# A Workshop Concerning The Application Of Integrated Pest Management (IPM) To Sea Lamprey Control In The Great Lakes


G. R. Spangler
L. D. Jacobson (editors)

P. A. Cochran
B. A. Henderson
M. E. Holey
D. B. Jester Jr.
R. L. Johnson
J. F. Koonce
C. C. Krueger
C. K. Minns
A. J. Sawyer



Special Publication No. 85-2

Great Lakes Fishery Commission

1451 Green Road
Ann Arbor, Michigan 48105

July, 1985

The Great Lakes Fishery Commission was established by the Convention on Great Lakes Fisheries between Canada and the United States, which was ratified on October 11, 1955. It was organized in April 1956 and assumed its duties as set forth in the Convention on July 1, 1956. The Commission has two major responsibilities: first, develop coordinated programs of research in the Great Lakes and, on the basis of the findings, recommend measures which will permit the maximum sustained productivity of stocks of fish of common concern; second, formulate and implement a program to eradicate or minimize sea lamprey populations in the Great Lakes.

The Commission is also required to publish or authorize the publication of scientific or other information obtained in the performance of its duties. In fulfillment of this requirement the Commission publishes the Technical Report Series, intended for peer-reviewed scientific literature, and Special Publications, designed primarily for dissemination of reports produced by working committees of the Commission. Technical Reports are most suitable for either interdisciplinary review and synthesis papers of general interest to Great Lakes fisheries researchers, managers, and administrators or more narrowly focused material with special relevance to a single but important aspect of the Commission's program. Special Publications, being working documents, may evolve with the findings of and charges to a particular committee. Sponsorship of Technical Reports or Special Publications does not necessarily imply that the findings or conclusions contained therein are endorsed by the Commission.

## TABLE OF CONTENTS

# A Workshop Concerning The Application Of Integrated Pest Management (IPM) To Sea Lamprey Control In The Great Lakes

## INTRODUCTION

This report describes the results of an Adaptive Environmental Assessment (adaptive management) workshop conducted in Sault Ste. Marie, Michigan in August, 1982. The purpose of the workshop was to examine the life history of the sea lamprey, *Petromyzon marinus* , and its interaction with other Great Lakes fishes, in order to explore the extent to which the principles of integrated pest management might be brought to bear on the problem of lamprey control. There were three circumstances that set the stage and provided motivation for this endeavor. The first circumstance was a previous workshop concerned with salmonid-lamprey interactions in the Great Lakes. Second was Heimbuch and Youngs' analysis of a decision-theoretic approach to sea lamprey control. The third factor was a general concern about continued reliance on lampricides as the primary means of lamprey control.

The workshop was conducted in the conventional format for adaptive management modelling efforts in which a group of technical experts develop a simulation model that describes the problem of interest within boundaries defined during a previous "scoping" session. Parties to the scoping session were the principle agents concerned with sea lamprey and fishery management in the cold-water systems of the upper Great Lakes.

The primary product of the workshop is the greater collective understanding of how the biological system of fish production interfaces with the management agencies' goals and activities. Understanding arises from personal interaction among participants that is facilitated by the intensive mutual effort to develop a simulation model of the biological and management aspects of the system. One of the results of this effort is the simulation model itself, a computer program intended to reflect the perceptions of workshop participants as to how the system works.

This report documents the workshop and content of the simulation model developed at Sault Ste. Marie. It includes descriptions of submodels and basic functional relationships thought by the participants to prevail in nature as well as flow charts and descriptive prose detailing the order of computation, parameter estimation procedures, and assumptions underlying the model. The section immediately following this introduction is a brief summary of some underlying principles of integrated pest management that relate to decision making and lamprey control. *There are two dangers inherent in preserving and describing the model. First, observers who were not part of the workshop may conclude that the model, with all of its errors of fact and perception, is the most important product of the workshop. Second, naive individuals may place too much faith in the outcome of particular simulations.* In spite of these inevitable misunderstandings, the model may serve to stimulate further efforts to understand the lamprey control problem, and it may provide a starting point for more nearly "correct" models yet to be developed. It is in this spirit that this report is written.

The model was implemented as a combination of four submodels, each describing a critical component of the overall lamprey control problem. The submodels, in order of computation, are entitled: the Parasitic Phase Submodel, Lake Trout-Prey Submodel, Lamprey Spawning Submodel, and the Ammocoetes and Transformers Submodel. The complete model, written in Applesoft™ Basic, and the simulation control program (MICRO SIMCON, developed at the University of British Columbia) that governs its execution are available on disk (APPLE™ DOS 3.3) from the Great Lakes Fishery Commission.

Workshop participants defined initial conditions for the model such that the program would simulate the historical experience with lamprey control in the upper Great Lakes. This version of the model is entitled LAMPREY BASELINE (see a catalog of the disk for this file and other files named below). Additionally, two scenarios were developed that reflect the participants' concern with the imminent lamprey control treatment of the Nipigon River. The first, entitled NIPIGON TREATMENT.SAV, projects lamprey control in Lake Superior with treatment of the Nipigon. The second, entitled NO NIPIGON.SAV projects lamprey control without treatment of the Nipigon. Few other scenarios were considered on site because of the very long run-time (approx. 70 minutes) required to execute a single simulation.

In the two months following the workshop, the original model was modified by Dr. C. K. Minns (Canada Centre for Inland Waters) in order to reduce the execution time. Minns' simplified the spatial structure of the Ammocoetes and Transformers submodel so that the program runs a 20-year scenario in approximately 12 minutes. The faster version of the model is named MODEL-OCT1 5 and is available on the program disk. It is offered as an example of the kind of changes that might be made in order to explore model behaviour under a variety of conditions not pursued during the workshop. Differences between the original model and Dr. Minns' version are briefly described in the appendix.

This report is a compilation of the documentation written on-site by the designated rapporteurs and programmers for each of the modelling sub-groups. Each of the submodel reports contains references to line numbers and variable names that occur in the computer code. Because the documentation was written synchronously with model development, these line numbers and variables may differ slightly from those in the final model. The reader is cautioned to accept these entries in the documentation as indicative of the approximate region in the program where the variables and functional relationships are used. Names of people are also cited occasionally. These are either names of workshop participants or authorities known to them who provided information for the model.

Editing of this report was limited to whatever was necessary to mould the final document into a single format. The editors are responsible for any errors of omission detectable from comparison with the original draft documents printed at the workshop. Other errors, mis-statements, unreasonable parameter estimates and untenable functional relationships remain the shared responsibility of participants involved in the working subgroups.

George R. Spangler
Lawrence D. Jacobson
University of Minnesota
30 July 1985

## **DECISION-MAKING** AND INTEGRATION OF TECHNIQUES

*(Editor's note: This material was discussed on-site as a means of characterizing the objectives of the workshop.)*

Control tactics for the sea lamprey fall into two broad classes with respect to implementation and decision-making.

Certain techniques, once implemented, serve to permanently reduce the general "equilibrium abundance" of the sea lamprey and do not require continual monitoring and decision-making. These include the erection of barrier dams, the development and release of lamprey-resistant lake trout and the introduction of biological control agents. This does not mean that the initial decision to implement the control tactic is not subject to economic analysis.

In the case of barrier dams, for example, the key question would be: Do the expected benefits from constructing and operating a set of barriers exceed the costs of doing so, over the life-span of the dams? To answer this and similar questions, assumptions would have to be made concerning the state of the lake trout resource, lamprey population pressure and average levels of other, more dynamic management activities over this extended time period.

Total replacement of existing lake trout stocks with a single, more resistant strain represents a situation similar to building barrier dams. In actual practice, however, a more complex stocking program might require periodic, if not annual, decisions to be made regarding the benefits and costs of different stocking strategies (numbers of two or more strains, each with different properties, released in different places).

Although the introduction of biological control agents is not currently being pursued, historical examples from other disciplines suggest that the result of a successful introduction would be a general decline in lamprey abundance with no continuing costs (except, possibly, imposing constraints on the use of other control methods). A single decision based on cost/benefit analysis of the initial introduction is all that would be required.

A second class of control tactics would require periodic decisions to be made, based on ecosystem monitoring and continuous economic evaluation. This includes the use of TFM, release of sterile males, setting of annual stocking rates and regulation of the fishery.

In each case, an assessment of lamprey abundance in relation to the economic injury level (EIL) would be made, followed by a control decision. The effects of the more permanent control tactics discussed above automatically enter into the decision-making process enacted here. Each control tactic affects the lamprey-trout system in a different way and different information enters into the cost/benefit analysis.

For example, since TFM eliminates ammocoetes of four or more year classes, the appropriate question is: what ratio of lampreys entering the parasitic phase to lake trout encountering these lampreys (integrated over the next 4-6 years) will result in losses exceeding the cost of applying TFM this year? This is clearly not a simple question. It requires an assessment of lake trout and lamprey abundance, a 4-6 year projection of dynamics and an understanding of the economic costs and benefits expected.

The decision to release sterile males requires different information. The question is: what ratio of lampreys currently spawning to the number of susceptible lake trout expected after transformation of the offspring of these spawners would result in losses exceeding the costs of implementing a sterile male release? This involves more uncertainty as a projection is made farther into the future.

Obviously, true integration of techniques into an economically favorable strategy is an extremely complex problem. Realistic models of the dynamics of sea lampreys and their prey and of economic relationships are essential to the pursuit of integrated management of the sea lamprey.

## Integrated Pest Management Defined

**Integrated Pest Management (IPM)** seeks to manage a resource in such a way that populations of its pests are maintained below the economic injury level. Control techniques are selected from all those available and are combined in an ecologically and sociologically compatible manner.

A pest is a species which at times reduced the quantity or quality of the managed resource.

The ecological basis of IPM places pest control in the context of comprehensive management of the affected resource. Artificial pest control measures are integrated with and complement the natural factors regulating pest populations.

The economic injury level (EIL) is the lowest pest population that would inflict losses to the existing resource exceeding the cost of implementing a given set of pest control measures. The EIL, therefore, is specific to each combination of controls and resource states.

The optimal control strategy integrates pest management tactics so as to maximize net benefits (marginal revenues - marginal costs). Societal and environmental, as well as economic, measures of the costs and benefits of pest control are internalized (accounted for) to the extent possible. .

Alan J. Sawyer
Cornell University
23 August 1982

## PARASITIC PHASE SUBMODEL

M. E. Holey
Wisconsin Department of Natural Resources

J. F. Koonce
Department of Biology
Case Western Reserve University

Assumptions And Functional Relationships

The parasitic phase submodel describes the dynamics of adult lamprey feeding, growth, induced mortality on lake trout and wounding rates on lake trout. The dynamics of the parasitic phase are driven by size and number of prey available for feeding.

Transformers entering the lake are grown through two size classes in the submodel. Feeding starts at 40g body weight in the first size group and the weight of the second size class is computed from growth curves. Consequently, attacks on prey and lamprey growth are computed for both lamprey size classes. The time of feeding for the small size class of lamprey, which produce primarily partially lethal attacks, is one month. The time of feeding for the large size class of lamprey is 7.1 months and they produce primarily lethal attacks.

The number of attacks (A) by a lamprey on a prey type was calculated using a multispecies disc equation indexed by prey type:

$$A = T*a/(1+H*a*N) \quad \text{1/yr line 2082}$$

where:

a = the rate of effective search by a lamprey for a given prey type
H = the handling time by a lamprey for a prey type
T = the total time spent feeding
N = the abundance of a prey type.

The rate of effective search was determined from swimming speed (SS), reactive distance (R) and probability of attack (P) relationships between lamprey and their prey:

a = SS*R*P    line 2025

where:

SS = 7.884*L    km/yr

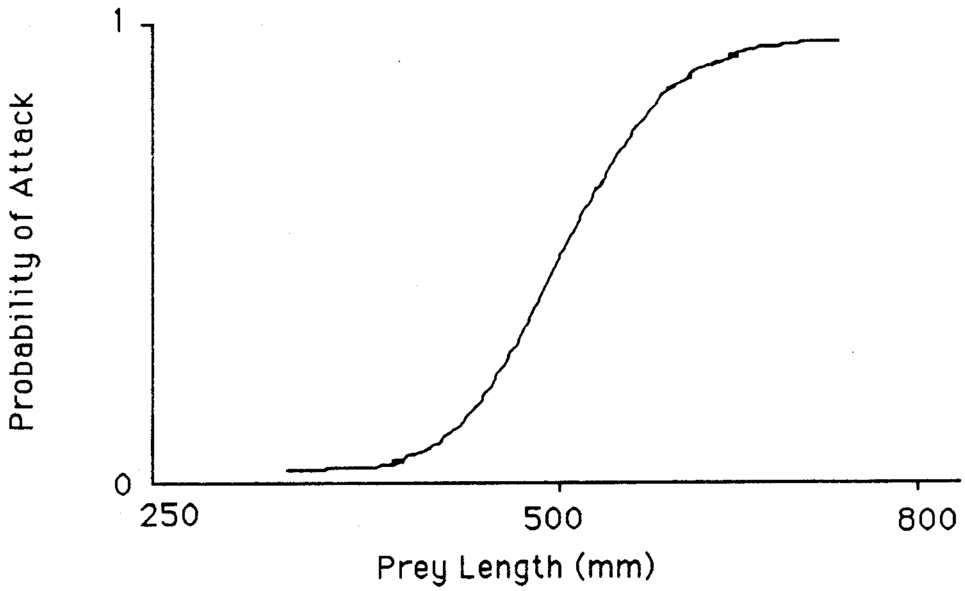
R = .0075E-3*L km line 202

P = oA*LAnl(BAn+LAn)

L = length of the prey (mm)

A,B and n = sigmoid curve coefficients.

The shape of the probability of attack curve is:




Handling time was determined by averaging the time spent between lethal and partially lethal attacks, indexed by lamprey size:

H = (HL*d)+(HP*e)    line 2050

where:

HL = handling time of a lethal attack
HP = handling time of a partially lethal attack
d = proportion of time spent attacking lethally
e = proportion of time spent attacking partially lethally.

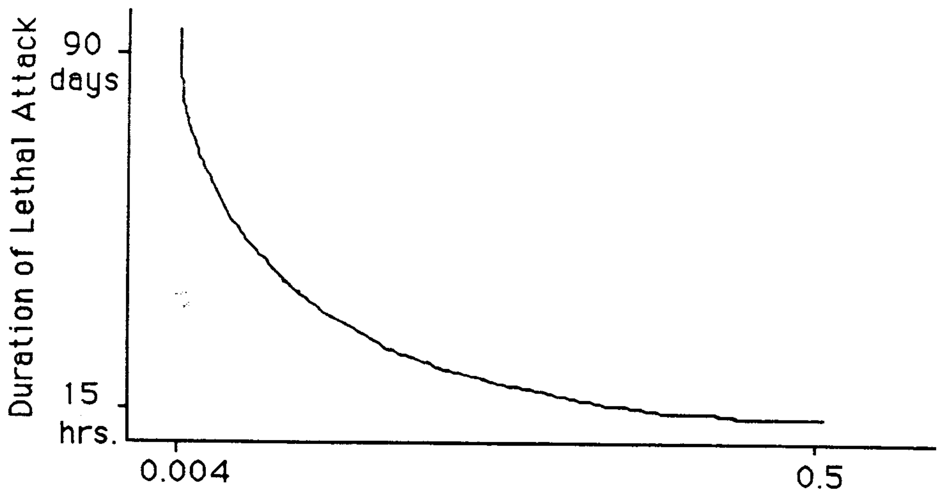
Handling time for the lethal attacks was determined from:

HL = ((SL/SP*f)^-1.1533)*2.6E-2 line 2050

where:

SL = weight of the lamprey
SP = weight of the prey
f = blood consumption
= 20% for small and 15% for large lamprey/body wt/day.

The lethal attack handling time curve is shaped as follows:

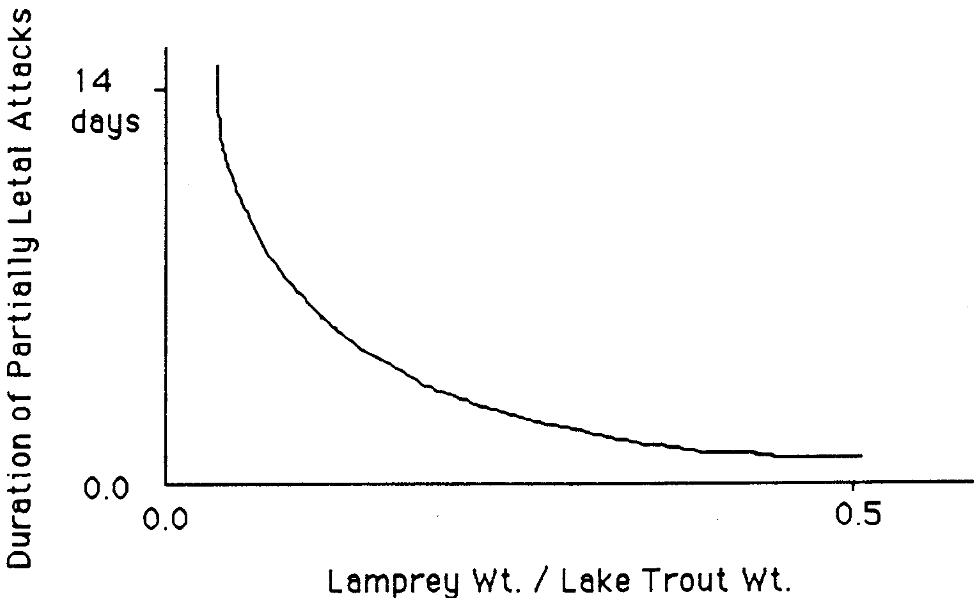



Lamprey Wt. / Lake Trout Wt.

Handling time for the partially lethal attacks was determined from:

HP = (SL/SP)“-1.15)*1.23E-3      line 2050

and graphically looks like:




Growth (G) of parasitic lamprey was formulated to be energetically related to lamprey food consumption (C), attack rate (A), handling time (H) and conversion efficiency:

C = A*SI*f*H     line 2082

and

G = (C*b*g)      line 2082

where:

b = blood energy content coefficient (= .625 Kitchell)
g = conversion efficiency = .3.

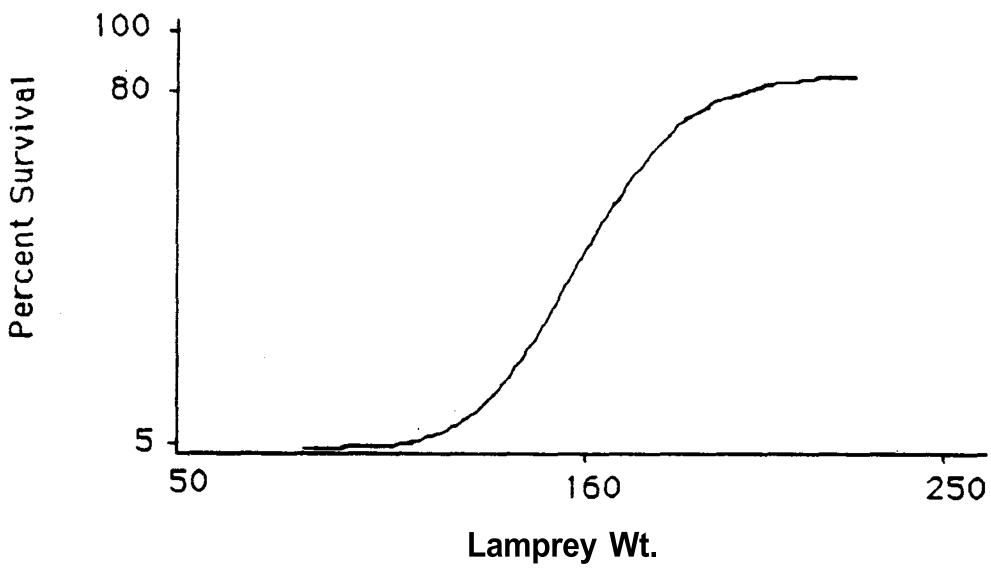
Upon entering the lake the parasitic phase lampreys grow to a maximum weight by around November and decrease in size by about *20%.*

Mortality rate (M) of the parasitic lamprey was determined as a function of lamprey body weight, assuming that when growth is good survival is high:

M = A*SL^n/(B^n+SL^n)      line 2200

where:

A,B and n are coefficients of the curve.




The observed wounding rates were formulated to approximate rates from spring assessment gear for a given number of attacks and prey density:

$$WR = A(d*HL+(1=d))*(PS*J+HP) \quad \text{line 2095}$$

where:

PS = probability of surviving a partially lethal attack
J = healing time on an A2 wound,
PS = a function of lamprey/prey body size:

$$PS = 1-(A(SL/SP)^2/(B^2+(SL/SP)^2)) \quad \text{line2090}$$

A and B are curve coefficients.

Lamprey induced instantaneous mortality (I) was indexed by prey type and calculated with the following equation:

$$I = N1*(1-(1-d)*PS)*A \quad \text{line 2090}$$

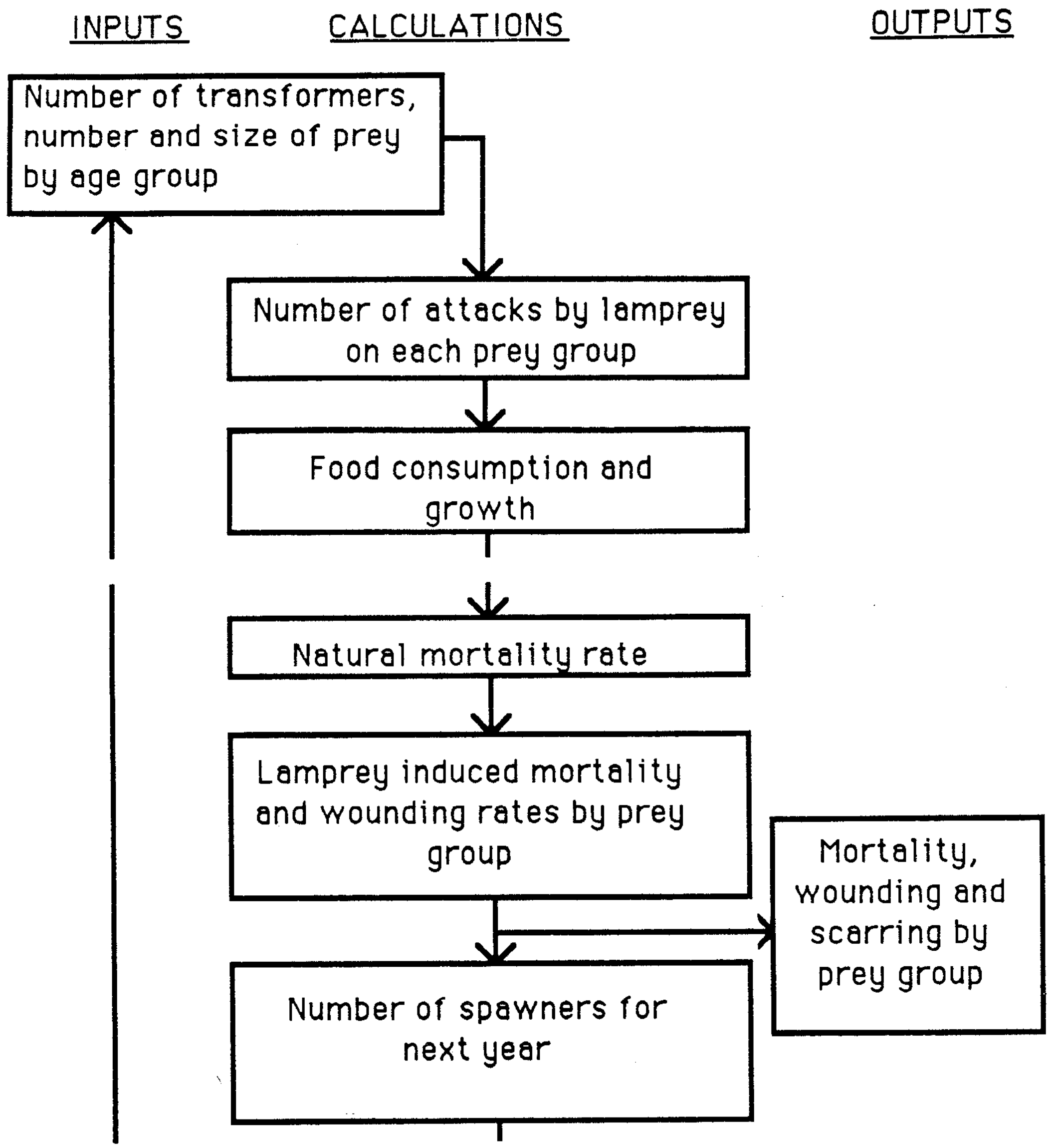
where:

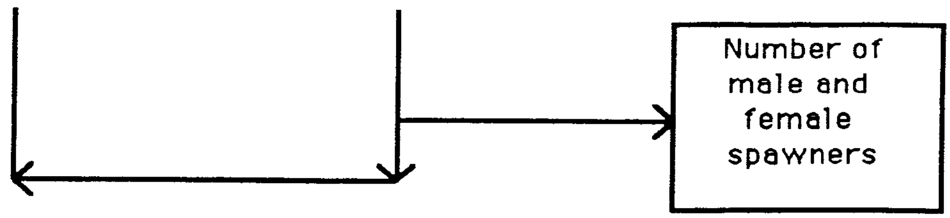
N1 = the number of lamprey.

For conversions from length to weight for lamprey and lake trout the following equations were used:

$$\text{Lamprey } w = 5*(l^2.75)$$
$$\text{Lake Trout } w = 2.3589E-6*(l^3.204)$$

Flow Chart For Parasitic Phase Submodel






Description Of Equations By Line Number

| Line Number | Description |
|---|---|
| 2000 | 1) Number of Male Spawning Lampreys: time t<br>2) Number of Female Spawning Lampreys: time t<br>3) Weight of a Spawning Lamprey: time t<br>4) Length of a Spawning Lamprey: time t<br>5) Total Number of Transformers Entering the Lake: time t-l and t |
| 2005 | 1) Begin Loop From Prey Type QU to QV<br>2,3,4) Set Arrays for Induced Mortality, Observed Wounding Rates and Number of Wounds/Prey to 0<br>5) Determine Length of Lake Trout From Weight at Each Age<br>6) Adjust Lake Trout Length for Prob. of Attack Equation<br>7) Set Adjusted Trout Length to 0 if Trout Length is Less Than the Threshold<br>8) Return to Beginning of the Loop if Adjusted Trout Length is Less Than 0 |
| 2020 | 1) Set Dummy Variable<br>2) Reactive Distance<br>3) If Reactive Distance is > 6m Set it at 6m |

| | |
|---|---|
| 2025 | 1) Prob. of Attack<br>2) Rate of Effective Search by Prey Type |
| 2030 | 1) Return to the Beginning of the Loop at 2005 and Start the Next Prey Type |
| 2035 | 1) Adjust Rate of Effective Search for Whitefish<br>2) Lamprey Weight at First Feeding |
| 2040 | 1) Begin Loop to Determine Attacks by Lamprey Size<br>2,3,4) Set Lamprey Growth, Attack Equation Denominator and Attack Rate to 0<br>5) Begin Loop by Prey Type<br>6) If Rate of Effective Search is 0 Go to Next Prey Type |
| 2050 | 1) Ratio of Lamprey WT to Lake Trout WT<br>2) Handling Time of Lethal Attacks by Prey Type<br>3) Handling Time of Partially Lethal Attacks by Prey Type<br>4) Sets Index<br>5) Percent of Attacks that are Lethal by Lamprey Size and Prey Type<br>6) Mean Handling Time by Prey Type<br>7) Attack Equation Denominator |
| 2070 | 1) Return to the Beginning of the Loop at 2040 (5) and Start the Next Prey Type |
| 2080 | 1) Begin Loop by Prey Type<br>2) If Rate of Effective Search is 0 Then go to the Next Prey Type |
| 2082 | 1) Attack Rate<br>2) Lamprey Growth |

| | |
|---|---|
| | 3) Ratio of Lamprey WT to Lake Trout WT |
| 2085 | 1,2,3) Adjusts Fraction of Lethal Attacks by Lamprey Size and Trout Resistance |
| 2090 | 1) Dummy Variable<br>2) Survival Prob. for Partially Lethal Attacks<br>3) Lamprey Induced Inst. Mortality Rate by Prey Type<br>4) Number of Wounds per Prey Type |
| 2095 | 1) Observed Wounding Rate by Prey Type |
| 2098 | 1) Return to the Beginning of the Loop at 2080 and Start the Next Prey Type |
| 2100 | 1) Lamprey Weight After Growth<br>2) Return to the Beginning of the Loop at 2040(1) and Start the Next Lamprey Type |
| 2200 | 1) Dummy Variable<br>2) Natural Mortality Rate of Lamprey<br>3) Percentage of Male Lampreys in the Population<br>4) Number of Spawners for the Next Year: time t+1 |
| 2210 | 1-5) Loads Indicator Variables for Graphics Display |

Variables Used In the Parasitic-Phase Submodel

| Name | Description | Value | Units |
|---|---|---|---|
| PA(20) | No. of Wounds Per Prey Type | Functional | no./pr |
| PB(20) | Lamprey Induced Inst. Mortality | Functional | rate/year |
| PC | Lake Trout Length/Weight Coef. | 2.3589e-3 | unitless |
| PD | Lake Trout Length/Weight Coef. | .3 | unitless |
| PE | Predator Swimming Coef. | 7.884 | km/yr/mm body length |
| PF | Length Corr. Factor for Attack Prob. | 300 | mm |
| PG | Reactive Distance Coef. | 7.5e-6 | ? |
| PH | Reactive Distance Functional | | m |
| PI | Prob. of Attack Coef. | 1 | unitless |
| PJ | Prob. of Attack Coef. | 250^PK | unitless |
| PK | Prob. of Attack Coef. | 2 | unitless |
| PL | Lamprey Length | Functional | mm |
| FM | Prob. of Attack/Dummy Variable | Functional | % |
| PN(20) | Rate of Eff. Search by Prey Type | Functional | km^2/yr |
| PQ | Lamprey Weight | Functional | Kg |
| PR | Lamprey WT to Lake Trout WT Ratio | Functional | unitless |
| PS(1) | No. of Spawners by Sex | Functional | unitless |
| PT(i) | Blood Consumption Coef. | PT(0)=.2, PT(1)=.3 | ? |
| PU | Lethal Attack Handling Time Coef. | 2.2e-5 | unitless |
| PV | Lethal Attack Handling Time Coef. | -1.533 | unitless |
| Pw | Partially Lethal Handling Time Coef. | 1.23e-3 | unitless |
| Px | Partially Lethal Handling Time Coef. | -1.15 | unitless |
| PY(20) | Lethal Attack Handling Time by Prey Type | Functional | Yr |
| PZ(20) | Partially Lethal Attack Handling Time by Prey Type | Functional | Yr |
| P1(1,1) | % of Lethal Attacks by Lake Trout Prey Type (initialized at .75 for all) | | unitless |

| | | | |
|---|---|---|---|
| P2(20) | Mean Handling Time by Prey Type | Functional | Yr |
| P3 | Disk Equation Denominator | Functional | unitless |
| P4(1) | Lamprey Feeding Time | P4(0)=0.06<br>P4(1)=0.30 | years |
| P5 | Growth Correction Coef. | 1 or large | unitless |
| P6 | Growth of Lamprey | Functional | Kg |
| P7 | Energy Content Coef. | .625 | unitless |
| P8 | Lamprey Feeding Efficiency | .3 | unitless |
| P9 | Lamprey WT | Functional | Kg |
| PO | Partial Mortality Coef. | 1.0 | unitless |
| QA | Partial Mortality Coef. | .25^2 | unitless |
| QB | Part. Lethal Attack Survival Prob. | Functional | unitless |
| QC | Observed Wounding Rate | Functional | no./prey |
| QD | Max Reactive Distance | 6 | m |
| QE | Max Lethal Attack Handling Time | .16 | years |
| QF | Dummy Variable (PR^2) | Functional | unitless |
| QG | Attack Rate | Functional | 1/yr |
| QH | Healing Time of A2 Wounds | .25 | years |
| QI | Lamprey Natural Mortality Rate | Functional | instantaneous |
| QI | Lamprey Natural Mortality Coef. | 1 | unitless |
| QK | Lamprey Natural Mortality Coef. | 160^4 | unitless |
| QL | Lamprey Male to Female Ratio | Functional | unitless |
| QM | No. of Spawners Next Year | Functional | unitless |
| QN | Lamprey % Weight Loss Prior to Spawning | ? | unitless |
| QP | Lamprey Length/Weight Coef. | 2.47e7 | unitless |
| QQ | Lamprey Length/Weight Coef. | .397 | unitless |
| QR | Lamprey Weight at First Feeding | .04 | Kg |
| QS | No. of Trans. Entering the Lake | Functional | ? |
| QU | Lower Lamprey Limit | ? | ? |
| QV | Upper Lamprey Limit | ? | ? |
| Q1 | Dummy Variable | None | unitless |
| Q2 | Preference of Whitefish to Lake Trout by Lamprey | 0.01 | % |
| Q3 | Post Transformation survival | 0.4 | unitless |

Z(1,TI) Observed Wounding Rate on Four Year Old Lake Trout-Ordinary Type

Z(2,TI) Observed Wounding Rate on Ten and Older Lake Trout-Ordinary Type

Z(3,TI) Observed Wounding Rate on Four Year Old Lake Trout-Resistant Type

Z(4,TI) Observed Wounding Rate on Ten and Older Lake Trout-Resistant Type

Z(5,TI) Lamprey Survival Rate

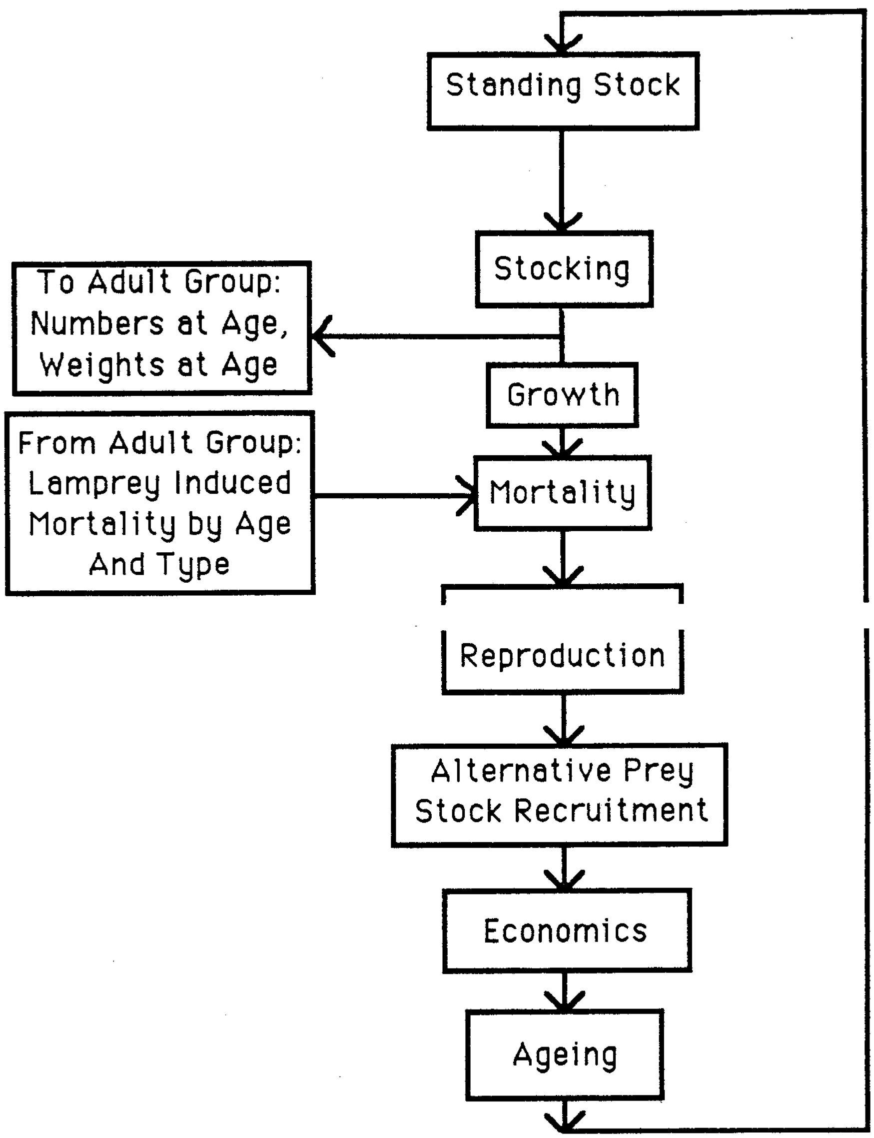
## LAKE TROUT - PREY SUBMODEL

R. L. Johnson
Department of Agricultural Economics
Michigan State University

B. A. Henderson
Ontario Ministry of Natural Resources

Introduction And Flow Chart

The lake trout - prey submodel provides a dynamic representation of the prey population that is vulnerable to lamprey attacks. The net economic benefits of the lake trout fishery are also approximated using very general relationships. The sequence of the model is as follows:


Standing Stock
Stocking
To Adult Group: Numbers at Age, Weights at Age
Growth
From Adult Group: Lamprey Induced Mortality by Age And Type
Mortality
Reproduction
Alternative Prey Stock Recruitment
Economics
Ageing

Submodel Description With Line Numbers

Standing Stock (line 3000)

There are 3 types of prey included in this submodel. There are natural ordinary lake trout, and there are stocked resistant lake trout. In addition, there is a buffer stock of alternative prey that lamprey will attack if lake trout are not available. The resistant character of the lake trout can come from 5 different sources, each of which is modelled in a separate scenario. The 5 options are:

1) A faster growing lake trout, i.e. one which spawns at an earlier age and therefore is exposed to lamprey for fewer years before spawning.

2) A lake trout that matures and spawns at a smaller size, i.e. before it becomes a target for lamprey.

3) A lake trout that can better survive lamprey attacks.

4) A lake trout whose habitat is in deeper water and therefore is not as available to the lamprey.

5) A lake trout with a higher survival rate from years O-l.

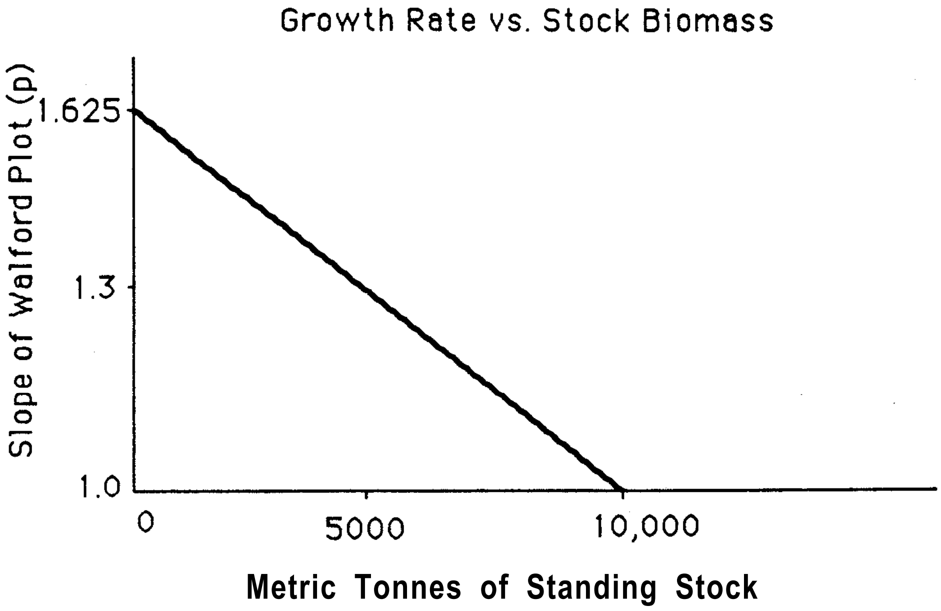
The total standing stock of the two types of lake trout was calculated by summing weight x numbers across the ten age categories for each type. The total standing stock for the entire lake trout population (both types) was then calculated by summation.

Stocking (line 3010)

The stocking decision is an “action” in this submodel and a constant number of yearlings to be stocked can be specified by the user. A level of 1000000 is used here as an initial value. A constant natural survival rate of .63 is then applied to the stocked fish from age 1 to 2, but after age 2 the natural survival rate is .75.

Growth (lines 3020 to 3040)

The rate of growth of lake trout was represented as being inversely related to the population stock size:




A linear. equation was used with an intercept of 1.41, slope of -.0000030, and maximum biomass of 10000 metric **tons. When standing** stock is greater than or equal to 10000 metric tons a new equation (intercept 2.64, slope -1.27E-4) was used. Values used to estimate this line were provided by Swanson.

To model a resistant type of lake trout with an increased rate of growth, the weight at each of the ten age classes was increased by a factor of 1.4.

The buffer species' growth was not modelled dynamically. An average length of this alternate prey of 450 mm was used to represent the buffer stock.

After growth calculations are made in this submodel, the numbers and weights of each of the types of prey are provided to the adult lamprey submodel. They in return provide this submodel with instantaneous mortality rates and wounding rates from lamprey attacks which are inputs to the next section.

Mortality And Scarring (lines 3070-3077.3098)

Wounding rates from adult lamprey are converted into total scars per fish in each age class:

$$scars_{i,j,t+1} = scars_{i,j-1,t} + wounds_{i,j,t+1}$$

where:

i = type of lake trout
j= age class of lake trout.

Mortality was modelled by assuming that natural, lamprey-induced and fishing mortality combined could not exceed .45 (finite rate) of the total -harvestable stock. This constant was decided upon by considering the level of mortality that could be sustained with lake trout rehabilitation still taking place. Natural mortality was described by a constant rate and lamprey-induced mortality rates were provided by the adult lamprey submodel. The fishing mortality was then the difference between the total and the natural and lamprey rates combined:

fishing = total - natural - lamprey

where all rates are instantaneous mortality rates.

The instantaneous rate of fishing mortality that is derived from this is applied to the individuals in the stock weighing more than 1.2 kg. (i.e. “knife-edge” vulnerability at the minimum size). The mortality rates affect the lake trout population in the following way:

$$pop_{i,j,t+1} = pop_{i,j-1,t} * \exp^{-(total)}$$

where:

i = type of lake trout
j = age class of lake trout
total = instantaneous mortality rate of .5978.

The total harvest is derived from the instantaneous rate of fishing mortality by:

$$h = \sum_{i=1}^{2} \sum_{j=1}^{10} ((fishing/total * .45) * pop_{i,j} * wt_{i,j}) / 1000$$

where h is harvest.

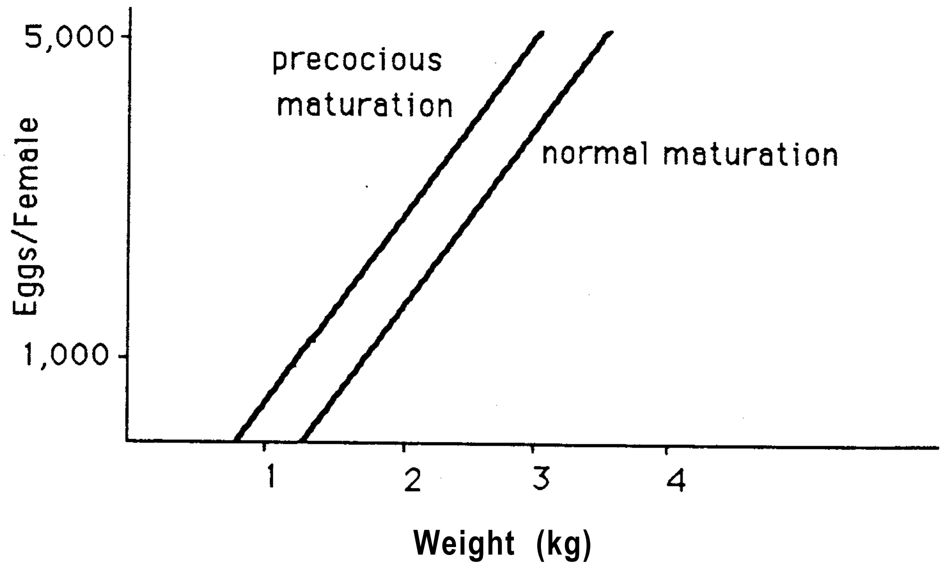
The mortality of the alternate prey is described by:

$$pop_{t+1} = pop_t * e^{-(lamprey + natural)}$$

where natural = .438.

<u>Reproduction (lines 3090 to 3095)</u>

Egg production was modelled as a function of the weight of the fish of each age class and each type. Maturation and fecundity equations were combined into one equation. At maximum maturity, it was assumed that only 80% of the mature stock spawned, because trout do not spawn each year (suggested by Swanson).




The equations for these lines are:

$$eggs_{i,j} = 2393 * kg_{i,j} - 2929 \quad \text{ordinary}$$
$$eggs_{i,j} = 2393 * kg_{i,j} - 1700 \quad \text{resistant (precocious).}$$

Total annual egg production for each type then, is:

$$eggs_i = \sum_{j=1}^{10} eggs_{i,j} * \text{(ave. \# of fish in jth age class)}$$
$$* \text{(ave. wt. in jth class)} * \text{(survival of 0-1 age class)}$$

The survival rate has an initial value of .0001, and this can be changed to simulate a different type of resistant lake trout.

Alternate Prey Stock Recruitment (line 3097)

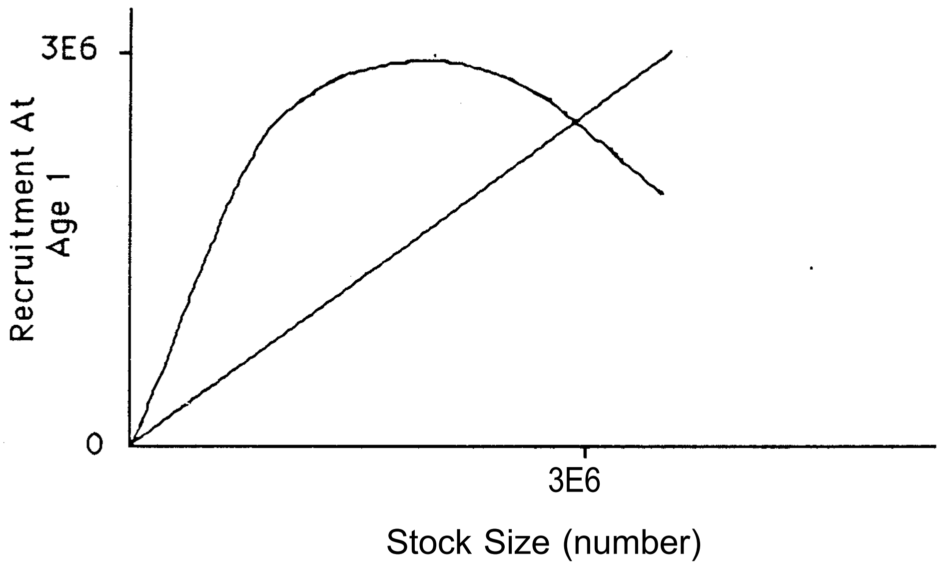
The alternate prey dynamics were modelled using a simple Ricker stock recruitment model:

$$\text{recruitment} = a(\text{pop} * \text{prop. spawners}) * e^{-b(\text{pop} * \text{prop. spawners})}$$

where:

$$a = 2.691$$
$$b = 3.3\ E\text{-}7.$$

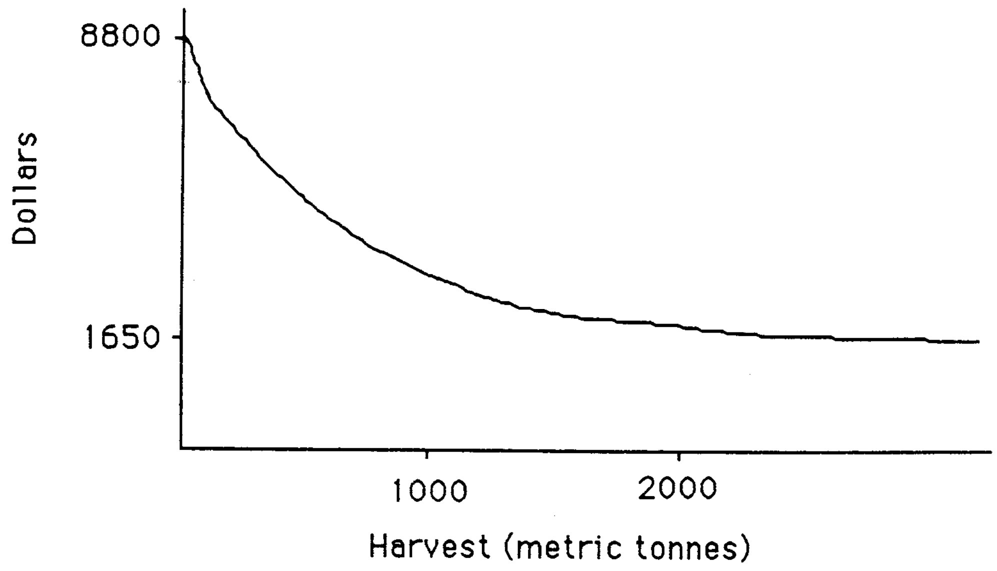
It was assumed that only .4 of the stock spawned.




Economics (lines 3150 to 3197)

While a complete analysis of the economics of the fishery could not be done within the scope of this model, some basic measurements of economic costs and benefits were calculated. In this submodel, we were concerned with the benefits and costs of the lake trout fishery only in Lake Superior. The costs of lamprey control were calculated by the other submodels.

The benefits of both the sport and commercial fishery were estimated by the following relationship:




This curve shows the diminishing returns per unit of harvest as harvest increases. The equation used to represent this curve is:

$$\text{Benefits} = a * \exp^{-b(\text{harvest})} + c$$

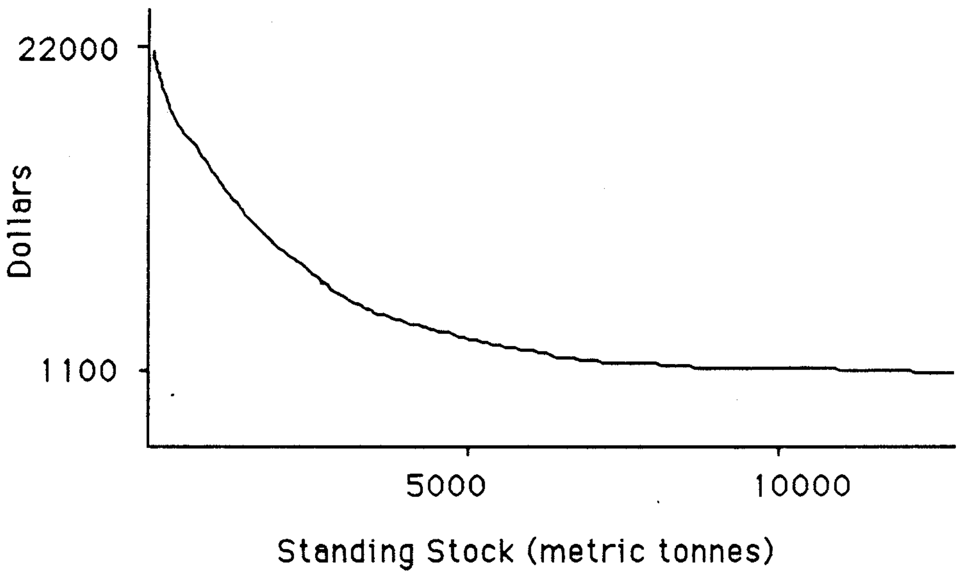
where:

a=7150
b = .003
c = 1 6 5 0 .

The range of values used to approximate this curve were provided by Chevalier, McCallum, Swanson and Talhelm.

To convert this measure of benefit/M.T. to total benefits, B was multiplied by the total harvest of the lake trout fishery.

The costs of the fishery included costs of fishing, costs of stocking, costs of other management activities (e.g. regulation, assessment, etc.), and a cost for endangering or extinction of the lake trout species. Costs of fishing are assumed to be a function of the level of standing stock of lake trout. If the standing stock is high, it takes less effort to catch a fish and the costs involved per unit caught should be low. If standing stock is low, costs per unit should be high and therefore the function should look like the following:




The same equation as for benefits, with different parameters, can be used to represent this curve:

$$\text{costs} = a * e^{-b(\text{stock})} + c$$

**where:**

a = 20900
b = .0004
c= 1100.

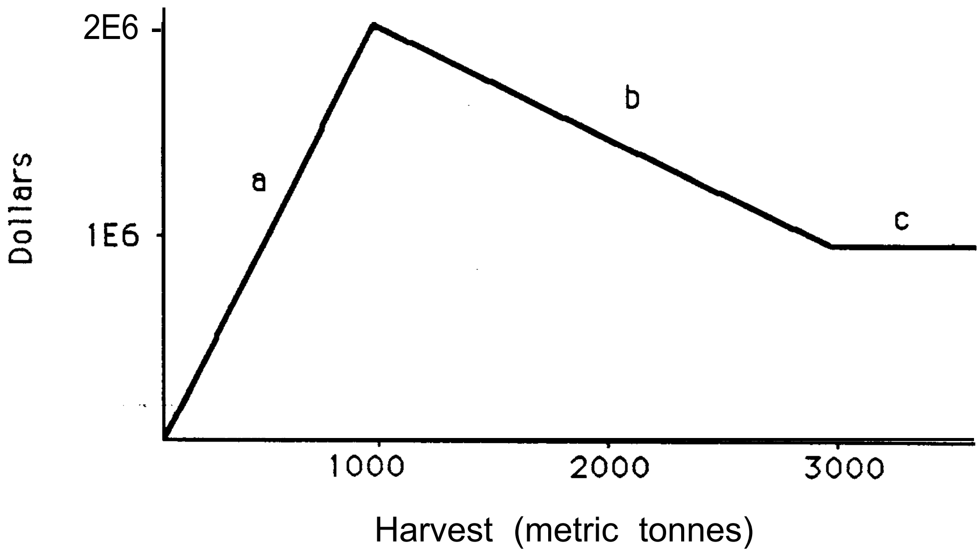
The range of values used to generate this curve were supplied by Talhelm.

The costs per unit from the above equation were multiplied by harvest to convert to total fishing costs.

Costs of stocking were assumed to be a constant value times the number of fish stocked:

costs = a * (number stocked).

The costs of managing the fishery, other than stocking costs and lamprey control costs, are hypothesized to be a function of harvest. Starting at zero harvest, management costs are likely to rise steeply as harvest grows, reach a peak at around 1000 M.T. of harvest, and then decline to a constant cost at high levels of harvest. Such a relationship was approximated by the series of linear segments below:

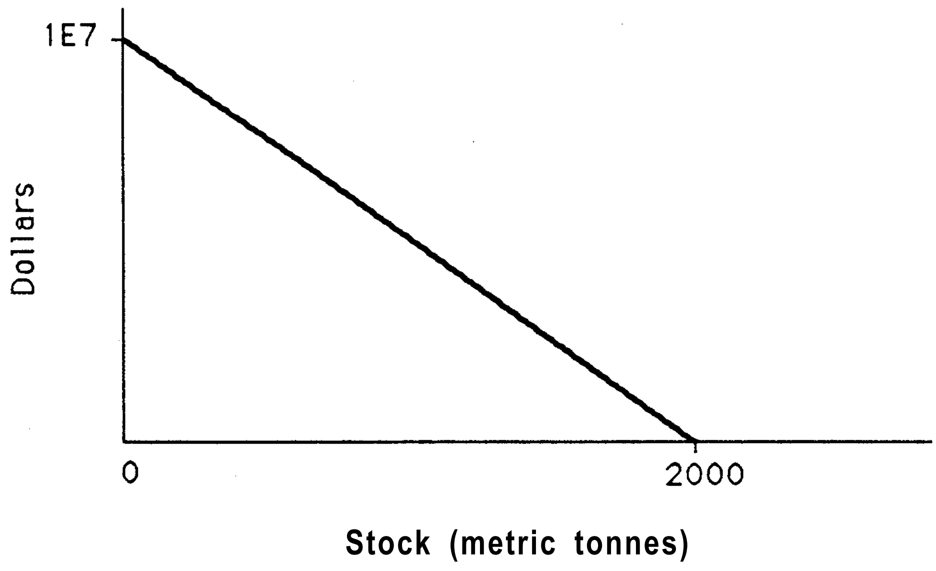


where:

slope of a = 2000
intercept of b = 2500000
slope of b = -500
level of c = 1000000
min. M.T. for b = 1000
min. M.T. for c = 3000.

These parameter values were suggested by H. Johnson.

The final cost to be considered is the cost to society if the lake trout species should become extinct. To incorporate this concept, the following relationship was used:




where the slope is 5. This relationship suggests that as lake trout become increasingly endangered, an increasing cost is suffered.

Aging (line 3200)

In this last section of the model, the weight and numbers at time t and age j are transformed to time t+1 and age j+1.

List Of Variables

| CODE | NAME |
|---|---|
| TN | Numbers at Age |
| TB | Total Biomass |
| TE | Total Annual Egg Production |
| TF | Egg Production by Weight of Fish |
| TH | Harvest |
| TK | Instantaneous Fishing Mortality Rate |
| TL | Lamprey Scars Accumulated |
| TP | Walford Plot Slope |
| TV | Biomass by Type |
| TW | Weight by Age Class |
| UC | Accumulator Function For Lake Trout |
| UH | Numbers of Each Type of Trout |
| UK | Counter |
| UL | Lamprey Mortality Rate on Each Type and Age |
| UP | Parent Stock |
| UQ | Proportion of Vulnerable Trout to Lamprey |
| UR | Recruitment |
| u z | Flag Counter |
| UI | Counter |
| VA | Net Benefits |
| v o | Cost of Endangering the Lake Trout Species |
| VR | Fisheries Management Costs |

List Of Constants

| Name | Description | Value | Units |
|---|---|---|---|
| TC | Slope of Egg Production Curve | 2393 | |
| T W | Intercept of Egg Production | | |
| | Curve - Normal | 2929 | |
| | - Precocious | 1700 | |
| TA | Natural Mortality Rate | .288 | instantaneous |
| TO | Egg Survival | .0001 | finite |
| TQ | Walford Plot Intercept | .1374 | |
| TR | Type 2 Growth Rate Correction | 1.4 | |
| TS | Stocking Level | 1e6 | numbers |
| TT | Stock Survival | .63 | finite |
| TU(I) | Minimum Size for Reproduction | | |
| | Normal | 1.22 | |
| | Precocious | .800 | kg |
| TY | Total Finite Rate of Mortality | .45 | |
| TZ | Total Instantaneous Rate | .5978 | |
| UA | Stock Recruitment Parameter | 2.691 | |
| UB | Stock Recruitment Parameter | 3.3 E-7 | |
| UD | Slope of Growth Curve | 6.25E-5 | |
| UE | Intercept of Growth Curve | 1.625 | |
| UF | Max. Biomass of Growth Curve | 10 E3 | M.T. |
| UJ | Initial Weight at Age 1 for Lake Trout | .023 | kg |
| UM | Minimum Size for Harvest | 1.2 | kg |
| UN | Initial Value for Number of | | |
| | Alternative Prey | 7.5 E6 | number |
| UO | Proportion of Population That Spawns | .4 | |
| US | Natural Mortality Rate for | | |
| | Alternate Prey | .438 | instantaneous |
| UT | Metric Tonne | 1000 | scalar |
| UJ | Million Dollars | 1 E6 | scalar |

| | | | |
|---|---|---|---|
| QL | Length of Alternate Prey | 450 | mm |
| VF | Min. of Total Benefit Curve | 1650 | $/M.T. |
| VG | Parameter of Benefit Curve | 7150 | |
| VH | Parameter of Benefit Curve | .003 | |
| VI | Minimum of Total Cost Curve | 1100 | $/M.T. |
| VJ | Parameter of Cost Curve | 20900 | |
| VK | Parameter of Cost Curve | .0004 | |
| VL | Constant for 2nd Segment of Endangered Cost Curve | 0 | $ |
| VM | Minimum Biomass Before Stock Becomes Endangered | 2000 | M.T. |
| VN | Stocking Cost per Fish | .25 | $/fish |
| VP | Slope of Cost Curve for Endangered Lake Trout | 5 | |
| VQ | Intercept of Cost Curve for Endangered Lake Trout | 1 E7 | $/M.T. |
| VS | Slope of 1st Segment of Management Cost Curve | 2000 | |
| VT | Intercept of 2nd Segment of Management Cost Curve | 2.5 E6 | |
| VU | Slope of 2nd Segment of Management Cost Curve | 500 | |
| VV | Constant for Management Cost Curve | 1 E6 | |
| VW | Harvest for Peak Costs | 1000 | M.T. |
| VX | Harvest for Constant Costs | 3000 | M.T. |
| VY | Discount Rate | .035 | |

# LAMPREY SPAWNING SUBMODEL

C. C. Krueger
Wisconsin Department of Natural Resources

C. K. Minns
Great Lakes Biolimnology Laboratory
Canada Centre for Inland Waters

## Summary

The spawning submodel describes the reproductive dynamics of lamprey from adult maturation through larval emergence and the effects of control measures intended to interrupt these processes.

## Input From Other Submodels

This submodel receives variable inputs from the Adult Submodel (the number, sex, and average female length of lamprey in a spawning run) and Ammocoete Submodel (the number of ammocoetes by stream type and the meters of stream chemically treated in the previous year) upon each iteration.

## Description

### Spawning Adult Allocation (lines 41054310)

Adult lampreys returning to spawn are allocated to stream types (small, medium, large, Nipigon) by equal weighting of the differences in stream discharge and ammocoete densities. The proportional allocation by stream type is calculated as follows:

$$\text{Proportion Allocated} = OS_{i}\,(\text{discharge/total discharge}) + 0.5 * (\text{larvae no./total}).$$

This proportion is then multiplied by the number of adults in the'spawning run:

No. Adults by Stream Type = Proportion Allocated * Adult Nos.

Adults are also allocated among streams within stream type on the basis of whether the streams are occupied or unoccupied by ammocoetes. The amount of unoccupied habitat that now becomes occupied due to allocation of spawning adults is calculated as:

Occupied Habitat = Proportion Habitat Changed to Occupied
* Amount Unoccupied.

The proportion of habitat changed is the ratio of spawner numbers to spawner numbers plus 25000. This rule causes 50% of the spawners to disperse to unoccupied habitat when spawner density is 25000 adults. The "occupied habitat" calculations are output to the Ammocoete Submodel and do not affect the reproduction dynamics in this submodel.

Barrier Control (lines 441 O-44501

The number of barriers to be constructed in each stream type and the period over which construction is to take place is set at the beginning of each scenario generation (lines 440-441). These inputs are checked in this segment of the submodel to determine whether barrier construction is to take place in a particular iteration. Barriers may not be constructed on small streams or the Nipigon River. The number of barriers allowed on medium and large rivers is constrained to 23 and 12, respectively (estimated by Mcdonald, Purvis, and Manion). The reduction of ammocoete habitat caused by barrier construction in this model uses data from barriers on Lake Superior streams. If barrier construction occurs within an iteration then the amount of ammocoete habitat transfered from occupied to unoccupied is calculated as:

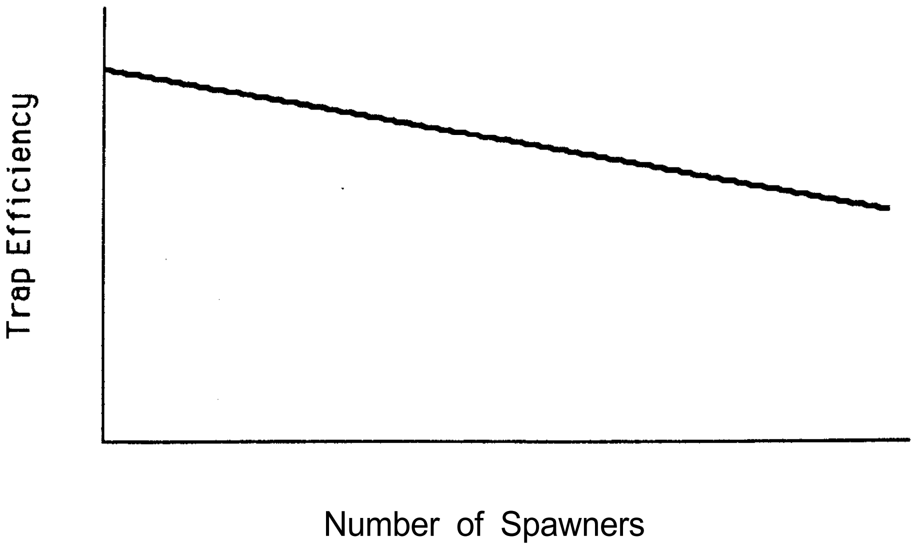
Occupied Habitat Changed to Unoccupied = # Barriers Constructed per Year * Average Stream Length Removed by a Barrier.

After a barrier is constructed the sum of the total number of barriers constructed within the simulation is updated. The costs of barrier construction are based on the average barrier cost in the Lake Superior basin (actual and proposed). These costs were then averaged over 25 years to develop a cost per barrier per year ($2240 medium, $18000 large; Manion, Purvis, and McDonald),

Barrier Costs = Number of Barriers * Cost per Barrier

Lamprey Trap Operation (lines 451 O-45451

The operation of lamprey traps at barriers is input at line 451 by stream type. If traps are to be operated then traps are operated on all barriers within a stream type. Traps may not be operated where no barriers exist. The efficiency of the traps was set at 0.5 of the spawning run in a stream. The efficiency of traps in capturing the total spawning run is a function of the number of spawners relative to the estimated numbers present in the 1950's. If the number of barriers is held constant then the trap efficiency in capturing the total spawning run is inversely related to the number of spawners (see Figure).




The equation used to define this trap efficiency was as follows:

Trap Efficiency = 0.5 * Number of Barriers / (Number 1982 Streams + (Number of Spawners / High Density Number Streams Unoccupied)).

A high density lamprey population was set at 500,000 adults (line 450). The number of males captured by traps is calculated as:

Number Lamprey Trapped = Trap Efficiency* Total Number in Spawning Run.

The numbers trapped are then subtracted from the numbers originally allocated in lines 4105-4310. The costs of trapping are calculated as:

Trapping Costs per Year = Number of Traps * Operating Costs per Trap.

The operating cost per trap is set at $4500 per year based on Canadian and US costs observed in 1982.

Sterile Male Program (lines 461 O-4645)

The year of implementation of a sterile male program is input at line 460.

The allocation of sterile males into streams is input at line 460 also and provides the options of no sterile males, medium streams only, large streams only, or the spawning adult allocation rules lines 4105-4310. The source of sterile males is also specified in line 460 as being either from Lake Superior or external. If an external source is specified, the number desired must be stored at line 460 also. If Lake Superior is specified as the source of males then the number of males captured by traps in the current iteration is used as the number released. When the allocation rule is specified sterile males are allocated among all stream types according to the proportions calculated in the current iteration.

Costs for this program when males from Lake Superior are used equals the overhead costs per year of $45,000 (Seelye and Hanson). Costs for this program when males from an external source are used equals:

Cost per Year (external males) = Number of Sterile Males Released* Trapping Costs per Male + Overhead Costs.

The overhead costs in this equation are the same as specified before. The trapping costs per male were calculated as $0.35 based on 1982 data (Purvis and Manion).

Emergent Larvae Calculations (line 4710)

The numbers of emergent larvae by stream type (small, medium, large, Nipigon) that result from the spawning run are calculated based on numbers of eggs, the egg to larvae ratio, and the proportion of sterile males in the population. The fecundity per female is determined from a linear regression of number eggs vs. female length (data from Manion 1972

TAFS pages 718-720). The regression coefficients are stored in line 470. The survival of eggs to larvae is set at 0.03 (Hanson and Manion) in line 470. The number of emergent larvae are calculated as:

Number of Emergent Larvae = 0.03 * Number of Eggs per Female
* Proportion of Successful Matings.

The proportion of successful matings is 1 when no sterile males are introduced. The proportion declines as the number of sterile males released increases. This proportion is calculated as:

Proportion of Successful Matings = Number of Fertile Males
/ (Number of Fertile Males + Number of Sterile Males).

Output To Other Submodels

The outputs of this spawning submodel are transfered to the Ammocoete Submodel. These outputs are:

1. The number of emergent larvae by stream type.

2. The amount of stream habitat removed due to barrier construction.

3. The amount of stream habitat now occupied by ammocoetes.

4. The amount of stream habitat now unoccupied by ammocoetes.

5. The amount of stream habitat transferred from unoccupied to occupied due to the allocation of spawning lamprey.

Output For Indicator Variables

The total costs (millions of dollars) of barriers, traps, and the sterile male program by year are sent to the control cost equation (line 6010).

Flow Diagram For Spawning Submodel

**INPUT**

**Variables from other submodels**

**1. Adult Submodel**

**a. Number by sex of spawning run**

**b. Average female length (mm)**

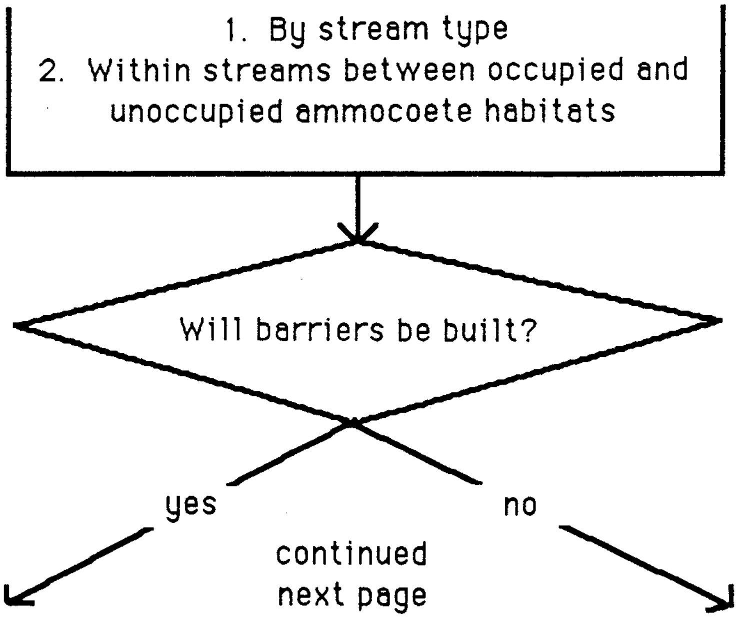
**2. Ammocoete submodel**

**a. Number ammocoetes by stream**

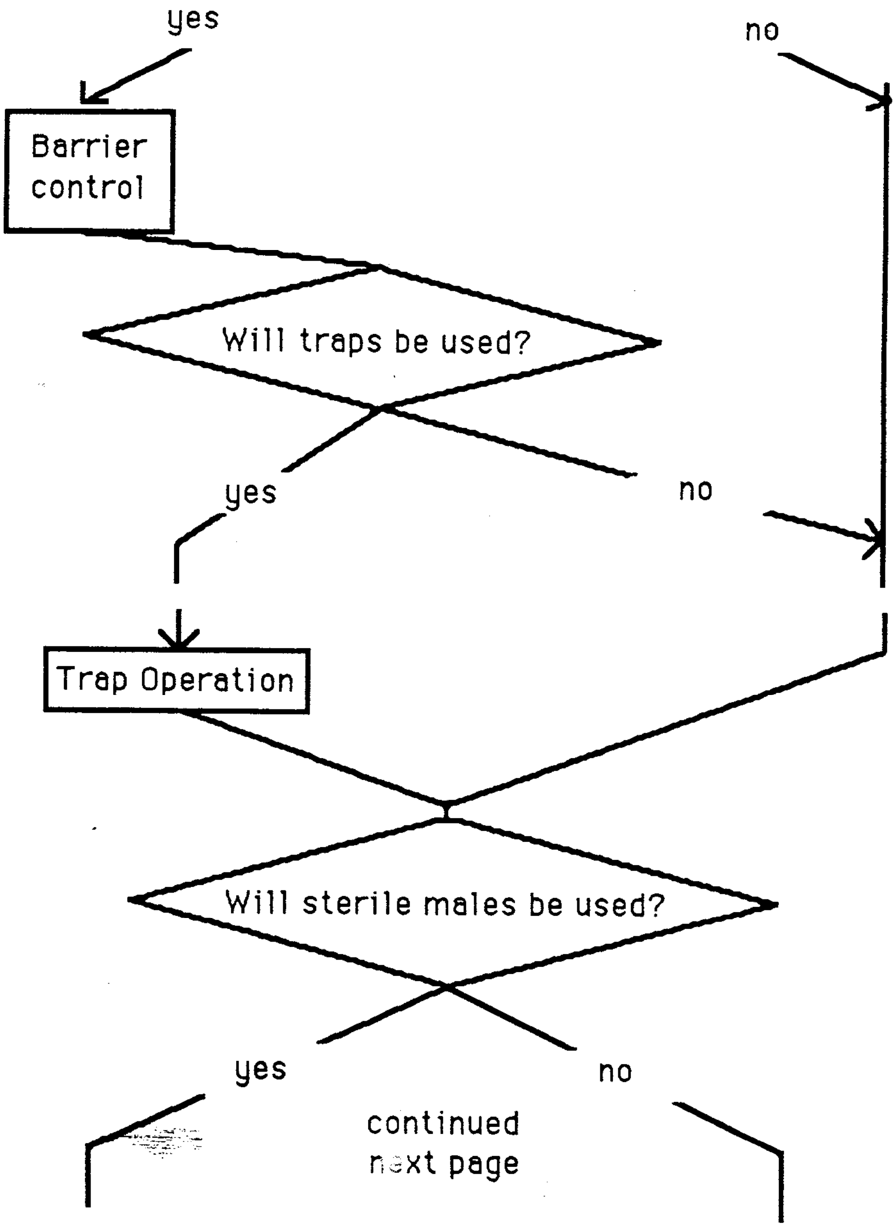
**b. Meters of stream chemically treated in previous years (unoccupied habitat).**

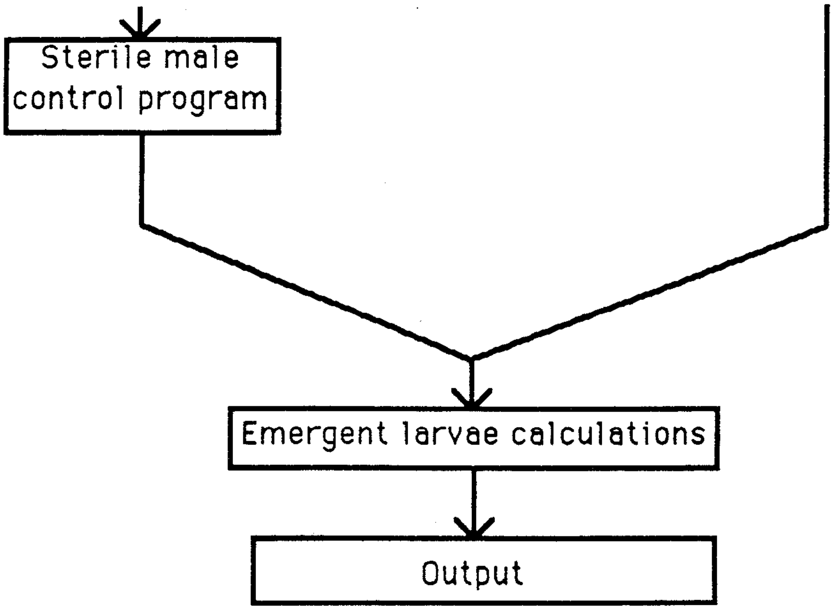
↓

**SPAWNING ADULT ALLOCATION**




continued next page


yes
no
Barrier control
Will traps be used?
yes
no
Trap Operation
Will sterile males be used?
yes
no
continued next page


Sterile male control program
Emergent larvae calculations
Output

Variables for the spawning adult submodel.

| Code | Identification |
| --- | --- |
| EC | Barrier cost in a year (dollars). |
| ED(1) | Amortized construction cost per barrier per year by stream type ($2240 medium, $18000 large). |
| EE | Cost of trapping at a barrier site per year ($4500). |
| EF | Trap cost in a year (dollars). |
| EN(3) | Emergent larvae by stream type in numbers. |
| EP | Input of year that sterile male program is started. |
| EQ | Cost per sterile male ($0.35 per lamprey). |
| ER | Overhead costs of the sterile male program ($45,000 per year). |
| Es | Sterile male control costs in a year. |
| ET(2) | Input of the presence or absence of traps on medium rivers, large rivers, or the Nipigon River. |
| EU(2) | Proportion of lamprey spawning run captured in traps by stream type (0.5). |
| FA(1,2) | Input by stream type of the number of barriers, year construction begins, and year construction ends. |
| FE | Number of barriers constructed per year. |

| | |
|---|---|
| FM(3) | Current number of barriers by stream type. |
| Fp(3) | Proportion of adults allocated by stream type. |
| FQ | Proportion that weights the effect of stream discharge and ammocoete density on adult allocation into streams (0.5). |
| FS(3) | Number of sterile males released by stream type. |
| GB(4,2) | Ammocoete habitat in stream length (meters) that is occupied, unoccupied, and removed by barrier construction in a year by stream type (only medium or large streams). |
| GC(3) | Number of lamprey producing streams in 1982 by small, medium, large, and Nipigon (Purvis and McDonald). |
| GD | Adult density where 50% of spawning adults are allocated to stream habitat unoccupied by ammocoetes (25000). |
| GE | Proportion of stream habitat unoccupied by ammocoetes that is changed to occupied by adult allocation rule. |
| GH | Definition of high adult density (500,000 adults). |
| GI | Temporary variable - proportion of lamprey removed by traps. |
| GM | Total number of males captured by traps in medium rivers, large rivers, and the Nipigon River. |

| | |
|---|---|
| GN(3,1) | Number of adults allocated by stream type by sex. |
| GO(3,5) | 1950's stream data matrix by stream type. Data are stream numbers, cfs, length, and width. Also stored are the maximum number of barriers and the average stream length in meters that are removed by barriers (McDonald and Purvis). |
| Gp | Sum of discharges (cfs) of all known lamprey producing streams (1950's data; Purvis and McDonald). |
| Gs | Input - identification of sterile male allocation. Options are: no sterile males, medium streams only, large streams only, Nipigon only, and the adult spawner distribution rule. |
| GT | Input - identification of source of males for sterile male program. If source is external then the number of males is stored here. |
| GU | Number of sterile males used in a year (either GN or GT). |
| GV | Sum of the ammocoete numbers among stream types. |
| GX | Regression intercept coefficient of number of eggs vs. female lamprey length (mm). (12107) |
| GY | Regression slope coefficient of number of eggs vs. female lamprey length (mm). (205.6) |
| GZ | Proportion of eggs that result in emergent larvae (0.03; Manion and Hanson). |
| Z(11,TI) | Percent of females in spawning run. |

Z(12,TI) Length (meters) of stream habitat occupied by ammocoetes.

Z(13,TI) Total number of lamprey spawners.

Z(14,TI) Average length (mm) of adult female lamprey.

Z(15,TI) Total control costs of all methods.

Description Of Spawning Submodel Line Numbers

| Line Number | Description |
|---|---|
| 400-498 | Initial conditions for spawning submodel. |
| 400-442 | Input of barrier control measures and costs. |
| 450-451 | Input of trapping control parameters and costs. |
| 455460 | Input sterile male program parameters and costs. |
| 470 | Reproductive dynamics specified and sum of discharges of all streams is entered. |
| 490-492 | 1950's stream data matrix by stream type is entered. |
| 493-494 | Number of lamprey producing streams in 1982 is entered. |
| 497-498 | Initial setting of the amount of ammocoete habitat by stream type that is occupied, unoccupied and removed by barrier construction. |
| 4105-4310 | Spawning adult allocation. |
| 4105-4110 | Calculation of the proportion of adults allocated by stream type. |
| 4210-4220 | Allocation of adults within stream type. |
| 4310 | Allocation of adults by stream type. |
| 4410-4450 | Barrier control. |
| 4410-4425 | Determination of whether barriers are to be constructed. |

Number of barriers to be constructed in current iteration is stored.

4425-4435 Amount of stream habitat removed due to barriers is determined.

4440-4450 Calculation of the barrier cost in the year and current number of barriers by stream type is determined.

451 O-4545 Lamprey trap operation.

451 O-4525 Calculation of proportion of adult lamprey removed by traps in a year.

4530-4545 Numbers trapped and costs of trapping determined.

461 O-4645 Sterile male program.

4610-4620 Determination of when to start program and how to distribute sterile males.

4625 Calculation of the number released and program costs if male source is from Lake Superior.

4630 Calculation of the number released and program costs if male source is external.

4635 Distribution of sterile males into medium or large streams.

4640-4645 Distribution of sterile males by spawning adult allocation rule.

4710 Emergent larvae calculations.

491 O-491 5 Output of indicator variables.

## AMMOCOETES AND TRANSFORMERS SUBMODEL

P. A. Cochran
Limnology Laboratory
University of Wisconsin, Madison

D. B. Jester Jr.
Michigan Department of Natural Resources

### Introduction

This submodel describes the dynamics of an age-structured population of ammocoetes which can be distributed among a number of habitat types and to which chemical treatments can be applied. The habitat types are small, medium, and large streams, the Nipigon River, and offshore lentic regions. Each habitat type is distributed among categories based on the number of years since chemical treatment.

### Sequence Of Calculations

The sequence of calculations during each time step follows the natural chronological order of events in the stream:

1. Apply natural mortality to last year's population, then age ammocoetes by one year. A proportion of the individuals removed through "natural mortality" are assumed to enter the lentic population. Natural mortality of age 0 and I ammocoetes is density dependent. During this step a year is added to the duration of time since last chemical treatment.

2. Emergent larvae (numbers provided by the Lamprey Spawning Submodel) are placed into age group 0.

3. Transformation to the adult stage takes place and sex ratios are established. A constant proportion of ammocoetes age V and older transform. The proportion of age IV's that transform, and the sex

ratio of all transformers, is density dependent.

4. Chemical treatment may be applied here, with an effect on both ammocoetes and transformers.

5. Numbers of surviving ammocoetes are supplied to the Lamprey Spawning submodel. Numbers of transformers by season and sex are supplied to the Parasitic Phase Submodel.

6. Proceed to next yearly increment.

General Description Of Calculations

Population Dynamics (lines 5000-5320)

The following age-structured population dynamics are incorporated into the submodel:

| Age | Survival | Transformation |
|---|---|---|
| 0 | -.006*(density)+.28 | 0 |
| I | -.013*(density)+.56 | 0 |
| II | .7 | 0 |
| III | .7 | 0 |
| IV | .7 | -.01 *(density)+. 1 |
| v+ | .7 | .1*(.02 in the lake) |

The proportion of transformers that are female is given by the relationship:

Proportion female = -.034*(density)+.67.

One-half of the ammocoetes removed from the stream due to "natural mortality" are assumed to enter the lentic ammocoete population, but the lentic population is not allowed to exceed a carrying capacity of 10/m$^2$. Summary data on Lake Superior streams (all streams which have ever been chemically treated plus the Nipigon River) were provided by Dustin, Rugen, and Schuldt. Areas computed from these numbers provided a basis for calculating ammocoete densities.

| Size | CFS | Total miles | Mean width |
|---|---|---|---|
| small | <10 | 59 | 3m |
| medium | 10-100 | 512 | 11 m |
| large | >100 | 825 | 31 m |
| Nipigon | 3800 | 12 | 60 m |

Chemical Treatment (lines 5550-5904)

Chemical treatment policy can be set externally to the model. The chemical dosage (% of minimum lethal dose) and bank length (exposure time) can be varied, as well as the minimum treatment threshold concentration of detectable transformers and the total cost budgeted for treatment. The relationships of mortality to chemical dosage and bank time were adapted from Heimbuch and Youngs (1982, report to the GLFC). Detection and treatment are assumed to increase in effectiveness with decreasing stream size. The Nipigon River and lentic populations are not treated.

Alternate chemical control policies can be specified and included in the model using the EXEC command.

Description By Line Number

| Line Number | Description |
|---|---|
| 510-555 | Setting initial conditions and parameter values |
| 512 | Initial lamprey densities by age and habitat type |
| 514 | Total area of each habitat type |
| 515 | Survival parameters |
| 516 | Transformation and sex ratio |
| 517 | Lentic carrying capacity; transformer and total ammocoete numbers |
| 518 | Proportion of males |
| 520-555 | Chemical treatment parameters |
| 520 | Proportion chemical bank length |
| 521 | Dosage |
| 522-525 | Calculates chemical/m treated, cost/$m^2$ treated, proportion killed |
| 555 | Threshold for transferring low density occupied habitat to unoccupied; treatment threshold detection densities |
| 5005 | Update area of newly occupied habitat |
| 5100-5160 | Calculate total ammocoete densities; update time since last treatment |
| 5170 | Update habitat areas |
| 5220 | Density dependent survival for age 0 and I |
| 5230 | Transfer of ammocoetes to lentic population |
| 5240 | Keeps lentic population under carrying capacity |
| 5250 | Application of survival rates |
| 5270 | Ages ammocoetes |
| 5280 | Assignment of emergent larvae to age 0 in non-lentic habitats |
| 5290 | Transformation (density dependent for age IV) |
| 5300-5310 | Establishment of transformer sex ratio |
| 5550 | Calculation of prey habitat occupied in densities greater than minimum treatment threshold |
| 5620 | Calculation of chemical cost, amount of chemical used, and total number of transformers killed |

5 6 3 0 Chemical induced mortality of ammocoetes and transformers
5640 Transfer of treated habitat
5645 Removal of dammed habitat from further consideration (chemically treated first)
5650 Reassignment of occupied to unoccupied habitat
5903-5904 Calculation of total transformers by season and sex
591 O-591 3 Variables to be plotted

Variable Names And Descriptions

| Variable Name | Description and Values | Units |
|---|---|---|
| I | Subscript for years since chemical treatment | |
| J | Subscript for age (5=V and up; 6,7=male and female transformers) | |
| K | Subscript for habitat (0-2=small-large; 3=Nipigon; 4=lentic) | |
| AA(I,K) | total ammocoete density, age I and up | $\#/m^2$ |
| AB(I,K) | habitat area | $m^2$ |
| AC(K) | upper loop limit | |
| AD(I,J,K) | ammocoete density | $\#/m^2$ |
| AE(J) | slope of survival line (-.006 - age 0 -.013 - age I) | |
| AF(J) | survival line intercept (.28 - age 0 - .56 - age I) | |
| AI | loop limit (I-I) | |
| AK | lentic carrying capacity (10) | $\#/m^2$ |
| AM(J) | proportion transforming (.1 in stream, .02 in lake) | |
| AN(K,0) | ammocoete number | |
| AP | constant (.1) | |
| AQ | proportion spring transformers (.5) | |
| AS | survival | |
| Ax | slope for density dependent proportion female transformers (-.034) | |
| AY | intercept (.67) | |
| BN(4) | number of transformers by sex and season (0=fall male;1 =fall female; 3=spring male; 4=spring female) | |
| CB(K) | chemical bank length (proportion) | |
| c c | chemical control cost | dollars |
| CD(K) | chemical dosage (proportion of minimum lethal dose) | |
| CH(I,K) | habitat area treated | $m^2$ |
| CK(K) | proportion killed by chemical | |
| CL(K) | minimum transformer density detection threshold | |

| | | |
|---|---|---|
| | for treatment | #/$m^2$ |
| CM(K) | treatment cost/ $m^2$ | \$/$m^2$ |
| CT(K) | chemical used/stream length | pounds/m |
| CU | total chemical used | pounds |
| CX | threshold ammocoete density for "unoccupied" habitat | #/$m^2$ |
| CZ | total number transformers killed | |
| Z16 | transformers/area | #/$m^2$ |
| Z17 | transformers killed/dollar | |
| Z18 | chemical treatment cost | dollars |
| z19 | amount chemical used | pounds |

## APPENDIX A: PARTICIPANTS

ADDIS, James - Bureau of Fish Management, Wisconsin Department of Natural Resources, P. 0. Box 7921, Madison, WI 53707, (608) 266-7025

BRAEM, Robert - Marquette Biological Station, U. S. Fish and Wildlife Service, 446 E. Crescent St., Marquette, MI 49855-3698, (906) 226-6571

CHEVALIER, John - Ontario Ministry of Natural Resources, 199 Larch St., Sudbury, ONT, Canada P3E 5P9, (705) 675-4135

COCHRAN, Phil - Limnology Laboratory, University of Wisconsin, Madison, WI 53706, (608) 262-3087

COLLINS, John - Ontario Ministry of Natural Resources, South Bay Fisheries Research Station, R. R. 1, Tehkummah, ONT, Canada POP 2C0, (705) 859-3 137

DUSTIN, Stan - Sea Lamprey Control Centre, Department of Fisheries and Oceans, Huron St., Ship Canal P. O., Sault Ste. Marie, ONT, Canada P6A 1P0, (705) 949-l 102

ESHENRODER, Randy - Great Lakes Fishery Commission, 1451 Green Road, Ann Arbor, MI 48105 , (313) 662-3209

FARMER, Gil - Department of Fisheries and Oceans, Resource Branch, P. 0. Box 550, Halifax, Nova Scotia, Canada P3J 2S7, (902) 426-7819

GRIMA, Lino - Institute for Environmental Studies, University of Toronto, Toronto, ONT, Canada M5S 1A4, (416) 978-3486

HANSON, Lee - U. S. Fish and Wildlife Service, Hammond Bay Biological Station, RFD, Box 263, Millersburg, MI 49759, (517) 734-2511

HEIMBUCH, Doug - Department of Natural Resources, Cornell University, Fernow Hall, Ithaca, NY 14853, (607) 256-5470

HENDERSON, Bryan - Ontario Ministry of Natural Resources, South Bay Fisheries Research Station, R. R. 1, Tehkummah, ONT, Canada POP 2C0, (705) 859-3137

HEWETT, Steve - Limnology Laboratory, University of Wisconsin, Madison, WI 53706, (608) 262-9512

HOLEY, Mark - Wisconsin Department of Natural Resources, 600 E. Greenfield, Milwaukee, WI 53204, (414) 224-3006

JACOBSON, Larry - Department of Fisheries and Wildlife, University of - Minnesota, 200 Hodson Hall, 1980 Folwell Ave., St. Paul, MN 55108, (612) 373-I 709

JESTER, Doug - Fisheries Division, Michigan Department of Natural Resources, Box 30028, Lansing, MI 48909, (517) 373-I 280

JOHNSON, Rebecca - Department of Agricultural Economics, Michigan State University, Room 203C, Cook Hall, East Lansing, MI 48824, (517) 353-7898

JOHNSON, Herb - Sea Lamprey Control Centre, Department of Fisheries and Oceans, Huron St., Ship Canal P. O., Sault Ste. Marie, ONT, Canada P6A 1 PO, (705) 949-I 102

KING, Louis - U. S. Fish and Wildlife Service, Hammond Bay Biological Station, RFD, Box 263, Millersburg, MI 49759, (517) 734-2511

KOONCE, Joseph F. - Department of Biology, Case Western Reserve University, Cleveland, OH 44106, (216) 368-3561

KRUEGER, Charles - Department of Natural Resources, 101 S. Webster, GEF

II, Box 7921, Madison, WI 53707, (608) 267-7501

LAMSA, Aarne - Great Lakes Fishery Commission, 1451 Green Road, Ann Arbor, MI 48105, (313) 662-3209

MACCALLUM, Wayne - Lake Superior Fisheries Assessment Unit, P.O. Box 5000, Thunder Bay, ONT, Canada P7C 5G6 (807) 475-1635

MCDONALD, Rod - Sea Lamprey Control Centre, Department of Fisheries and Oceans, Huron St., Ship Canal P. O., Sault Ste. Marie, ONT, Canada P6A 1 PO, (705) 949-l 102

MCNAMEE, Peter - ESSA Ltd., 7678 W. Broadway, Vancouver, B.C., Canada V5Z 1C6, (604) 872-0691

MANION, Patrick - Sea Lamprey Control, U. S. Fish and Wildlife Service, Federal Building, Fort Snelling, St Paul, MN 5511 I, (612) 725-3447

MINNS, C. K. - Great Lakes Biolimnology Laboratory, Canada Centre for Inland Waters, P.O. Box 5050, Burlington, ONT, Canada L7R 4A6, (416) 637-4730

MOORE, Harry - Marquette Biological Station, U. S. Fish and Wildlife Service, 446 E. Crescent St., Marquette, MI 49855-3698, (906) 226-6571

PURVIS, Harry - Marquette Biological Station, U. S. Fish and Wildlife Service, 446 E. Crescent St., Marquette, MI 49855-3698, (906) 226-6571

PYCHA, Dick - U. S. Fish and Wildlife Service, 1606 St. Clair St., Ashland, WI 54806, (715) 682-6163

REGIER, Henry - Institute for Environmental Studies, Haultain Building, University of Toronto, Toronto, ONT, Canada M5S 1A4, (416) 978-7338

RUGEN, Paul - Marquette Biological Station, U. S. Fish and Wildlife Service, 446 E. Crescent St., Marquette, MI 49855-3698, (906) 226-6571

RYBICKI, R. W. - Charlevoix Great Lakes Fisheries Station, Michigan Department of Natural Resources, P. 0. Box 205, Charlevoix, MI 49720, (616) 547-2914

SAWYER, Alan - Department of Entomology, Cornell University, Ithaca, NY 14853

SCHULDT, Rick - Marquette Biological Station, U. S. Fish and Wildlife Service, 446 E. Crescent St., Marquette, MI 49855-3698, (906) 226-6571

SEELYE, James - U. S. Fish and Wildlife Service, Hammond Bay Biological Station, RFD, Box 263, Millersburg, MI 49759, (517) 734-2511

SPANGLER, George - Department of Fisheries and Wildlife, University of Minnesota, 200 Hodson Hall, 1980 Folwell Ave., St. Paul, MN 55108, (612) 376-2929

SWANSON, Bruce - Lake Superior Office, Department of Natural Resources, P. 0. Box 589, Bayfield, WI 54814, (715) 779-3346

TALHELM, Dan - Fish and Wildlife Extension, Department of Natural Resources, Michigan State University, East Lansing, MI 48824, (517) 355-7493

TIBBLES, James J. - Sea Lamprey Control Centre, Department of Fisheries and Oceans, Huron St., Ship Canal P. O., Sault Ste. Mane, ONT, Canada P6A 1 PO, (705) 949-I 102

WEISE, Jerry G. - Sea Lamprey Control Centre, Department of Fisheries and Oceans, Huron St., Ship Canal P. O., Sault Ste. Marie, ONT, Canada P6A 1 PO, (705) 949-I 102

## APPENDIX B: INSTRUCTIONS FOR EXECUTING MODELS

1. Insert the disk containing the programs into drive 1 of an Apple™ II series microcomputer.

2. Close the door on the drive and turn the machine on. Make sure that the monitor is turned on also.

3. Wait until the disk drive stops whirring and a cursor appears on the screen.

4. Type “CATALOG” -cr- (-cr- stands for a carriage return) to see the names of all files on the disk.

5. To load a program and run it type “RUN filename” -cr-. For example, to run the program called “LAMPREY BASELINE”, type “RUN LAMPREY BASELINE” -cr-. The program will be loaded into the computer and begin to execute.

6. When the prompt “ENTER COMMANDS THEN CONT” appears you have two options. You may either type “CONT” -cr- to continue the simulation or you may execute Applesoft™ BASIC commands in immediate mode (in order, for example, to change the initial values of variables). When you are done executing commands in immediate mode type “CONT” to continue the simulation.

8. When the prompt “SIM FROM” appears you must specify the number of years of output that you wish to plot. Enter two numbers. The first number is the first year to be plotted, the second number is the last year to be plotted. For example type “0 35” -cr- to plot output for years 0 to 35. Your best bet is to use zero for the lower limit. If you set the upper limit too high, then the computer will run out of memory and the program will bomb. Use a smaller upper limit next time.

9. Eventually, the screen will clear and a rectangle for plotting will appear. At the bottom of the screen will be the prompt “PLOT VAR#“. At this point you have a number of options:

   a Type “n” -cr- where 1<=n<=20 to plot variable number n. For example, typing “4” -cr- will cause variable number 4 to be plotted.

   b. Type “n/x” -cr-. This response will cause variable n to be plotted with the Y-axis scaled from 0 to x. For example, typing “5/10” -cr- will cause variable 5 to be plotted with the Y-axis scaled from 0 to 10 Variable numbers and names are given below for the models LAMPREY BASELINE and MODEL-OCT1 5.

   c. Type “n P”. -cr- This response will cause variable n to be plotted as a series of points, rather than as a line.

   d. Type “G”. -cr- This causes previous plots to be cleared from the screen.

   e. Type “x Ci”. Variable x will be plotted in color number i. The argument for color “i” is an integer between 1 and 6. Color number 4 is black and will not be visible. You can erase a single line from the screen by replotting a variable in black.

   f. Type “S filename” -cr-. This response saves the current plotting screen to a file called filename. For example “S MYPLOTS” -cr- will save the current screen to a disk file named MYPLOTS.

   g. Type “R filename” -cr-. This reponse replaces the current screen with a screen previously stored in the file called filename. For example “R OLDPLOTS” restores the plot saved to the disk file OLDPLOTS.

h. Type -cr-. A carriage return by itself starts the simulation over from the beginning. Be careful not to do this accidently or you will have to start over!

Most of these commands can be used in combination (e.g. “4/100 P” -cr- to plot variable 4 on a scale of 0 to 100 in point form.

A more complete description of these procedures can be found in Great Lakes Fishery Commission Special Publication 82-2, p. 33-40.

Output Variables In LAMPREY BASELINE and MODEL-OCT15 For Plotting

| Variable Number | Description |
|---|---|
| 1 | Wounding rate on age IV lake trout (ordinary type) |
| 2 | Wounding rate on age X and older lake trout (ordinary type) |
| 3 | Wounding rate on age IV lake trout (resistant type) |
| 4 | Wounding rate on age X and older lake trout (resistant type) |
| 5 | Lamprey survival rate |
| 6 | Total biomass of lake trout |
| 7 | Total harvest of lake trout |
| 8 | Total number age 0 lake trout (recruitment) |
| 9 | Lamprey scarring rate |
| 10 | Net benefits (millions of dollars) |
| 11 | Percent female lampreys in spawning1 run |
| 12 | Length (meters) of steam habitat occupied by ammocoetes |
| 13 | Total number of spawning sea lamprey |
| 14 | Average length (mm) of adult female lamprey |
| 15 | Total cost of all lamprey control methods in dollars |
| 16 | Transforming sea lamprey per surface area (number |

| | |
|---|---|
| | transformers / $m^2$) |
| 17 | Transformers killed per dollar |
| 18 | Cost of chemical treatments in dollars |
| 19 | Amount lampricide used in pounds |

# APPENDIX C. LISTING OF THE WORKSHOP MODEL (LAMPREY BASELINE)

LAMPREY BASELINE

```
-------------------
1   DIM Z(20,35),ZM(20)
-------------------
2   DIM
PA(20),PB(20),PN(20),PQ(1),PT(1),PS(1),PY(20),PZ(20),P1(1,1),
P2(20),
     P4(1),QC(20),QN(20),QW(20),QL(20)
-------------------
3   DIM TN(1,9),TW(1,9),TL(1,9),TD(1),TR(1),TM(1)
-------------------
4   DIM
GO(4,5),GC(3),GB(4,3),FA(1,2),ED(1),ET(2),EU(2),FP(3),GN(3,1),F
M(3),
     EN(4)
-------------------
5   DIM
AA(6,4),AD(6,7,4),AB(6,4),AE(5),AF(5),AM(4),BN(3),AN(4,1),AC(
4),CH(6
     ,4),CL(4)
-------------------
10  NV = 20
-------------------
11  IF PEEK (104) < > 64 THEN POKE 104,64
 * : POKE 103,1
 * : POKE 16384,0
 * : PRINT CHR$ (4);"RUN SIMCON3"
-------------------
12  PRINT "ENTER COMMANDS,THEN CONT"
  : END
-------------------
14  INPUT "SIM FROM ";A$
  : IF LEFT$ (A$,2) = "GR" THEN 10100
-------------------
15  ZS = VAL ( LEFT$ (A$,2))
  : NT = VAL ( RIGHT$ (A$,3))
-------------------
16  IF NT < 1 THEN NT = VAL ( RIGHT$ (A$,2))
-------------------
18  FOR TIME = ZS TO NT
-------------------
100   IF TI > 0 THEN 1000
-------------------
110   I0 = 0
  :  I1 = 1
  :  I2 = 2
  :  I3 = 3
  :  I4 = 4
  :  I5 = 5
  :  I6 = 6
  :  I7 = 7
  :  I8 = 8
  :  I9 = 9
  :  J0 = 10
  :  J1 = 11
  :  J2 = 12
  :  J3 = 13
  :  J4 = 14
  :  J5 = 15
  :  J6 = 16
  :  J7 = 17
  :  J8 = 18
  :  J9 = 19
  :  K0 = 20
-------------------
115   SD = 365
-------------------
150   SA = 82400
  :  SV = 12200
  :  SL = 16300
  :  SR = SL / SA
  :  ZZ = 1E - 10
-------------------
160   QU = 0
  :  QV = 10
-------------------
210   QM = 5E3 / SL
  :  P9 = .20
-------------------
220   PC = 2.36E - 9
  :  PD = .31
  :  PE = 7.884
  :  PG = 7.5E - 6
  :  PI = 1
-------------------
230   PK = 2
  :  PJ = 150 ^ PK
  :  PT(0) = .2
  :  PT(1) = .15
-------------------
235   PU = 2.2E - 5
  :  PV = - 1.533
  :  PW = 1.23E - 3
  :  PX = - 1.15
-------------------
240   P1(1,1) = .75
  :  P1(1,0) = .75
  :  P1(0,1) = .75
  :  P1(0,0) = .75
-------------------
245   P4(0) = .06
```

```
  : P4(1) = .3
  : P7 = .625
--------------------
250   P8 = .3
  : P0 = 1
  : QA = .25 * .25
  : QD = 6
  : QE = 1
--------------------
255   QH = .25
  : QJ = .8
  : QK = .160 ^ 4
  : Q3 = .4
--------------------
260   QP = 2.47E7
  : QQ = .397
  : QR = .03
--------------------
270   PF = 300
  : QN = .85
  : Q2 = .01
--------------------
300   UQ = 1
  : VY = 0.035
--------------------
301   TC = 2393
  : TD(0) = 2929
  : TD(1) = 1700
  : TA = .288
  : UK = 0
  : T0 = 1E - 4
  : TQ = .1374
  : TT = .63
  : TX = .1
  : TZ = .69
  : UA = 2.691
  : UB = 3.3E - 6
  : UN = 7.5E6
  : U0 = .4
--------------------
302   US = .436
  : VF = 1650
  : VG = 7150
  : VH = 3E - 3
  : VI = 1.1E3
  : VJ = 2.2E7
  : UU = 1E6
--------------------
305   VK = 4E - 4
  : VN = .25
  : VP = 5
  : VM = 2E3
  : VQ = 2E6
  : VS = 2E3
  : VT = 2.5E6
  : VU = 5E2
  : VV = 1E6
  : VW = 1E3
  : VX = 3E3
  : TS(0) = 2E6
  : TS(1) = 0
  : UD = 1.00E - 5
--------------------
307   UE = 1
  : UF = 10E3
  : UM = 1.2
  : TY = .45
  : UT = 1E3
  : QL(10) = 450
  : QW(10) = 1
--------------------
310   DATA
3E6,2.3E6,1.6E6,1.1E6,9.1E5,6.8E5,4.2E5,2.5E5,1.5E5,1.2E5
  : FOR J = I0 TO I9
  :   READ TN(0,J)
  :    TN(0,J) = TN(0,J) / I2
  :    TN(1,J) = TN(0,J)
  :    QN(J) = TN(0,J) / SL
  :    QN(J + 1 + J0) = TN(1,J) / SL
  : NEXT
  : QN(10) = UN / SL
--------------------
320   DATA      .023,.041,.114,.255,.5
,.8180,1.290,1.923,2.732,5.000
  : FOR J = I0 TO I9
  :   READ TW(0,J)
  :    TW(1,J) = TW(0,J)
  :    QW(J) = TW(0,J)
  :    QW(J + 1 + J0) = TW(1,J)
  : NEXT
--------------------
360   TM(0) = 0
  : TM(1) = 11
--------------------
361   TR(0) = 1
  : TR(1) = 1
--------------------
365   UJ = .023
--------------------
400   REM   SPAWNERS DATA
--------------------
440   FOR I = 0 TO 1
```

```
  : FOR J = 0 TO 2
  :   READ FA(I,J)
  : NEXT
  :   FA(I,0) = FA(I,0) / (FA(I,2) - FA(I,1) + 1)
  : NEXT
--------------------
441   DATA  0,0,0,0,0,0
--------------------
442   ED(0) = 2240
  : ED(1) = 18000
--------------------
450   EE = 4500
  : FQ = 0.5
  : GD = 25000
  : GH = 5E5
--------------------
451   ET(0) = 0
  : ET(1) = 0
  : ET(2) = 0
  : EU(0) = 0.5
  : EU(1) = 0.5
  : EU(2) = 0.1
--------------------
455   ER = 45000
  : EQ = 0.35
--------------------
460   EP = 0
  : GS = 0
  : GT = 0
--------------------
470   GZ = 0.03
  : GX = 12107
  : GY = 205.6
  : GP = 19175
--------------------
490   FOR I = 0 TO 3
  : FOR J = 0 TO 5
  :   READ GO(I,J)
  : NEXT J
  : NEXT I
--------------------
491   DATA  24,125,94400,3,0,0,71,2520,819000,11,23,740
--------------------
492   DATA
27,12730,1320000,31,12,39100,1,3800,19200,60,0,0
--------------------
493   FOR I = 0 TO 3
  : READ GC(I)
  : NEXT I
--------------------
494   DATA  3,42,22,1
--------------------
497   FOR I = 0 TO 3
  : FOR J = 0 TO 1
  :   READ GB(I,J)
  : NEXT J
  : NEXT I
--------------------
498   DATA
4800,89600,633600,186400,1304000,16000,19200,0
--------------------
510   AC(I0) = I6
  : AC(I1) = I6
  : AC(I2) = I6
  : AC(I3) = I0
  : AC(I4) = I0
  : FOR K = I0 TO I4
--------------------
511   IZ = AC(K)
--------------------
512   AD(IZ,I0,K) = 5.5
  : AD(IZ,I1,K) = 1.2
  : AD(IZ,I2,K) = .5
  : AD(IZ,I3,K) = .35
  : AD(IZ,I4,K) = .22
  : AD(IZ,I5,K) = .5
  : AD(IZ,I6,K) = .037
  : AD(IZ,I7,K) = .013
  : FOR I = I0 TO I7
  :   AD(IZ,I,K) = AD(IZ,I,K) / 20
  : NEXT
--------------------
513   NEXT
--------------------
514   AB(I6,I0) = 285000
  : AB(I6,I1) = 9E6
  : AB(I6,I2) = 41E6
  : AB(I0,I3) = 4.6E6
  : AB(I0,I4) = 5E6
--------------------
515   AE(I0) = - .006
  : AF(I0) = .28
  : AE(I1) = - .013
  : AF(I1) = .56
  : FOR J = I2 TO I5
  :   AF(J) = .7
  : NEXT J
  : AQ = .5
--------------------
516   AM(I0) = .1
  : AM(I1) = .1
  : AM(I2) = .1
```

```
    : AM(I3) = .1
    : AM(I4) = .02
    : AP = .1
    : AY = .67
    : AX = - .034
    : A2 = .5
--------------------
517   AK = 10
    : BN(0) = 900000
    : BN(1) = 325000
    : BN(2) = 900000
    : BN(3) = 325000
    : AN(0,0) = 2850000
    : AN(1,0) = 9E7
    : AN(2,0) = 41E7
    : AN(3,0) = 4.6E7
    : AN(4,0) = 0
--------------------
518   QL = (BN(0) + BN(2)) / (BN(0) + BN(1) + BN(2) + BN(3) +
ZZ)
    : FOR I = I0 TO I3
    :   BN(I) = BN(I) / 20
    : NEXT
--------------------
519   FOR I = 0 TO 3
    :   BN(I) = BN(I) / SL
    : NEXT
--------------------
520   CB(I0) = I1
    : CB(I1) = I1
    : CB(I2) = I1
    : CB(I3) = I1
    : CB(I3) = I1
    : CB(I4) = I1
--------------------
521   CD(I0) = I1
    : CD(I1) = I1
    : CD(I2) = I1
    : CD(I3) = I1
    : CD(I4) = I1
--------------------
522   CT(I0) = CB(I0) * CD(I0) * .004
    : CM(I0) = .0776 + CT(I0) * 7.3
    : CK(I0) = 1.078 * CD(I0) ^ 3 * CB(I0) ^ 4 / ((.015625 + CD(I0)
^ 3) *
      (.0625 + CB(I0) ^ 4))
    : IF CK(I0) > I1 THEN CK(I0) = I1
--------------------
523   CT(I1) = CB(I1) * CD(I1) * .002
    : CM(I1) = .0338 + CT(I1) * 7.3
    : CK(I1) = 1.025 * CD(I1) ^ 3 * CB(I1) ^ 4 / ((.015625 + CD(I1)
^ 3) *
      (.0625 + CB(I1) ^ 4))
    : IF CK(I1) > I1 THEN CK(I1) = I1
--------------------
524   CT(I2) = CB(I2) * CD(I2) * .0009
    : CM(I2) = .0194 + CT(I2) * 7.3
    : CK(I2) = 0.917 * CD(I2) ^ 3 * CB(I2) ^ 4 / ((.015625 + CD(I2)
^ 3) *
      (.0625 + CB(I2) ^ 4))
    : IF CK(I2) > I1 THEN CK(I2) = I1
--------------------
525   CT(I3) = CB(I3) * CD(I3) * .004
    : CM(I3) = .0854 + CT(I3) * 7.3
    : CK(I3) = 0.917 * CD(I3) ^ 3 * CB(I3) ^ 4 / ((.015625 + CD(I3)
^ 3) *
      (.0625 + CB(I3) ^ 4))
    : IF CK(I3) > I1 THEN CK(I3) = I1
--------------------
555   CX = .0001
    : CL(I0) = 5E - 4
    : CL(I1) = 5E - 4
    : CL(I2) = 5E - 4
    : CL(I3) = 5E - 4
--------------------
1000  REM  MODEL BEGINS
--------------------
1001  PRINT TI;"ADULT LAMPREY"
--------------------
2000  PS(I0) = QM * QL
    : PS(I1) = (I1 - QL) * QM
    : P9 = P9 * QN
    : PL = (P9 * QP) ^ QQ
    : QS = (BN(I0) + BN(I1) + BN(I2) + BN(I3)) * Q3
--------------------
2005  FOR I = QU TO QV
    :   PB(I) = I0
    :   QC(I) = I0
    :   PA(I) = I0
    :   QL(I) = (QW(I) / PC) ^ PD
    :   PM = QL(I) - PF
    :   IF PM < = 0 THEN PN(I) = 0
    :   GOTO 2030
--------------------
2020    PM = PM ^ PK
    :   PH = PG * QL(I)
    :   IF PH > QD THEN PH = QD
--------------------
2025    PM = PI * PM / (PJ + PM)
    :   PN(I) = QL(I) * PE * PH * PM * SR
--------------------
2030  NEXT
```

```
--------------------
2035  PN(J0) = PN(J0) * Q2
  :  P9 = QR
--------------------
2040  FOR J = I0 TO I1
  :    P6 = 0
  :    P3 = 0
  :   FOR I = QU TO QV
  :     IF PN(I) = 0 GOTO 2070
--------------------
2050      PR = P9 / QW(I)
  :     PY(I) = (PR * PT(J)) ^ PV * PU
  :     PZ(I) = PR ^ PX * PW
  :     Q1 = I > I9
  :     Q1 = P1(J,Q1)
  :     P2(I) = Q1 * PY(I) + (I1 - Q1) * PZ(I)
  :     P3 = P3 + P2(I) * PN(I) * QN(I)
--------------------
2070   NEXT I
--------------------
2080   FOR I = QU TO QV
  :     IF PN(I) = I0 GOTO 2098
--------------------
2082      QG = P4(J) * PN(I) / (P3 + 1)
  :     P6 = P6 + QG * QN(I) * P9 * PT(J) * P2(I) * P7 * P8 * SD
  :     PR = P9 / (QW(I) + ZZ)
--------------------
2085      Q1 = I > I9
  :     Q1 = P1(J,Q1)
  :    IF PY(I) > QE THEN Q1 = 0
--------------------
2090      QF = PR ^ I2
  :     QB = I1 - P0 * QF / (QF + QA)
  :     PB(I) = PB(I) + QS * (I1 - (I1 - Q1) * QB) * QG
  :     PA(I) = QS * QG * QB * (I1 - Q1) + PA(I)
--------------------
2095      QC(I) = QC(I) + QG * QS * (Q1 * PY(I) + (I1 - Q1) * (QB
* QH + PZ(
           I)))
--------------------
2098   NEXT I
--------------------
2100    P9 = P9 + P6
  :  NEXT J
--------------------
2200  Q1 = P9 ^ 4
  :   QI = QJ * Q1 / (QK + Q1)
  :   QL = (BN(I0) + BN(I2)) / (BN(I0) + BN(I1) + BN(I2) + BN(I3) +
ZZ)
  :  QM = QS * QI
--------------------
2210  Z(1,TI) = QC(I3)
  :   Z(2,TI) = QC(I9)
  :   Z(3,TI) = QC(J4)
  :   Z(4,TI) = QC(K0)
  :   Z(5,TI) = 1 - QI
--------------------
2999  PRINT "TROUT"
--------------------
3000  TW(I0,I0) = UJ
  :   TW(I1,I0) = UJ
  :   T8 = I0
  :   TB = 0
  :  FOR I = I0 TO UK
  :    U9 = I0
  :    TV(I) = 0
  :   FOR J = I0 TO I9
  :     TV(I) = TV(I) + TW(I,J) * TN(I,J)
  :     IF TW(I,J) > UM THEN U9 = U9 + TW(I,J) * TN(I,J) / UT
--------------------
3005    NEXT
  :    TB = TB + TV(I) / UT
  :    T8 = T8 + U9
  :  NEXT
--------------------
3010  FOR I = I0 TO UK
  :    TN(I,0) = TN(I,0) + TS(I)
  :  NEXT
--------------------
3020  TP = I1 - UD * TB
  :  IF TP < I0 THEN TP = I0
--------------------
3025  REM
--------------------
3026  REM
--------------------
3030  TH = 0
  :  FOR I = I0 TO UK
  :    UZ = TM(I)
  :    TE(I) = 0
  :   FOR J = I0 TO I9
  :     U1 = UZ + J
  :     IF J > I4 THEN TW(I,J) = TP * TR(I) * (TW(I,J) + UE)
--------------------
3040      QN(U1) = QN(U1) + TN(I0,J) * UQ / SL
  :     QW(U1) = TW(I,J)
--------------------
3070      TK = 0
  :     TL(I,J) = TL(I,J) + PA(U1)
  :    IF TW(I,J) > = UM THEN TK = TZ - (TA + PB(U1))
--------------------
3075     IF TK < = 0 THEN TK = 0
```

```
--------------------
3077    TN(I,J) = TN(I,J) * EXP ( - (PB(U1) + TA + TK))
  :   TH = TH + (((TK / TZ * TY) * TN(I,J) * TW(I,J)) / UT)
--------------------
3090    TF = TC * TW(I,J) - TD(I)
  :   IF TF < I0 THEN TF = I0
--------------------
3095    TE(I) = TE(I) + TF * (TN(I,J) / I2) * T0
  :  NEXT
  : NEXT
--------------------
3097  UR = UA * U0 * UN * EXP ( - UB * UN)
  : UN = UN + UR
--------------------
3098  QN(10) = UN / SL
  : UN = UN * EXP ( - (PB(10) + US))
--------------------
3100  FOR I = I0 TO UK
  :   UC = TN(I,I9) + TN(I,8) + ZZ
  :   TW(I,I9) = (TW(I,I9) * TN(I,I9) + TW(I,I8) * TN(I,I8)) / UC
  :   TN(I,I9) = UC
  : NEXT
--------------------
3150  IF TH < VW GOTO 3170
--------------------
3160  IF TH > VX GOTO 3180
--------------------
3165  GOTO 3190
--------------------
3170  VR = VS * TH
  : GOTO 3195
--------------------
3180  VR = VV
  : GOTO 3195
--------------------
3190  VR = VT - VU * TH
--------------------
3195  IF TB > I0 THEN VO = VQ
--------------------
3196  IF TB > VM THEN VO = VL
--------------------
3197  VA = ((VG * EXP ( - VH * TH) + VF) * TH + VO) - (VN * TS
+ VR + (VJ /
      (TB + ZZ) * TH))
--------------------
3200  FOR I = I0 TO UK
  :   FOR J = I7 TO I0 STEP - 1
  :     TN(I,J + I1) = TN(I,J)
  :     IF J > I3 THEN TW(I,J + I1) = TW(I,J)
--------------------
3205    TL(I,J + I1) = TL(I,J)
  :   NEXT
  :   TN(I,I0) = TE(I) + TS(I)
  : NEXT
--------------------
3310  Z(0,TI) = Z(0,TI) + (VA * (1 - VY) ^ TI) / UU
  : Z(6,TI) = T8
  : Z(7,TI) = TH
  : Z(8,TI) = TN(0,0)
  : Z(9,TI) = TL(0,7)
  : Z(10,TI) = VA / UU
--------------------
4000  REM   SPAWNERS MODEL
--------------------
4001  PRINT "SPAWNERS"
--------------------
4105  GV = 0
  : FOR I = I0 TO I1
  :   PS(I) = SL * PS(I)
  : NEXT
--------------------
4106  FOR I = I0 TO I3
  :   GV = GV + AN(I,I0)
  : NEXT
--------------------
4110  FOR I = I0 TO I3
  :   FP(I) = FQ * GO(I,I1) / GP + (I1 - FQ) * AN(I,I0) / (GV + ZZ)
  : NEXT
--------------------
4210  GE = (PS(I0) + PS(I1)) / (GD + PS(I0) + PS(I1))
--------------------
4220  FOR I = I0 TO I3
  :   GB(I,I0) = GB(I,I0) + GE * GB(I,I1)
  :   GB(I,I3) = GE * GB(I,I1)
  :   GB(I,I1) = (I1 - GE) * GB(I,I1)
--------------------
4310    FOR J = I0 TO I1
  :     GN(I,J) = FP(I) * PS(J)
  :   NEXT
  : NEXT
--------------------
4410  FOR I = I1 TO I2
  :   GB(I,I2) = 0
--------------------
4415    IF TI < FA(I - I1,I1) OR TI > FA(I - I1,I2) THEN 4450
--------------------
4425    FE = FA(I - I1,I0)
  :   GB(I,I2) = FE * GO(I,I5)
--------------------
4430    IF GB(I,I2) < GB(I,I0) THEN GB(I,I0) = GB(I,I0) - GB(I,I2)
  :   GOTO 4440
--------------------
```

```
4435    GB(I,I2) = GB(I,I0)
  :   GB(I,I0) = I0
--------------------
4440    EC = EC + ED(I - I1) * FE
  :   FM(I) = FM(I) + FE
--------------------
4450   NEXT
--------------------
4510   EF = I0
  :  GM = I0
  :  FM(I3) = I0
  :  IF ET(I2) = I1 THEN FM(I3) = I1
--------------------
4515   FOR I = I1 TO I3
  :    IF ET(I - I1) < 1 THEN 4545
--------------------
4525    GI = EU(I - I1) * FM(I) / (GC(I) + (GO(I,I0) - GC(I)) *
(PS(I0) + PS
       (I1)) / GH)
--------------------
4530    GM = GM + GI * GN(I,I0)
  :   EF = EF + FM(I) * EE
--------------------
4535    FOR J = I0 TO I1
  :     GN(I,J) = GN(I,J) * (I1 - GI)
  :   NEXT
--------------------
4545   NEXT
--------------------
4610   IF TI < EP THEN 4645
--------------------
4615   FOR I = I0 TO I3
  :    FS(I) = I0
  :  NEXT
--------------------
4620   IF GS < I1 OR GS > I4 THEN 4645
--------------------
4625   IF GT = I0 THEN GU = GM
 *:  ES = ER
 *:  GOTO 4635
--------------------
4630   GU = GT
  :  ES = ER + GT * EQ
--------------------
4635   IF GS < I4 THEN FS(GS) = GU
 *:  GOTO 4645
--------------------
4640   FOR I = I0 TO I3
  :    FS(I) = FP(I) * GU
  :  NEXT
--------------------
4645   REM    KEEP THIS IN
--------------------
4710   FOR I = I0 TO I3
  :    EN(I) = GZ * (GX + GY * PL) * GN(I,I1) * GN(I,I0) / (GN(I,I0)
+ FS(
       I) + ZZ)
  :  NEXT
--------------------
4910   Z(11,TI) = PS(I1) / (PS(I1) + PS(I0) + ZZ)
  :    Z(12,TI) = I0
  :    Z(13,TI) = PS(I0) + PS(I1)
  :    Z(14,TI) = PL
--------------------
4915   FOR I = I0 TO I3
  :    Z(12,TI) = Z(12,TI) + GB(I,I0)
  :  NEXT
--------------------
4999   PRINT "AMMOCETES"
--------------------
5000   FOR K = I0 TO I4
--------------------
5005     AB(I0,K) = AB(I0,K) + GB(K,I3) * GO(K,I3)
--------------------
5010     IZ = AC(K)
--------------------
5020     FOR I = I0 TO IZ
  :      AA(I,K) = I0
  :    NEXT
--------------------
5100     FOR J = I0 TO I7
--------------------
5110       IF K > I2 THEN AA(I0,K) = AA(I0,K) + AD(I0,J,K)
 *:      GOTO 5160
--------------------
5120       AD(I6,J,K) = (AD(I6,J,K) * AB(I6,K) + AD(I5,J,K) *
AB(I5,K)) / (AB
         (I6,K) + AB(I5,K) + ZZ)
  :      AA(I6,K) = AA(I6,K) + AD(I6,J,K)
--------------------
5130       FOR I = I5 TO I1 STEP - 1
  :          AI = I - I1
  :          AD(I,J,K) = AD(AI,J,K)
  :          AA(I,K) = AA(I,K) + AD(I,J,K)
  :        NEXT
--------------------
5140       GOTO 5160
--------------------
5150       AA(I0,K) = AA(I0,K) + AD(I0,J,K)
--------------------
5160     NEXT
--------------------
```

```
5170   AB(I6,K) = AB(I6,K) + AB(I5,K)
  :  FOR I = I5 TO I1 STEP - 1
  :    AB(I,K) = AB(I - I1,K)
  :  NEXT
--------------------
5200   FOR I = I0 TO IZ
--------------------
5210    FOR J = I0 TO I5
--------------------
5220     AS = AE(J) * AA(I,K) + AF(J)
  :    IF AS < I0 THEN AS = I0
--------------------
5230     IF K < I4 THEN AD(I0,J,I4) = AD(I0,J,I4) + (A2 * (I1 -
AS) * AD(
         I,J,K) * AB(I,K)) / (AB(I0,I4) + ZZ)
--------------------
5240     IF K = I4 AND AA(I0,K) > AK THEN AS = AS * AK /
(AA(I0,K) + ZZ)
--------------------
5250     AD(I,J,K) = AD(I,J,K) * AS
--------------------
5260    NEXT
--------------------
5270    AD(I,I5,K) = AD(I,I5,K) + AD(I,I4,K)
  :   FOR J = I4 TO I1 STEP - 1
  :     AD(I,J,K) = AD(I,J - I1,K)
  :   NEXT J
--------------------
5280    IF K < I4 THEN AD(I,I0,K) = EN(K) / (GB(K,I0) *
GO(K,I3) + ZZ)
--------------------
5287    AS = I1 - AP * AA(I,K)
  :   IF AS < I0 THEN AS = I0
--------------------
5290    AD(I,I6,K) = AM(K) * (AD(I,I5,K) + AS * AD(I,I4,K))
  :   AD(I,I4,K) = AD(I,I4,K) * (I1 - AM(K) * AS)
  :   AD(I,I5,K) = (I1 - AM(K)) * AD(I,I5,K)
--------------------
5300    AD(I,I7,K) = (AY + AX * AA(I,K)) * AD(I,I6,K)
  :   IF AD(I,I7,K) < I0 THEN AD(I,I7,K) = I0
--------------------
5310    AD(I,I6,K) = AD(I,I6,K) - AD(I,I7,K)
--------------------
5320   NEXT
  : NEXT
--------------------
5550  FOR K = I0 TO I3
  :  IZ = AC(K)
  :  AN(K,I0) = 0
  :  AS = CL(K) * CL(K)
  :  FOR I = I0 TO IZ
  :    CH(I,K) = AB(I,K) * (I1 - AS / ((AD(I,I6,K) + AD(I,I7,K)) *
(AD(I
       I6,K) + AD(I,I7,K)) + AS + ZZ))
  :    AN(K,I0) = AN(K,I0) + AA(I,K) * AB(I,K)
  :  NEXT
  : NEXT
--------------------
5600  CC = I0
  :  CU = I0
  :  CZ = I0
  :  FOR K = I0 TO I3
  :   IZ = AC(K)
  :   AB(I0,K) = I0
--------------------
5610   FOR I = I0 TO IZ
--------------------
5620    CC = CC + CM(K) * CH(I,K)
  :   CU = CU + CT(K) * CH(I,K)
  :   CZ = CZ + CK(K) * (AD(I,I6,K) + AD(I,I7,K)) * AB(I,K)
--------------------
5630    FOR J = I0 TO I7
  :    AD(I0,J,K) = (AD(I0,J,K) * AB(I0,K) + AD(I,J,K) * CH(I,K)
* (I1
         - CK(K))) / (AB(I0,K) + CH(I,K) + ZZ)
  :    AD(I,J,K) = AD(I,J,K) * CL(K) / (AD(I,I6,K) + AD(I,I7,K) +
ZZ)
  :   NEXT
--------------------
5640    AB(I0,K) = AB(I0,K) + CH(I,K)
  :   AB(I,K) = AB(I,K) - CH(I,K)
--------------------
5645   NEXT
  :  AB(I0,K) = AB(I0,K) - GB(K,I2) * GO(K,I3)
--------------------
5650   AA(I0,K) = I0
  :  FOR J = I0 TO I7
  :    AA(I0,K) = AA(I0,K) + AD(I0,J,K)
  :  NEXT
  :  IF AA(I0,K) < CX THEN GB(K,I0) = GB(K,I0) - AB(I0,K) /
(GO(K,I3) +
      ZZ)
 * :  GB(K,I1) = GB(K,I1) + AB(I0,K) / (GO(K,I3) + ZZ)
--------------------
5660  NEXT
--------------------
5900  BN(I0) = I0
  : BN(I1) = I0
--------------------
5901  FOR K = I0 TO I4
--------------------
5902   IZ = AC(K)
```

```
--------------------
5903    FOR I = I0 TO IZ
  :       BN(I0) = BN(I0) + AD(I,I6,K) * AB(I,K) / (SL + ZZ)
  :       BN(I1) = BN(I1) + AD(I,I7,K) * AB(I,K) / (SL + ZZ)
  :    NEXT
  :   NEXT
  :   BQ = BN(I0) + BN(I1)
  :   IF BQ > 5E6 THEN BN(I0) = BN(I0) * (5E6 / BQ)
 * :   BN(I1) = BN(I1) * 5E6 / BQ
--------------------
5904    BN(I2) = AQ * BN(I0)
  :   BN(I0) = (I1 - AQ) * BN(I0)
  :   BN(I3) = AQ * BN(I1)
  :   BN(I1) = (I1 - AQ) * BN(I1)
--------------------
5910   Z(J6,TI) = BN(I0) + BN(I1) + BN(I2) + BN(I3)
--------------------
5911   Z(J7,TI) = CZ / (CC + ZZ)
--------------------
5912   Z(J8,TI) = CC
--------------------
5913   Z(J9,TI) = CU
--------------------
6000  REM   SHARED ZVARS
--------------------
6010  Z(15,TI) = (ES + EF + EC + CC) / 1E6
--------------------
10000  FOR J = 0 TO NV
--------------------
10020    IF Z(J,TI) > ZM(J) THEN ZM(J) = Z(J,TI)
--------------------
10040    IF Z(J,TI) < 0 THEN Z(J,TI) = 0
--------------------
10060  NEXT
--------------------
10080 NEXT TIME
--------------------
10100 HGR
--------------------
10110 ZF = 0
  : GOTO 10240
--------------------
10120 CALL 62450
  : HCOLOR= 3
  : HPLOT 0,0 TO 0,159 TO 279,159 TO 279,0 TO 0,0
  : VTAB 24
--------------------
10125 IF ZF > 1 THEN ZF = 1
--------------------
10130 IF ZF > 0 THEN HPLOT 0,80 TO 279,80
 * : HPLOT 140,0 TO 140,159
--------------------
10140 INPUT "PLOT VAR#";A$
--------------------
10145 IF LEFT$ (A$,1) = "D" THEN 14000
--------------------
10200 IF LEFT$ (A$,1) < > "F" THEN 10280
--------------------
10220 ZF = VAL ( RIGHT$ (A$,1))
  : IF ZF > 1 THEN ZF = 1
--------------------
10240 IF ZF = 0 THEN ZA = 279
 * : ZB = 0
 * : ZE = 159
 * : ZD = 159
 * : GOTO 10120
--------------------
10260 GOTO 10130
--------------------
10280 IF ZF = 0 THEN 10320
--------------------
10282 ZA = 139
  : ZE = 79
  : ZB = 0
  : ZD = 79
--------------------
10284 IF ZF = 2 OR ZF = 4 THEN ZB = 140
--------------------
10286 IF ZF > 2 THEN ZD = 159
--------------------
10288 ZF = ZF + 1
  : IF ZF > 4 THEN ZF = 1
--------------------
10320 Z6 = 0
  : Z5 = 0
--------------------
10340 REM   SEARCH FOR MAX Z3ALES
--------------------
10360 FOR I = 1 TO LEN (A$)
--------------------
10380   IF MID$ (A$,I,1) = "/" THEN 10420
--------------------
10400 NEXT
  : GOTO 10600
--------------------
10420 FOR J = I + 1 TO LEN (A$)
--------------------
10440   IF MID$ (A$,J,1) = " " THEN 10480
--------------------
10460 NEXT
  : J = LEN (A$)
  : A$ = A$ + " "
```

```
--------------------
10480 Z3 = VAL ( MID$ (A$,I + 1,J - I))
--------------------
10500 IF Z6 > 0 THEN 10540
--------------------
10520 Z6 = 1
  : Z4 = Z3
  : GOTO 10560
--------------------
10540 Z5 = 1
  : XT = Z3
--------------------
10560 A$ = LEFT$ (A$,I - 1) + RIGHT$ (A$, LEN (A$) - J)
--------------------
10580 GOTO 10360
--------------------
10600 IF LEN (A$) < 3 THEN 10640
--------------------
10610 IF MID$ (A$, LEN (A$) - 1,1) < > "C" THEN 10640
--------------------
10620 ZC = VAL ( RIGHT$ (A$,1))
  : A$ = LEFT$ (A$, LEN (A$) - 2)
--------------------
10630 FOR I = 1 TO LEN (A$)
  :   IF RIGHT$ (A$,1) < > " " THEN 10640
--------------------
10635   A$ = LEFT$ (A$, LEN (A$) - 1)
  : NEXT
--------------------
10640 ZP = 0
  : IF RIGHT$ (A$,1) < > "P" THEN GOTO 10700
--------------------
10660 ZP = 1
--------------------
10680 A$ = LEFT$ (A$, LEN (A$) - 1)
--------------------
10700 IF LEN (A$) > 4 THEN GOTO 11060
--------------------
10720 I = VAL ( LEFT$ (A$,2))
--------------------
10740 IF A$ = "G" THEN GOTO 10120
--------------------
10760 IF A$ = "" THEN 10140
--------------------
10770 IF A$ = "ZZ" THEN 10
--------------------
10780 IF I < 7 THEN HCOLOR= I
  : IF I = 4 THEN HCOLOR= 5
--------------------
10790 IF ZC < > 0 THEN HCOLOR= ZC
  : ZC = 0
--------------------
10800 PRINT "MAX=";ZM(I)
--------------------
10820 Z1 = ZA / (NT - ZS)
  : Z2 = 1
  : IF ZM(I) > 0 THEN Z2 = ZE / ZM(I)
--------------------
10840 IF Z6 > 0 AND Z4 > ZM(I) THEN Z2 = ZE / Z4
--------------------
10860 HPLOT ZB,ZD - Z(I,ZS) * Z2
  : Z8 = ZS + 1
--------------------
10900 IF ZP > 0 THEN GOTO 10980
--------------------
10920 FOR J = Z8 TO NT
  :   Z8 = J - ZS
--------------------
10940   HPLOT TO ZB + Z8 * Z1,ZD - Z(I,J) * Z2
  : NEXT
--------------------
10960 GOTO 10140
--------------------
10980 FOR J = Z8 TO NT
--------------------
11000   Z8 = J - ZS
--------------------
11020   HPLOT ZB + Z8 * Z1,ZD - Z(I,J) * Z2
  : NEXT
--------------------
11040 GOTO 10140
--------------------
11060 HCOLOR= 5
  : I = VAL ( LEFT$ (A$,2))
  : J = VAL ( RIGHT$ (A$,2))
  : Z1 = 1
  : Z2 = 1
--------------------
11140 IF ZM(J) > 0 THEN Z1 = ZA / ZM(J)
--------------------
11160 IF Z5 > 0 AND XT > ZM(J) THEN Z1 = ZA / XT
--------------------
11200 IF ZM(I) > 0 THEN Z2 = ZE / ZM(I)
--------------------
11220 IF Z6 > 0 AND Z4 > ZM(I) THEN Z2 = ZE / Z4
--------------------
11240 HCOLOR= 3
--------------------
11250 IF ZC < > 0 THEN HCOLOR= ZC
  : ZC = 0
--------------------
11260 HPLOT ZB + Z(J,ZS) * Z1,ZD - Z(I,ZS) * Z2
```

```
  : Z7 = ZS + 1
--------------------
11300 IF ZP > 0 THEN GOTO 11380
--------------------
11320 FOR Z8 = Z7 TO NT
--------------------
11340  HPLOT TO ZB + Z(J,Z8) * Z1,ZD - Z(I,Z8) * Z2
  : NEXT
--------------------
11360 GOTO 10140
--------------------
11380 FOR Z8 = Z7 TO NT
--------------------
11400  HPLOT ZB + Z(J,Z8) * Z1,ZD - Z(I,Z8) * Z2
  : NEXT
--------------------
11420 GOTO 10140
--------------------
11440 END
--------------------
12000 ZR = 2 * RND (15) - 1
  : ZR = LOG ((1 + ZR) / (1 - ZR)) / 1.82
  : RETURN
--------------------
14000 D$ = CHR$ (4)
--------------------
14010 INPUT "ENTER FILE NAME--";B$
--------------------
14020 PRINT D$;"OPEN ";B$
--------------------
14030 IF MID$ (A$,2,1) = "L" THEN 14150
--------------------
14040 IF MID$ (A$,2,1) < > "S" THEN 10140
--------------------
14050 PRINT D$;"DELETE ";B$
--------------------
14060 PRINT D$;"OPEN ";B$
--------------------
14070 PRINT D$;"WRITE ";B$
--------------------
14080 PRINT ZS
  : PRINT NT
  : PRINT NV
--------------------
14090 FOR I = 1 TO NV
  :  PRINT ZM(I)
--------------------
14100  FOR J = ZS TO NT
  :    PRINT Z(I,J)
--------------------
14110  NEXT J
  : NEXT I
--------------------
14120 PRINT D$;"CLOSE ";B$
--------------------
14130 GOTO 10140
--------------------
14150 PRINT D$;"READ ";B$
--------------------
14160 INPUT ZS
  : INPUT NT
  : INPUT NV
--------------------
14170 FOR I = 1 TO NV
  :  INPUT ZM(I)
--------------------
14180  FOR J = ZS TO NT
  :    INPUT Z(I,J)
--------------------
14190  NEXT J
  : NEXT I
--------------------
14200 GOTO 14120
-------END OF FILE-------
```

APPENDIX D. DESCRIPTION OF MODEL-OCT15

*(Editor's note: The following materials relate to a version of the workshop model called MODEL-OCT15 that was developed by Dr. C. K. Minns after the workshop as described in the introduction. The principal advantage of MODEL-OCT15 is reduced execution time. Note that the Ammocoete and Transformers Submodel is the only section of code that differs between the workshop mode/ and MODEL-OCT15. Descriptions of variables and code are, therefore, given below only for the Ammocoete and Transformers submodel of MODEL OCT- 15.)*

Uncertainties

The following uncertainties in MODEL-OCT15 were identified by Dr. Minns.

1) I am not sure I have the correct version of the trout-prey submodel. I attempted to reconstruct the weight-growth model for lake trout from the workshop notes.

2) In the workshop model, lentic habitat was assigned no size. I guesstimated a length of 3E5 m and a width of 10 m (line 496). This is what keeps the populations going in my version of the model.

3) In the workshop model, there was a piece of code which ensured that the total numbers of transformers would never exceed 5E5, I eliminated this.

4) In the workshop model, it appeared that all new eggs were assigned to newly occupied habitat - this would by itself induce 5 year cycles. The new model cycles also but I think this is induced by the dynamics of the habitat.

5) I had thought I would have to check the control program by the frequency of treatment but apparently the existing rule works as is. I

did fix it so that the Nipigon is only treated once every five years but I suspect that the change was unnecessary.

6) The cost factors have dropped significantly but I have not been able to figure out why. Costs in the workshop model always seemed high. The cost should be in the $1.52 million range.

Variable List For MODEL-OCT1 5

| Variable | Description |
|---|---|
| J | Subscript for age (5 = V and up; 6,7 = male and female transformers) |
| K | Subscript for habitat (O-2 = small-large; 3 = Nipigon; 4 = lentic) |
| AA | Total ammocoete density (#/m$^2$) |
| AD(J,K) | Ammocoete density (#/m$^2$) |
| AE(J) | Slope of survival line (-0.006 for age 0, -0.013 for age I, rest = zero) |
| AF(J) | Survival line intercept (0.28 for age 0, 0.56 for age I, rest = 0.7) |
| AK | Lentic carrying capacity (10/m$^2$) |
| AM(J) | Proportion transforming (0.1 in stream, 0.02 in lake) |
| AN(K,Ø) | Ammocoete numbers |
| AP | Density constant (0.1 for transforming age IV ammocoetes) |
| AQ | Proportion of spring transformers (0.5) |
| AS | Survival |
| Ax | Slope for density dependent proportional female transformers (-0.036) |
| AY | Intercept for above (0.67) |
| A3 | Proportion of age 4 transformers |
| A4 | Proportion of female transformers |
| BN(4) | number transformers by sex and season (0 = fall male; 1 = fall female; 2,3 = spring) |
| CB(K) | Chemical bank length (proportion of stream length) |
| CC | Chemical control cost |
| CD(K) | Chemical dosage (proportion of minimum lethal dose) |
| CF(K) | Frequency of treatment (years) |
| CH(K) | Habitat - area treated (m$^2$) |
| CK(K) | Proportion killed by chemical |
| CL(K) | Minimum transformer density detection threshold for |

| | |
|---|---|
| | treatment (number/$m^2$) |
| CM(K) | Treatment cost (\$/$m^2$) |
| CT(K) | Chemical used (pound/$m^2$) |
| CU | Total chemical used |
| C2 | Total transformers killed |
| z15 | Total control cost |
| Z16 | Transformers |
| Z17 | Transformers killed per cost (number/$) |
| Z18 | Chemical treatment cost |
| z19 | Chemical used (pounds) |

Description By Line Number

| Line number(s) | Operation |
|---|---|
| 5100-5170 | Basic population dynamics |
| 5100 | Open habitat loop |
| 5105 | Rescale densities for newly occupied habitat |
| 5110 | Sum ammocoete density age 0 to age V |
| 5115 | Calculate proportion of age IV transformers |
| 5120 | Calculate proportion female transformers |
| 5125 | Open age loop, calculate natural survival |
| 5130 | Add part of stream pop loss to lentic habitat |
| 5135 | Adjust lentic survival if over carrying capacity |
| 5140 | Ensure survival < = 1 |
| 5145 | Update age class for survival, close age loop |
| 5150 | Move age classes up one year |
| 5155 | Calculate number of transformers and remove from ammocoete classes |
| 5160 | Calculate number male, female transformers |
| 5165 | Place new eggs in 0 age class |
| 5170 | Close habitats loop |
| 5200-5270 | Treatment calculations |
| 5200 | Initialize CC, CU and CZ |
| 5205 | Open habitat loop, test if to be treated |
| 5210 | Calculate habitat to be treated |
| 5216 | Adjust for treatment frequency - habitat O-2 |
| 5217 | Adjust for treatment frequency - Nipigon |
| 5220 | Close habitat loop |
| 5225 | Open habitat loop, test if to be treated |
| 5230 | Accumulate CC, CU and CZ |
| 5235 | Calculate treatment mortality rate |
| 5240 | Apply treatment mortality to age classes O-VII |
| 5250 | Calculate habitat-rendered unoccupied |

| | |
|---|---|
| 5255 | Rescale remaining ammocoete densities |
| 5260 | Adjust occupied and unoccupied categories |
| 5270 | Close habitat loop |
| 5300-5340 | Complete transformer calculations, calculate habitat, abundance and Z vars. |
| 5300 | Initialize transformers |
| 5305 | Open habitat-loop, calculate habitat area |
| 5310 | Calculate male transformers per lake area |
| 5315 | Calculate female tranformers per lake area |
| 5320 | Calculate ammocoete populations |
| 5325 | Close habitat loop |
| 5330 | Calculate spring-fall tranformers and sum of transformers/area |
| 5335 | Calculate total transformers |
| 5340 | Calculate Z vars. |

# APPENDIX E. LISTING OF MODEL-OCT15

MODEL-OCT15

```
--------------------
1    DIM Z(20,30),ZM(20)
--------------------
2    DIM
PA(20),PB(20),PN(20),PQ(1),PM(20),PT(1),PS(1),PY(20),PZ(20),P
1(1,1),
      P2(20),P4(1),QC(20),QN(20),QW(20),QL(20)
--------------------
3    DIM TN(1,9),TW(1,9),TL(1,9),TD(1),TR(1),TM(1)
--------------------
4    DIM
GO(4,5),GC(3),GB(4,3),FA(1,2),ED(1),ET(2),EU(2),FP(3),GN(3,1),F
M(3),
      EN(4)
--------------------
5    DIM
AD(7,4),A9(4),AE(5),AF(5),AM(4),BN(3),CK(4),CL(4),CD(4),CB(4),
CM(4),
      CT(4),AN(4,1),CH(4),CF(4)
--------------------
10   NV = 20
--------------------
12   PRINT "ENTER COMMANDS,THEN CONT"
  : END
--------------------
14   INPUT "SIM FROM ";A$
  : IF LEFT$ (A$,2) = "GR" THEN 10100
--------------------
15   ZS = VAL ( LEFT$ (A$,2))
  : NT = VAL ( RIGHT$ (A$,3))
--------------------
16   IF NT < 1 THEN NT = VAL ( RIGHT$ (A$,2))
--------------------
18   FOR TIME = ZS TO NT
--------------------
100   IF TI > 0 THEN 1000
--------------------
110   I0 = 0
  : I1 = 1
  : I2 = 2
  : I3 = 3
  : I4 = 4
  : I5 = 5
  : I6 = 6
  : I7 = 7
  : I8 = 8
  : I9 = 9
  : J0 = 10
  : J1 = 11
  : J2 = 12
  : J3 = 13
  : J4 = 14
  : J5 = 15
  : J6 = 16
  : J7 = 17
  : J8 = 18
  : J9 = 19
  : K0 = 20
--------------------
115   SD = 365
--------------------
150   SA = 82400
  : SV = 12200
  : SL = 16300
  : SR = SL / SA
  : ZZ = 1E - 25
--------------------
160   QU = 0
  : QV = 10
--------------------
210   QM = 5E4 / SL
  : P9 = .20
--------------------
220   PC = 2.36E - 9
  : PD = .31
  : PE = 7.884
  : PG = 7.5E - 6
  : PI = 1
--------------------
230   PK = 2
  : PJ = 150 ^ PK
  : PT(0) = .2
  : PT(1) = .15
--------------------
235   PU = 2.2E - 5
  : PV = - 1.533
  : PW = 1.23E - 3
  : PX = - 1.15
--------------------
240   P1(1,1) = .75
  : P1(1,0) = .75
  : P1(0,1) = .75
  : P1(0,0) = .75
--------------------
245   P4(0) = .06
  : P4(1) = .3
  : P7 = .625
--------------------
250   P8 = .3
  : P0 = 1
  : QA = .25 * .25
  : QD = .006
  : QE = 1
```

```
--------------------
255   QH = .25
  :  QJ = .5
  :  QK = .160 ^ 4
  :  Q3 = .4
--------------------
260   QP = 2.47E7
  :  QQ = .397
  :  QR = .03
--------------------
270   PF = 300
  :  QN = .85
  :  PM(10) = .01
--------------------
300   UQ = 1
  :  VY = 0.035
--------------------
301   TC = 2393
  :  TD(0) = 2929
  :  TD(1) = 1700
  :  TA = .288
  :  UK = 0
  :  T0 = 1E - 3
  :  TQ = .1374
  :  TT = .63
  :  TX = .1
  :  TZ = .6
  :  UA = 2.691
  :  UB = 3.3E - 6
  :  UN = 7.5E6
  :  U0 = .4
--------------------
302   US = .436
  :  UU = 1E6
--------------------
305   TS(0) = 1E6
  :  TS(1) = 0
--------------------
307   UE = 150
  :  UF = 12E3
  :  UM = 1.2
  :  TY = .45
  :  UT = 1E3
  :  QL(10) = 450
  :  QW(10) = 1
--------------------
310   DATA
3E6,2.3E6,1.6E6,1.1E6,9.1E5,6.8E5,4.2E5,2.5E5,1.5E5,1.2E5
  :  FOR J = I0 TO I9
  :    READ TN(0,J)
  :    TN(0,J) = I2 * TN(0,J)
  :    TN(1,J) = TN(0,J)
  :    QN(J) = TN(0,J) / SL
  :    QN(J + 1 + J0) = TN(1,J) / SL
  :  NEXT
--------------------
311   QN(10) = UN / SL
--------------------
320   DATA      .023,.041,.114,.255,.5
,.8180,1.290,1.923,2.732,5.000
  :  FOR J = I0 TO I9
  :    READ TW(0,J)
  :    TW(1,J) = TW(0,J)
  :    QW(J) = TW(0,J)
  :    QW(J + 1 + J0) = TW(1,J)
  :  NEXT
--------------------
360   TM(0) = 0
  :  TM(1) = 11
--------------------
361   TR(0) = .808
  :  TR(1) = .808
--------------------
365   UJ = .023
--------------------
370   T9 = 10000
  :  U2 = 3E - 6
  :  UE = 1.41
  :  U3 = 1.27E - 4
  :  UY = 2.64
--------------------
371   TR(0) = 1.41
  :  TR(1) = 1.41
  :  TQ = 0.1374
--------------------
400   REM   SPAWNERS DATA
--------------------
440   FOR I = 0 TO 1
  :    FOR J = 0 TO 2
  :      READ FA(I,J)
  :    NEXT
  :    FA(I,0) = FA(I,0) / (FA(I,2) - FA(I,1) + 1)
  :  NEXT
--------------------
441   DATA  0,0,0,0,0,0
--------------------
442   ED(0) = 2240
  :  ED(1) = 18000
--------------------
450   EE = 4500
  :  FQ = 0.5
  :  GD = 25000
  :  GH = 5E5
--------------------
451   ET(0) = 0
```

```
  :  ET(1) = 0
  :  ET(2) = 0
  :  EU(0) = 0.5
  :  EU(1) = 0.5
  :  EU(2) = 0.1
--------------------
455   ER = 45000
  :  EQ = 0.35
--------------------
460   EP = 0
  :  GS = 0
  :  GT = 0
--------------------
470   GZ = 0.01
  :  GX = 12107
  :  GY = 205.6
  :  GP = 19175
--------------------
480   G1 = 0
--------------------
490   FOR I = 0 TO 3
  :    FOR J = 0 TO 5
  :      READ GO(I,J)
  :    NEXT J
  :  NEXT I
--------------------
491   DATA   24,125,94400,3,0,0,71,2520,819000,11,23,740
--------------------
492   DATA
27,12730,1320000,31,12,39100,1,3800,19200,60,0,0
--------------------
493   FOR I = 0 TO 3
  :    READ GC(I)
  :  NEXT I
--------------------
494   DATA   3,42,22,1
--------------------
495   FOR I = 0 TO 3
  :    GB(I,0) = GO(I,2)
  :  NEXT I
--------------------
496   GO(4,3) = 10
  :  GB(4,0) = 3E5
--------------------
515   AE(I0) = - .006
  :  AF(I0) = .28
  :  AE(I1) = - .013
  :  AF(I1) = .56
  :  FOR J = I2 TO I5
  :    AF(J) = .7
  :  NEXT J
  :  AQ = .5
--------------------
516   AM(I0) = .1
  :  AM(I1) = .1
  :  AM(I2) = .1
  :  AM(I3) = .1
  :  AM(I4) = .02
  :  AP = .1
  :  AY = .67
  :  AX = - .034
  :  A2 = .5
--------------------
517   AK = 10
  :  BN(0) = 9E5
  :  BN(1) = 325E3
  :  BN(2) = 9E5
  :  BN(3) = 325E3
  :  AN(0,0) = 285E4
--------------------
518   QL = (BN(0) + BN(2)) / (BN(0) + BN(1) + BN(2) + BN(3) +
ZZ)
  :  FOR I = I0 TO I3
  :    BN(I) = BN(I) / 20
  :  NEXT
--------------------
519   FOR I = 0 TO 3
  :    BN(I) = BN(I) / SL
  :  NEXT
--------------------
520   CB(I0) = I1
  :  CB(I1) = I1
  :  CB(I2) = I1
  :  CB(I3) = I1
  :  CB(I3) = I1
  :  CB(I4) = I1
--------------------
521   CD(I0) = I1
  :  CD(I1) = I1
  :  CD(I2) = I1
  :  CD(I3) = I1
  :  CD(I4) = I1
--------------------
522   CT(I0) = CB(I0) * CD(I0) * .004
  :  CM(I0) = .0776 + CT(I0) * 7.3
  :  CK(I0) = 1.078 * CD(I0) ^ 3 * CB(I0) ^ 4 / ((.015625 + CD(I0)
^ 3) *
       (.0625 + CB(I0) ^ 4))
  :  IF CK(I0) > I1 THEN CK(I0) = I1
--------------------
523   CT(I1) = CB(I1) * CD(I1) * .002
  :  CM(I1) = .0338 + CT(I1) * 7.3
  :  CK(I1) = 1.025 * CD(I1) ^ 3 * CB(I1) ^ 4 / ((.015625 + CD(I1)
^ 3) *
       (.0625 + CB(I1) ^ 4))
```

```
  : IF CK(I1) > I1 THEN CK(I1) = I1
--------------------
524    CT(I2) = CB(I2) * CD(I2) * .0009
  : CM(I2) = .0194 + CT(I2) * 7.3
  : CK(I2) = 0.917 * CD(I2) ^ 3 * CB(I2) ^ 4 / ((.015625 + CD(I2)
^ 3) *
     (.0625 + CB(I2) ^ 4))
  : IF CK(I2) > I1 THEN CK(I2) = I1
--------------------
525    CT(I3) = CB(I3) * CD(I3) * .004
  : CM(I3) = .0854 + CT(I3) * 7.3
  : CK(I3) = 0.917 * CD(I3) ^ 3 * CB(I3) ^ 4 / ((.015625 + CD(I3)
^ 3) *
     (.0625 + CB(I3) ^ 4))
  : IF CK(I3) > I1 THEN CK(I3) = I1
--------------------
555    CX = .0001
  : CL(I0) = 5E - 4
  : CL(I1) = 5E - 4
  : CL(I2) = 5E - 4
  : CL(I3) = 5E - 4
--------------------
560    FOR K = I0 TO I4
  :  FOR J = I0 TO I5
  :    READ AD(J,K)
  :  NEXT
  : NEXT
--------------------
565    DATA  0.2,0.033,0.0102,3.66E-3,1.31E-3,8.90E-4
--------------------
566    DATA  0.134,0.0245,8.34E-3,3.4E-3,1.31E-3,9.38E-4
--------------------
567    DATA  0.148,0.027,9.23E-3,3.76E-3,1.45E-3,1.03E-3
--------------------
568    DATA  5.41,1.30,0.58,0.39,0.26,0.46
--------------------
569    DATA  0,0.583,0.465,0.368,0.285,0.892
--------------------
570    CF(0) = 1
  : CF(1) = 1
  : CF(2) = 1
--------------------
597    A9(0) = 1
  : A9(1) = 1
  : A9(2) = 1
  : A9(3) = 0
--------------------
999    PRINT "PLEASE BE PATIENT,FOLKS"
--------------------
1000   REM  MODEL STARTS
--------------------
1005   PRINT "TIME IS ";TI
--------------------
2000   PS(I0) = QM * QL
  : PS(I1) = (I1 - QL) * QM
  : P9 = P9 * QN
  : PL = (P9 * QP) ^ QQ
  : QS = (BN(I0) + BN(I1) + BN(I2) + BN(I3)) * Q3
  : PP = ZZ
--------------------
2005   FOR I = QU TO QV
  :    PB(I) = I0
  :    QC(I) = I0
  :    PA(I) = I0
  :    QL(I) = (QW(I) / PC) ^ PD
  :    PM = QL(I) - PF
  :    IF PM < = 0 THEN PN(I) = 0
* :    GOTO 2030
--------------------
2020     PM = PM ^ PK
  :    PH = PG * QL(I)
  :  IF PH > QD THEN PH = QD
--------------------
2025     PM(I) = PI * PM / (PJ + PM)
  :    PN(I) = QL(I) * PE * PH * SR
  :    PP = PP + PM(I)
--------------------
2030   NEXT
--------------------
2035   PN(J0) = PN(J0) * Q2
  : P9 = QR
--------------------
2040   FOR J = I0 TO I1
  :    P6 = 0
  :    P3 = 0
  :  FOR I = QU TO QV
  :    IF PN(I) = 0 GOTO 2070
--------------------
2045       PN(I) = PN(I) * PM(I) / PP
--------------------
2050       PR = P9 / QW(I)
  :      PY(I) = (PR * PT(J)) ^ PV * PU
  :      PZ(I) = PR ^ PX * PW
  :      Q1 = I > I9
  :      Q1 = P1(J,Q1)
  :      P2(I) = Q1 * PY(I) + (I1 - Q1) * PZ(I)
  :      P3 = P3 + P2(I) * PN(I) * QN(I)
--------------------
2070    NEXT
--------------------
2080    FOR I = QU TO QV
  :    IF PN(I) = I0 GOTO 2098
--------------------
2082       QG = P4(J) * PN(I) / P3
  :      P6 = P6 + QG * QN(I) * P9 * PT(J) * P2(I) * P7 * P8 * SD
```

```
  :     PR = P9 / (QW(I) + ZZ)
--------------------
2085      Q1 = I > I9
  :     Q1 = P1(J,Q1)
  :    IF PY(I) > QE THEN Q1 = 0
--------------------
2090      QF = PR ^ I2
  :     QB = I1 - P0 * QF / (QF + QA)
  :     PB(I) = PB(I) + QS * ((I1 - Q1) * (I1 - QB) + Q1) * QG
  :     PA(I) = QS * QG * QB * (I1 - Q1) + PA(I)
--------------------
2095      QC(I) = QC(I) + QG * QS * (Q1 * PY(I) + (I1 - Q1) * (QB
* QH + PZ(
          I)))
2098    NEXT
--------------------
2100    P9 = P9 + P6
  : NEXT
--------------------
2200  Q1 = P9 ^ 4
  :  QI = QJ * Q1 / (QK + Q1)
  :  QL = (BN(I0) + BN(I2)) / (BN(I0) + BN(I1) + BN(I2) + BN(I3) +
ZZ)
  : QM = QS * QI
--------------------
2210  Z(I1,TI) = QC(I3)
  :  Z(I2,TI) = QC(I9)
  :  Z(I3,TI) = QC(J4)
  :  Z(I4,TI) = QC(K0)
  :  Z(I5,TI) = I1 - QI
--------------------
3000  T8 = I0
  :  TB = 0
  : FOR I = I0 TO UK
  :    U9 = I0
  :    TV(I) = 0
  :   FOR J = I0 TO I9
  :     TV(I) = TV(I) + TW(I,J) * TN(I,J)
  :     IF TW(I,J) > UM THEN U9 = U9 + TW(I,J) * TN(I,J) / UT
--------------------
3005    NEXT
  :   TB = TB + TV(I) / UT
  :   T8 = T8 + U9
  : NEXT
--------------------
3020  IF TB < T9 THEN TP = UY - U3 * TB
 *:  GOTO 3030
--------------------
3021  TP = UE - U2 * TB
--------------------
3030  TH = 0
  : FOR I = I0 TO UK
  :   UZ = TM(I)
  :   U4 = TP * TR(I)
  :   TE(I) = 0
  :   FOR J = I0 TO I9
  :    U1 = UZ + J
  :    TW(I,J) = TP * TW(I,J) + UJ
--------------------
3040      QN(U1) = QN(U1) + TN(I0,J) * UQ / SL
  :    QW(U1) = TW(I,J)
--------------------
3070      TK = 0
  :    TL(I,J) = TL(I,J) + PA(U1)
  :    IF TW(I,J) > = UM THEN TK = TZ - (TA + PB(U1))
--------------------
3075     IF TK < = 0 THEN TK = 0
--------------------
3077      TH = TH + (TK / TZ) * TY * TN(I,J) * TW(I,J) / UT
  :    TN(I,J) = TN(I,J) * EXP ( - (PB(U1) + TA + TK))
--------------------
3090      TF = TC * TW(I,J) - TD(I)
  :    IF TF < I0 THEN TF = I0
--------------------
3095      TE(I) = TE(I) + TF * (TN(I,J) / I2) * T0
  :  NEXT
  : NEXT
--------------------
3097  UR = UA * U0 * UN * EXP ( - UB * UN)
  : UN = UN + UR
--------------------
3098  QN(10) = UN / SL
  : UN = UN * EXP ( - (PB(10) + US))
--------------------
3100  FOR I = I0 TO UK
  :   UC = TN(I,I9) + TN(I,8) + ZZ
  :   TW(I,I9) = (TW(I,I9) * TN(I,I9) + TW(I,I8) * TN(I,I8)) / UC
  :   TN(I,I9) = UC
  :   U5 = TM(I) + I9
  :   U4 = TM(I) + I8
  :   QN(U5) = TN(I,I9)
  :   QN(U4) = TN(I,I8)
  :   QW(U5) = TW(I,I9)
  :   QW(U4) = TW(I,I8)
  : NEXT
--------------------
3200  FOR I = I0 TO UK
  :   FOR J = I7 TO I0 STEP - 1
  :     TN(I,J + I1) = TN(I,J)
  :     TW(I,J + I1) = TW(I,J)
  :     QN(TM(I) + J + I1) = TN(I,J)
  :     QW(TM(I) + J + I1) = TW(I,J)
--------------------
3205      TL(I,J + I1) = TL(I,J)
  :  NEXT
```

```
   :  TN(I,I0) = TE(I) + TS(I)
   :  TW(I,I0) = UJ
   :  QW(TM(I)) = UJ
   : NEXT
--------------------
3210  FOR I = I0 TO K0
   :  QN(I) = QN(I) / SL
   : NEXT
--------------------
3310  Z(J0,TI) = T8
   : Z(I7,TI) = TH
   : Z(I8,TI) = TN(0,0)
   : Z(I9,TI) = TL(0,7)
   : Z(I6,TI) = TB
--------------------
4105  GV = 0
   : FOR I = I0 TO I1
   :  PS(I) = SL * PS(I)
   : NEXT
--------------------
4106  FOR I = I0 TO I3
   :  GV = GV + AN(I,I0)
   : NEXT
--------------------
4110  FOR I = I0 TO I3
   :  FP(I) = FQ * GO(I,I1) / GP + (I1 - FQ) * AN(I,I0) / (GV + ZZ)
   : NEXT
--------------------
4210  GE = (PS(I0) + PS(I1)) / (GD + PS(I0) + PS(I1))
--------------------
4220  FOR I = I0 TO I3
   :  GB(I,I0) = GB(I,I0) + GE * GB(I,I1)
   :  GB(I,I3) = GE * GB(I,I1)
   :  GB(I,I1) = (I1 - GE) * GB(I,I1)
--------------------
4310   FOR J = I0 TO I1
   :   GN(I,J) = FP(I) * PS(J)
   :  NEXT
   : NEXT
--------------------
4410  FOR I = I1 TO I2
   :  GB(I,I2) = 0
--------------------
4415   IF TI < FA(I - I1,I1) OR TI > FA(I - I1,I2) THEN 4450
--------------------
4425   FE = FA(I - I1,I0)
   :  GB(I,I2) = FE * GO(I,I5)
--------------------
4430   IF GB(I,I2) < GB(I,I0) THEN GB(I,I0) = GB(I,I0) - GB(I,I2)
  * :  GOTO 4440
--------------------
4435   GB(I,I2) = GB(I,I0)
   :  GB(I,I0) = I0
--------------------
4440   EC = EC + ED(I - I1) * FE
   :  FM(I) = FM(I) + FE
--------------------
4450  NEXT
--------------------
4510  EF = I0
   : GM = I0
   : FM(I3) = I0
   : IF ET(I2) = I1 THEN FM(I3) = I1
--------------------
4515  FOR I = I1 TO I3
   :  IF ET(I - I1) < 1 THEN 4545
--------------------
4525   GI = EU(I - I1) * FM(I) / (GC(I) + (GO(I,I0) - GC(I)) *
(PS(I0) + PS
       (I1)) / GH)
--------------------
4530   GM = GM + GI * GN(I,I0)
   :  EF = EF + FM(I) * EE
--------------------
4535   FOR J = I0 TO I1
   :   GN(I,J) = GN(I,J) * (I1 - GI)
   :  NEXT
--------------------
4545  NEXT
--------------------
4610  IF TI < EP THEN 4645
--------------------
4615  FOR I = I0 TO I3
   :  FS(I) = I0
   : NEXT
--------------------
4620  IF GS < I1 OR GS > I4 THEN 4645
--------------------
4625  GU = GT + GM * G1
   : ES = ER + GT * EQ
--------------------
4635  IF GS < I4 THEN FS(GS) = GU
  * : GOTO 4645
--------------------
4640  FOR I = I0 TO I3
   :  FS(I) = FP(I) * GU
   : NEXT
--------------------
4645  REM   KEEP THIS IN
--------------------
4710  FOR I = I0 TO I3
   :  EN(I) = GZ * (GX + GY * PL) * GN(I,I1) * GN(I,I0) / (GN(I,I0)
+ FS(
       I) + ZZ)
   : NEXT
```

```
--------------------
4910   Z(11,TI) = PS(I1) / (PS(I1) + PS(I0) + ZZ)
  :  Z(12,TI) = I0
  :  Z(13,TI) = PS(I0) + PS(I1)
  :  Z(14,TI) = PL
--------------------
4915   FOR I = I0 TO I3
  :   Z(12,TI) = Z(12,TI) + GB(I,I0)
  :  NEXT
--------------------
5099   REM BASIC POP DYN
--------------------
5100   FOR K = I0 TO I4
--------------------
5105     FOR J = I0 TO I5
  :    AD(J,K) = AD(J,K) * (GB(K,I0) - GB(K,I3)) / GB(K,I0)
  :   NEXT J
--------------------
5110     AA = I0
  :   FOR J = I0 TO I5
  :     AA = AA + AD(J,K)
  :   NEXT J
--------------------
5115     A3 = I1 - AP * AA
  :    IF A3 < I0 THEN A3 = I0
--------------------
5120     A4 = AY + AX * AA
  :    IF A4 < I0 THEN A4 = I0
--------------------
5125     FOR J = I0 TO I5
  :     AS = AE(J) * AA + AF(J)
  :     IF AS < I0 THEN AS = I0
--------------------
5130       IF K < I4 THEN AD(J,I4) = AD(J,I4) + A2 * (I1 - AS) *
AD(J,K)
--------------------
5135       IF K = I4 AND AA > AK THEN AS = AS * AK / (AA + ZZ)
--------------------
5140       IF AS > I1 THEN AS = I1
--------------------
5145       AD(J,K) = AS * AD(J,K)
  :    NEXT J
--------------------
5150     AD(I5,K) = AD(I5,K) + AD(I4,K)
  :   FOR J = I4 TO I1 STEP - 1
  :     AD(J,K) = AD(J - 1,K)
  :   NEXT J
--------------------
5155     AD(I6,K) = AM(K) * (AD(I5,K) + A3 * AD(I4,K))
  :   AD(I4,K) = AD(I4,K) * (I1 - A3 * AM(K))
  :   AD(I5,K) = AD(I5,K) * (I1 - AM(K))
--------------------
5160     AD(I7,K) = A4 * AD(I6,K)
  :   AD(I6,K) = AD(I6,K) - AD(I7,K)
--------------------
5165     AD(I0,K) = EN(K) / (GB(K,I0) * GO(K,I3) + ZZ)
--------------------
5170   NEXT K
--------------------
5199   REM TREATMENTS
--------------------
5200   CC = I0
  :  CU = I0
  :  CZ = I0
--------------------
5205   FOR K = I0 TO I3
  :    CH(K) = I0
  :    AS = CL(K) * CL(K) * A9(K)
  :    IF AS = I0 THEN 5220
--------------------
5210     CH(K) = GB(K,I0) * GO(K,I3) * (I1 - AS / ((AD(I6,K) +
AD(I7,K)) * (A
        D(I6,K) + AD(I7,K)) + AS + ZZ))
--------------------
5215   REM MAY MODIFY THIS FOR FREQ AND COST LIMITS
--------------------
5216     IF K < I3 THEN CH(K) = CH(K) / CF(K)
--------------------
5217     IF K = I3 AND TI > I0 AND TI - CF(K) * INT (TI / CF(K)) <
> I0 THEN
        CH(K) = 0.0
--------------------
5220   NEXT K
--------------------
5225   FOR K = I0 TO I3
  :   IF CH(K) = I0 THEN 5270
--------------------
5230     CC = CC + CM(K) * CH(K)
  :   CU = CU + CT(K) * CH(K)
  :   CZ = CZ + CK(K) * CH(K) * (AD(I6,K) + AD(I7,K))
--------------------
5235     A1 = I1 - CH(K) * CK(K) / (GB(K,I0) * GO(K,I3) + ZZ)
  :   IF A1 < I0 THEN A1 = I0
--------------------
5240     FOR J = I0 TO I7
  :     AD(J,K) = A1 * AD(J,K)
  :   NEXT J
--------------------
5250     A6 = CH(K) * CK(K) / GO(K,I3)
--------------------
5255     FOR J = I0 TO I7
  :     AD(J,K) = AD(J,K) * GB(K,I0) / (GB(K,I0) - A6 + ZZ)
  :   NEXT J
--------------------
5260     GB(K,I0) = GB(K,I0) - A6
```

```
  :  GB(K,I1) = GB(K,I1) + A6
--------------------
5270  NEXT K
--------------------
5299  REM TRANSFORMATION
--------------------
5300   BN(I0) = I0
  :  BN(I1) = I0
--------------------
5305   FOR K = I0 TO I4
  :    AA = GB(K,I0) * GO(K,I3)
--------------------
5310     BN(I0) = BN(I0) + AD(I6,K) * AA / (SL + ZZ)
--------------------
5315     BN(I1) = BN(I1) + AD(I7,K) * AA / (SL + ZZ)
--------------------
5320     AN(K,I0) = I0
  :    FOR J = I0 TO I5
  :      AN(K,I0) = AN(K,I0) + AD(J,K) * AA
  :    NEXT J
--------------------
5325   NEXT K
--------------------
5330   Z(J6,TI) = I0
  :  FOR I = I0 TO I1
  :    BN(I + I2) = AQ * BN(I)
  :    BN(I) = BN(I) - BN(I + I2)
  :    Z(J6,TI) = Z(J6,TI) + BN(I) + BN(I + I2)
  :  NEXT I
--------------------
5335   Z(J6,TI) = Z(J6,TI) * SL
--------------------
5340   Z(J7,TI) = CZ / (CC + ZZ)
  :  Z(J8,TI) = CC
  :  Z(J9,TI) = CU
  :  Z(J5,TI) = (ES + EF + EC + CC) / 1E6
--------------------
10000  FOR J = 0 TO NV
--------------------
10020    IF Z(J,TI) > ZM(J) THEN ZM(J) = Z(J,TI)
--------------------
10040    IF Z(J,TI) < 0 THEN Z(J,TI) = 0
--------------------
10060  NEXT
--------------------
10080 NEXT TIME
--------------------
10100 HGR
--------------------
10110 ZF = 0
  : GOTO 10240
--------------------
10120 CALL 62450
  : HCOLOR= 3
  : HPLOT 0,0 TO 0,159 TO 279,159 TO 279,0 TO 0,0
  : VTAB 24
--------------------
10125 IF ZF > 1 THEN ZF = 1
--------------------
10130 IF ZF > 0 THEN HPLOT 0,80 TO 279,80
 * : HPLOT 140,0 TO 140,159
--------------------
10140 INPUT "PLOT VAR#";A$
--------------------
10145 IF LEFT$ (A$,1) = "D" THEN 14000
--------------------
10200 IF LEFT$ (A$,1) < > "F" THEN 10280
--------------------
10220 ZF = VAL ( RIGHT$ (A$,1))
  : IF ZF > 1 THEN ZF = 1
--------------------
10240 IF ZF = 0 THEN ZA = 279
 * : ZB = 0
 * : ZE = 159
 * : ZD = 159
 * : GOTO 10120
--------------------
10260 GOTO 10130
--------------------
10280 IF ZF = 0 THEN 10320
--------------------
10282 ZA = 139
  : ZE = 79
  : ZB = 0
  : ZD = 79
--------------------
10284 IF ZF = 2 OR ZF = 4 THEN ZB = 140
--------------------
10286 IF ZF > 2 THEN ZD = 159
--------------------
10288 ZF = ZF + 1
  : IF ZF > 4 THEN ZF = 1
--------------------
10320 Z6 = 0
  : Z5 = 0
--------------------
10340 REM  SEARCH FOR MAX ZSALES
--------------------
10360 FOR I = 1 TO LEN (A$)
--------------------
10380   IF MID$ (A$,I,1) = "/" THEN 10420
--------------------
10400 NEXT
  : GOTO 10600
--------------------
```

```
10420 FOR J = I + 1 TO LEN (A$)
--------------------
10440   IF MID$ (A$,J,1) = " " THEN 10480
--------------------
10460 NEXT
   : J = LEN (A$)
   : A$ = A$ + " "
--------------------
10480 Z3 = VAL ( MID$ (A$,I + 1,J - I))
--------------------
10500 IF Z6 > 0 THEN 10540
--------------------
10520 Z6 = 1
   : Z4 = Z3
   : GOTO 10560
--------------------
10540 Z5 = 1
   : XT = Z3
--------------------
10560 A$ = LEFT$ (A$,I - 1) + RIGHT$ (A$, LEN (A$) - J)
--------------------
10580 GOTO 10360
--------------------
10600 IF LEN (A$) < 3 THEN 10640
--------------------
10610 IF MID$ (A$, LEN (A$) - 1,1) < > "C" THEN 10640
--------------------
10620 ZC = VAL ( RIGHT$ (A$,1))
   : A$ = LEFT$ (A$, LEN (A$) - 2)
--------------------
10630 FOR I = 1 TO LEN (A$)
   :   IF RIGHT$ (A$,1) < > " " THEN 10640
--------------------
10635   A$ = LEFT$ (A$, LEN (A$) - 1)
   : NEXT
--------------------
10640 ZP = 0
   : IF RIGHT$ (A$,1) < > "P" THEN GOTO 10700
--------------------
10660 ZP = 1
--------------------
10680 A$ = LEFT$ (A$, LEN (A$) - 1)
--------------------
10700 IF LEN (A$) > 4 THEN GOTO 11060
--------------------
10720 I = VAL ( LEFT$ (A$,2))
--------------------
10740 IF A$ = "G" THEN GOTO 10120
--------------------
10760 IF A$ = "" THEN 10140
--------------------
10770 IF A$ = "ZZ" THEN 10
--------------------
10780 IF I < 7 THEN HCOLOR= I
   : IF I = 4 THEN HCOLOR= 5
--------------------
10790 IF ZC < > 0 THEN HCOLOR= ZC
   : ZC = 0
--------------------
10800 PRINT "MAX=";ZM(I)
--------------------
10820 Z1 = ZA / (NT - ZS)
   : Z2 = 1
   : IF ZM(I) > 0 THEN Z2 = ZE / ZM(I)
--------------------
10840 IF Z6 > 0 AND Z4 > ZM(I) THEN Z2 = ZE / Z4
--------------------
10860 HPLOT ZB,ZD - Z(I,ZS) * Z2
   : Z8 = ZS + 1
--------------------
10900 IF ZP > 0 THEN GOTO 10980
--------------------
10920 FOR J = Z8 TO NT
   :   Z8 = J - ZS
--------------------
10940   HPLOT TO ZB + Z8 * Z1,ZD - Z(I,J) * Z2
   : NEXT
--------------------
10960 GOTO 10140
--------------------
10980 FOR J = Z8 TO NT
--------------------
11000   Z8 = J - ZS
--------------------
11020   HPLOT ZB + Z8 * Z1,ZD - Z(I,J) * Z2
   : NEXT
--------------------
11040 GOTO 10140
--------------------
11060 HCOLOR= 5
   : I = VAL ( LEFT$ (A$,2))
   : J = VAL ( RIGHT$ (A$,2))
   : Z1 = 1
   : Z2 = 1
--------------------
11140 IF ZM(J) > 0 THEN Z1 = ZA / ZM(J)
--------------------
11160 IF Z5 > 0 AND XT > ZM(J) THEN Z1 = ZA / XT
--------------------
11200 IF ZM(I) > 0 THEN Z2 = ZE / ZM(I)
--------------------
11220 IF Z6 > 0 AND Z4 > ZM(I) THEN Z2 = ZE / Z4
--------------------
11240 HCOLOR= 3
--------------------
```

```
11250 IF ZC < > 0 THEN HCOLOR= ZC
 * : ZC = 0
--------------------
11260 HPLOT ZB + Z(J,ZS) * Z1,ZD - Z(I,ZS) * Z2
  : Z7 = ZS + 1
--------------------
11300 IF ZP > 0 THEN GOTO 11380
--------------------
11320 FOR Z8 = Z7 TO NT
--------------------
11340  HPLOT TO ZB + Z(J,Z8) * Z1,ZD - Z(I,Z8) * Z2
  : NEXT
--------------------
11360 GOTO 10140
--------------------
11380 FOR Z8 = Z7 TO NT
--------------------
11400  HPLOT ZB + Z(J,Z8) * Z1,ZD - Z(I,Z8) * Z2
  : NEXT
--------------------
11420 GOTO 10140
--------------------
11440 END
--------------------
12000 ZR = 2 * RND (15) - 1
  : ZR = LOG ((1 + ZR) / (1 - ZR)) / 1.82
  : RETURN
--------------------
14000 D$ = CHR$ (4)
--------------------
14010 INPUT "ENTER FILE NAME--";B$
--------------------
14020 PRINT D$;"OPEN ";B$
--------------------
14030 IF MID$ (A$,2,1) = "L" THEN 14150
--------------------
14040 IF MID$ (A$,2,1) < > "S" THEN 10140
--------------------
14050 PRINT D$;"DELETE ";B$
--------------------
14060 PRINT D$;"OPEN ";B$
--------------------
14070 PRINT D$;"WRITE ";B$
--------------------
14080 PRINT ZS
  : PRINT NT
  : PRINT NV
--------------------
14090 FOR I = 1 TO NV
  :   PRINT ZM(I)
--------------------
14100  FOR J = ZS TO NT
  :    PRINT Z(I,J)
--------------------
14110  NEXT J
  : NEXT I
--------------------
14120 PRINT D$;"CLOSE ";B$
--------------------
14130 GOTO 10140
--------------------
14150 PRINT D$;"READ ";B$
--------------------
14160 INPUT ZS
  : INPUT NT
  : INPUT NV
--------------------
14170 FOR I = 1 TO NV
  :   INPUT ZM(I)
--------------------
14180  FOR J = ZS TO NT
  :    INPUT Z(I,J)
--------------------
14190  NEXT J
  : NEXT I
--------------------
14200 GOTO 14120
--------------------
60000 REM  CAPTURES PROGRAMS IN SEQUENTIAL FILES
--------------------
60020 D$ = CHR$ (4)
--------------------
60030 INPUT "ENTER FILE NAME--";A$
--------------------
60040 PRINT D$;"OPEN ";A$
--------------------
60050 PRINT D$;"DELETE ";A$
--------------------
60060 PRINT D$;"OPEN ";A$
--------------------
60065 PRINT D$;"WRITE ";A$
--------------------
60070 POKE 33,30
--------------------
60080 LIST 60000,60110
--------------------
60090 PRINT D$;"CLOSE ";A$
--------------------
60100 TEXT
--------------------
60110 END
------END OF FILE------
```

AVAILABLE PUBLICATIONS

GREAT LAKES FISHERY COMMISSION
1451 Green Road
Ann Arbor, MI 48105

Mimeo. Reports (disontinued series)

79-1 TALHELM, D. R. et al. 1979. Current estimates of Great Lakes fisheries values: 1979 status report. Great Lakes Fish. Comm. Mimeo. Rep. 79-l. 18 p.

Special Publications

79-l KING, E. L., Jr. and T. A. EDSALL. 1979. Illustrated field guide for the classification of sea lamprey attack marks on Great Lakes lake trout. Great Lakes Fish. Comm. Spec. Pub. 79-1. 41 p.

92-l BERST, A. H., and G. R. SPANGLER. 1962. Recommendations for freshwater fisheries research and management from the Stock Concept Symposium (STOCS). Great Lakes Fish. Comm. Spec. Pub. 82-l. 24 p.

82-2 KOONCE, J. F. (ed.), L. GREIG, B. HENDERSON, D. TESTER, K. MINNS, and G. SPANGLER. 1982. A review of the adaptive management workshop addressing salmonid/lamprey management in the Great Lakes. Great Lakes Fish. Comm. Spec. Pub. 82-2. 40 p.

82-3 AUER, N. A. (ed.). 1982. Identification of larval fishes of the Great Lakes basin with emphasis on the Lake Michigan drainage. Great Lakes Fish. Comm. Spec. Pub. 82-3. 744 p.

83-1 KOONCE, J. F. (ed.), D. JESTER, B. HENDERSON, R. HATCH, and of. JONES. 1983. Quota management of Lake Erie fisheries. Great Lakes Fish. Comm. Spec. Pub. 83-l. 39 p.

83-2 MEYER, F. P., J. W. WARREN, and T. G. CAREY feds.). 1983. A guide to integrated fish health management in the Great Lakes basin. Great Lakes Fish. Comm. Spec. Pub. 83-2. 262 p.

**84-1** ESHENRODER, R. L., and J. F. KOONCE. 1984. Recommendations for standardizing the reporting of sea lamprey marking data. Great Lakes Fish. Comm. Spec. Pub. 84-l. 21 p.

84-2 ESHENRODER, R. L., T. P. POE, and C. H. OLVER (eds.). 1985. Working papers developed at the August 1983 conference on lake trout research. Great Lakes Fish. Comm. Spec. Pub. 84-2.

84-3 MINNS, C. K., J. U. COOLEY, and J. E. FORNEY. 1984. Analysis of the response to the use of “Adaptive Environmental Assessment Methodology” by the Great Lakes Fishery Commission. Great Lakes Fish. Comm. Spec. Pub. 84-3. 21 p.

**85-1** PAINE, J. R., and R. B. KENYON (eds.). 1985. Lake Erie fish community workshop (report of the April 4-5, 1979 meeting). Great Lakes Fish. Comm. Spec. Pub. 85-l: 58 p.

Technical Reports

Great Lakes Fishery Commission Technical Report Series No. l-40. 1956-1984.

Annual Reports

Great Lakes Fishery Commission Annual Reports. 1956-1982.

other

CROWE, W. R. 1975. Great Lakes Fishery Commission history, program, and progress. 22 p.

Proceedings of the session on fish genetics - fundamentals and implications to fish management. Supported by the Great Lakes Fishery Commission and co-sponsored with the American Fisheries Society. Slay 1O-12, 1977. 62 p.

Contaminant research needs in the Great Lakes. June 28, 1979. 55 p.

Report of the audit of the Great Lakes Fishery Commission’s program of sea lamprey control and research. 1980. 157 p.

The effect of environmental issues and programs on Great Lakes fisheries: Directions for the future. 1980. 55 p.

A joint strategic plan for management of Great Lakes fisheries. 1980. 23 p.

HEALEY, P. J., and P.W.H. BEAMISH. 1984. The cyclostomata an annotated bibliography supplement 1979-1983. 320 p.

Brochures

Sea lamprey management program
Lake trout rehabilitation in the Great Lakes

July 1985