3,86) IS,SUM1(IS),SUM2(IS),SUM3(IS),SUM4(IS)
86    FORMAT(1X,I2,3X,4(F13.5,2X))
20   CONTINUE
     RETURN
     END
```

TRIP NO. 1 OF SEASON 3

CATCH DISTRIBUTION -----> P(Y)

AREA 1			AREA 2			AREA 3			AREA 4	
G	A	P	G	A	P	G	A	P	G	A
0.31	0.29	0.31	0.30	0.31	0.31	0.35	0.39	0.37	0.35	0.37

DRAG DISTRIBUTION -----> P(D)

AREA 1		AREA 2		AREA 3		AREA 4
FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.
0.381	0.379	0.381	0.380	0.381	0.381	0.268

OPTIMAL DECISION NO. 11 WILL YIELD AN EXPECTED REVENUE OF ******* DO

OPTIMAL DECISION NO. 11 IS APPLIED

WHICH MEANS --------> TEST DRAG AREA 5

ACTUAL CATCH YIELDS Y= 0.1 -----> POOR
FOR A REVENUE OF 49.95 DOLLARS
ACCUMULATED REVENUE UPDATED AT 990.60 DOLLARS

AFTER THIS TRIP, THE PI-VECTOR IS UPDATED......

G	A	P
0.204	0.470	0.326

AREA 4			AREA 5			AREA 6		
G	A	P	G	A	P	G	A	P
0.35	0.37	0.37	0.34	0.33	0.31	0.35	0.31	0.31

AREA 4		AREA 5		AREA 6	
FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNFAV.
0.268	0.269	0.269	0.267	0.269	0.269

******* DOLLARS

===========

===========
===========

TRIP NO. 2 OF SEASON 3

CATCH DISTRIBUTION -----> P(Y)

AREA 1			AREA 2			AREA 3			AREA 4		
G	A	P	G	A	P	G	A	P	G	A	P
0.31	0.27	0.30	0.30	0.30	0.30	0.35	0.39	0.37	0.35	0.37	0.37

DRAG DISTRIBUTION -----> P(D)

AREA 1		AREA 2		AREA 3		AREA 4	
FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNFAV
0.381	0.377	0.380	0.380	0.380	0.380	0.267	0.269

OPTIMAL DECISION NO. 11 WILL YIELD AN EXPECTED REVENUE OF ******* DOLLAR

OPTIMAL DECISION NO. 11 IS APPLIED

WHICH MEANS --------> TEST DRAG AREA 5

ACTUAL CATCH YIELDS Y= 0.1 -----> GOOD
FOR A REVENUE OF 31.96 DOLLARS
ACCUMULATED REVENUE UPDATED AT 1022.56 DOLLARS

==

AFTER THIS TRIP, THE PI-VECTOR IS UPDATED......

G	A	P
0.268	0.464	0.268

==
==

AREA 4			AREA 5			AREA 6		
G	A	P	G	A	P	G	A	P
.35	0.37	0.37	0.35	0.34	0.32	0.35	0.32	0.32

AREA 4		AREA 5		AREA 6	
FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNFAV.
0.267	0.269	0.269	0.267	0.269	0.269

****** DOLLARS

==========

===========
===========

TRIP NO. 3 OF SEASON 3

ATCH DISTRIBUTION -----> P(Y)

AREA 1			AREA 2			AREA 3			AREA 4		
G	A	P	G	A	P	G	A	P	G	A	P
0.31	0.29	0.31	0.30	0.31	0.31	0.35	0.38	0.37	0.35	0.37	0..

RAG DISTRIBUTION -----> P(D)

AREA 1		AREA 2		AREA 3		AREA 4	
FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNF/
0.381	0.379	0.381	0.380	0.381	0.381	0.268	0.26

PTIMAL DECISION NO. 11 WILL YIELD AN EXPECTED REVENUE OF

PTIMAL DECISION NO. 11 IS APPLIED

HICH MEANS --------> TEST DRAG AREA 5

CTUAL CATCH YIELDS Y= -----> AVERAGE
OR A REVENUE OF 16.95 DOLLARS
CCUMULATED REVENUE UPDATED AT 1039.51 DOLLARS

===

TER THIS TRIP, THE PI-VECTOR IS UPDATED......

G	A	P
).251	0.497	0.251

===
===

AREA 4			AREA 5			AREA 6		
G	A	P	G	A	P	G	A	P
.35	0.37	0.37	0.34	0.33	0.31	0.35	0.31	0.31

AREA 4		AREA 5		AREA 6	
FAVOR.	UNFAV.	FAVOR.	UNFAV.	FAVOR.	UNFAV.
0.268	0.269	0.269	0.267	0.269	0.269

:=========

:=========
:=========

Figure C.2: Myopic Algorithm: Fishing Simulation

```
STATISTICS ON ****        OVERALL TOTAL CATCH **** SEASON NO.    5

STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
                    MAXIMUM VALUE =           1.02
                    MINIMUM VALUE =           0.45
                    AVERAGE VALUE =           0.74
                    VARIANCE      =           0.01
                    STD DEVIATION =           0.12

                                      HISTOGRAM
 CLASS  MARK
 1(       0.51 ),....      1
 2(       0.57 ),....      2
 3(       0.62 ),....      7
 4(       0.68 ),....      6
 5(       0.74 ),....      6
 6(       0.79 ),....     11
 7(       0.85 ),....      6
 8(       0.91 ),....      5
 9(       0.96 ),....      3
10(       1.02 ),....      1


STATISTICS ON ****        OVERALL TOTAL COSTS **** SEASON NO.    5

STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
                    MAXIMUM VALUE =         189.81
                    MINIMUM VALUE =          74.83
                    AVERAGE VALUE =         124.80
                    VARIANCE      =         557.50
                    STD DEVIATION =          23.85

                                      HISTOGRAM
 CLASS  MARK
 1(      86.33 ),....      2
 2(      97.83 ),....      3
 3(     109.33 ),....      6
 4(     120.82 ),....     15
 5(     132.32 ),....     10
 6(     143.82 ),....      3
 7(     155.31 ),....      6
 8(     166.81 ),....      3
 9(     178.31 ),....      0
10(     189.81 ),....      2
```

```
STATISTICS ON ****          OVERALL NET INCOME **** SEASON NO.    5

STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
                   MAXIMUM VALUE =          947.17
                   MINIMUM VALUE =          387.12
                   AVERAGE VALUE =          649.36
                   VARIANCE      =        15629.88
                   STD DEVIATION =          126.29

                                        HISTOGRAM
 CLASS  MARK
  1(      443.13  ),....     2
  2(      499.13  ),....     6
  3(      555.14  ),....     5
  4(      511.14  ),....     6
  5(      667.15  ),....     7
  6(      723.15  ),....    10
  7(      779.16  ),....     7
  8(      835.16  ),....     2
  9(      891.17  ),....     4
 10(      947.17  ),....     1
```

```
STATISTICS ON ****          OVERALL LANDED VALUE **** SEASON NO.    5

STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
                   MAXIMUM VALUE =         1073.05
                   MINIMUM VALUE =          471.76
                   AVERAGE VALUE =          774.66
                   VARIANCE      =        15933.75
                   STD DEVIATION =          127.51

                                        HISTOGRAM
 CLASS  MARK
  1(      531.89  ),....     1
  2(      592.02  ),....     2
  3(      652.15  ),....     7
  4(      712.28  ),....     6
  5(      772.41  ),....     6
  6(      832.54  ),....    11
  7(      892.67  ),....     8
  8(      952.79  ),....     5
  9(     1012.92  ),....     3
 10(     1073.05  ),....     1
```

STATISTICS ON ACCUMULATED TOTAL CATCH
MAXIMUM VALUE = 1.49
MINIMUM VALUE = 0.00

HISTOGRAM

CLASS MARK															
0.15	0	0	0	0	0	0	37	50	50	50	50	50	50	50	50
0.30	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0
0.45	0	0	0	0	1	7	0	0	0	0	0	0	0	0	0
0.59	0	0	0	3	13	23	0	0	0	0	0	0	0	0	0
0.74	3	3	7	17	22	17	0	0	0	0	0	0	0	0	0
0.89	13	19	18	18	11	3	0	0	0	0	0	0	0	0	0
1.04	23	15	19	10	3	0	0	0	0	0	0	0	0	0	0
1.19	10	10	6	2	0	0	0	0	0	0	0	0	0	0	0
1.34	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0
1.49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

STATISTICS ON ACCUMULATED TOTAL COSTS
MAXIMUM VALUE = 184.62
MINIMUM VALUE = 0.00

HISTOGRAM

CLASS MARK															
18.45	0	0	0	0	0	0	18	50	50	50	50	50	50	50	50
36.92	0	0	0	0	0	0	32	0	0	0	0	0	0	0	0
55.39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
73.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
92.31	0	0	0	0	0	25	0	0	0	0	0	0	0	0	0
110.77	0	0	0	0	24	25	0	0	0	0	0	0	0	0	0
129.23	0	0	0	16	25	0	0	0	0	0	0	0	0	0	0
147.70	4	4	15	30	1	0	0	0	0	0	0	0	0	0	0
166.16	31	31	30	4	0	0	0	0	0	0	0	0	0	0	0
184.62	14	14	5	0	0	0	0	0	0	0	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

STATISTICS ON ACCUMULATED LANDED VALUE

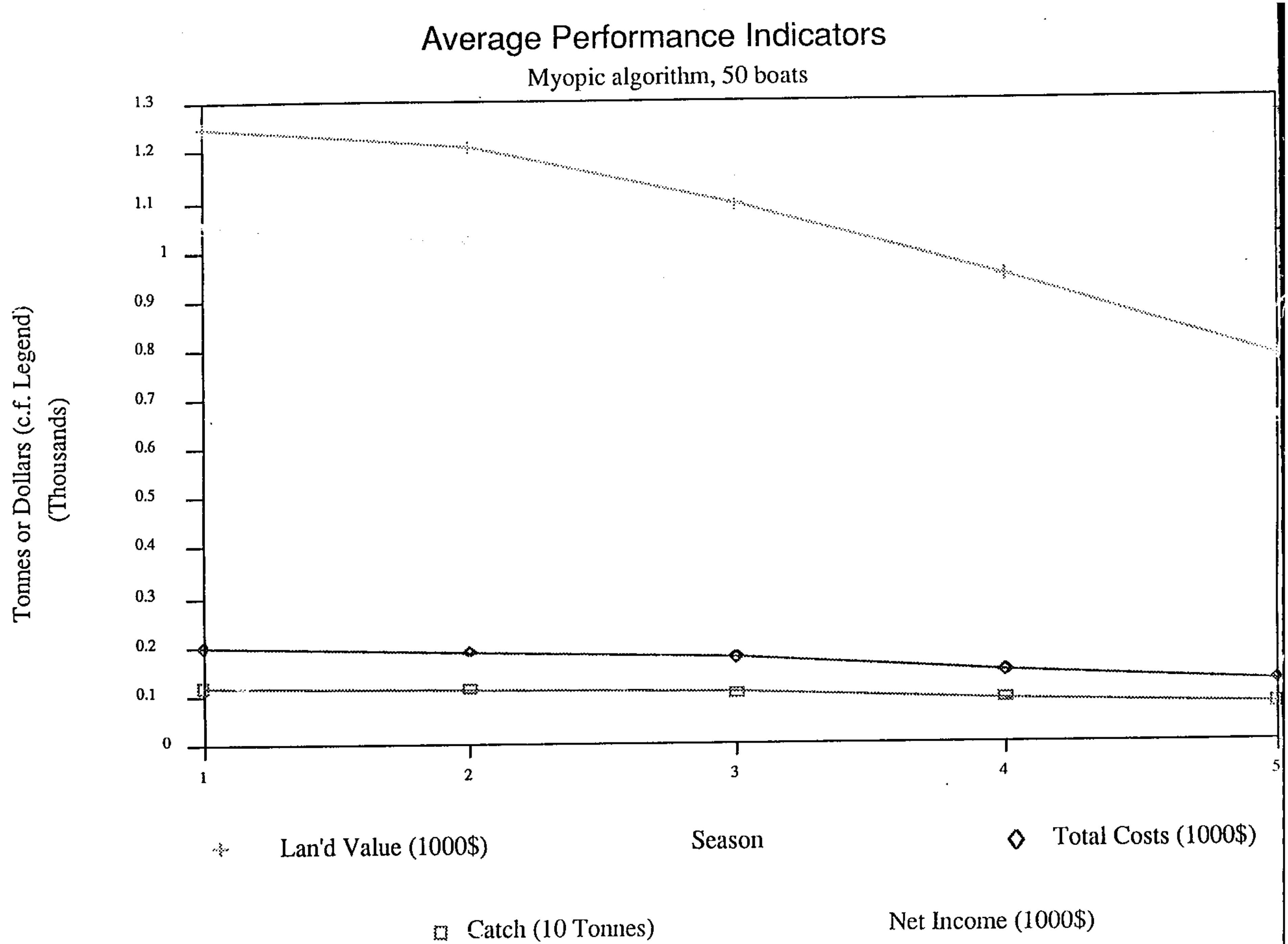
MAXIMUM VALUE = 1522.15

MINIMUM VALUE = 0.00

HISTOGRAM

CLASS MARK															
152.21	0	0	0	0	0	0	32	50	50	50	50	50	50	50	50
304.43	0	0	0	0	0	0	17	0	0	0	0	0	0	0	0
456.64	0	0	0	0	0	4	1	0	0	0	0	0	0	0	0
608.86	0	0	0	2	9	17	0	0	0	0	0	0	0	0	0
761.07	3	3	4	13	16	20	0	0	0	0	0	0	0	0	0
913.29	13	19	17	18	18	8	0	0	0	0	0	0	0	0	0
1065.50	23	15	19	13	6	1	0	0	0	0	0	0	0	0	0
1217.72	10	10	10	3	1	0	0	0	0	0	0	0	0	0	0
1369.93	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0
1522.15	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Figure C.3: Seasonal performance indicators


Average Performance Indicators
Myopic algorithm, 50 boats
Tonnes or Dollars (c.f. Legend)
(Thousands)
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1
2
3
4
5
Season
Lan'd Value (1000$)
Total Costs (1000$)
Catch (10 Tonnes)
Net Income (1000$)

Appendix D

Adaptive Algorithm

This appendix presents the key computer code for the adaptive algorithm (Figure D.1). The code is written in FORTRAN-VS.
Also included are sample outputs for the developed policy, the fishing activity,

the tallied statistics and the performance indicators' graph for a sample run of the simulation model.

Figure D.1: Adaptive Algorithm: Main program

```
      PROGRAM ADAPT
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER AREA,CATCHN,IREAL,ISEED,NAREAS,NBOATS,NS,NSEASN,NTRIPS
      INTEGER REFER,NT(10),JALPHA(15,6)
      CHARACTER*8 CATCH(3)
      CHARACTER*8 ACTUAL
      DIMENSION P(3,3),PI(3),PYX(3,3,6),CF(6),QCATCH(3,6),ZALPHA(15,6,3)
      DIMENSION PIB(50,3),TCATCH(5,15,50),TCOST(5,15,50)
      DIMENSION VALAND(5,15,50),PRODEV(50),NINC(5,15,50)
      DIMENSION TTCATC(5,15,50),TTCOST(5,15,50),NNINC(5,15,50)
      DIMENSION VVALAN(5,15,50)
      COMMON/BLOC2/PYX,QCATCH,CF
      COMMON/BLOC3/TCATCH,TCOST,VALAND,NINC
      COMMON/BLOC4/TTCATC,TTCOST,VVALAN,NNINC
      DATA XTM1/100.0/,RT/40.0/,RTM1/40.0/
      DATA ISEED/6325779/
      CATCH(1)='GOOD    '
      CATCH(2)='AVERAGE '
      CATCH(3)='POOR    '
      CALL LIRE(IREAL,NAREAS,NBOATS,NTRIPS,NSEASN)
C     WRITE(6,*) ' LIRE TERMINE'
      DO 300 I=1,NSEASN
       DO 300 II=1,NTRIPS
        DO 300 III=1,NBOATS
         TTCATC(I,II,III)=0.0
         TTCOST(I,II,III)=0.0
         NNINC(I,II,III)=0.0
         VVALAN(I,II,III)=0.0
300   CONTINUE
      DO 5 IR=1,IREAL
      DO 100 I=1,NBOATS
100    PRODEV(I)=1.0
      DO 200 I=1,3
200    PI(I)=1.0/3.0
      DO 20 J=1,NBOATS
       DO 20 I=1,3
20      PIB(J,I)=PI(I)
      CALL HCOMP(H)
      XT=100.0
      PRICE=600.0
C     CALL TRANS(P,XTM1,RTM1)
C     WRITE(6,*) ' TRANS TERMINE'
C     CALL ECRIR1(X,PI,P,CATCH,PRICE,NAREAS,NBOATS,NTRIPS,NSEASN)
C     WRITE(6,*) ' ECRIR1 TERMINE'
      CALL BACKW(JALPHA,ZALPHA,PRICE,NAREAS,NTRIPS)
C     WRITE(6,*) ' BACKW TERMINE'
      DO 10 NS=1,NSEASN
       ISEED=ISEED+2
       ISTOP=0
       WRITE(6,*) ' DOING SEASON NUMBER....',NS
C
C  LOCATE MATRICES' ROW ASSOCIATED WITH CURRENT STATE
        CALL REFERE(XT,REFER,ACTUAL)
C
        DO 15 IINT=1,NTRIPS
```

```
      NT(NS)=IINT
C     WRITE(6,*) ' DOING TRIP NUMBER....',NT(NS)
C     WRITE(8,500) NT(NS),NS
500   FORMAT('1',30X,'TRIP NO. ',I3,' OF SEASON ',I2,/,25X,37('-'),//)
      DO 30 IBOAT=1,NBOATS
       DO 40 I=1,3
40      PI(I)=PIB(IBOAT,I)
       CALL OPTIM(AREA,IR,ISEED,NAREAS,NS,NT,NTRIPS,JALPHA,PI,ZALPHA)
       CALL FISHIN(REFER,AREA,CATCH,CATCHN,IBOAT,IR,IREAL,ISEED,
     *            NS,NT,PRICE,PRCDEV)
       CALL UPDPI(PI,CATCHN,AREA)
       DO 50 I=1,3
50      PIB(IBOAT,I)=PI(I)
30    CONTINUE
      CALL CHECK(IR,ISTOP,NBOATS,NS,NT,H,TCATCH,XT)
      IF(ISTOP.EQ.1) GO TO 70
15   CONTINUE
70   IF(NT(NS).EQ.16) NT(NS)=15
     WRITE(4,1)'SEASONAL TOTAL CATCH',NS,XT,ACTUAL,IR
     CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,TCATCH)
     WRITE(4,1)'SEASONAL LANDED VALUE',NS,XT,ACTUAL,IR
     CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,VALAND)
     WRITE(4,1)'SEASONAL TOTAL COSTS',NS,XT,ACTUAL,IR
     CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,TCOST)
     WRITE(4,1)'SEASONAL NET INCOME',NS,XT,ACTUAL,IR
     CALL STATS(IFLAG,IR,NBOATS,NS,NT,NTRIPS,NINC)
     CALL DEVIA(NBOATS,NS,NT,PRCDEV,TCOST,VALAND,NINC)
     CALL UPDST(H,P,PIB,RT,RTM1,XT,XTM1,ACTUAL,ISEED,NBOATS,REFER)
10   CONTINUE
5    CONTINUE
C    ****************
     IFLAG=1
     DO 80 NS=1,NSEASN
      WRITE(4,2)'OVERALL TOTAL CATCH',NS
      CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,TTCATC)
      WRITE(4,2)'OVERALL LANDED VALUE',NS
      CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,VVALAN)
      WRITE(4,2)'OVERALL TOTAL COSTS',NS
      CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,TTCOST)
      WRITE(4,2)'OVERALL NET INCOME',NS
      CALL STATS(IFLAG,IREAL,NBOATS,NS,NT,NTRIPS,NNINC)
80   CONTINUE
C    ****************
     CALL STAT2(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,VVALAN)
1    FORMAT('1','STATISTICS ON **** ',A25,' **** SEASON NO. ',I2,
    *  ', CURRENT STATE: X= ',F6.1,' -----> ',A8,/,46X,'****',
    *  'REALIZATION:',I2)
2    FORMAT('1','STATISTICS ON **** ',A25,' **** SEASON NO. ',I2)
     CALL STAT3(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,VVALAN)
     STOP
     END
C
C
      SUBROUTINE TRANS(P,XTM1,RTM1)
      IMPLICIT REAL *8(A-H,M-Z)
```

```
      DIMENSION P(3,3)
      DOUBLE PRECISION XX,Y
      DATA GL/125.0/,AL/75.0/,X/50.0/
      DATA R/0.95776147/,M/0.1/,ALPHA/0.4/
      DISC=SQRT((R*(ALPHA-1.0)-1.0)**2-4.0*R*(1.0-ALPHA))
      F1=-LOG((1.0-R*(ALPHA-1.0)+DISC)/(2.0*R))-M
      F2=-LOG((1.0-R*(ALPHA-1.0)-DISC)/(2.0*R))-M
      IF(F1.GT.0.0.AND.F1.LT.1.0) THEN
       F=F1+0.05
      ELSE
       F=F2+0.05
      ENDIF
      PI=4.0*ATAN(1.0)
      S=EXP(-F-M)
      H=F*(1.0-S)/(F+M)
      I=0
99    I=I+1
C     R=0.0
C     THE FOLLOWING BOUNDS ARE FOR THE DERISO MODEL
      AA=((GL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/(ALPHA*X))
      IF(AA.LE.0.0) WRITE(6,1) X,XTM1,RTM1,AL,R,S,ALPHA,F
1     FORMAT(1X,4(F5.0,1X),4(F5.3,1X))
      AA=((AL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/(ALPHA*X))
      IF(AA.LE.0.0) WRITE(6,1) X,XTM1,RTM1,AL,R,S,ALPHA,F
      RHSL=-LOG((GL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/
     *(ALPHA*X))
      RHSR=-LOG((AL-(1.0+R)*S*X*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1)/
     *(ALPHA*X))
      XX=RHSL
      CALL MDNORD(XX,Y)
C     P(I,1)=L'INTEGRALE DE -L'INFINI A RHSL DE F(X) - DENSITE NORMALE
      P(I,1)=Y
      XX=RHSR
C     P(I,2)=L'INTEGRALE DE RHSL A RHSR DE F(X) - DENSITE NORMALE
      CALL MDNORD(XX,Y)
      P(I,2)=Y
      P(I,2)=P(I,2)-P(I,1)
C     P(I,3)=L'INTEGRALE DE RHSR A L'INFINI DE F(X) - DENSITE NORMALE
      P(I,3)=1.0-P(I,1)-P(I,2)
      IF(I.LT.3) THEN
       X=X+50.0
       GOTO 99
      ENDIF
      RETURN
      END
C
C
      SUBROUTINE LIRE(IREAL,NAREAS,NBOATS,NTRIPS,NSEASN)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER IREAL,NAREAS,NBOATS,NTRIPS,NSEASN
      DIMENSION PI(3),CF(10),PYX(3,3,6),QCATCH(3,6)
      COMMON/BLOC2/PYX,QCATCH,CF
C
C  READ IN GIVEN DATA:
C
```

```
C      PYX(I,J) CATCH GIVEN STATE DISTRIBUTION                           NNS01660
C                                                                        NNS01670
       DO 15 I=1,6                                                       NNS01680
        READ(5,81)                                                       NNS01690
        DO 20 J=1,3                                                      NNS01700
         READ(5,*)(PYX(J,K,I),K=1,3)                                     NNS01710
20     CONTINUE                                                          NNS01720
15     CONTINUE                                                          NNS01730
C                                                                        NNS01740
C       READ NUMBER OF AREAS, NO OF TRIPS/SEASON                         NNS01750
C        AND NUMBER OF SEASONS FOR SIMULATION.                           NNS01760
C                                                                        NNS01770
        READ(5,81)                                                       NNS01780
        READ(5,*) NAREAS,NBOATS,NTRIPS,NSEASN,IREAL                      NNS01790
C                                                                        NNS01800
C       READ THE COSTS OF FISHING FOR EACH AREA                          NNS01810
C                                                                        NNS01820
        READ(5,81)                                                       NNS01830
        READ(5,*)(CF(I),I=1,NAREAS)                                      NNS01840
C                                                                        NNS01850
C      READ AREA'S QUANTITATIVE CATCHES ASSOCIATED WITH GOOD,AVG AND POORNNS01860
C                                                                        NNS01870
        READ(5,81)                                                       NNS01880
        DO 30 I=1,3                                                      NNS01890
30       READ(5,*)(QCATCH(I,J),J=1,6)                                    NNS01900
C                                                                        NNS01910
81     FORMAT(A50)                                                       NNS01920
C                                                                        NNS01930
       RETURN                                                            NNS01940
       END                                                               NNS01950
C                                                                        NNS01960
C                                                                        NNS01970
       SUBROUTINE RAND(ISEED,U)                                          NNS01980
C *********     MODIFIED RANDOM NUMBER GENERATOR (V.DYCK,J.LAWSON,J.SMITH NNS01990
C *********       "INTRODUCTION TO COMPUTING", RESTON,VA.(1979))         NNS02000
C               ======================================================== NNS02010
C                                                                        NNS02020
C                LINEAR CONGRUENTIAL GENERATOR FOR 32-BIT WORD COMPUTER. NNS02030
C                    CYCLE IS 2**29 IN LENGTH: VARIABLES ARE U(0,1)      NNS02040
       INTEGER ISEED                                                     NNS02050
       ISEED=ISEED*843314861+453816693                                   NNS02060
       IF(ISEED.LT.0)ISEED=ISEED+2147483647+1                            NNS02070
       U=ISEED*0.4656612E-9                                              NNS02080
       RETURN                                                            NNS02090
       END                                                               NNS02100
C                                                                        NNS02110
C                                                                        NNS02120
       SUBROUTINE REFERE(XT,REFER,ACTUAL)                                NNS02130
       IMPLICIT REAL *8(A-H,M-Z)                                         NNS02140
       INTEGER REFER                                                     NNS02150
       CHARACTER*8 ACTUAL                                                NNS02160
       IF(XT.LT.75.0) THEN                                               NNS02170
        REFER=3                                                          NNS02180
        ACTUAL= 'POOR'                                                   NNS02190
C       XT=50.0                                                          NNS02200
```

```
      ELSE
       IF(XT.LT.125.0) THEN
        REFER=2
        ACTUAL='AVERAGE'
C       XT=100.0
       ELSE
        REFER=1
        ACTUAL='GOOD'
C       XT=150.0
       ENDIF
      ENDIF
      RETURN
      END
C
C
      SUBROUTINE ECRIR1(X,PI,P,CATCH,PRICE,NAREAS,NBOATS,NTRIPS,NSEASN)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NAREAS,NBOATS,NTRIPS,NS
      CHARACTER*8 CATCH(3)
      DIMENSION CF(6),PI(3),PYX(3,3,6),P( 3, 3),QCATCH( 3,6)
      COMMON/BLOC2/PYX,QCATCH,CF
C     WRITE(8,91) NBOATS,NAREAS,NTRIPS
C     WRITE(8,92)(CF(I),I=1,NAREAS)
C     WRITE(8,93)
C     WRITE(8,94)
CC    WRITE(8,95) X,NS
CC    WRITE(8,96) (P(1,J),J=1,3),
CC   *             (P(2,J),J=1,3),
CC   *             (P(3,J),J=1,3)
CC    WRITE(8,97)(PI(I),I=1,3)
C     WRITE(8,980)
C     DO 10 I=1,3
C10    WRITE(8,98) CATCH(I),((PYX(I,J,K),J=1,3),K=1,NAREAS)
C     WRITE(8,990)(CF(I),I=1,NAREAS),PRICE
C     WRITE(8,995)(CATCH(I),(QCATCH(I,J),J=1,NAREAS),I=1,3)
C
91    FORMAT('1','        ADAPTIVE DECISION MAKING BY ',I3,' FISHERMEN
     *WITH: ',//,15X,I3,3X,'FISHING AREAS,',/,15X,I3,3X,
     *'TRIPS PER SEASON',//)
92    FORMAT(' THE COSTS ARE DISTRIBUTED AS FOLLOWS (*)',//,25X,
     *'A R E A',/,15X,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,'6',/,
     *14X,35('-'),/,1X,'  FISHING:    ',6(F5.1,1X),///)
93    FORMAT(1X,'STATE DEFINITION:',/,10X,'GOOD ----------->  OVER 125',
     */,10X,'AVERAGE ------->  BETWEEN 75 AND 125',/,10X,'POOR ---------'
     *,'-->  BELOW 75',//,1X,20('-'),//)
94    FORMAT(' INITIAL STATE: X(0) = 100   ------> AVERAGE',//,1X,
     *20('-'),//)
C95   FORMAT(1X,'ACTUAL STATE LEVEL ---------> ',F5.1,5X,'SEASON NUMBER'
C    *,1X,I2,//,1X,56('-'),//)
C96   FORMAT(' STATE TRANSITION MATRIX',/,1X,23('-'),//,13X,'TO',/,
C    *' FROM :    G        A        P',/,1X,26('-'),/,3X,'G   :',
C    *3(F6.2,1X),/,3X,'A   :',3(F6.2,1X),/,3X,'P   :',3(F6.2,1X),//,
C    *1X,26('-'),//)
C97   FORMAT(1X,'PI VECTOR',/,1X,9('-'),/,3X,'G      A      P',/,1X,
C    *20('-'),/,2X,3(F5.3,1X),//,1X,20('-'),//)
```

```
98    FORMAT(3(1X,A8,'‡',6(3(F4.2,1X),1X),/),/)
980   FORMAT('1','CONDITIONAL DISTRIBUTION OF CATCH GIVEN STATE',/,1X,
     *45('-'),//,14X,'AREA 1',10X,'AREA 2',10X,'AREA 3',10X,'AREA 4',
     *10X,'AREA 5',10X,'AREA 6',/,1X,105('-'),/,4X,'Y',4X,
     *'‡  ',6('G',4X,'A',4X,'P',5X),/,1X,105('-'),/)
990   FORMAT(//,' COSTS OF FISHING PER AREA',/,1X,
     *37('-'),//,32X,'A R E A',/,11X,52('-'),/,11X,'‡     1',8X,'2',8X,
     *'3',8X,'4',8X,'5',8X,'6',/,11X,52('-'),/,1X,'FISHING   ‡',
     *6(F6.1,3X),//,1X,'THE UNIT PRICE NOW STANDS AT ',F7.2,//)
995   FORMAT(//,' QUANTITATIVE VALUES FOR CATCHES PER AREA',/,1X,
     *40('-'),//,32X,'A R E A',/,11X,46('-'),/,11X,'‡    1',7X,'2',7X,
     *'3',7X,'4',7X,'5',7X,'6',/,1X,56('-'),/,3(1X,A10,'‡',6(F5.2,3X),
     */),//)
      RETURN
      END
C
C
      SUBROUTINE OPTIM(AREA,IR,ISEED,NAREAS,NS,NT,NTRIPS,JALPHA,
     *                 PI,ZALPHA)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER A,AREA,JALPHA(15,6),NT(10),NAREAS,NS,NTRIPS
      DIMENSION PI(3),ZALPHA(15,6,3)
      COMMON/BLOC2/PYX,QCATCH,CF
      EXPECT=0.0
      IINT=NT(NS)
      DO 10 A=1,NAREAS
       SUM=0.0
       IF(JALPHA(NTRIPS-IINT+1,A).EQ.1) THEN
        DO 20 I=1,3
20        SUM=SUM+(ZALPHA(NTRIPS-IINT+1,A,I)*PI(I))
        IF(SUM.GT.EXPECT) THEN
          AREA=A
          EXPECT=SUM
        ENDIF
       ENDIF
10    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE FISHIN(REFER,AREA,CATCH,CATCHN,IBOAT,IR,IREAL,ISEED,
     *                  NS,NT,PRICE,PRCDEV)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER AREA,CATCHN,ISEED,NAREAS,NS,NT(10),NTRIPS,NSEASN
      INTEGER QBEST,REFER
      CHARACTER*8 CATCH(3)
      DIMENSION CF(10),PYX(3,3,6),QCATCH(3,6),PRCDEV(50),NINC(5,15,50)
      DIMENSION TCATCH(5,15,50),TCOST(5,15,50),VALAND(5,15,50)
      DIMENSION TTCATC(5,15,50),TTCOST(5,15,50),VVALAN(5,15,50)
      DIMENSION NNINC(5,15,50)
      COMMON/BLOC2/PYX,QCATCH,CF
      COMMON/BLOC3/TCATCH,TCOST,VALAND,NINC
      COMMON/BLOC4/TTCATC,TTCOST,VVALAN,NNINC
      PK=1.0*NT(NS)
      PPRICE=15.0*(10.0*(PK**2.0)-160.0*K+3335.0)/49.0
```

```
      CALL RAND(ISEED,Y)
      IF(Y.LE.PYX(1,REFER,AREA)) THEN
       CATCHN=1
      ELSE
       IF(Y.LE.(PYX(1,REFER,AREA)+PYX(2,REFER,AREA))) THEN
        CATCHN=2
       ELSE
        CATCHN=3
       ENDIF
      ENDIF
      YY=QCATCH(CATCHN,AREA)
      TCATCH(NS,NT(NS),IBOAT)=YY
      TTCATC(NS,NT(NS),IBOAT)=TTCATC(NS,NT(NS),IBOAT)+
     *                   (TCATCH(NS,NT(NS),IBOAT)/IREAL)
      TCOST(NS,NT(NS),IBOAT)=CF(AREA)*PRCDEV(IBOAT)
      TTCOST(NS,NT(NS),IBOAT)=TTCOST(NS,NT(NS),IBOAT)+
     *                   (TCOST(NS,NT(NS),IBOAT)/IREAL)
      NINC(NS,NT(NS),IBOAT)=YY*PPRICE-CF(AREA)*PRCDEV(IBOAT)
      NNINC(NS,NT(NS),IBOAT)=NNINC(NS,NT(NS),IBOAT)+
     *                   (NINC(NS,NT(NS),IBOAT)/IREAL)
      VALAND(NS,NT(NS),IBOAT)=YY*PPRICE
      VVALAN(NS,NT(NS),IBOAT)=VVALAN(NS,NT(NS),IBOAT)+
     *                   (VALAND(NS,NT(NS),IBOAT)/IREAL)
      APRFT=YY*PPRICE-CF(AREA)
C     WRITE(8,580) IBOAT,AREA,YY,CATCH(CATCHN),APRFT
C580  FORMAT(1X,'BOAT ',I3,' FISHES IN AREA ',I1,' CATCHES Y= ',F6.3,
C    *' -----> ',A8,' FOR A REVENUE OF ',F8.2,'$')
      RETURN
      END
C
C
      SUBROUTINE UPDPI(PI,CATCHN,AREA)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER CATCHN,AREA
      DIMENSION PI(3),PYX(3,3,6)
      COMMON/BLOC2/PYX,QCATCH,CF
      BOTTOM=0.0
      DO 600 I=1,3
600    BOTTOM=BOTTOM+(PI(I)*PYX(CATCHN,I,AREA))
      DO 610 I=1,3
610    PI(I)=PI(I)*PYX(CATCHN,I,AREA)/BOTTOM
C     WRITE(8,680) (PI(I),I=1,3)
C680  FORMAT(10X,'PI-VECTOR FOR THIS BOAT NEXT TRIP -----> ',
C    *       3(F6.3,1X)/)
      RETURN
      END
C
C
      SUBROUTINE UPDST(H,P,PIB,RT,RTM1,XT,XTM1,
     *                 ACTUAL,ISEED,NBOATS,REFER)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER ISEED,NBOATS,REFER
      DOUBLE PRECISION X,Y
      CHARACTER*8 ACTUAL
      DIMENSION P(3,3),PIB(50,3)
```

```
      DATA GL/125.0/,AL/75.0/,R/0.95776147/,M/0.1/,ALPHA/0.4/
      DATA A0/0.699/,A1/-0.698/,A2/0.236/,A3/-0.029/
      DISC=SQRT((R*(ALPHA-1.0)-1.0)**2-4.0*R*(1.0-ALPHA))
C     COMPUTATION OF "F ZERO POINT ONE"
      MM=M/(-LOG(R))
      FZSM=EXP(A0+A1*MM+A2*(MM**2.0)+A3*(MM**3.0))
      F=M*FZSM
C     END OF COMPUTATION OF F0.1 (NATURAL MORT. SET AT 0.1... COULD
C     BE SET HIGHER.
      F=F+0.5
C     WRITE(6,*) ' F0.1 (+0.5) NOW SET AT ',F
      I=1
99    WRITE(8,2)
2     FORMAT('1','SEASON UPDATING (DERISO"S MODEL)',//,1X,
     *'POOR -------> (0,75) ...... XT =  50',/,1X,
     *'AVERAGE ----> (75,125) .... XT = 100',/,1X,
     *'GOOD -------> (125, +) .... XT = 150',//)
      CALL REFERE(XT,REFER,ACTUAL)
C     WRITE(8,3) ACTUAL, XT
C3    FORMAT(1X,'ACTUAL STATE : ', A8,'  ( XT = ',F5.1,' )',/)
C     WRITE(8,4) ACTUAL
C4    FORMAT(1X,'P(XTP1 = NEWSTATE ; XT = ',A8,' ) = ',/,
C    *20X,'- NEW STATE -',/,
C    *10X,'  GOOD  ',2X,'AVERAGE',2X,'  POOR  ',/,
C    *10X,8('-'),2X,8('-'),2X,8('-'),/)
C     R=0.0
      PII=4.0*ATAN(1.0)
      SS=EXP(-F-M)
      S=1.0-(F/(F+M))*(1.0-EXP(-F-M))
      H=F*(1.0-S)/(F+M)
C     THE FOLLOWING BOUNDS ARE FOR THE DERISO MODEL
      AA=((GL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1 )/(ALPHA*XT))
      IF(AA.LE.0.0) WRITE(6,*) I
      AA=((AL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1 )/(ALPHA*XT))
      IF(AA.LE.0.0) WRITE(6,*) I
      RHSL=-LOG((GL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1 )/
     *(ALPHA*XT))
      RHSR=-LOG((AL-(1.0+R)*S*XT*(1.0-H)+R*S**2.0*XTM1+R*S*RTM1 )/
     *(ALPHA*XT))
      X=RHSL
      CALL MDNORD(X,Y)
C     P(1)=L'INTEGRALE DE -L'INFINI A RHSL DE F(X) - DENSITE NORMALE
      P1=Y
      X=RHSR
C     P(2)=L'INTEGRALE DE RHSL A RHSR DE F(X) - DENSITE NORMALE
      CALL MDNORD(X,Y)
      P2=Y
      P2=P2-P1
C     P(3)=L'INTEGRALE DE RHSR A L'INFINI DE F(X) - DENSITE NORMALE
      P3=1.0-P1-P2
C     WRITE(8,5) P1,P2,P3
C5    FORMAT(11X,3(F6.4,4X),///)
C ****GENERATE RECRUITMENT ERROR (NORMAL, MU=0, SIGMA=1)
70    CALL RAND(ISEED,U1)
      U1=4.0*U1-2.0
```

```
      FU1=(1.0/SQRT(2.0*PII))*EXP(-(U1**2)/2.0)
      CALL RAND(ISEED,U2)
      U2=U2*(1.0/SQRT(2.0*PII))
      IF(U2.LE.FU1) THEN
C      VARIABILITY IS MODIFIED HERE ********************
       VAR=0.4
       ERROR=VAR*U1
      ELSE
       GOTO 70
      ENDIF
C     WRITE(8,7) ERROR
      RT=ALPHA*XT*EXP(-ERROR)
      XTP1=(1.0+R)*S*XT-R*S**2.0*XTM1+RT-R*S*RTM1
      RTM1=RT
      XTM1=XT
      XT=XTP1
      WRITE(6,6) XT
6     FORMAT(1X,'NEW STATE VALUE .....: ',F6.1)
C7    FORMAT(1X,'ASSOCIATED EPSILON ..: ',F9.6,/)
C     UPDATE TRANSITION MATRIX
      IF(ACTUAL.EQ.'GOOD')THEN
       P(1,1)=P1
       P(1,2)=P2
       P(1,3)=P3
      ELSE
       IF(ACTUAL.EQ.'AVERAGE')THEN
        P(2,1)=P1
        P(2,2)=P2
        P(2,3)=P3
       ELSE
        P(3,1)=P1
        P(3,2)=P2
        P(3,3)=P3
       ENDIF
      ENDIF
C
C     UPDATE PI-VECTOR FOR NEXT SEASON
      DO 10 IBOAT=1,NBOATS
       BOTTOM=0.0
       DO 20 I=1,3
        SUMP=0.0
        DO 30 J=1,3
30       SUMP=SUMP+P(I,J)
        BOTTOM=BOTTOM+(PIB(IBOAT,I)*SUMP)
20     CONTINUE
       DO 40 I=1,3
        TOP=0.0
        DO 50 J=1,3
50       TOP=TOP+(P(I,J)*PIB(IBOAT,I))
        PIB(IBOAT,I)=TOP/BOTTOM
40     CONTINUE
10    CONTINUE
      RETURN
      END
C
```

```
C
      SUBROUTINE BACKW( JALPHA,ZALPHA,PRICE,NAREAS,NTRIPS)
      IMPLICIT REAL *8( A-H,M-Z)
      INTEGER NAREAS,NTRIPS,NSEASN,NDOMIN(1000),FLAGD,JALPHA(15,6)
      DIMENSION PYX(3,3,6),CF(10),QCATCH( 3,6),G(3,6),ALPHA(3,10000)
      DIMENSION ZALPHA(15,6,3)
      COMMON/BLOC1/ALPHA,NDOMIN
      COMMON/BLOC2/PYX,QCATCH,CF
C
C     COMPUTE "G" FUNCTIONS FOR EACH STATE, FOR EACH AREA.
      CALL GCOMP( CF,PYX,QCATCH,G,K,PRICE,NAREAS)
C     COMPUTE ALPHA VALUES
       CALL INACO( G,ZALPHA,NAREAS,NTRIPS)
C      WRITE(6,*) ' INACO COMPLETED'
C
C     CREATE INITIAL SET OF NON-DOMINANT ALPHA VALUES
      K=1
      IQ=6
      CALL SETCO( K,FLAGD,IQ,JALPHA)
C
C*****************************************************************
C
C     REPEAT ABOVE "INDENTED" PROCEDURE FOR EACH TRIPS.
      DO 5 K=2,NTRIPS
C      WRITE(6,*) ' K=',K
       WRITE(7,1)
       CALL ACOMP( G,PYX,K,FLAGD,IQ,ZALPHA,NTRIPS)
       CALL SETCO( K,FLAGD,IQ,JALPHA )
5     CONTINUE
1     FORMAT('1')
      RETURN
      END
C
C
      SUBROUTINE GCOMP( CF,PYX,QCATCH,G,K, PRICE,NAREAS )
      IMPLICIT REAL *8( A-H,M-Z)
      INTEGER A,NAREAS
      DIMENSION CF(6),PYX( 3,3,6),QCATCH(3,6),G( 3,6)
C     PPRICE=PRICE
C     PPRICE=(10.0/7.0)*ABS(1.0*K-8.0)+PRICE
      PK=1.0*K
      PPRICE=15.0*(10.0*(PK**2.0)-160.0*K+3335.0)/49.0
      DO 10 A=1,NAREAS
       DO 10 I=1,3
        SUM=0.0
        DO 20 J=1,3
20       SUM=SUM+QCATCH(J,A)*PYX(J,I,A)
        G(I,A)=PPRICE*SUM-CF(A)
10    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE INACO( G,ZALPHA,NAREAS,NTRIPS)
      IMPLICIT REAL *8( A-H,M-Z)
```

```
      INTEGER A,NAREAS,NTRIPS,NDOMIN(1000)
      DIMENSION ALPHA(3,10000),G(3,6),ZALPHA(15,6,3)
      COMMON/BLOC1/ALPHA,NDOMIN
      DO 10 I=1,3
      DO 10 A=1,NAREAS
       ALPHA(I,A)=G(I,A)
       ZALPHA(1,A,I)=G(I,A)
10    CONTINUE
      WRITE(7,6)
      DO 20 A=1,6
20     WRITE(7,7)A,(ALPHA(I,A),I=1,3)
6     FORMAT('0','INITIAL ALPHA VALUES (= G(I,A)):',/,
     *       1X,'A   ','      I=1   ','      I=2   ','      I=3',/)
7     FORMAT(1X,I1,2X,3(F8.2,1X))
      RETURN
      END
C
C
      SUBROUTINE ACOMP(G,PYX,K,FLAGD,IQ,ZALPHA,NTRIPS)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER A,APRIM,APP,FLAGD,FLINS,NTRIPS,NDOMIN(1000)
      DIMENSION ALPHA(3,10000),G(3,6),BRAK(6,3,1000,3),PYX(3,3,6)
      DIMENSION ALPHIN(3,20,20,20),AVERG(3),ZALPHA(15,6,3)
      COMMON/BLOC1/ALPHA,NDOMIN
      DO 10 A=1,6
      DC 10 L=1,3
       DO 20 APRIM=1,FLAGD
       DO 20 I=1,3
        BRAK(A,L,APRIM,I)=ALPHA(I,NDOMIN(APRIM))*PYX(L,I,A)
20     CONTINUE
10    CONTINUE
      WRITE(7,4)
      DO 40 A=1,6
       IQ=0
       CO 5J IP1=1,FLAGD
       DO 50 IP2=1,FLAGD
       CO 50 IP3=1,FLAGD
        DO 60 I=1,3
60       ALPHIN(I,IP1,IP2,IP3)=BRAK(A,1,IP1,I)
     *                        +BRAK(A,2,IP2,I)
     *                        +BRAK(A,3,IP3,I)
        IQ=IQ+1
50     CONTINUE
       CALL MOYEN(AVERG,ALPHIN,IQ,FLAGD)
       DO 70 I=1,3
        ALPHA(I,A)=G(I,A)+AVERG(I)
        ZALPHA(K,A,I)=ALPHA(I,A)
70     CONTINUE
       WRITE(7,6)A,(ALPHA(I,A),I=1,3)
40    CONTINUE
3     FORMAT(1X,'A=',I1,1X,'L=',I1,1X,'APP=',I2,1X,3(F10.2,1X))
4     FORMAT(1X,'CURRENT   ',' ALPHA VALUES (I=1,2,3)',/,
     *       1X,41('-'))
6     FORMAT(4X,I1,10X,3(F8.2))
      RETURN
```

```
      END                                                               NNS06060
C                                                                       NNS06070
C                                                                       NNS06080
      SUBROUTINE SETCO( K,FLAGD,IQ,JALPHA)                              NNS06090
      IMPLICIT REAL *8(A-H,M-Z)                                         NNS06100
      INTEGER A,AA,NDOMIN(1000),FLAGD,JALPHA(15,6)                      NNS06110
      DIMENSION ALPHA(3,10000)                                          NNS06120
      COMMON/BLOC1/ALPHA,NDOMIN                                         NNS06130
      FLAGD=0                                                           NNS06140
      DO 5 A=1,6                                                        NNS06150
5       JALPHA(K,A)=0                                                   NNS06160
      DO 10 A=1,IQ                                                      NNS06170
        DO 20 AA=1,IQ                                                   NNS06180
          IF((ALPHA(1,A).LT.ALPHA(1,AA)) .AND.                          NNS06190
     *       (ALPHA( 2,A).LT.ALPHA( 2,AA)) .AND.                        NNS06200
     *       (ALPHA( 3,A).LT.ALPHA( 3,AA))) GO TO 10                    NNS06210
20      CONTINUE                                                        NNS06220
        FLAGD=FLAGD+1                                                   NNS06230
        NDOMIN(FLAGD)=A                                                 NNS06240
        JALPHA(K,A)=1                                                   NNS06250
10    CONTINUE                                                          NNS06260
      WRITE(7,1) K,(NDOMIN(A),A=1,FLAGD)                                NNS06270
      WRITE(7,2)(NDOMIN(A),(ALPHA(J,NDOMIN(A)),J=1,3),A=1,FLAGD)        NNS06280
1     FORMAT('0','WITH ',I2,' PERIODS TO GO, THE NON-DOMINATED ',       NNS06290
     *       'DECISIONS ARE',/,40(1X,I4)/)                              NNS06300
2     FORMAT('0',/,40(1X,I4,' -----> ',3(F8.2,1X),/))                   NNS06310
      RETURN                                                            NNS06320
      END                                                               NNS06330
C                                                                       NNS06340
C                                                                       NNS06350
      SUBROUTINE INSDOM(ALPHIN,NINS,IP1,IP2,IP3)                        NNS06360
      IMPLICIT REAL *8(A-H,M-Z)                                         NNS06370
      INTEGER NINS(20,20,20)                                            NNS06380
      DIMENSION ALPHIN( 3,20,20,20)                                     NNS06390
      DO 10 I=1,IP1                                                     NNS06400
      DO 10 II=1,IP2                                                    NNS06410
      DO 10 III=1,IP3                                                   NNS06420
       IF(NINS(I,II,III).EQ.0) GO TO 10                                 NNS06430
       IF((ALPHIN(1,I,II,III).LT.ALPHIN(1,IP1,IP2,IP3)).AND.            NNS06440
     *    (ALPHIN( 2,I,II,III).LT.ALPHIN( 2,IP1,IP2,IP3)).AND.          NNS06450
     *    (ALPHIN( 3,I,II,III).LT.ALPHIN( 3,IP1,IP2,IP3))) THEN         NNS06460
        NINS(I,II,III)=0                                                NNS06470
        GO TO 10                                                        NNS06480
       ELSE                                                             NNS06490
        IF((ALPHIN(1,I,II,III).GT.ALPHIN( 1,IP1,IP2,IP3)).AND.          NNS06500
     *      (ALPHIN( 2,I,II,III).GT.ALPHIN( 2,IP1,IP2,IP3)).AND.        NNS06510
     *      (ALPHIN(3,I,II,III).GT.ALPHIN( 3,IP1,IP2,IP3))) THEN        NNS06520
        NINS(IP1,IP2,IP3)=0                                             NNS06530
        GO TO 20                                                        NNS06540
        ENDIF                                                           NNS06550
       ENDIF                                                            NNS06560
10    CONTINUE                                                          NNS06570
      NINS(IP1,IP2,IP3)=1                                               NNS06580
20    RETURN                                                            NNS06590
      END                                                               NNS06600
```

```
C
C
      SUBROUTINE MOYEN( AVERG, ALPHIN, IQ, FLAGD)
      IMPLICIT REAL *8( A-H, M-Z)
      INTEGER APRIM,NDOMIN(1000),FLAGD
      DIMENSION ALPHIN( 3,20,20,20),AVERG( 3)
      COMMON/BLOC1/ALPHA,NDOMIN
      DO 10  I=1,3
       SUM=0.0
       DO 20  IP1=1,FLAGD
        DO 20  IP2=1,FLAGD
         DO  20  IP3=1,FLAGD
20        SUM=SUM+ALPHIN(I,IP1,IP2,IP3)
10     AVERG(I)=SUM/IQ
      RETURN
      END
C
C
      SUBROUTINE STATS( IFLAG, IR, NBOATS, NS, NT, NTRIPS, VECTOR)
      IMPLICIT REAL *8( A-H, M-Z)
      INTEGER NBOATS,NS,NT(10),NTRIPS,IMARK(10)
      DIMENSION VECTOR( 5,15,50),CLASS(10)
      DO 50  I=1,10
50     IMARK(I)=0
      IINT=NT(NS)
      IF(IFLAG.EQ.1) THEN
       IINT=NTRIPS
      ENDIF
      CALL MAXMIN( VECTOR,IR,NBOATS,IINT,NS,MAX,MIN)
      CALL MEANC( VECTOR,IR,NBOATS,IINT,NS,MEAN)
      CALL VARNCE( VECTOR,IR,NBOATS,IINT,NS,MEAN,VAR)
      CALL STANDV( VECTOR,IR,NBOATS,IINT,NS,MEAN,STD)
      WRITE(4,2) MAX,MIN,MEAN,VAR,STD
2     FORMAT('0','STATISTICS ON THE ACCUMULATED ABOVE VECTOR:',/,
     *         15X,'MAXIMUM VALUE = ',F13.2,/,
     *         15X,'MINIMUM VALUE = ',F13.2,/,
     *         15X,'AVERAGE VALUE = ',F13.2,/,
     *         15X,'VARIANCE      = ',F13.2,/,
     *         15X,'STD DEVIATION = ',F13.2,/)
      DO 10  K=1,10
10     CLASS(K)=MIN+K*((MAX-MIN)/10.0)
      DO 20  I=1,NBOATS
       SUM=0.0
       DO 60  II=1,IINT
60      SUM=SUM+VECTOR( NS,II,I)
       DO 30  K=1,10
        IF( SUM.LT.CLASS(K)) THEN
         IMARK(K)=IMARK(K)+1
         GOTO 20
        ENDIF
30     CONTINUE
20    CONTINUE
      IMARK(10)=IMARK(10)+1
C ***** WRITE THE HISTOGRAM *******
      WRITE(4,*) '                                      HISTOGRAM'
```

```
      WRITE(4,*) ' CLASS MARK'
      DO 40 K=1,10
       WRITE(4,1) K,CLASS(K),IMARK(K)
40    CONTINUE
1     FORMAT(1X,I2,'( ',F8.2,' )',',...   ',I3)
      RETURN
      END
C
C
      SUBROUTINE STAT2(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,
     *                 VVALAN)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NSEASN,NTRIPS,NT(10)
      DIMENSION NNINC(5,15,50),TTCATC(5,15,50),TTCOST(5,15,50)
      DIMENSION VVALAN(5,15,50)
      CALL MAXM2(NNINC,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      WRITE(4,2) 'NET INCOME',MAX,MIN
      CALL ST2ANX(NNINC,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      CALL MAXM2(TTCATC,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      WRITE(4,2) 'TOTAL CATCH',MAX,MIN
      CALL ST2ANX(TTCATC,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      CALL MAXM2(TTCOST,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      WRITE(4,2) 'TOTAL COSTS',MAX,MIN
      CALL ST2ANX(TTCOST,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      CALL MAXM2(VVALAN,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      WRITE(4,2) 'LANDED VALUE',MAX,MIN
      CALL ST2ANX(VVALAN,NBOATS,NSEASN,NTRIPS,MAX,MIN)
2     FORMAT('1','STATISTICS ON ACCUMULATED ',A15,/,
     *        15X,'MAXIMUM VALUE = ',F13.2,/,
     *        15X,'MINIMUM VALUE = ',F13.2,/)
      RETURN
      END
C
C
      SUBROUTINE ST2ANX(VECTOR,NBOATS,NSEASN,NTRIPS,MAX,MIN)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NSEASN,NTRIPS,NT(10),IIMARK(10,15),IAXIS(15)
      DIMENSION VECTOR(5,15,50),CLASS(10)
      DO 5 I=1,10
       DO 5 II=1,15
5       IIMARK(I,II)=0
      DO 10 K=1,10
10     CLASS(K)=MIN+K*((MAX-MIN)/10.0)
       DO 20 I=1,NTRIPS
        DO 30 II=1,NBOATS
         SUM=0.0
         DO 40 III=1,NSEASN
C         IF(I.GT.NI(III)) GO TO 40
          SUM=SUM+VECTOR(III,I,II)
40       CONTINUE
         DO 50 K=1,10
          IF(SUM.LT.CLASS(K)) THEN
           IIMARK(K,I)=IIMARK(K,I)+1
           GO TO 30
          ENDIF
```

```
50        CONTINUE
30       CONTINUE
20      CONTINUE
       DO 60  J=1,15
C       IIMARK(10,J)=IIMARK(10,J)+1
        IAXIS(J)=J
60     CONTINUE
C ***** WRITE THE HISTOGRAM *******
       WRITE(4,*) '                                          HISTOGRAM'
       WRITE(4,*) ' CLASS MARK'
       WRITE(4,2)(CLASS(K),(IIMARK(K,I),I=1,15),K=1,10)
       WRITE(4,3)'TRIPS',(IAXIS(I),I=1,15)
2      FORMAT(/,10(F8.2,5X,15(I3,2X),//))
3      FORMAT(/,A8,4X,15(I3,2X))
       RETURN
       END
C
C
       SUBROUTINE MAXMIN(VECTOR,IR,NBOATS,IINT,NS,MAX,MIN)
       IMPLICIT REAL *8(A-H,M-Z)
       INTEGER NBOATS,NS,NT(10)
       DIMENSION SUM(50),VECTOR(5,15,50)
       MAX=0.0
       MIN=100000000.0
       DO 5  I=1,NBOATS
5       SUM(I)=0.0
       DO 10  I=1,NBOATS
        DO 15  II=1,IINT
15        SUM(I)=SUM(I)+VECTOR(NS,II,I)
        IF(SUM(I).GE.MAX) THEN
         MAX=SUM(I)
        ENDIF
10     CONTINUE
       DO 20  I=1,NBOATS
        IF(SUM(I).LE.MIN) THEN
         MIN=SUM(I)
        ENDIF
20     CONTINUE
       RETURN
       END
C
C
       SUBROUTINE MAXM2(VECTOR,NBOATS,NSEASN,NTRIPS,MAX,MIN)
       IMPLICIT REAL *8(A-H,M-Z)
       INTEGER NBOATS,NSEASN,NTRIPS
       DIMENSION SUM(15,50),VECTOR(5,15,50)
       MAX=0.0
       MIN=100000000.0
       DO 5  I=1,NBOATS
        DO 5  II=1,15
5        SUM(II,I)=0.0
       DO 10  III=1,NSEASN
        DO 10  I=1,NBOATS
         DO 10  II=1,NTRIPS
10         SUM(II,I)=SUM(II,I)+VECTOR(III,II,I)
```

```
      DO 20 III=1,NSEASN                                           NNS08260
       DO 20 I=1,NBOATS                                            NNS08270
        DO 20 II=1,NTRIPS                                          NNS08280
         IF( SUM(II,I).GE.MAX) THEN                                NNS08290
          MAX=SUM(II,I)                                            NNS08300
         ENDIF                                                     NNS08310
20    CONTINUE                                                     NNS08320
      DO 30 III=1,NSEASN                                           NNS08330
       DO 30 I=1,NBOATS                                            NNS08340
        DO 30 II=1,NTRIPS                                          NNS08350
         IF( SUM(II,I).LE.MIN) THEN                                NNS08360
          MIN=SUM(II,I)                                            NNS08370
         ENDIF                                                     NNS08380
30    CONTINUE                                                     NNS08390
      RETURN                                                       NNS08400
      END                                                          NNS08410
C                                                                  NNS08420
C                                                                  NNS08430
      SUBROUTINE MEANC( VECTOR,IR,NBOATS,IINT,NS,MEAN)             NNS08440
      IMPLICIT REAL *8(A-H,M-Z)                                    NNS08450
      INTEGER NBOATS,NS,NT(10)                                     NNS08460
      DIMENSION VECTOR(5,15,50)                                    NNS08470
      SUM=0.0                                                      NNS08480
      DO 10 I=1,NBOATS                                             NNS08490
       DO 10 II=1,IINT                                             NNS08500
10      SUM=SUM+VECTOR(NS,II,I)                                    NNS08510
      MEAN=SUM/NBOATS                                              NNS08520
      RETURN                                                       NNS08530
      END                                                          NNS08540
C                                                                  NNS08550
C                                                                  NNS08560
      SUBROUTINE VARNCE(VECTOR,IR,NBOATS,IINT,NS,MEAN,VAR)         NNS08570
      IMPLICIT REAL *8(A-H,M-Z)                                    NNS08580
      INTEGER NBOATS,NS,NT(10)                                     NNS08590
      DIMENSION SUM(50),VECTOR(5,15,50)                            NNS08600
      DO 5 I=1,NBOATS                                              NNS08610
5      SUM( I )=0.0                                                NNS08620
      SUM1=0.0                                                     NNS08630
      DO 10 I=1,NBOATS                                             NNS08640
       DO 10 II=1,IINT                                             NNS08650
10      SUM( I)=SUM(I)+VECTOR(NS,II,I)                             NNS08660
      DO 20 I=1,NBOATS                                             NNS08670
20     SUM1=SUM1+(ABS(SUM(I)-MEAN)**2.0)                           NNS08680
      VAR=SUM1/NBOATS                                              NNS08690
      RETURN                                                       NNS08700
      END                                                          NNS08710
C                                                                  NNS08720
C                                                                  NNS08730
      SUBROUTINE STANDV(VECTOR,IR,NBOATS,IINT,NS,MEAN,STD)         NNS08740
      IMPLICIT REAL *8(A-H,M-Z)                                    NNS08750
      INTEGER NBOATS,NS,NT(10)                                     NNS08760
      DIMENSION SUM(50),VECTOR(5,15,50)                            NNS08770
      DO 5 I=1,NBOATS                                              NNS08780
5      SUM( I )=0.0                                                NNS08790
      SUM1=0.0                                                     NNS08800
```

```
      DO 10  I=1,NBOATS
       DO 10  II=1,IINT
10      SUM(I)=SUM(I)+VECTOR(NS,II,I)
      DO 20  I=1,NBOATS
20     SUM1=SUM1+(ABS(SUM(I)-MEAN)**2.0)
      IF(NBOATS.EQ.1)THEN
       STD=0.0
       GO TO 30
      ENDIF
      STD=(SUM1/((NBOATS*1.0)-1.0))**0.5
30    RETURN
      END
C
C
      SUBROUTINE DEVIA(NBOATS,NS,NT,PRCDEV,TCOST,VALAND,NINC)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10)
      DIMENSION TCOST(5,15,50),VALAND(5,15,50),PRCDEV(50)
      DIMENSION NINC(5,15,50)
      SUM1=0.0
      DO 10  I=1,NBOATS
       DO 10  II=1,NT(NS)
10      SUM1=SUM1+NINC(NS,II,I)
      AVERA=SUM1/NBOATS
      DO 20  I=1,NBOATS
       SUM2=0.0
       DO 30  II=1,NT(NS)
30      SUM2=SUM2+NINC(NS,II,I)
       PRCDEV(I)=((SUM2-AVERA)/AVERA)+1.0
20    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE CHECK(IR,ISTOP,NBOATS,NS,NT,H,VECTOR,XT)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NS,NT(10)
      DIMENSION VECTOR(5,15,50)
      SUM=0.0
      DO 10  I=1,NBOATS
       DO 10  II=1,NT(NS)
10      SUM=SUM+VECTOR(NS,II,I)
      ACTHAR=SUM/XT
      IF(ACTHAR.GT.H) THEN
       ISTOP=1
C      WRITE(6,2) ACTHAR,H,NT(NS)
C      WRITE(8,2) ACTHAR,H,NT(NS)
      ENDIF
C2    FORMAT(1X,60('*'),/,1X,'ACTUAL HARVEST EXCEEDS QUOTA:...',/,
C    *        1X,'CATCH/STATE ABUN. = ',F7.4,' H = ',F7.4,//,
C    *        1X,'TOTAL NUMBER OF TRIPS THIS SEASON: ',I3,/,
C    *        1X,60('*'))
      RETURN
      END
C
```

```
C
      SUBROUTINE HCCMP(H)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER ISEED,NBOATS,REFER
      DATA R/0.95776147/,M/0.1/
      DATA A0/0.699/,A1/-0.698/,A2/0.236/,A3/-0.029/
C     COMPUTATION OF "F ZERO POINT ONE"
      MM=M/(-LOG(R))
      FZSM=EXP(A0+A1*MM+A2*(MM**2.0)+A3*(MM**3.0))
      F=M*FZSM
C     END OF COMPUTATION OF F0.1 (NATURAL MORT. SET AT 0.1... COULD
C     BE SET HIGHER.
      F=F+0.5
      S=EXP(-F-M)
      H=F*(1.0-S)/(F+M)
      RETURN
      END
C
C
      SUBROUTINE STAT3(NBOATS,NTRIPS,NSEASN,NNINC,TTCATC,TTCOST,
     *                 VVALAN)
      IMPLICIT REAL *8(A-H,M-Z)
      INTEGER NBOATS,NSEASN,NTRIPS,NT(10)
      DIMENSION NNINC(5,15,50),TTCATC(5,15,50),TTCOST(5,15,50)
      DIMENSION VVALAN(5,15,50),AVER(4)
      DIMENSION SUM1(5),SUM2(5),SUM3(5),SUM4(5)
      AVR1=0.0
      AVR2=0.0
      AVR3=0.0
      AVR4=0.0
      DO 5 IS=1,5
       DO 10 IT=1,15
        DO 10 IB=1,50
         AVR1=AVR1+NNINC(IS,IT,IB)/5.0
         AVR2=AVR2+TTCATC(IS,IT,IB)/5.0
         AVR3=AVR3+TTCOST(IS,IT,IB)/5.0
         AVR4=AVR4+VVALAN(IS,IT,IB)/5.0
10     CONTINUE
       SUM1(IS)=0.0
       SUM2(IS)=0.0
       SUM3(IS)=0.0
       SUM4(IS)=0.0
5     CONTINUE
      WRITE(3,80) AVR1,AVR2,AVR3,AVR4
80    FORMAT('1','GLOBAL AVERAGE OF PERFORMANCE INDICATORS',/,
     *       1X,'NET INCOME',5X,'TOTAL CATCH',4X,'TOTAL COST',5X,
     *       'LANDED VALUE',/,1X,4(F13.5,2X),/,1X,60('='))
       WRITE(3,83) IS
83     FORMAT('1','PERFORMANCE INDICATORS',/,
     *       1X,'SN',3X,'NET INCOME',5X,'TOTAL CATCH',4X,
     *       'TOTAL COST',5X,'LANDED VALUE')
      DO 20 IS=1,5
       DO 30 IT=1,15
       DO 30 IB=1,50
        SUM1(IS)=SUM1(IS)+NNINC(IS,IT,IB)
```

```
         SUM2(IS)=SUM2(IS)+TTCATC(IS,IT,IB)
         SUM3(IS)=SUM3(IS)+TTCOST(IS,IT,IB)
         SUM4(IS)=SUM4(IS)+VVALAN(IS,IT,IB)
30      CONTINUE
C       SUM1(IS)=SUM1(IS)/NSEASN
C       SUM2(IS)=SUM2(IS)/NSEASN
C       SUM3(IS)=SUM3(IS)/NSEASN
C       SUM4(IS)=SUM4(IS)/NSEASN
        WRITE(3,86) IS,SUM1(IS),SUM2(IS),SUM3(IS),SUM4(IS)
86      FORMAT(1X,I2,3X,4(F13.5,2X))
20    CONTINUE
      RETURN
      END
```

INITIAL ALPHA VALUES (= G(I,A)):

A	I=1	I=2	I=3
1	28.05	27.00	23.85
2	31.55	25.70	21.35
3	30.85	27.40	24.10
4	29.55	26.85	24.60
5	30.35	26.90	23.60
6	30.35	26.90	23.60

WITH 1 PERIODS TO GO, THE NON-DOMINATED DECISIONS ARE
2 3 4

2 ----->	31.55	25.70	21.35
3 ----->	30.85	27.40	24.10
4 ----->	29.55	26.85	24.60

CURRENT # ALPHA VALUES (I=1,2,3)

1	58.70	53.65	47.20
2	62.20	52.35	44.70
3	61.50	54.05	47.45
4	60.20	53.50	47.95
5	61.00	53.55	46.95
6	61.00	53.55	46.95

WITH 2 PERIODS TO GO, THE NON-DOMINATED DECISIONS ARE
2 3 4

2 ----->	62.20	52.35	44.70
3 ----->	61.50	54.05	47.45
4 ----->	60.20	53.50	47.95

CURRENT # ALPHA VALUES (I=1,2,3)

1	89.35	80.30	70.55
2	92.85	79.00	68.05
3	92.15	80.70	70.80
4	90.85	80.15	71.30
5	91.65	80.20	70.30
6	91.65	80.20	70.30

WITH 3 PERIODS TO GO, THE NON-DOMINATED DECISIONS ARE
2 3 4

2 ----->	92.85	79.00	68.05
3 ----->	92.15	80.70	70.80
4 ----->	90.85	80.15	71.30

CURRENT #	ALPHA VALUES (I=1,2,3)		
1	395.35	346.80	304.05
2	399.35	345.50	301.55
3	398.65	347.20	304.30
4	397.35	345.65	304.80
5	398.15	346.70	303.80
6	398.15	346.70	303.80

WITH 13 PERIODS TO GO, THE NON-DOMINATED DECISIONS ARE
2 3 4

2 ----->	399.35	345.50	301.55
3 ----->	398.65	347.20	304.30
4 ----->	397.35	346.65	304.80

CURRENT #	ALPHA VALUES (I=1,2,3)		
1	426.50	373.45	327.40
2	430.00	372.15	324.90
3	429.30	373.85	327.65
4	428.00	373.30	328.15
5	428.30	373.35	327.15
6	428.30	373.35	327.15

WITH 14 PERIODS TO GO, THE NON-DOMINATED DECISIONS ARE
2 3 4

2 ----->	430.00	372.15	324.90
3 ----->	429.30	373.85	327.65
4 ----->	428.00	373.30	328.15

Figure D.2: Adaptive policy output

TRIP NO. 3 OF SEASON 2

```
BOAT   1 FISHES IN AREA 1 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.224    0.220   0.556
BOAT   2 FISHES IN AREA 3 CATCHES Y=   0.4 -----> POOR     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.156    0.101   0.744
BOAT   3 FISHES IN AREA 1 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.202    0.260   0.538
BOAT   4 FISHES IN AREA 3 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.221    0.322   0.457
BOAT   5 FISHES IN AREA 3 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.363    0.410   0.227
BOAT   6 FISHES IN AREA 3 CATCHES Y=   0.4 -----> POOR     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.651    0.287   0.062
BOAT   7 FISHES IN AREA 3 CATCHES Y=   1.2 -----> GOOD     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.548    0.368   0.084
BOAT   8 FISHES IN AREA 3 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.513    0.362   0.125
BOAT   9 FISHES IN AREA 1 CATCHES Y=   0.4 -----> POOR     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.010    0.202   0.788
BOAT  10 FISHES IN AREA 1 CATCHES Y=   0.4 -----> POOR     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.114    0.263   0.623
BOAT  11 FISHES IN AREA 1 CATCHES Y=   0.4 -----> POOR     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.089    0.284   0.627
BOAT  12 FISHES IN AREA 1 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.165    0.157   0.679
BOAT  13 FISHES IN AREA 3 CATCHES Y=   0.4 -----> POOR     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.619    0.286   0.094
BOAT  14 FISHES IN AREA 1 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.190    0.724   0.086
BOAT  15 FISHES IN AREA 3 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.232    0.468   0.300
BOAT  16 FISHES IN AREA 3 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.534    0.385   0.081
BOAT  17 FISHES IN AREA 3 CATCHES Y=   0.4 -----> POOR     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.469    0.347   0.183
BOAT  18 FISHES IN AREA 3 CATCHES Y=   1.2 -----> GOOD     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.396    0.438   0.166
BOAT  19 FISHES IN AREA 1 CATCHES Y=   0.8 -----> AVERAGE  FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.140    0.345   0.515
BOAT  20 FISHES IN AREA 3 CATCHES Y=   1.2 -----> GOOD     FOR A REVENUE
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.619    0.286   0.094
BOAT  21 FISHES IN AREA 1 CATCHES Y=   1.2 -----> GOOD     FOR A REVENUE
```

SEASON 2

```
---> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   23450.00$
--->  0.224   0.220   0.556
---> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   23455.50$
--->  0.156   0.101   0.744
---> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   23482.50$
--->  0.202   0.260   0.538
---> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   23509.00$
--->  0.221   0.322   0.457
---> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   23535.50$
--->  0.363   0.410   0.227
---> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   23541.00$
--->  0.651   0.287   0.062
---> GOOD     FOR A REVENUE OF    50.50$ ...ACCUMULATED:   23591.50$
--->  0.548   0.368   0.084
---> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   23618.00$
--->  0.513   0.362   0.125
---> POOR     FOR A REVENUE OF     6.00$ ...ACCUMULATED:   23624.00$
--->  0.010   0.202   0.788
---> POOR     FOR A REVENUE OF     6.00$ ...ACCUMULATED:   23630.00$
--->  0.114   0.263   0.623
---> POOR     FOR A REVENUE OF     6.00$ ...ACCUMULATED:   23636.00$
--->  0.089   0.284   0.627
---> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   23663.00$
--->  0.165   0.157   0.679
---> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   23668.50$
--->  0.619   0.286   0.094
---> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   23695.50$
--->  0.190   0.724   0.086
---> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   23722.00$
--->  0.232   0.468   0.300
---> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   23748.50$
--->  0.534   0.385   0.081
---> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   23754.00$
--->  0.469   0.347   0.183
---> GOOD     FOR A REVENUE OF    50.50$ ...ACCUMULATED:   23804.50$
--->  0.396   0.438   0.166
---> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   23831.50$
--->  0.140   0.345   0.515
---> GOOD     FOR A REVENUE OF    50.50$ ...ACCUMULATED:   23882.00$
--->  0.619   0.286   0.094
---> GOOD     FOR A REVENUE OF    51.00$ ...ACCUMULATED:   23933.00$
```

```
BOAT   22 FISHES IN AREA 1 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.223   0.479  0.298
BOAT   23 FISHES IN AREA 1 CATCHES Y=    0.4 -----> POOR     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.162   0.514  0.323
BOAT   24 FISHES IN AREA 1 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.183   0.137  0.680
BOAT   25 FISHES IN AREA 3 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.348   0.482  0.169
BOAT   26 FISHES IN AREA 1 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.098   0.259  0.644
BOAT   27 FISHES IN AREA 3 CATCHES Y=    0.4 -----> POOR     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.469   0.347  0.183
BOAT   28 FISHES IN AREA 3 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.423   0.166  0.411
BOAT   29 FISHES IN AREA 3 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.651   0.287  0.062
BOAT   30 FISHES IN AREA 3 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.269   0.462  0.269
BOAT   31 FISHES IN AREA 3 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.276   0.539  0.185
BOAT   32 FISHES IN AREA 3 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.656   0.261  0.082
BOAT   33 FISHES IN AREA 3 CATCHES Y=    1.2 -----> GOOD     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.578   0.281  0.141
BOAT   34 FISHES IN AREA 1 CATCHES Y=    1.2 -----> GOOD     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.233   0.266  0.500
BOAT   35 FISHES IN AREA 2 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.748   0.210  0.042
BOAT   36 FISHES IN AREA 3 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.269   0.462  0.269
BOAT   37 FISHES IN AREA 3 CATCHES Y=    0.4 -----> POOR     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.230   0.259  0.511
BOAT   38 FISHES IN AREA 1 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.141   0.458  0.401
BOAT   39 FISHES IN AREA 3 CATCHES Y=    0.4 -----> POOR     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.469   0.347  0.183
BOAT   40 FISHES IN AREA 1 CATCHES Y=    1.2 -----> GOOD     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.163   0.567  0.270
BOAT   41 FISHES IN AREA 3 CATCHES Y=    0.4 -----> POOR     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.605   0.316  0.079
BOAT   42 FISHES IN AREA 1 CATCHES Y=    0.8 -----> AVERAGE  FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.248   0.255  0.496
BOAT   43 FISHES IN AREA 3 CATCHES Y=    0.4 -----> POOR     FOR A REVENUE OF
          PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.314   0.372  0.314
```

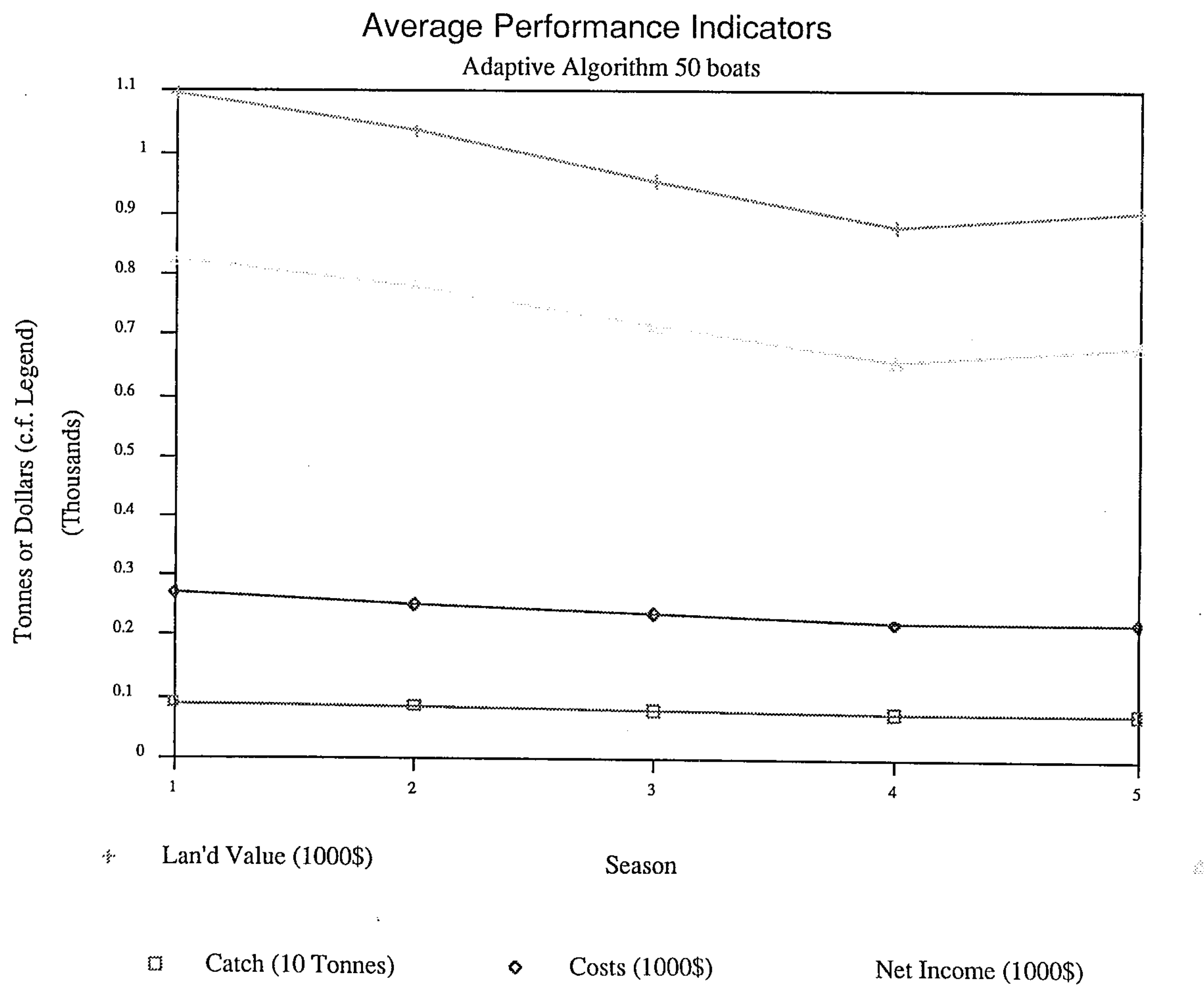
```
3 -----> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   23960.00$
P ----->  0.223   0.479   0.298

4 -----> POOR     FOR A REVENUE OF     6.00$ ...ACCUMULATED:   23966.00$
P ----->  0.162   0.514   0.323

3 -----> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   23993.00$
P ----->  0.183   0.137   0.680

3 -----> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   24019.50$
P ----->  0.348   0.482   0.169

3 -----> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   24046.50$
P ----->  0.098   0.259   0.644

4 -----> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   24052.00$
P ----->  0.469   0.347   0.183

3 -----> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   24078.50$
P ----->  0.423   0.166   0.411

3 -----> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   24105.00$
P ----->  0.651   0.287   0.062

3 -----> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   24131.50$
P ----->  0.269   0.462   0.269

3 -----> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   24158.00$
P ----->  0.276   0.539   0.185

3 -----> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   24184.50$
P ----->  0.656   0.261   0.082

2 -----> GOOD     FOR A REVENUE OF    50.50$ ...ACCUMULATED:   24235.00$
P ----->  0.578   0.281   0.141

2 -----> GOOD     FOR A REVENUE OF    51.00$ ...ACCUMULATED:   24286.00$
P ----->  0.233   0.266   0.500

3 -----> AVERAGE  FOR A REVENUE OF    26.00$ ...ACCUMULATED:   24312.00$
P ----->  0.748   0.210   0.042

3 -----> AVERAGE  FOR A REVENUE OF    26.50$ ...ACCUMULATED:   24338.50$
P ----->  0.269   0.462   0.269

4 -----> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   24344.00$
P ----->  0.230   0.259   0.511

3 -----> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   24371.00$
P ----->  0.141   0.458   0.401

4 -----> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   24376.50$
P ----->  0.469   0.347   0.183

2 -----> GOOD     FOR A REVENUE OF    51.00$ ...ACCUMULATED:   24427.50$
P ----->  0.163   0.567   0.270

4 -----> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   24433.00$
P ----->  0.605   0.316   0.079

3 -----> AVERAGE  FOR A REVENUE OF    27.00$ ...ACCUMULATED:   24460.00$
P ----->  0.248   0.255   0.496

4 -----> POOR     FOR A REVENUE OF     5.50$ ...ACCUMULATED:   24465.50$
P ----->  0.314   0.372   0.314
```

```
BOAT  45 FISHES IN AREA 1 CATCHES Y=    1.2 -----> GOOD        FOR A REVENUE OF
         PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.021   0.229   0.750
BOAT  46 FISHES IN AREA 3 CATCHES Y=    0.4 -----> POOR        FOR A REVENUE OF
         PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.314   0.372   0.314
BOAT  47 FISHES IN AREA 1 CATCHES Y=    1.2 -----> GOOD        FOR A REVENUE OF
         PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.043   0.405   0.553
BOAT  48 FISHES IN AREA 1 CATCHES Y=    0.8 -----> AVERAGE     FOR A REVENUE OF
         PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.191   0.400   0.409
BOAT  49 FISHES IN AREA 3 CATCHES Y=    1.2 -----> GOOD        FOR A REVENUE OF
         PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.513   0.362   0.125
BOAT  50 FISHES IN AREA 3 CATCHES Y=    1.2 -----> GOOD        FOR A REVENUE OF
         PI-VECTOR FOR THIS BOAT NEXT TRIP -----> 0.741   0.214   0.044
```

```
GOOD        FOR A  REVENUE OF      51.00$ ...ACCUMULATED:    24543.50$
 0.021     0.229   0.750
POOR        FOR A  REVENUE OF       5.50$ ...ACCUMULATED:    24549.00$
 0.314     0.372   0.314
GOOD        FOR A  REVENUE OF      51.00$ ...ACCUMULATED:    24600.00$
 0.043     0.405   0.553
AVERAGE     FOR A  REVENUE OF      27.00$ ...ACCUMULATED:    24627.00$
 0.191     0.400   0.409
GOOD        FOR A  REVENUE OF      50.50$ ...ACCUMULATED:    24677.50$
 0.513     0.362   0.125
GOOD        FOR A  REVENUE OF      50.50$ ...ACCUMULATED:    24728.00$
 0.741     0.214   0.044
```

Figure D.3: Fishing simulation output

Figure D.4: Seasonal performance indicators


Average Performance Indicators
Adaptive Algorithm 50 boats
Tonnes or Dollars (c.f. Legend)
(Thousands)
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1
2
3
4
5
Season
Lan'd Value (1000$)
Catch (10 Tonnes)
Costs (1000$)
Net Income (1000$)

```
STATISTICS ON ****        OVERALL TOTAL CATCH **** SEASON NO.    5

STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
              MAXIMUM VALUE =           0.82
              MINIMUM VALUE =           0.69
              AVERAGE VALUE =           0.75
              VARIANCE      =           0.00
              STD DEVIATION =           0.02

                              HISTOGRAM
CLASS MARK
 1(      0.70  ),...     2
 2(      0.72  ),...     3
 3(      0.73  ),...     4
 4(      0.74  ),...    10
 5(      0.76  ),...    11
 6(      0.77  ),...    12
 7(      0.78  ),...     5
 8(      0.79  ),...     2
 9(      0.81  ),...     0
 0(      0.82  ),...     1
```

```
STATISTICS ON ****        OVERALL TOTAL COSTS **** SEASON NO.    5

STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
              MAXIMUM VALUE =         255.18
              MINIMUM VALUE =         195.91
              AVERAGE VALUE =         224.48
              VARIANCE      =         143.12
              STD DEVIATION =          12.08

                              HISTOGRAM
CLASS MARK
 1(     201.84  ),...     1
 2(     207.76  ),...     2
 3(     213.69  ),...     7
 4(     219.62  ),...     9
 5(     225.55  ),...     8
 6(     231.47  ),...     9
 7(     237.40  ),...     7
 8(     243.33  ),...     4
 9(     249.26  ),...     1
10(     255.18  ),...     2
```

STATISTICS ON ****   OVERALL NET INCOME **** SEASON NO.   5

```
STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
              MAXIMUM VALUE =          769.31
              MINIMUM VALUE =          619.37
              AVERAGE VALUE =          681.19
              VARIANCE      =          946.38
              STD DEVIATION =           31.08

                                  HISTOGRAM
 CLASS MARK
  1(    634.36  ),...    3
  2(    649.36  ),...    6
  3(    664.35  ),...    4
  4(    679.35  ),...   10
  5(    694.34  ),...   13
  6(    709.34  ),...    5
  7(    724.33  ),...    6
  8(    739.33  ),...    1
  9(    754.32  ),...    1
 10(    769.31  ),...    1
```

STATISTICS ON ****   OVERALL LANDED VALUE **** SEASON NO.   5

```
STATISTICS ON THE ACCUMULATED ABOVE VECTOR:
              MAXIMUM VALUE =          990.10
              MINIMUM VALUE =          826.50
              AVERAGE VALUE =          905.68
              VARIANCE      =          877.22
              STD DEVIATION =           29.92

                                  HISTOGRAM
 CLASS MARK
  1(    842.86  ),...    2
  2(    859.22  ),...    1
  3(    875.58  ),...    4
  4(    891.94  ),...    9
  5(    908.30  ),...    8
  6(    924.66  ),...   13
  7(    941.02  ),...    8
  8(    957.38  ),...    4
  9(    973.74  ),...    0
 10(    990.10  ),...    1
```

STATISTICS ON ACCUMULATED LANDED VALUE
MAXIMUM VALUE = 417.37
MINIMUM VALUE = 99.16

HISTOGRAM

CLASS MARK															
130.98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35
162.80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15
194.63	0	0	0	0	0	0	0	0	0	0	0	0	0	17	0
226.45	0	0	0	0	0	0	0	0	0	0	0	0	0	29	0
258.27	0	0	0	0	0	0	0	0	0	0	1	0	2	4	0
290.09	0	0	0	0	0	0	0	0	0	0	17	1	27	0	0
321.91	0	0	0	0	0	0	0	0	5	7	27	33	21	0	0
353.73	14	11	3	9	2	10	8	6	26	33	5	14	0	0	0
385.55	33	34	34	33	36	28	31	26	18	10	0	2	0	0	0
417.37	3	5	8	8	12	12	11	17	1	0	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

STATISTICS ON ACCUMULATED TOTAL COSTS

MAXIMUM VALUE = 110.23

MINIMUM VALUE = 20.21

HISTOGRAM

CLASS MARK															
29.21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50
38.21	0	0	0	0	0	0	0	0	0	0	0	0	0	47	0
47.22	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0
56.22	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0
65.22	0	0	0	0	0	0	0	0	0	0	43	43	11	0	0
74.22	0	0	0	0	0	0	0	0	0	1	7	7	0	0	0
83.22	0	0	0	0	0	0	0	0	49	49	0	0	0	0	0
92.23	0	0	0	0	0	0	0	34	1	0	0	0	0	0	0
101.23	0	0	0	1	14	49	50	16	0	0	0	0	0	0	0
110.23	50	50	49	49	36	1	0	0	0	0	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

STATISTICS ON ACCUMULATED TOTAL CATCH
MAXIMUM VALUE = 0.40
MINIMUM VALUE = 0.06

HISTOGRAM

CLASS MARK															
0.09	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50
0.13	0	0	0	0	0	0	0	0	0	0	0	0	0	30	0
0.16	0	0	0	0	0	0	0	0	0	0	0	0	1	20	0
0.19	0	0	0	0	0	0	0	0	0	0	4	0	37	0	0
0.23	0	0	0	0	0	0	0	0	0	1	40	41	12	0	0
0.26	0	0	0	0	0	0	0	0	14	31	6	9	0	0	0
0.30	0	0	0	0	0	0	4	10	32	18	0	0	0	0	0
0.33	2	5	4	9	9	30	37	36	4	0	0	0	0	0	0
0.36	36	32	33	38	35	20	9	4	0	0	0	0	0	0	0
0.40	11	13	13	3	5	0	0	0	0	0	0	0	0	0	0
TRIPS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Appendix E

Initial Data

This appendix presents the initial data used for the myopic and adaptive algorithms. One key data input is the catch conditional on state matrix system ('Y given X'). The actual catchability facility in any fishing area is measured in this system. For example, fishermen might be very seldomly observing a high catch in a given area, regardless of the overall state of the system. Steepness of the Bank, meteorological components, undersea particularities, and so on constitute the issues on which the probability measures would be based. Table E.1 presents the input used for modelling both scenarios.

The quantitative catch class marks for each possible outcome levels was assumed particular to each fishing area and algorithm. Table E.2 gives these values for each of the six fishing areas.

The cost of fishing per area was, in the absence of real data and for simplification purposes, input as the same (22 units of money) for all 6 areas. The costs of test fishing (myopic algorithm only) was set at 10 units. Also used in the myopic algorithm was the catch conditional on test result probability matrix.

Catch	*Area 1*			*Area 2*			*Area 3*		
	j=G	j=A	j=P	j=G	j=A	j=P	j=G	j=A	j=P
l=G	0.30	0.35	0.25	0.45	0.25	0.20	0.40	0.30	0.25
l=A	0.45	0.30	0.35	0.30	0.45	0.35	0.35	0.40	0.35
l=P	0.25	0.35	0.40	0.25	0.30	0.45	0.25	0.30	0.40
Catch	*Area 4*			*Area 5*			*Area 6*		
	j=G	j=A	j=P	j=G	j=A	j=P	j=G	j=A	j=P
l=G	0.36	0.30	0.25	0.40	0.30	0.25	0.40	0.30	0.25
l=A	0.35	0.35	0.35	0.35	0.40	0.35	0.35	0.40	0.35
l=P	0.29	0.35	0.40	0.25	0.30	0.40	0.25	0.30	0.40

Table E.1: Catch conditional on State probability matrix system

Myopic Algorithm						
	Area					
	1	2	3	4	5	6
l=Good	0.25	0.25	0.25	0.25	0.25	0.25
l=Aver.	0.15	0.15	0.15	0.15	0.15	0.15
l=Poor	0.08	0.08	0.08	0.08	0.08	0.08
Adaptive Algorithm						
	Area					
	1	2	3	4	5	6
l=Good	0.1	0.1	0.1	0.1	0.1	0.1
l=Aver.	0.085	0.085	0.085	0.085	0.085	0.085
l=Poor	0.05	0.05	0.05	0.05	0.05	0.05

Table E.2: Value (in Tonnes) of catches per area

	Catch		
	l=G	l=A	l=P
t=favor.	0.60	0.55	0.25
t=unfav.	0.40	0.45	0.75

Table E.3: Catch conditional on Test fishing result matrix

The values are given in Table E.3.

The stock-recruitment initial data was established as follows:

Good stock level .... over 125 tonnes

Average stock level .... between 75 and 125 tonnes

Poor stock level .... under 75 tonnes

Initial recruitment (R_0) .... 40 tonnes

Brody growth coefficient .... $\varrho = 0.958$

Natural mortality rate .... $M = 0.10$

Recruitment rate .... $\alpha = 0.40$

Variability .... $\nu = 0.30$

The initial π-vector assumed each state level to be equally likely to occur (i.e. $\pi_G = 1/3, \pi_A = 1/3$ and $\pi_P = 1/3$)

Bibliography

[1] Yaakov Bar-Shalom and Edison Tse. Concepts and Methods in Stochastic Control. In C.T. Leondes, editor, *Control and Dynamic Systems: Advances in Theory and Applications*, 99-172, Academic Press, New York, New York, 1975.

[2] D.J. Bartholomew. *Stochastic Models for Social Processes.* John Wiley & Sons, London, England, 2nd edition, 1973.

[3] Dimitri P. Bertsekas. *Dynamic Programming and Stochastic Control.* Volume 125 of *Mathematics in Science and Engineering*, Academic Press, Inc., New York, New York, 1976.

[4] R.J.H. Beverton and S.J. Holt. On the Dynamics of Exploited Fish Populations. *Ministry of Agriculture, Fisheries and Food (London)*, 2(19), 1957.

[5] Peter J. Bickel and Kjell A. Doksum. *Mathematical Statistics.* Holden-Day, Inc., Oakland, California, 1977.

[6] J. F. Caddy. Spatial Model for an Exploited Shellfish Population, and its Application to the Georges Bank Scallop Fishery. *Journal of the Fisheries Research Board of Canada*, 32:1305–1328, 1975.

[7] Canada. Advice on the Management of Scallop Resources. *Department of Fisheries and Oceans*, 1985.

[8] Canada. Canadian Fisheries - Landings, 1986-88. *Department of Fisheries and Oceans*, 10(1), 1989.

[9] Canada. Offshore Scallop Fishery Management Plan. *Department of Fisheries and Oceans*, 1988.

[10] Anthony T. Charles. Fisheries Investment with Imperfect Information: Effects of Parameter Uncertainty and Bayesian Updating. 1983. Working Paper: Fisheries Research Branch, Fisheries and Oceans, Canada.

[11] Anthony T. Charles. Nonlinear Costs and Optimal Fleet Capacity in Deterministic and Stochastic Fisheries. *Mathematical Biosciences*, 73:271–299, 1985.

[12] Anthony T. Charles. Optimal Fisheries Investment: Comparative Dynamics for a Deterministic Seasonal Fishery. *Canadian Journal of Fisheries and Aquatic Sciences*, 40(12):2069–2079, 1983.

[13] Anthony T. Charles. Optimal Fisheries Investment Under Uncertainty. *Canadian Journal of Fisheries and Aquatic Sciences*, 40(12):2080–2091, 1983.

[14] Anthony T. Charles and Greg R. Munro. Irreversible Investment and Optimal Fisheries Management: A Stochastic Analysis. *Marine Resource Economics*, 1(3):247–264, 1985.

[15] Colin W. Clark. *Bioeconomic Modelling and Fisheries Management.* John Wiley & Sons, New York, New York, 1985.

[16] Colin W. Clark. *Mathematical Bioeconomics: The optimal management of renewable resources.* John Wiley & Sons, New York, New York, 1976.

[17] Colin W. Clark, Anthony T. Charles, J.R. Beddington, and Mark Mangel. Optimal Capacity Decisions in a Developing Fishery. *Marine Resource Economics*, 2(1):25–53, 1985.

[18] J. Collie and Carl J. Walters. Alternative Recruitment Models of Adams River Sockeye. *Canadian Journal of Fisheries and Aquatic Sciences*, 44:1551–1561, 1987.

[19] J. Crutchfield and A. Zellner. Economic Aspects of the Pacific Halibut Fishery. *Fish. Indus. Res.*, 1, 1962.

[20] R.B. Deriso. Harvesting Strategies and Parameter Estimation for an Age-structured Model. *Canadian Journal of Fisheries and Aquatic Sciences*, 37:268–282, 1980.

[21] R.B. Deriso. Optimal $F_{0.1}$ Criteria and their Relationship to Maximum Sustainable Yield. *Canadian Journal of Fisheries and Aquatic Sciences*, 44:339–348, 1987.

[22] W.G. Doubleday, D. Rivard, and W.D. McKone. Estimation of Partial Recruitment and Yield per Recruit for an Otter Trawl Fishery for Deepwater Redfish. *North American Journal of Fisheries Management*, 4:15–31, 1984.

[23] F.A. Droppo and Steven Guthrie. Underwater Klondike? *Canadian Banker*, 95(4):6–12, 1988.

[24] B.A.T. Edwards. *Dispatch Rule and Batch Size Effects on the Performance of Flow Shops Characterized by Multiple Revisit Workstations.* Master's thesis, University of Ottawa, Ottawa (Ontario), 1988.

[25] Michael Golomb and Merrill Shanks. *Elements of Ordinary Differential Equations.* McGraw-Hill Book Company, New York, New York, 1950.

[26] H.S. Gordon. The Economic Theory of a Common Property Resource. *Journal of Political Economy*, 62:124–142, 1954.

[27] J.A. Gulland. The Concept of Marginal Yield from Exploited Fish Stocks. *Journal du conseil international de l'exploration de la mer*, 32:256–261, 1968.

[28] J.A. Gulland. *Fish Stock Assessment: A Manual of Basic Methods.* Volume 1 of *FAO/ Wiley Series on Food and Agriculture*, John Wiley & Sons, Chichester, England, 1983.

[29] J.A. Gulland and L.K. Boerema. Scientific Advice on Catch Levels. *Fishery Bulletin*, 71(2):325–335, 1973.

[30] Rögnvaldur Hannesson and Stein Ivar Steinshamn. How to set catch quotas: Constant Effort or Constant Catch. *Discussion paper series of the Institute of Fisheries Economics*, 3, 1989. Norwegian School of Economics and Business Administration.

[31] L. Harris. Independent Review of the State of the Northern Cod Stock. May 1989. Prepared for the Minister of Fisheries and Oceans, Canada.

[32] Ray Hilborn. A comparison of alternative harvest tactics for inverterbrate fisheries. In G.S. Jamieson and N. Bourne, editors, *Canadian Special Publications of Fisheries and Aquatic Sciences: 92*, pages 313–317, North Pacific Workshop on Stock Assessment and Management of Invertebrates, 1986.

[33] Ray Hilborn. Fleet Dynamics and Individual Variation: Why Some People Catch Fish More than Others. *Canadian Journal of Fisheries and Aquatic Science*, 42(1):2–13, 1985.

[34] Ray Hilborn and Carl J. Walters. A General Model for Simulation of Stock and Fleet Dynamics in Spatially Heterogeneous Fisheries. *Canadian Journal of Fisheries and Aquatic Science*, 44:1366–1369, 1987.

[35] Paul G. Hoel, Sidney C. Port, and Charles J. Stone. *Introduction to Probability Theory.* Houghton Mifflin Company, Boston, Massachusetts, 1971.

[36] Robert V. Hogg and Elliot A. Tanis. *Probability and Statistical Inference.* Macmillan Publishing Company, New York, New York, 3rd edition, 1988.

[37] C.S. Holling. *Adaptive Environmental Assessment and Management.* John Wiley & Sons, New York, New York, 1978.

[38] R.C. Holt, J.N.P. Hume, Jean Lavoie, and Pierre N. Robillard. *Fortran Watfiv-S, Programmation structurée et exercices.* gaëtan morin & associés, Chicoutimi (Québec), 1980.

[39] Charles E. Hutchinson and T. R. Fischer. Stochastic Control Theory Applied to Fishery Management. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-9(5):253–259, 1979.

[40] G.S. Jamieson and M.J. Lundy. Bay of Fundy Scallop Stock Assessment - 1978. *Fisheries and Marine Service Technical Report*, 1979.

[41] G.S. Jamieson, N.B. Witherspoon, and M.J. Lundy. Assessment of Northumberland Strait Scallop Stock - 1980. *Canadian Technical Report of Fisheries and Aquatic Sciences*, 1017, 1981.

[42] J. Johnston. *Econometric Methods.* McGraw-Hill Book Company, New York, New York, 1984.

[43] M.J.L. Kirby. Navigating Troubled Waters, A New Policy for Atlantic Fisheries. *Report of the Task Force on Atlantic Fisheries*, 1982.

[44] Leslie Lamport. LaTeX *; A Document Preparation System.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1986.

[45] Daniel E. Lane. *Dynamic Models of Decision Making by Fishermen.* PhD thesis, University of British Columbia, Vancouver (British Columbia), July 1986.

[46] Daniel E. Lane. *Economics of* $F_{0.1}$. Technical Report FP802-7-2068, Department of Fisheries and Oceans, Economic and Commercial Analysis Directorate, Ottawa (Ontario), March 1989.

[47] Daniel E. Lane. Investment Decision Making by Fishermen. *Canadian Journal of Fisheries and Aquatic Sciences*, 45(5):782–796, 1988.

[48] Daniel E. Lane. Operational Research and Fisheries Management. *European Journal of Operational Research*, 42(3):229–242, October 1989.

[49] Daniel E. Lane. *Optimal Capacity Expansion with Infrastructure Considerations; Time Phasing and Plant Size.* Master's thesis, University of Waterloo, Waterloo (Ontario), 1979.

[50] Daniel E. Lane. A Partially Observable Model of Decision Making by Fishermen. *Operations Research*, 37(2):240–254, 1988.

[51] Daniel E. Lane. A Simulation Model of Lobster Fishery Dynamics. *University of Ottawa Working Paper 85-49*, 1985.

[52] A.M. Law and W.D. Kelton. *Simulation Modeling and Analysis.* McGraw-Hill Book Company, Toronto (Ontario), 1982.

[53] Sang M. Lee, Laurence J. Moore, and Bernard W. Taylor. *Management Science.* Wm. C. Brown Company, Dubuque, Iowa, 1981.

[54] Richard I. Levin, David S. Rubin, and Joel P. Stinson. *Quantitative Approaches to Management.* McGraw-Hill Book Company, New York, New York, 6th edition, 1986.

[55] Mark Mangel. *Decision and Control in Uncertain Resource Systems.* Academic Press, New York, New York, 1985.

[56] Mark Mangel and Colin W. Clark. Uncertainty, Search and Information in Fisheries. *Journal du conseil international de l'exploration de la mer*, 41:93–103, 1982.

[57] Patricia Marchak, Neil Guppy, and John McMullan, editors. *Uncommon Property: The fishing and fish processing industries in British Columbia.* Methuen, Toronto, Ontario, 1987.

[58] Richard J. Meinhold and Nozer D. Singpurwalla. Understanding the Kalman Filter. *The American Statistician*, 37(2):123–127, May 1983.

[59] Robert K. Mohn. Yield per recruit analysis. In Denis Rivard, editor, *Collected papers on stock assessment methods*, pages 147–167, Canadian Atlantic Fisheries Scientific Advisory Committee, 1988.

[60] Robert K. Mohn and R.W. Elner. A simulation of the Cape Breton snow crab, *Chionoecetes opilio*, fishery for testing the robustness of the Leslie method. *Canadian Journal of Fisheries and Aquatic Sciences*, 44:2002–2008, 1987.

[61] Robert K. Mohn, Ginette Robert, and D.L. Roddick. Georges Bank Scallop Stock Assessment - 1984. *CAFSAC Research Document 85/36*, 1985.

[62] Robert K. Mohn, Ginette Robert, and D.L. Roddick. Georges Bank Scallop Stock Assessment - 1986. *CAFSAC Research Document 87/9*, 1987.

[63] Robert K. Mohn, Ginette Robert, and D.L. Roddick. Georges Bank Scallop Stock Assessment - 1987. *CAFSAC Research Document 88/3*, 1988.

[64] Robert K. Mohn, D.L. Roddick, and Ginette Robert. Biological considerations in the definition of fishing strategies for Georges Bank scallops. 1987. Draft written for the Canadian Atlantic Fisheries Scientific Advisory Committee.

[65] K.S. Naidu. Reproduction and breeding cycle of the giant Scallop *Placopecten magellanicus* Gmelin) in Port au Port Bay, newfoundland. *Canadian Journal of Zoology*, 48:1003–1012, 1970.

[66] Michel Nedzela. *Introduction à la science de la gestion.* Presses de l'Université du Québec, Sillery (Québec), 3e edition, 1986.

[67] Michel Nedzela. *Modèles probabilistes d'aide à la décision.* Presses de l'Université du Québec, Sillery (Québec), 1987.

[68] Francis Neelamkavil. *Computer Simulation and Modelling.* John Wiley & Sons, Chichester, England, 1987.

[69] Robert N. O'Boyle and Peter A. Koeller. The management of marine fisheries: An historical perspective. *BIO review '86*, 13–16, 1987.

[70] Peter H. Pearse. Turning the Tide, A New Policy for Canada's Pacific Fisheries. *Report of the Commission of Pacific Fisheries Policy*, 1982.

[71] Howard Raiffa. *Decision Analysis* Introductory Lectures on Choices under Uncertainty. *Behavioral Science: Quantitative Methods*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1968.

[72] William E. Ricker. Computation and Interpretation of Biological Statistics of Fish Population. *Bulletin of the Fisheries Research Board of Canada*, 235–240, 1975.

[73] William E. Ricker. Stock and recruitment. *Journal of the Fisheries Research Board of Canada*, 11:559–623, 1954.

[74] Denis Rivard. Biological production and production models. In Denis Rivard, editor, *Collected papers on stock assessment methods*, pages 44–110, Canadian Atlantic Fisheries Scientific Advisory Committee, 1988.

[75] Ginette Robert, G.S. Jamieson, and M.J. Lundy. Profile of the Canadian Offshore Scallop Fishery on Georges Bank. *CAFSAC Research Document 82/15*, 1982.

[76] D.L. Roddick and R.K. Mohn. Use of Age-Length Information in Scallop Assessments. *CAFSAC Research Document 85-37*, 1985.

[77] Sheldon Ross. *Introduction to Probability Models.* Academic Press, Inc., Orlando, Florida, 3rd edition, 1985.

[78] M.B. Schaefer. A Study of the Dynamics of the Fishery for Yellowfin Tuna in the Eastern Tropical Pacific Ocean. *Bulletin IATTC*, 2:247–285, 1957.

[79] Jon Schnute. A General Theory for Analysis of Catch and Effort Data. *Canadian Journal of Fisheries and Aquatic Science*, 42:414–429, 1985.

[80] A. Scott and P.A. Neher. The Public Regulation of Commercial Fisheries in Canada. *A study prepared for the Economic Council of Canada*, 1981.

[81] Ross Shotton. *Search and Decision in Fishing Systems.* Master's thesis, University of British Columbia, Vancouver (British Columbia), December 1973.

[82] M. Sinclair, Robert K. Mohn, Ginette Robert, and D.L. Roddick. Considerations for the Effective Management of Atlantic Scallops. *Canadian Technical Report of Fisheries and Aquatic Sciences - 1382*, 1985.

[83] R.D. Smallwood and E.J. Sondik. The Optimal Control of Partially Observable Markov Processes over a Finite Horizon. *Operations Research*, 1071–1088, 1971.

[84] Carl J. Walters. *Adaptive Management of Renewable Resources.* Macmillan Publishing Company, New York, New York, 1986.

[85] G.H. Winters. The development and utility of sequential population analysis in stock assessment. In Denis Rivard, editor, *Collected papers on stock assessment methods*, pages 111–134, Canadian Atlantic Fisheries Scientific Advisory Committee, 1988.