PICES Scientific Report No. 27
2004

PICES-GLOBEC INTERNATIONAL PROGRAM ON CLIMATE CHANGE AND CARRYING CAPACITY

REPORT OF THE MODEL TASK TEAM SECOND WORKSHOP TO DEVELOP A MARINE ECOSYSTEM MODEL OF THE NORTH PACIFIC OCEAN INCLUDING PELAGIC FISHES

Edited by
Michio J. Kishi

November 2004
Secretariat / Publisher
North Pacific Marine Science Organization (PICES)
c/o Institute of Ocean Sciences, P.O. Box 6000, Sidney, B.C., Canada. V8L 4B2
E-mail: secretariat@pices.int   Home Page: http://www.pices.in

Table of Contents

Executive Summary

The Second International Workshop to Develop a Marine Ecosystem Model of the North Pacific Ocean including Pelagic Fishes was held at the Frontier Research System for Global Change in Yokohama, Japan, from March 3 to 6, 2003. The workshop was convened to build upon the results of three earlier marine ecosystem model workshops. The first was the NEMURO (**N**orth Pacific **E**cosystem **M**odel for **U**nderstanding **R**egional **O**ceanography) Workshop, held in Nemuro, Japan, in January 2000, to develop a marine ecosystem model based on lower trophic levels. Next was the MODEL Workshop, which took place in Hakodate, Japan during the PICES Ninth Annual Meeting in October 2000, to look at viable strategies for linking lower trophic level models to higher trophic models of the North Pacific ecosystem at regional and basin scales. The third was the NEMURO.FISH (NEMURO **F**or **I**ncluding **S**aury and **H**erring) Workshop, held in Nemuro, January 2002, in which a higher trophic level bioenergetic model for fish was linked with the lower trophic level NEMURO model.

Twenty-four scientists with expertise in plankton studies, modelling, and fisheries biology from Canada, Japan, Korea, and the United States participated in the Yokohama workshop chaired by Dr. Michio J. Kishi and Dr. Bernard A. Megrey.

The goals of the Yokohama 2003 MODEL Workshop were:

1. to develop a dynamically coupled two-way model that would include prey-predator interactions between lower trophic levels and pelagic fish, and
2. to develop a Lagrangian model that would include fish migration and fish population dynamics which could be embedded into a basin-scale three-dimensional circulation model.

Models were parameterized for Pacific saury (*Cololabis saira*) and herring (*Clupea harengus pallasi*).

This workshop produced a number of significant achievements:

1. A saury version of NEMURO.FISH was coupled with a population dynamics model and the effects of a two-way dynamics linkage between lower and higher trophic levels were tested.
2. A herring bioenergetics model was expanded to the population level and coupled to the lower trophic levels of the NEMURO model.
3. A Lagrangian model of NEMURO.FISH describing fish migration was developed and adapted to saury in the western North Pacific Ocean.

Based on these accomplishments, recommendations were made for future modelling activities. The importance of these achievements will be determined by how well the PICES-GLOBEC Climate Change and Carrying Capacity Program embraces these models and uses them as a basis for increasing our understanding of marine ecosystems and fisheries management in the North Pacific.

1. Workshop overview

1.1 Introduction

PICES supports the PICES/GLOBEC Climate Change and Carrying Capacity (CCCC) program in the temperate and subarctic regions of the North Pacific Ocean. Ecosystem modelling is one of five key research activities defined by the CCCC Implementation Panel. At a workshop held in 2000 in Nemuro, Japan, the MODEL Task Team developed NEMURO (**N**orth Pacific **E**cosystem **M**odel for **U**nderstanding **R**egional **O**ceanography), a lower trophic level marine ecosystem model. Since its initial development, NEMURO has become recognized internationally (*e.g.*, Fujii *et al.*, 2002). A MODEL Workshop, to extend the discussion initiated in Nemuro, was convened in Hakodate, Japan, in 2000. The focus of this workshop was to develop a conceptual model of the North Pacific ecosystem at regional and basin scales as well as examining strategies for linking lower trophic level (LTL) to higher trophic level (HTL) models of the ecosystem. Following these 2000 workshops, Dr. Michio J. Kishi submitted a proposal to the Heiwa Nakajima Foundation of Japan to help fund attendance at a subsequent workshop. The proposal was accepted and planning began for another workshop in Nemuro in 2002. At that workshop, an HTL bioenergetic model for fish was linked with the LTL NEMURO model. The new model was named NEMURO.FISH (NEMURO **F**or **I**ncluding **S**aury and **H**erring).

At the PICES Eleventh Annual Meeting held in Qingdao, China, in October 2002, the MODEL Task Team met and agreed it would be useful to extend NEMURO.FISH to include population dynamics and to incorporate the fish model into numerical circulation models using a Lagrangian approach. It was also considered important to implement feedback from the fish model to a lower trophic model. The Task Team agreed to have a follow-up workshop focusing on NEMURO.FISH in 2003, in Yokohama, Japan.

1.2 Goals and objectives of the workshop

The goals of the Yokohama 2003 MODEL Workshop were to develop a two-way model that would include prey-predator interactions between the lower trophic ecosystem and fish, and to produce a Lagrangian model that would include fish migration and fish population dynamics. These models were parameterized for Pacific herring (*Clupea harengus pallasi*) and Pacific saury (*Cololabis saira*).

1.3 Organizing committee, participants, sponsors and venue

Drs. Shin-ichi Ito, Michio J. Kishi, Bernard A. Megrey and Francisco E. Werner organized the meeting. Drs. Kishi and Megrey served as workshop co-chairmen. Twenty-four scientists from Canada, Korea, Japan, and the United States participated at the meeting (Fig. 1.1, Appendix 6.1). The group included plankton scientists, modellers, and fisheries biologists familiar with the growth and dynamics of herring and saury. Travel expenses of some participants were partly supported by the Heiwa Nakajima Foundation. A core group of scientists with knowledge about various aspects of the problem was assembled for intensive discussions and knowledge sharing, thereby providing a key to the success of the workshop.

The venue was a small lecture room (Fig. 1.2) at the offices of the Frontier Research System for Global Change (FRSGC), in Yokohama, Japan. FRSGC has a modern large computer "Earth Simulator" (Fig. 1.3) but there was no opportunity to use it although the group had a chance to tour the Earth Simulator facility during the workshop.

1.4 Workshop activity

First Session

In the morning session on the first day (see Appendix 6.2 for schedule), after the self-introductions, Dr. Kishi introduced NEMURO and showed some results of a coupled NEMURO-physical circulation model. Dr. Megrey introduced the concept of NEMURO.FISH, a coupling of NEMURO with fish bioenergetics models. Presentations on the adaptation of the model to saury and herring were made by Dr. Ito

Fig. 1.1 Saying "Cheese" by the workshop participants at the entrance hall of FRSGC. Back Row – Left to right: Goh Onitsuka, Kazuaki Tadokoro, Yasuhiro Yamanaka, Naoki Yoshie, Francisco E. Werner, Taketo Hashioka, Douglas E. Hay, Fei Chai, Kenneth A. Rose, Makoto Kashiwai. Front Row: Sinjae Yoo, Michio J. Kishi, Shin-ichi Ito, Toshio Katsukawa, Bernard A. Megrey, Daiki Mukai, Sachie Yoshimoto.

Fig. 1.2 Presentations during the workshop at FRSGC.

and Dr. Douglas E. Hay, respectively. Dr. Megrey explained the PICES MODEL Task Team website which he had made on a volunteer basis.

In the afternoon, Dr. Toshio Katsukawa explained his idea on population dynamics and Dr. Yasuhiro Yamanaka showed the results from a global model which includes NEMURO. After tea break, the group divided into task groups (TG) in separate breakout sessions.

Second Session

The second day was taken up primarily by the task groups dealing with their specific assignments:

Fig. 1.3 “Earth simulator” tour.

TG1 on a Population Dynamics Model

Build a 2-way model using a population dynamics coupled model (Ito, leader, and 13 participants).

TG2 on a Lagrangian Model

Make a model of NEMURO.FISH (Kishi and Yamanaka, leaders, and 7 participants).

During the meetings of task groups, a tour of the Earth Simulator was conducted (Fig. 1.3). The work of these groups continued until noon of the third day.

Third Session

The third session was held in the afternoon of the third day. The group discussed the structure and organization of the final report, issued writing assignments, generated a list of workshop recommendations, discussed where the MODEL Task Team should go next, and talked about the possibility of convening future workshops. Several individual seminars were presented by workshop participants dealing with their personal research topics.

2. NEMURO.FISH coupled with a population dynamics model (Saury)

Shin-ichi Ito[1], Bernard A. Megrey[2], Toshio Katsukawa[3], Naoki Yoshie[4], Goh Onitsuka[5], Kenneth A. Rose[6], Francisco E. Werner[7], Douglas E. Hay[8], Fei Chai[9], Lan Smith[10] and Michio J. Kishi[11]

[1] Tohoku National Fisheries Research Institute, 3-27-5 Shinhama-cho, Shiogama-city, Miyagi, 985-0001, Japan. E-mail: goito@affrc.go.jp
[2] Alaska Fisheries Science Center, National Marine Fisheries Service, NOAA, 7600 Sand Point Way NE, Seattle, WA, 98115-0070, U.S.A. E-mail: bern.megrey@noaa.gov
[3] Ocean Research Institute, University of Tokyo, 1-15-1 Minamidai, Nakano-ku, Tokyo, 164-8639, Japan. E-mail: katukawa@ori.u-tokyo.ac.jp
[4] Graduate School of Environmental Earth Science, Hokkaido University, Kita 10, Nishi 5, Kita-ku, Sapporo, Hokkaido, 060-0810, Japan. E-mail: naoki@ees.hokudai.ac.jp
[5] Prefectural University of Kumamoto, 3-1-100 Tsukide, Kumamoto, 862-8502, Japan
E-mail: onizuka@pu-kumamoto.jp
[6] Department of Oceanography and Coastal Sciences, 218 Wetlands Resources Building, Louisiana State University, Baton Rouge, LA, 70803, U.S.A. E-mail: karose@lsu.edu
[7] Department of Marine Sciences, University of North Carolina, Chapel Hill, NC, 27599-3300, U.S.A. E-mail: cisco@unc.edu
[8] Pacific Biological Station, Fisheries and Oceans Canada, 3190 Hammond Bay Rd, Nanaimo, BC, V9R 5K6, Canada. E-mail: hayd@pac.dfo-mpo.gc.ca
[9] School of Marine Science, University of Maine, 5471 Libby Hall, Orono, ME, 04469-5741, U.S.A. E-mail: fchai@maine.edu
[10] Frontier Research System for Global Change, 3173-25 Showa-machi, Kanazawa-ku, Yokohama, Kanagawa, 236-001, Japan. E-mail: lanimal@jamstec.go.jp
[11] Hokkaido University / Frontier Research System for Global Change, 3-1-1 Minato-cho, Hakodate, Hokkaido, 041-8611, Japan. E-mail: kishi@salmon.fish.hokudai.ac.jp

2.1 Introduction

Ito *et al*. (2002) developed a bioenergetics model for Pacific saury coupled with an ecosystem model. It was named NEMURO.FISH (**N**orth Pacific **E**cosystem **M**odel for **U**nderstanding **R**egional **O**ceanography. **F**or **I**ncluding **S**aury and **H**erring) (Megrey and Kishi, 2002). In that model, the growth rate of an individual Pacific saury is represented by the weight increment per unit of wet weight per time and is defined as the following:

$$\frac{dW}{W \cdot dt} = \left[C - (R + S + F + E + EP)\right] \cdot \frac{CAL_z}{CAL_f}$$

where W: wet weight of the fish (g), t: time (days), C: consumption (g prey g fish^{-1} d^{-1}), R: respiration or losses through metabolism (g prey g fish^{-1} d^{-1}), S: specific dynamic action or losses due to energy costs of digesting food (g prey g fish^{-1} d^{-1}), F: egestion or losses due to feces (g prey g fish^{-1} d^{-1}), E: excretion or losses of nitrogenous excretory wastes (g prey g fish^{-1} d^{-1}) and EP: egg production or losses due to reproduction (g prey g fish^{-1} d^{-1}). CAL_z and CAL_f are caloric equivalents of zooplankton (cal g zooplankton^{-1}) and fish (cal g fish^{-1}) respectively. This kind of model is very useful for analyzing the mechanism of fish growth variability. But to manage the stock of a specific fish, we need information on the biomass of the fish. As Megrey and Kishi (2002) pointed out, the fish biomass density B (g m^{-3}) can be written as the product of fish density N (m^{-3}) and the mean wet weight of individuals as

$$B = N \cdot W$$

and the rate of change of fish biomass can be written as

$$\frac{dB}{dt} = N\frac{dW}{dt} + W\frac{dN}{dt}$$

where the first term on the right side of the equation is due to changing individual weight (growth), and the second term is due to changing numbers of fish (through mortality and recruitment). The first term can be calculated by NEMURO.FISH if an estimate of fish density is available. The second term requires information about the rate of change of fish density that can be calculated from a population dynamics model such as

$$\frac{dN}{dt} = \left(-F - M\right) \cdot N + f(EP)$$

where *F:* instantaneous fishing mortality, *M:* instantaneous natural mortality, and *f(EP):* reproduction (defined as a function of egg production). If we define the parameters of the population dynamics model, we can calculate the change of fish biomass.

But in this case, the energy flow is only one way. The zooplankton density is not reduced by predation by fish, so there is no feedback into the internal dynamics of NEMURO. Nor does the depletion of prey by fish influence subsequent consumption by fish. To realize the two-way dynamic linkage of energy flow between higher trophic level (fish) and lower trophic level (phytoplankton and zooplankton), we made the following assumptions:

- zooplankton density is decreased by predation pressure from fish according to $C \cdot N$,
- excretion by fish $E \cdot N$ is converted to NH_4,
- egestion by fish $F \cdot N$ is converted to particulate organic nitrogen (PON),
- specific dynamic action $S \cdot N$ and respiration $R \cdot N$ by fish are not converted to any components.

During this workshop, we coupled the saury version of NEMURO.FISH with the population dynamics model and tested the effect of the two-way dynamics linkage to lower and higher trophic levels. We also investigated the model sensibility for mortality and reproduction rate. To make the problems simpler, we used a one-box NEMURO.FISH model. We drove NEMURO with the idealized seasonal forcing by prescribing SST (Sea Surface Temperature) and light intensity (Lint0) at the surface (Fig. 2.1). The temperature of the mixed layer is defined by the SST and the temperature at the bottom of the mixed layer (BLT) is set to be constant as the bottom boundary condition. The thickness of the mixed layer increases (decreases) when SST is lower (higher) than BLT. The exchange rate of NO_3 and $Si(OH)_4$ between the mixed layer and the bottom layer is changed as a function of the stability defined by the difference between SST and BLT. As a result, the mixed layer thickness showed seasonal variation as in Figure 2.2. We will investigate the effect of two-way linkage under this simple forcing case in the following sections.

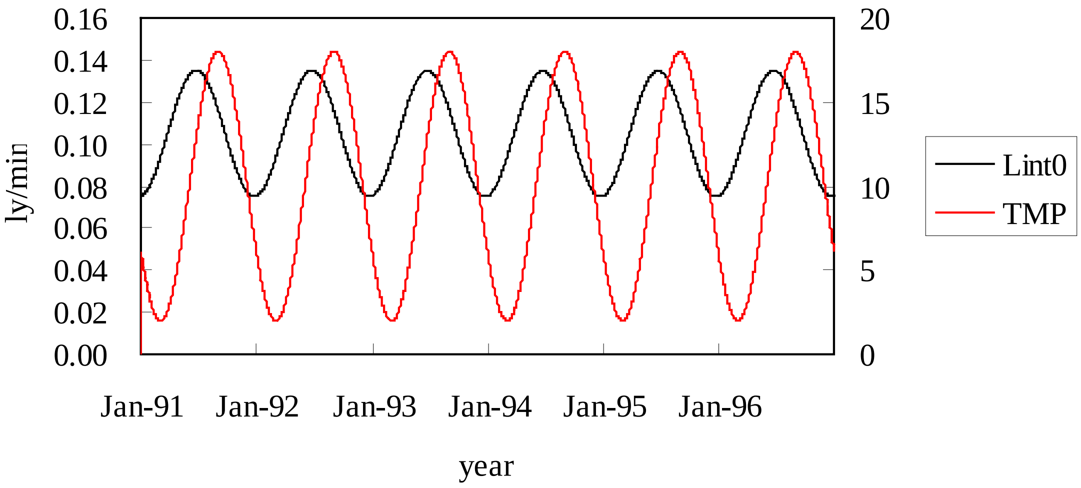

Fig. 2.1 Idealized seasonal forcing of light (Lint0) and temperature (TMP) for NEMURO.

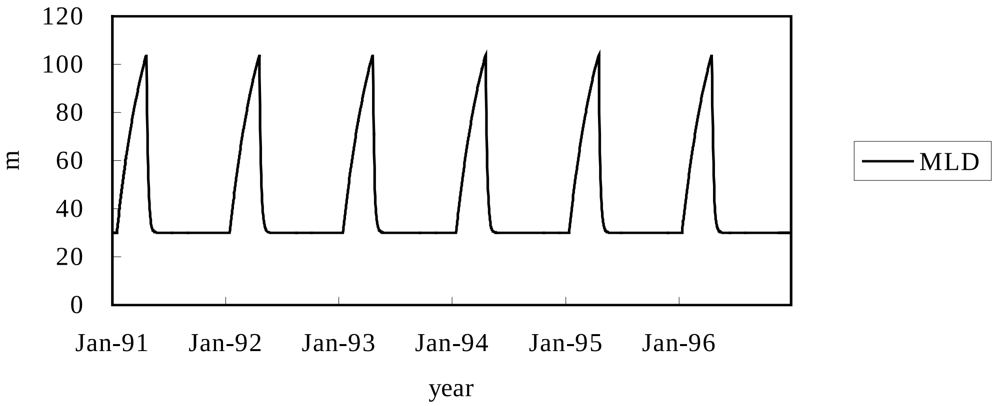

Fig. 2.2 Seasonal change of mixed layer thickness simulated in a NEMURO one-box model.

2.2 One cohort case with no reproduction

First, to test the feedback from the higher trophic level to the lower trophic level, we included only the effect of predatory pressure of fish on zooplankton, conversion of excretion to NH_4, and egestion to PON. The parameters of the saury bioenergetics model are almost identical to those used in Ito *et al.* (2002). The first experiment is simplified by ignoring reproduction and assuming a constant mortality rate so that population dynamics are represented by

$$\frac{dN}{dt} = \left(-F - M\right)\cdot N$$

assuming F: 0.20 year^{-1} and M: 0.15 year^{-1}. Although the life span of Pacific saury is assumed to be two years and two cohorts are usually incorporated into the model, we started with one non-overlapping cohort to make the problem simpler. We prescribe 0.001 m^{-3} as the initial value of N which corresponds to one fish for 10 m^2 area with a 100-m mixed layer depth. Differences between two-way and one-way linkages are shown in Figures 2.3 and 2.4.

The life history stages of saury were defined in Table 2.1. We assumed that saury feed only on small zooplankton (ZS) during stage 1, while in stage 2 they feed on ZS and large zooplankton (ZL), and then on ZL and predatory zooplankton (ZP) in stage 3.

Saury begin to eat ZL and ZP after July in the first year, but at the beginning of September, the ZL component migrates to the deep layer and saury should feed on ZP only. Then, the predatory pressure from saury on ZP increases and the density of ZP decreases rapidly after the beginning of September in the two-way case. This decrease of ZP in the two-way case continues until the beginning of February of the second year when the saury cohort reaches the end of its lifespan. The reduction of ZP causes a trophic cascade to their prey, so ZL and ZS increase. The increase of ZL is only slight because saury also feed on ZL.

The decrease of ZP and the increases of ZS and ZL in the two-way case were not significant from the middle of February to the end of April (spring bloom season). In this season, the water temperature is <5ºC (Fig. 2.1) and the consumption rate decreases rapidly because the Thorton and Lessem (1978) temperature-adjustment function is applied to the consumption-temperature dependency and the lower limit temperature parameter is defined as 5ºC. Due to the low temperatures in spring, there was no significant difference between the one-way and two-way cases in the spring bloom.

Table 2.1 Life history stages assumed for Pacific saury.

Stage	Period	Age
1	February 1 of the first year to March 1 in the first year	0-30 days
2	March 1 in the first year to July 1 in the first year	30-150 days
3	July 1 in the first year to February 1 in the second year	150-730 days

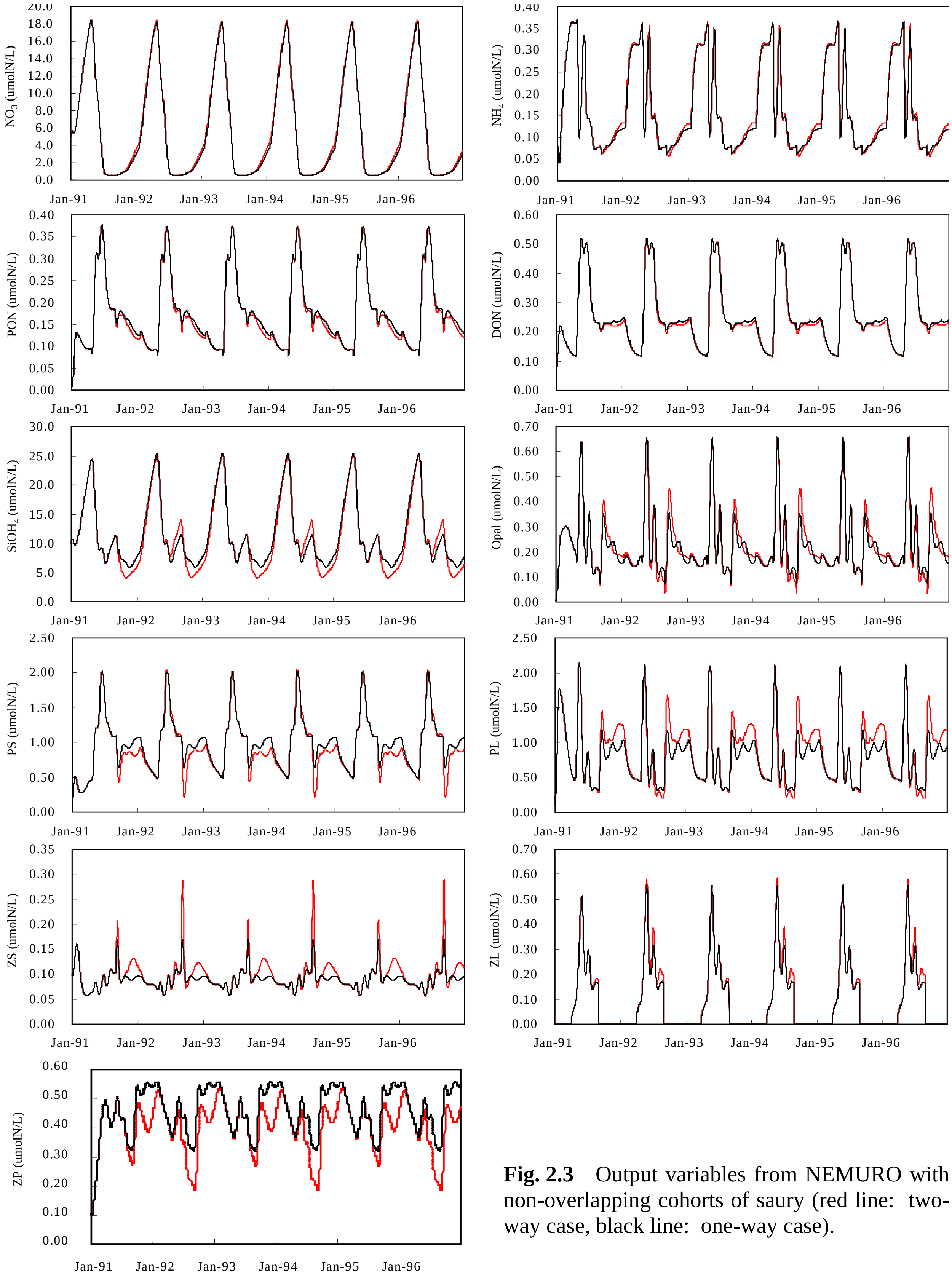

Fig. 2.3 Output variables from NEMURO with non-overlapping cohorts of saury (red line: two-way case, black line: one-way case).

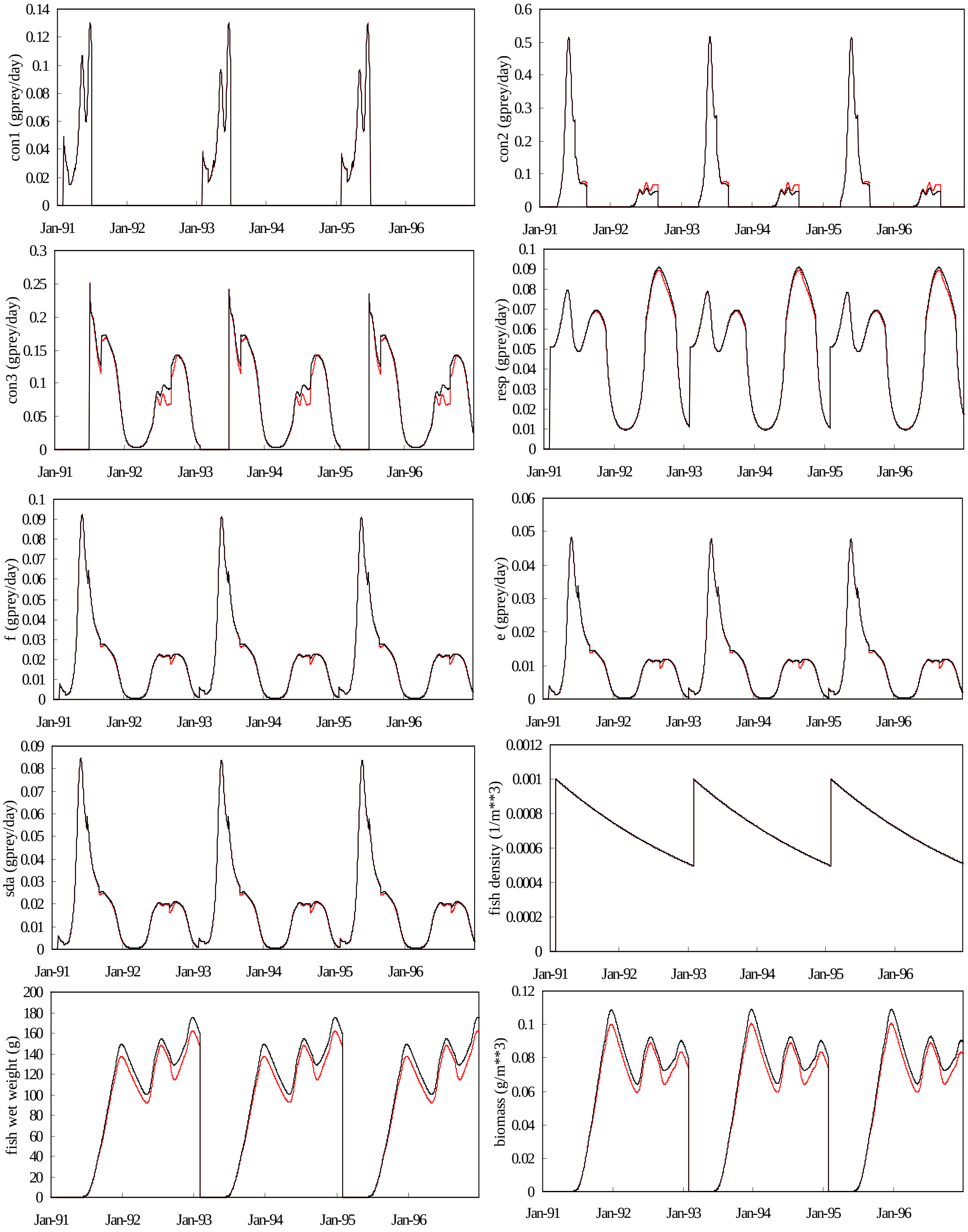

Fig. 2.4 Each component of the fish bioenergetics model coupled with the population dynamics model and output fish wet weight, fish density m^{-3}, and fish biomass m^{-3} for the non-overlapping saury cohort scenario (red line: two-way case, black line: one-way case).

Small phytoplankton (PS) decreased in autumn and winter, large phytoplankton (PL) decreased in summer and increased in autumn and winter in the two-way case compared with the one-way case. The increase of ZL was thought to cause an increase in the predatory pressure on PL in summer. The increase of ZS increased the predatory pressure on PS, and the decrease of ZP decreased the predatory pressure on PL in autumn and winter in the two-way case.

NO_3 and NH_4 increased slightly and PON and dissolved organic nitrogen (DON) decreased slightly in winter in the two-way case, though saury produced PON and DON in the two-way case. The increases of NO_3 and NH_4 can be explained by a decrease of total amount of phytoplankton. The decreases of PON and DON can be also explained by the decrease in the total amount of zooplankton. $SiOH_4$ decreased and opal increased in winter in the two-way case and these can be also explained by the increase of PL in autumn and winter in the two-way case.

In the saury bioenergetics model, the consumption term did not change significantly for stages 1 and 2 because the predatory pressure on zooplankton was low in these stages and the zooplankton density in the two-way case was almost the same as that in the one-way case. But in stage 3, the consumption rate decreased in the two-way case compared with that in the one-way case because the predatory pressure was high and the zooplankton density was reduced. For stage 3, consumption of ZL increased and ZP decreased according to the density change of ZL and ZP. Total consumption rate decreased slightly in the two-way case and resulted in slightly slower saury growth. In this case, where the mortality rates were fixed, the number of surviving fish were identical, and the biomass decrease in the two-way case was due solely to reduced growth rate.

2.3 Two (overlapping) cohort scenario with no reproduction

Though only one cohort was included in the previous section, the life span of saury was assumed to be two years. Thus, at any given time, there will be an overlapping of two cohorts and attention was focused on this situation.

The output from NEMURO.FISH coupled with a population dynamics model with two overlapping cohorts of saury is shown in Figure 2.5. In the two-overlapping cohort case, the effects of the two-way dynamic linkage were emphasized, especially in the summer to autumn seasons. Saury becomes stage 3 at the beginning of July in this model. From July to January, two adult cohorts exist, which are simultaneously exploiting the zooplankton prey. Again, in late winter the temperature becomes too low to maintain high predatory pressure so fish growth was slowed more drastically in the two-cohort case.

2.4 Two-cohort case with no reproduction and body size-dependent mortality

Though mortality rates were fixed in previous sections for simplicity, they are usually considered to be a function of body length (or life stages) and environment. For example, Miller *et al.* (1988) proposed a size-dependent probability function for prey capture success per attempt (*Cs*) as

$$Cs = \left(\left(Lp / Ll + 3.37\right) / 44.76\right)^{-2.28} \quad (\%)$$

where *Lp* is body size of a predator and *Ll* is body size of larvae. For saury, knob length (hereafter *KL*) is commonly used instead of total body length and we adapted the following mortality function using a Miller *et al.* (1988) type formulation;

$$M = \frac{100 - \left(\left(Lp / KL + 3.37\right) / 44.76\right)^{-2.28}}{100} \quad (\text{year}^{-1})$$

The body size of a predator is thought to depend on prey body size and we assumed the dependency to take the following form

$$Lp = 20 + 2 \cdot KL \quad (\text{cm}),$$

where *KL* is calculated from wet weight using

$$KL = 6.13 \cdot W^{0.33} / 10 \quad (\text{cm}),$$

although *KL* does not shrink even if the wet weight decreases and keeps its value until the wet weight recovers to the previous value. In this case, the *Lp* and *M* become like that shown in Figure 2.6.

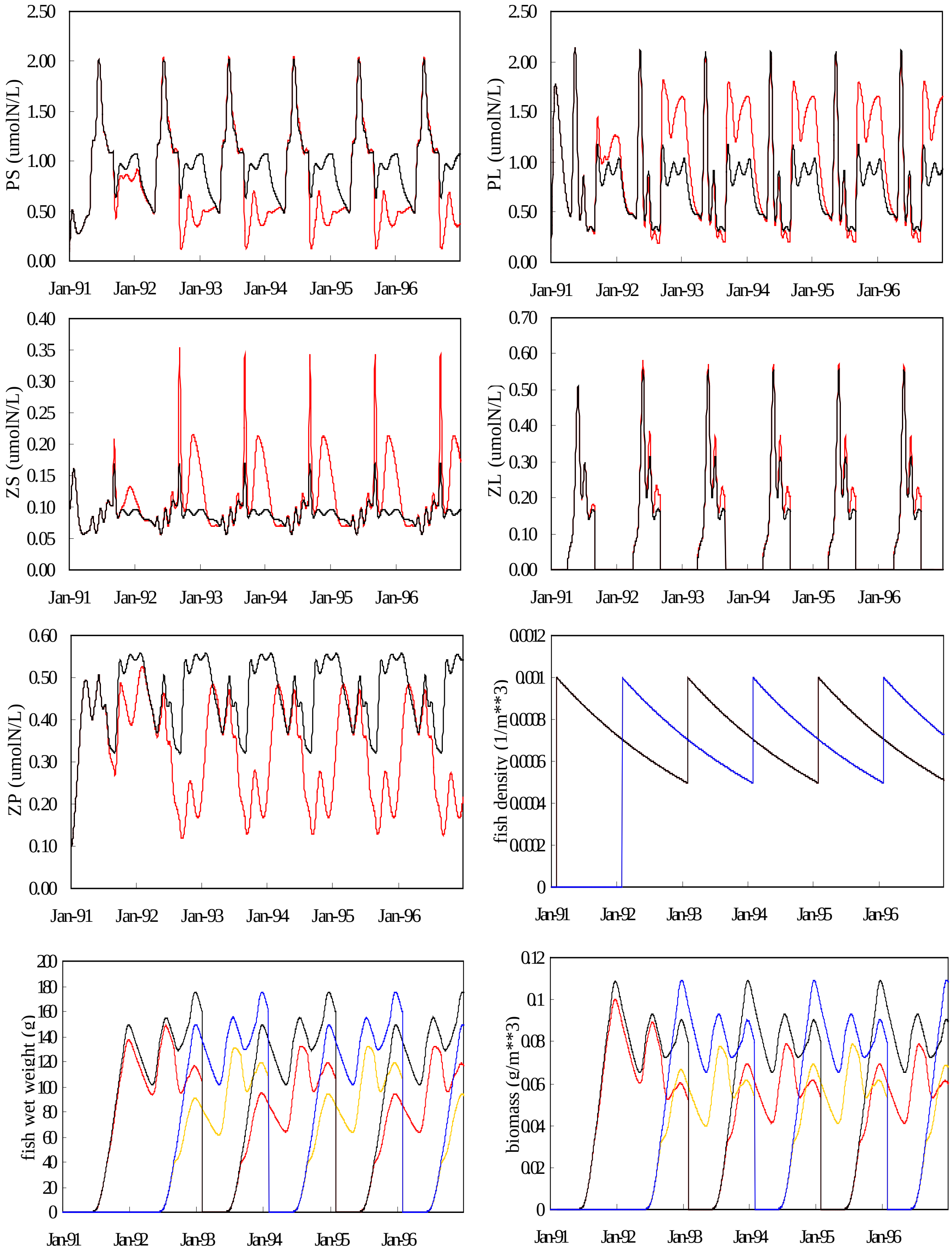

Fig. 2.5 Phytoplankton (PS, PL) and zooplankton (ZS, ZL, ZP) density and numbers, wet weight, biomass of saury calculated by NEMURO.FISH with two overlapping cohorts, saury (red: two-way first cohort, orange: two-way second cohort, black: one-way first cohort, blue: one-way second cohort).

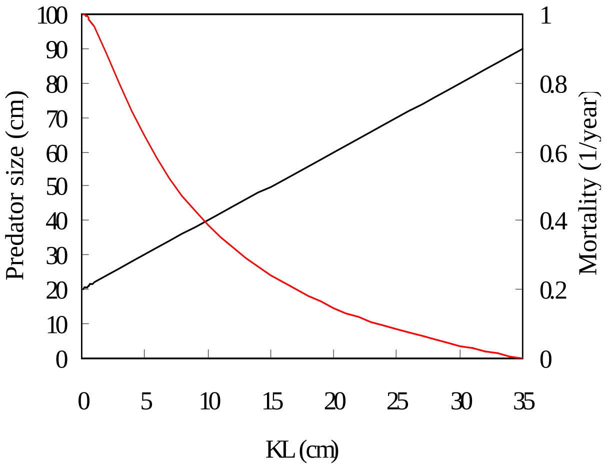

Fig. 2.6 Predator size (black line) and mortality (red line) as functions of knob length of saury.

Using this pattern for natural mortality rather than a constant value, 0.15 year^{-1}, we compared the results of NEMURO.FISH coupled with a population dynamics model between two-way dynamic linkage and one-way linkage (Fig. 2.7). The differences between the two-way and one-way models diminished in all compartments except for the numbers of fish. Natural mortality rate depended on *KL* and as a result, the natural mortality increased compared to the constant mortality case (Fig. 2.9). Then, the numbers of fish decreased in the *KL*-dependent mortality case compared with the constant mortality case. As a result, the predatory pressure from fish to zooplankton decreased and the effect of the two-way linkage was decreased.

It should be noted that the numbers of fish decreased in the two-way case when natural mortality depended on *KL*. The mechanism was thought to be because of the two-way dynamic linkage. ZP density decreased and the growth of saury also slowed down in stage 3 during the first year (Figs. 2.7 and 2.8). From spring to summer of the second year, the growth rate increased in the two-way linkage because the fish size was smaller in the two-way linkage case. Growth was again reduced after the ZL migrated to the deeper layer. This slowing of growth exposed the saury to higher mortality rates for a longer period, thereby reducing survival (Fig. 2.9). So the numbers of fish decreased in the two-way case. This mechanism seems to be important because it reflects environmental effects; low prey density brings low growth and survival. So, if this model was driven by inter-annual forcing and the zooplankton densities varied from year to year, the model results will show interannual variation of mortality and growth of fish. If the parameters in the model are estimated correctly, this model will become very useful for fish stock management.

2.5 Two-cohort case with reproduction and *KL*-dependent mortality

In the previous sections, reproduction was ignored and the initial density of renewal recruitment was fixed to 0.001 m^{-3} for simplicity. Ito *et al.* (2004) parameterized egg production as a constant fraction of consumption rate, but in this section a simple reproduction process is included. We assume that each surviving 1-year-old saury spawns 1.4 larvae on February 1 and loses 20 g wet weight. Each surviving 2-year-old saury spawns 1.4 larvae on February 1 and dies. The value 1.4 was estimated from the ratio of the initial value of *N* (0.001 m^{-3}) to the sum of 1-year-old (0.00440 m^{-3}) and 2-year-old (0.00352 m^{-3}) population densities; 0.001/(0.00352 + 0.0044). This estimation was made only to keep the population dynamics model stable. Observations showed that a female saury contains, on average, 2,400 eggs in one batch and spawns once in 3.7 days (Kurita, personal comm.). However, we don’t know the success rate of larval production from eggs. Therefore, we cannot judge whether the value 1.4 is appropriate or not.

Using the reproduction processes above, we integrated the model and got the results in Figure 2.10 showing that fish numbers decreased slowly and the maximum fish wet weight was increasing slowly as generations are changed. Now, there should be a negative feedback mechanism where the decrease of fish numbers decreases predatory pressure on zooplankton by fish and brings an increase of zooplankton. It brings high growth of fish and decreases the mortality and increases the fish numbers, but the feedback mechanism is very weak and the numbers of fish continued to decrease.

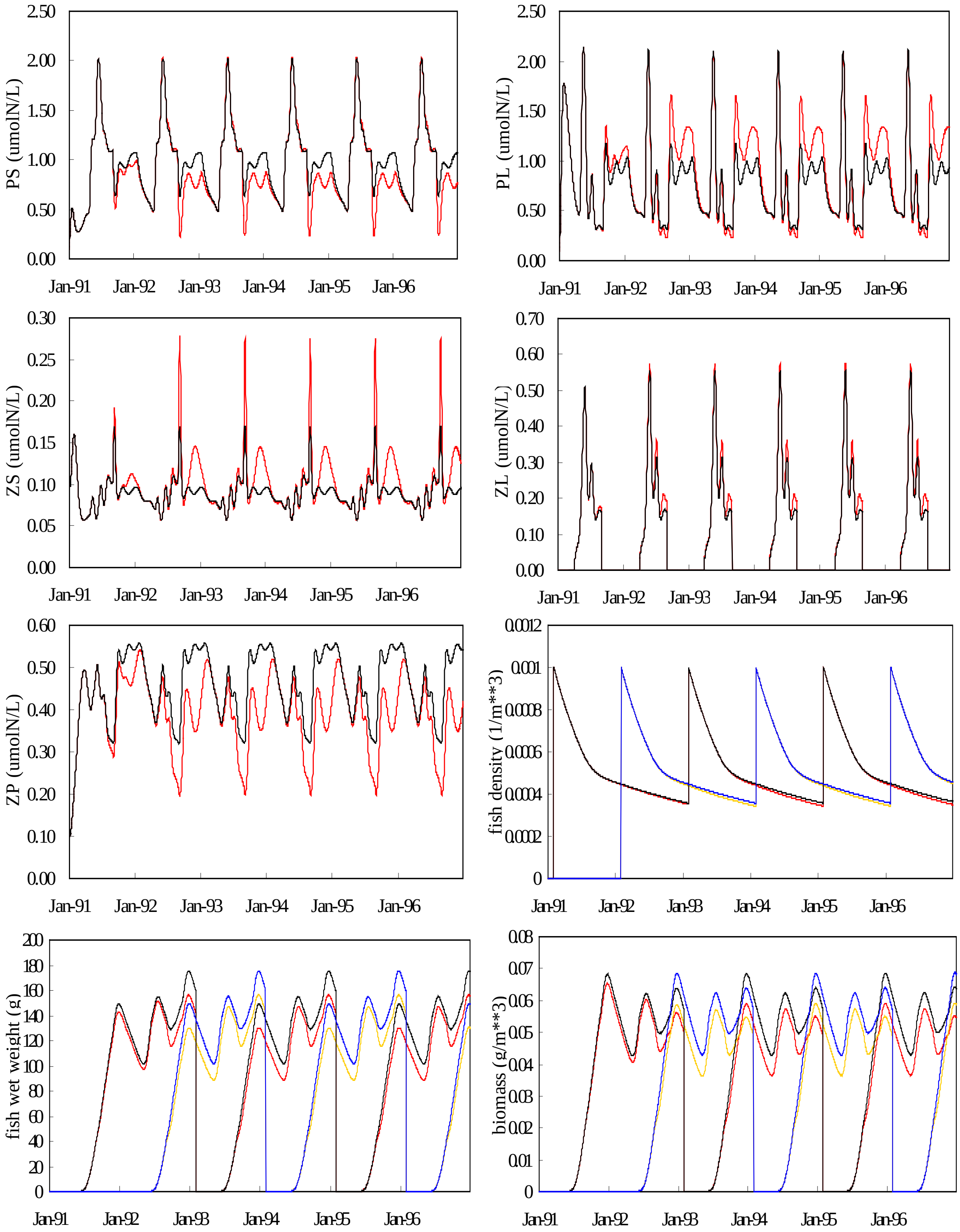

Fig. 2.7 Phytoplankton and zooplankton density and numbers, wet weight, biomass of saury calculated by NEMURO.FISH with two-cohort saury using *KL*-dependent natural mortality (red: two-way first cohort, orange: two-way second cohort, black: one-way first cohort, blue: one-way second cohort).

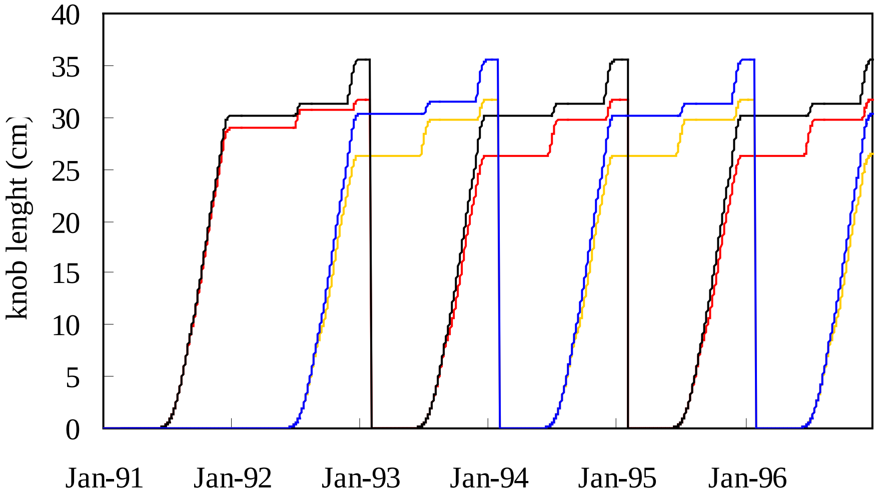

Fig. 2.8 Saury knob length calculated by NEMURO.FISH from a two-cohort model using *KL*-dependent natural mortality (red: two-way first cohort, orange: two-way second cohort, black: one-way first cohort, blue: one-way second cohort).

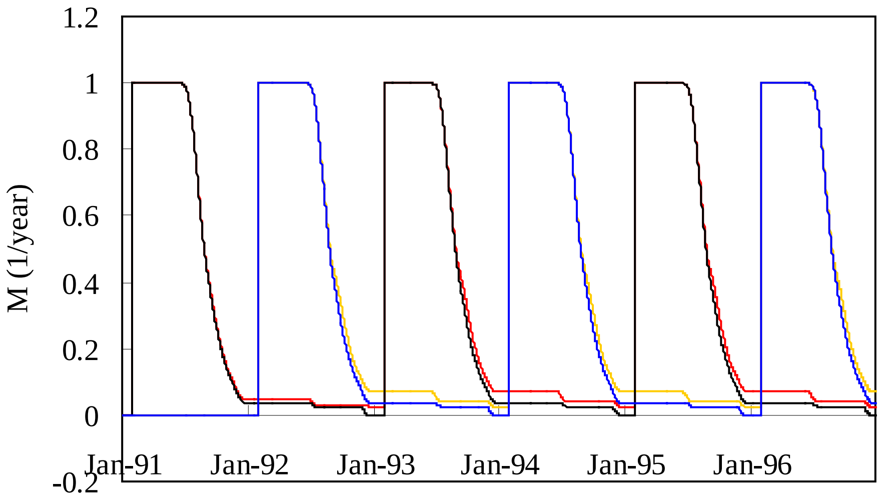

Fig. 2.9 Saury natural mortality rate derived from *KL* calculated by NEMURO.FISH (two-cohort model with *KL*-dependent natural mortality) (red: two-way first cohort, orange: two-way second cohort, black: one-way first cohort, blue: one-way second cohort).

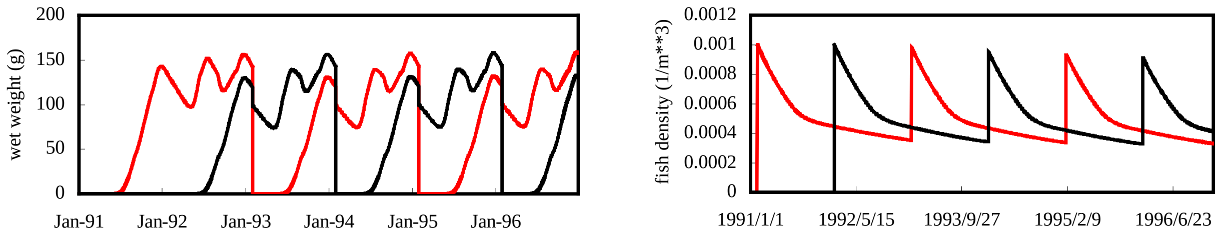

Fig. 2.10 Wet weight and numbers of saury calculated by NEMURO.FISH coupled with a population dynamics model including a reproduction process. Since the life span of saury is two years, there are two cohorts in the model (red: odd year cohort, black: even year cohort).

2.6 Conclusions and future perspectives

We coupled a lower-higher trophic ecosystem model (NEMURO.FISH) with a population dynamics model and tested the basic features of the model. The two-way dynamic linkage of the model showed that:

a) ZP decreased and ZL and ZS increased,
b) PS decreased in autumn and winter and PL decreased in summer and increased in autumn,
c) NO_3 and NH_4 increased slightly and PON and DON decreased slightly in winter,
d) $SiOH_4$ decreased and opal increased in winter,
e) saury wet weight decreased.

All of these results could be explained by the predatory pressure on ZP by saury. Including two cohorts emphasized the effect of the two-way dynamic linkage. The mortality dependency on *KL* (body size) also results in a decrease in fish numbers which modifies the effect of the two-way linkage by a negative feedback mechanism. Finally, a simple reproduction process was included though it only showed the possibility of involving a reproduction process.

The model described here is a prototype and the parameters used are not necessarily those found in nature. In addition, the model consists of only a single box and at least a three-box model is needed for Pacific saury (Ito *et al.*, 2004). Also, the winter temperature is too low in this one-box model so the response may change when an appropriate forcing is applied. As future perspectives, we recommend the following:

a) use a three-box model,
b) determine an appropriate initial fish density from observational data,
c) evaluate an appropriate mortality function from observational data,
d) include a proper reproduction process,
e) include the dependency of reproductive success on adult fish condition, prey density, temperature, *etc*.,
f) adjust the unknown parameters by adjusting the resultant biomass to observational data.

This kind of model may be very useful to fish stock management because the model includes both biological mechanisms and environmental effects. We eagerly recommend the development of this model in the future.

3. Lagrangian model of NEMURO.FISH

Michio J. Kishi[1] and Yasuhiro Yamanaka[2]

[1] Hokkaido University / Frontier Research System for Global Change, 3-1-1 Minato-cho, Hakodate, Hokkaido, 041-8611, Japan. E-mail: kishi@salmon.fish.hokudai.ac.jp

[2] Graduate School of Environmental Earth Science, Hokkaido University, Kita 10, Nishi 5, Kita-ku, Sapporo, Hokkaido, 060-0810, Japan. E-mail: galapen@ees.hokudai.ac.jp

3.1 Tasks and members

This task group developed a Lagrangian model of NEMURO.FISH and applied it to Pacific saury as an example. The members of the TG were: Yasuhiro Yamanaka, Michio J. Kishi, Goh Onitsuka, Taketo Hashioka, Sinjae Yoo, Naoki Yoshie and Daiki Mukai. Maki Aita-Noguchi of the Frontier Research System for Global Change provided monthly averaged sea surface currents, SST, SSS and ecosystem compartments (biomasses) from a 3-D coupled biological-physical model in which the biology is given by NEMURO.

3.2 Description of model and preliminary results

The Lagrangian model is coded as follows:

(1) The spawning area is indicated;
(2) Hatched fish are advected by surface currents (in the case of Pacific saury);
(3) Small and large phytoplankton, small, large and predatory zooplankton (PS, PL, ZS, ZL, ZP) of the grid where the advected fish exist are given as the food of fish;
(4) Fish swim toward the maximum temperature gradient at a speed determined by NEMURO.FISH.

In the case of winter-spawned saury, eggs were assumed to be spawned on February 1. Eggs that will hatch in a few days are transported by advection of currents only. After growing (June 1 in this case) the saury begin a northward migration towards the maximum temperature gradient (it was assumed that saury have a preference for cold water) with a speed that is a function of body size (output from NEMURO.FISH) and also advection (current velocity). After October 1 saury were assumed to migrate southward toward their original spawning site. The trajectory of the saury is shown in Figure 3.1. Saury are advected eastward by Kuroshio until June 1 and then migrate northward reaching Kamchatka, and return to their original position to spawn. (On the migration route, if a fish interacts with land, it is relocated to an offshore position.) On the way, they feed on zooplankton (ZS, ZL, ZP) concentrations provided by the 3-D physical-biological coupled model (Aita *et al.*, 2003). Figure 3.2 shows the time-dependent value of swimming speed (Migration speed is given as a function of body length; Appendix 6.3), temperature where the saury exist, together with body length, body weight and number. The numbers decay exponentially, the details of which are shown in the previous paper. Figure 3.2 also shows the concentration of zooplankton at the point where the saury are found which is given from the results calculated by the 3-D NEMURO model. The files of the 3-D NEMURO are saved as monthly averaged values at each grid point, the

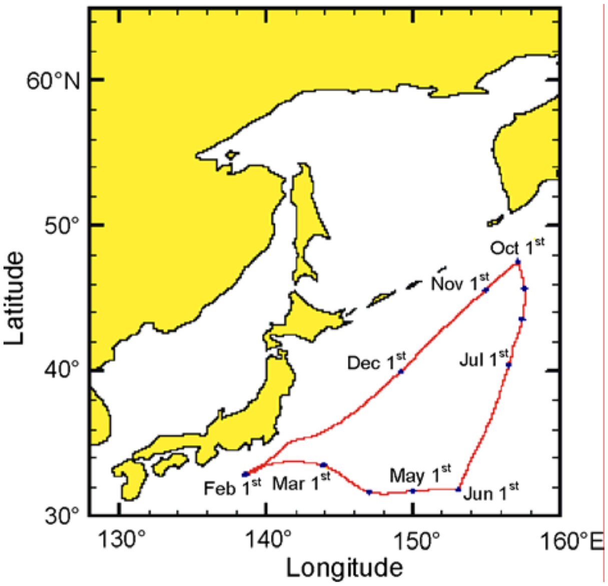

Fig. 3.1 Saury track from a Lagrangian model.

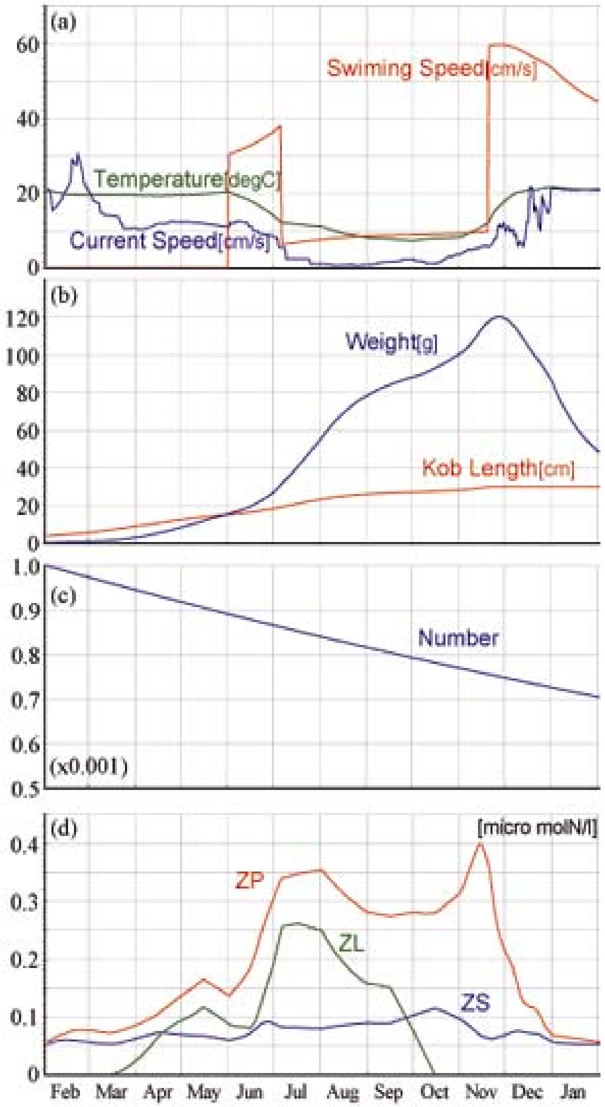

Fig. 3.2 Time-dependent values of (a) swimming speed and temperature where the saury exist, (b) body length and body weight and (c) number of saury. (d) concentration of zooplankton at the point where the saury exist which is given from the results files calculated by the 3-D NEMURO model.

size of which is 1° in the horizontal. Spatially and temporally interpolated values of the upper 100 m are used as a food of saury.

The FORTRAN program for the Lagrangian model is given in Appendix 6.3. To run this model, files containing horizontal velocity, temperature, ZS, ZL, and ZP (output from the 3-D model) are necessary:

```
open( 30, file='japan-u.dat', form='formatted',
      access='sequential')
open( 31, file='japan-v.dat', form='formatted',
      access='sequential')
open( 32, file='japan-t.dat', form='formatted',
      access='sequential')
open( 33, file='japan-zoos.dat', form='formatted',
      access='sequential')
open( 34, file='japan-zool.dat', form='formatted',
      access='sequential')
open( 35, file='japan-zoop.dat', form='formatted',
      access='sequential')
```

For a better understanding, the grid of our 3-D model is shown in Figure 3.3 and topography is shown in Figure 3.4.

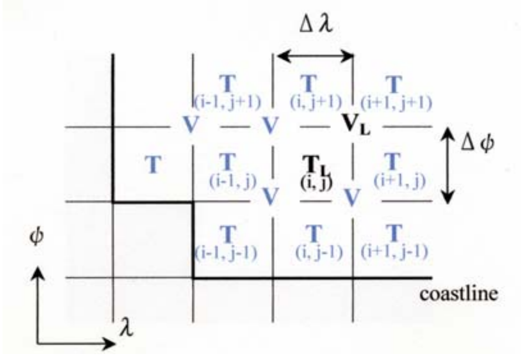

Fig. 3.3 Distribution of grid point (*cf*., Appendix 6.3). V: velocity point, T: tracers point.

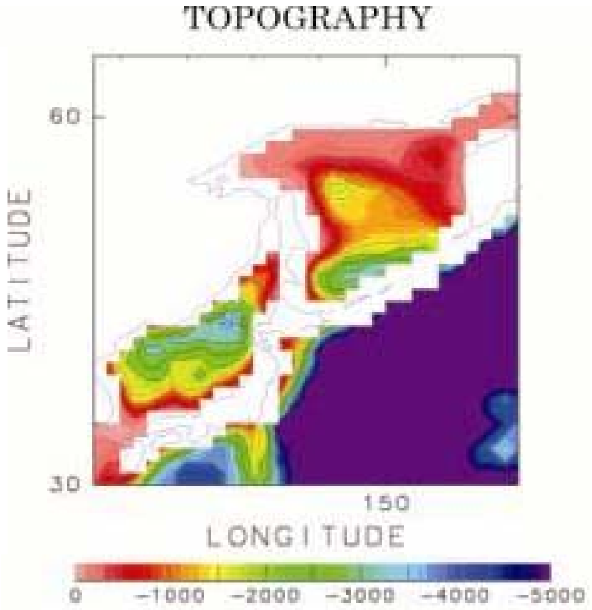

Fig. 3.4 Topography around Japan (modelled spatial domain of the saury Lagrangian model).

3.3 Future tasks

The feedback from saury grazing and egestion were not considered. Two-way coupling, *e.g.*, including feedback from fish via resource depletion and egestion is very difficult in linked Eulerian-Lagrangian models. However, this feedback must be included in the 3-D NEMURO model in order to consider the response of fish to climate change.

4. Coupling NEMURO to herring bioenergetics

Kenneth A. Rose[1], Bernard A. Megrey[2], Douglas E. Hay[3] and Francisco E. Werner[4]

[1] Department of Oceanography and Coastal Sciences, 218 Wetlands Resources Building, Louisiana State University, Baton Rouge, LA, 70803, U.S.A. E-mail: karose@lsu.edu

[2] Alaska Fisheries Science Center, National Marine Fisheries Service, NOAA, 7600 Sand Point Way NE, Seattle, WA, 98115-0070, U.S.A. E-mail: bern.megrey@noaa.gov

[3] Pacific Biological Station, Fisheries and Oceans Canada, 3190 Hammond Bay Rd, Nanaimo, BC, V9R 5K6, Canada. E-mail: hayd@pac.dfo-mpo.gc.ca

[4] Department of Marine Sciences, University of North Carolina, Chapel Hill, NC, 27599-3300, U.S.A. E-mail: cisco@unc.edu

4.1 Overview

During the 2003 Yokohama Workshop (March 3–6, 2003), a herring bioenergetics model was expanded to the population level and fully coupled to the lower trophic levels of the NEMURO model. The coupled model (denoted NEMURO_Herring) is now capable of simulating the daily dynamics of the lower trophic levels, and the daily average weight and numbers of individual herring in each of 10 age-classes over multiple years. The lower trophic level represented in NEMURO and the herring population dynamics are fully coupled in NEMURO_Herring. The dynamics of the three zooplankton groups in NEMURO determine the consumption rate of the average herring in each age-class, and thereby influence the growth rates and sizes of the herring. The densities of zooplankton eaten by all classes of herring are removed as a mortality rate on the zooplankton groups. Herring excretion contributes to the nitrogen dynamics by adding to the ammonia compartment, and herring egestion adds to the particulate organic nitrogen (PON) compartment. Thus, the lower trophic level dynamics and the herring dynamics are solved simultaneously in NEMURO_Herring.

The bioenergetics model that simulated the growth rate of an individual herring was developed at the NEMURO Workshop held January 25–27, 2002 (Megrey *et al.*, 2002a and 2002b). At this workshop, the bioenergetics model was also coupled to NEMURO model, but the two models were not solved simultaneously (*i.e.* a static link, Megrey *et al.*, 2002a and 2002b). The bioenergetics model simulated the growth (weight over time) of an individual herring, but there were no feedbacks between herring consumption, excretion and egestion, and the dynamics of the lower trophic levels. The amount of zooplankton prey eaten was not removed from zooplankton densities as a predation mortality, and ammonia and PON were unaffected by herring processes.

At the 2003 Yokohama Workshop, the participants used the most up-to-date NEMURO FORTRAN code, which was the same code used to couple NEMURO to saury bioenergetics, and developed a single code (NEMURO_Herring) that included a population-level version of the herring bioenergetics model and included the feedbacks between herring and the lower trophic levels. We roughly configured the NEMURO_Herring model for the coastal waters off Vancouver Island, a region where herring have been well-studied. We modified the NEMURO model by eliminating some of the features that had been recently added to NEMURO to simulate the lower trophic levels appropriate for saury (*e.g.*, vertical migration of zooplankton, mixing between deep and upper water layers). We also continued to synthesize the information on the life history of herring in the Vancouver Island area which was started at the 2002 NEMURO Workshop, including where (inshore versus offshore) and when (months) during the year different life stages, and mortality rates, life stage densities, and weights-at-age of herring are found. More fine-tuning is required for the NEMURO_Herring model to accurately mimic the known temporal patterns of nitrogen, plankton, and herring in the Vancouver Island region.

4.2 Details of the NEMURO_Herring model

The NEMURO model simulates the daily dynamics of 12 compartments, which comprise the lower trophic levels of the marine food web. These compartments include: nitrate, nitrite, ammonia, dissolved organic nitrogen, particulate organic nitrogen, silicate, opal, small phytoplankton, large phytoplankton, small zooplankton, large zooplankton, and predatory zooplankton. Processes represented include photosynthesis, respiration, excretion, nitrification, and detrital recycling of nitrogen. NEMURO has been fully described in previous reports (Eslinger *et al.*, 2000; Megrey *et al.*, 2000; Kishi *et al.*, 2001).

The previously developed bioenergetics model simulated the daily weight of an individual herring, using as output from the NEMURO model daily densities of the small, large, and predatory zooplankton used as prey for the herring. The bioenergetics model was described in detail in Megrey *et al.* (2002a, 2002b); we briefly summarize the bioenergetics model here. The growth rate in weight (grams wet weight, denoted g ww) of an individual herring was computed as:

$$\frac{dW}{dt} = \left[C - (R + S + F + E)\right] \cdot \frac{CAL_z}{CAL_f} \cdot W \qquad (4.1)$$

where the symbols are defined as C: consumption, E: excretion or losses of nitrogenous excretory wastes, F: egestion or losses due to feces, R: respiration or losses through metabolism, S: specific dynamic action or losses due to energy costs of digesting food, W: the weight of the fish (g ww), t: time (days), CAL_z: the caloric equivalent of zooplankton (cal·g zooplankton^{-1}) and CAL_f: the caloric equivalent of fish (cal·g fish^{-1}). The units of C, E, F, S, and R are g prey·g fish^{-1}·d^{-1}. The ratio of zooplankton to herring caloric densities converts all of the process rates from g prey·g fish^{-1}·d^{-1} to g fish·g fish^{-1}·d^{-1}.

Consumption was computed as a multispecies functional response (Rose *et al.*, 1999) with the small, large, and predatory zooplankton as prey types:

$$C_j = \frac{C_{MAX} \cdot \dfrac{PD_j \cdot v_{ij}}{K_{ij}}}{1 + \sum_{k=1}^{n} \dfrac{PD_k \cdot v_{ik}}{K_{ik}}} \qquad (4.2)$$

where C_j is the consumption rate of the jth zooplankton group (j = 1 is small, j = 2 is large, and j = 3 is predatory zooplankton) by the ith herring, PD_j is the density of the jth zooplankton group (in g ww·m^{-3}) on each day, C_{MAX} is the maximum consumption rate (g prey·g fish^{-1}·d^{-1}) determined from fish weight and water temperature, v_{ij} is the vulnerability of the jth zooplankton group to the ith herring (assumed one in our application), and K_{ij} is the half-saturation coefficient of the jth zooplankton group to the ith herring (g ww·m^{-3}). Notations v and K are subscripted for each herring to allow for vulnerability and half-saturation coefficients to be specified by herring size or age. However, in our application to date of the NEMURO_Herring model, v and K only vary by zooplankton group and are assumed the same for all herring. Consumption by the herring (C in equation 4.1) is the sum of the consumption rates over the three zooplankton groups ($C = \sum C_j$). Respiration was computed dependent on herring weight and temperature. Egestion was computed as a fraction of consumption, and excretion and S were computed as fractions of assimilated energy (consumption minus egestion).

In NEMURO_Herring we represented the herring population as 10 age-classes, and we tracked the daily number of individuals in each age class and the average weight of an individual in each age class. The previously developed bioenergetics model of an individual herring was used to simulate the weight of an average herring in each age-class over time. All herring in an age-class were assumed to be identical, and so all grew as if they all weighed the average weight.

The numbers of individuals in each class were updated daily based on specified mortality rates. The initial number of age-0 individuals was specified as a fixed number entering each year (*i.e.*, assumes constant recruitment of age-0 to the population). We began a simulation with specified numbers in each class as initial conditions. An important bookkeeping issue is ensuring that the herring in the model "see" the appropriate lower trophic level conditions, which vary seasonally. A representative herring life cycle for our region of interest is shown in Figure 4.1. Herring spawn in late winter, which results in age-0 individuals that begin model simulations to enter in mid-July (assumed July 17) at 2.0 grams wet weight. On July 17 of each subsequent year, surviving individuals are promoted to the next age-class, and a new number of age-0 individuals are specified.

Jan	Feb	Mar	Apr	May	June
	eggs		**larvae**		**postlarvae**

0+ juvenile

Month	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	June
Age (mo)	4	5	6	7	8	9	10	11	12	13	14	15
Length (mm)	50	70	80	90	90	90	90	90	90	100	110	120
Weight (g)	2	5	8	9	10	10	10	10	10	12	15	18
Habitat	A		B							C		

1+ juvenile

Month	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	June
Age (mo)	16	17	18	19	20	21	22	23	24	25	26	27
Length (mm)	122	125	128	130	130	130	130	130	130	132	134	138
Weight (g)	22	26	28	30	30	30	30	30	30	35	40	45
Habitat	D			E	F					G		

2+ pre-recruit

Month	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	June
Age (mo)	28	29	30	31	32	33	34	35	36	37	38	39
Length (mm)	142	146	150	155	158	160	160	160	160	160	165	170
Weight (g)	50	60	65	70	70	70	70	70	60	70	80	90
Habitat	H			I	J				K	L	M	

ages-3 and older

Month	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	June
Age (mo)	40	41	42	43	44	45	46	47	48	49	50	51
Length (mm)	175	180	180	180	180	180	180	180	180	150	150	150
Weight (g)	100	110	115	115	115	115	115	115	115	115	120	125
Habitat	H			I	J				K	L	M	

Fig. 4.1 Schematic representation of the herring life cycle showing calendar months, age in months, life stages, typical weights and length of individuals, and general habitat information. Note that herring live to age 7 to 10; the information shown for ages-3 and older is generally applicable to older individuals. Habitat letter codes on the figure are: 0+ juveniles: *A* = nearshore (mainly <50 m deep), *B* = deeper, nearshore waters (about 100 m deep), *C* = nearshore, deep and shallow; 1+ juveniles: *D* = nearshore, deep and shallow, *E* = deeper, inshore waters, *F* = deeper, nearshore waters, *G* = nearshore, deep and shallow; 2+ pre-recruit and age-3+: *H* = mainly shelf waters up to 200 m deep, *I* = beginning migrations to nearshore over-wintering areas, *J* = deep nearshore waters, *K* = moving to nearshore spawning areas, *L* = migrating to shelf waters, *M* = feeding on shelf waters.

The feedbacks between the herring and the lower trophic levels were included in NEMURO_ Herring. At each time step in the numerical integration scheme, densities of small, large, and predatory zooplankton were used to compute the consumption rate of the average herring in each age-class. The average weight of herring in each age-class was then updated, and the numbers in each age-class were decremented based on mortality rates. The total predation rate of herring on each zooplankton group (in g prey·m^{-3}·d^{-1}) was computed as:

$$\sum_{i=1}^{10} C_{j,i} \cdot N_i \cdot W_i \tag{4.3}$$

where $C_{j,i}$ is the consumption rate of the ith age class of herring on the jth zooplankton group, and N_i and W_i are the numbers and average weight of individuals, respectively, in the ith age-class. This loss rate was included in the NEMURO differential equation that describes the rate of change of each zooplankton group. The same coupling was done for ammonia and PON. Total herring excretion as ammonia was computed by summing over age-classes the excretion rate of the average individual times the number in an age-class; PON was added by summing over egestion. The contribution by excretion was included in the ammonia differential equation of NEMURO, and the contribution by herring egestion was included in the PON differential equation of NEMURO. We converted between the NEMURO units of µM N·liter^{-1} for zooplankton densities, ammonia concentration, and PON concentration and the herring bioenergetics units of g ww of zooplankton per m^3 for consumption, excretion, and egestion. The conversion from µM N·liter^{-1} to g ww·m^{-3} is:

$$\frac{uM\ N}{liter} \cdot \frac{14\ ug\ N}{uM} \cdot \frac{10^{-6} g}{ug} \cdot \frac{1\ g\ dw}{0.07\ g\ N\ dw} \cdot \frac{1\ g\ ww}{0.2\ g\ dw} \cdot \frac{10^3 liters}{m^3} = \frac{g\ ww}{m^3} \tag{4.4}$$

4.3 Example simulation of NEMURO_Herring

We present two 11-year simulations of the NEMURO_Herring model. The simulations were identical except that one simulation (denoted "coupled") included the feedbacks (herring consumption of three zooplankton groups, excretion on ammonia, and egestion on PON) while the other simulation (denoted "uncoupled") did not include the feedbacks. The NEMURO component begins on January 1, 1991; we do not introduce the herring into the model until July 17 of year 2 to let the lower trophic level dynamics spin up to their regular seasonal cycles. We used a natural mortality rate of 0.44·year^{-1} and a fishing mortality rate of 0.25·year^{-1}, applied to all age-classes of herring. Recruitment of herring to age-0 in the model was assumed to be 0.0005·m^{-3} each year in the simulation. The initial conditions for the numbers in age-classes 2 through 10 on July 17 of the first year were computed by the applying the annual mortality rates to the assumed recruitment to age-0. Initial weights of the average herring (g ww) in each age-class on July 17 of year 1 was set to: 2.0, 60, 80, 125, 140, 150, 170, 180, 190, and 200. Note that the initial weights differ somewhat from those shown in Figure 4.1. The simulations are for illustrative purposes and subsequent analyses will use initial conditions based on field data.

The simulations illustrate the new capability of the NEMURO_Herring model. We recognize that the dynamics of the lower trophic levels and herring require additional fine-tuning. We are not yet ready to compare model predictions to field data for the lower trophic levels or for herring from the Vancouver Island area. We include these results to illustrate the capabilities of the NEMURO_ Herring model developed at the 2003 Yokohama Workshop. We therefore compare predictions between the coupled and uncoupled simulations in this report.

Predicted mean weights by age of the herring were lower under the coupled simulation as compared to the uncoupled simulation (Fig. 4.2), while as expected because of the assumed constant mortality rates, predicted numbers in each age-class over time were identical for the two simulations (Fig. 4.3). Including herring consumption as a dynamic mortality term on the zooplankton resulted in a density-dependent feedback and lower herring growth rates in the coupled simulation.

The effects of including the feedbacks were also apparent in the predicted dynamics of the lower trophic levels. Predicted densities of the large zooplankton group over time were generally lower under the coupled simulation than under the uncoupled simulation (Fig. 4.4b). Much smaller effects of herring consumption were predicted for the small and predatory zooplankton groups (Figs. 4.4a and 4.4c), as the large zooplankton group was the dominant prey in the diet of simulated herring. The effects of herring excretion on ammonia (Fig. 4.5a) and the effects of egestion on PON (Fig. 4.5b) were both relatively small.

Interestingly, the shift in zooplankton densities in the coupled simulation caused by herring selectively consuming the large zooplankton group cascaded downward in the food web resulting in consistent (albeit small) changes in the densities of the large phytoplankton group (Fig. 4.5c).

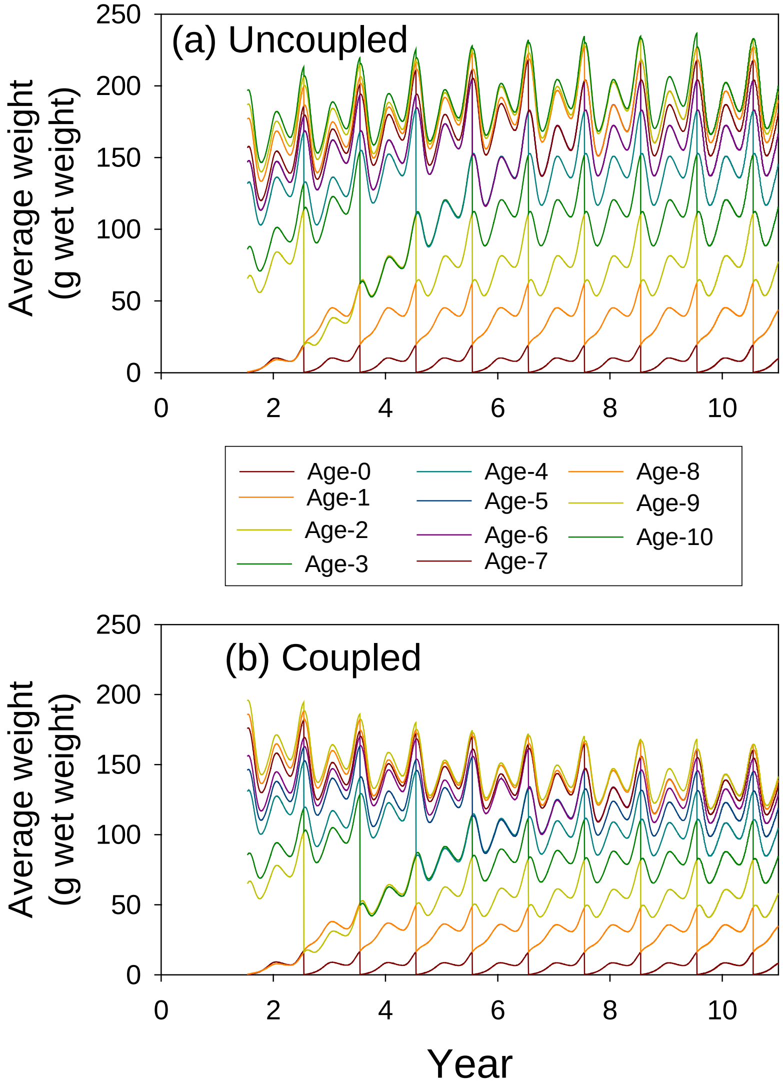

Fig. 4.2 Predicted daily average weights per individual by age-class of herring from the 11-year (a) uncoupled and (b) coupled simulations of the NEMURO_Herring model.

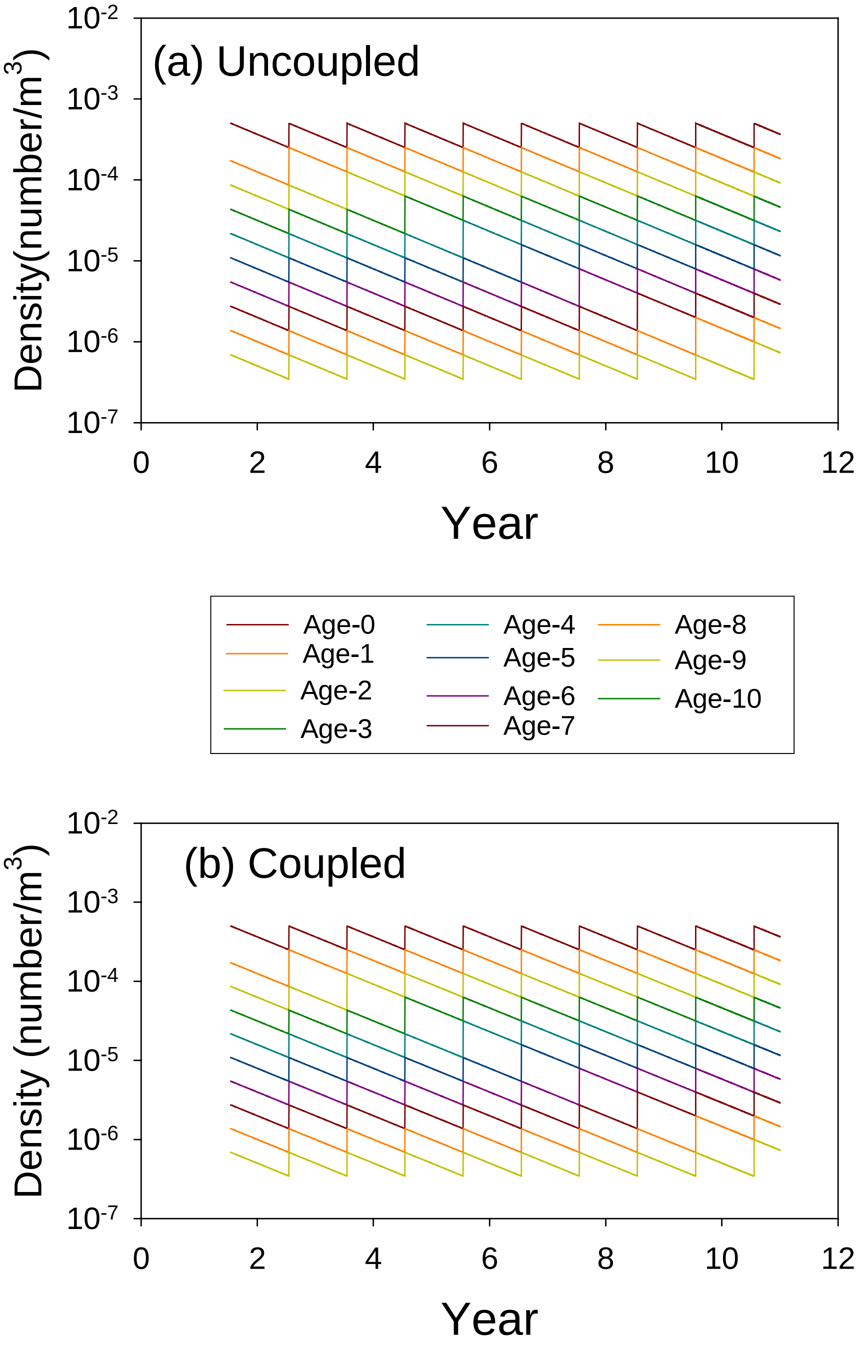

Fig. 4.3 Predicted daily herring densities by age-class of herring from the 11-year (a) uncoupled and (b) coupled simulations of the NEMURO_Herring model.

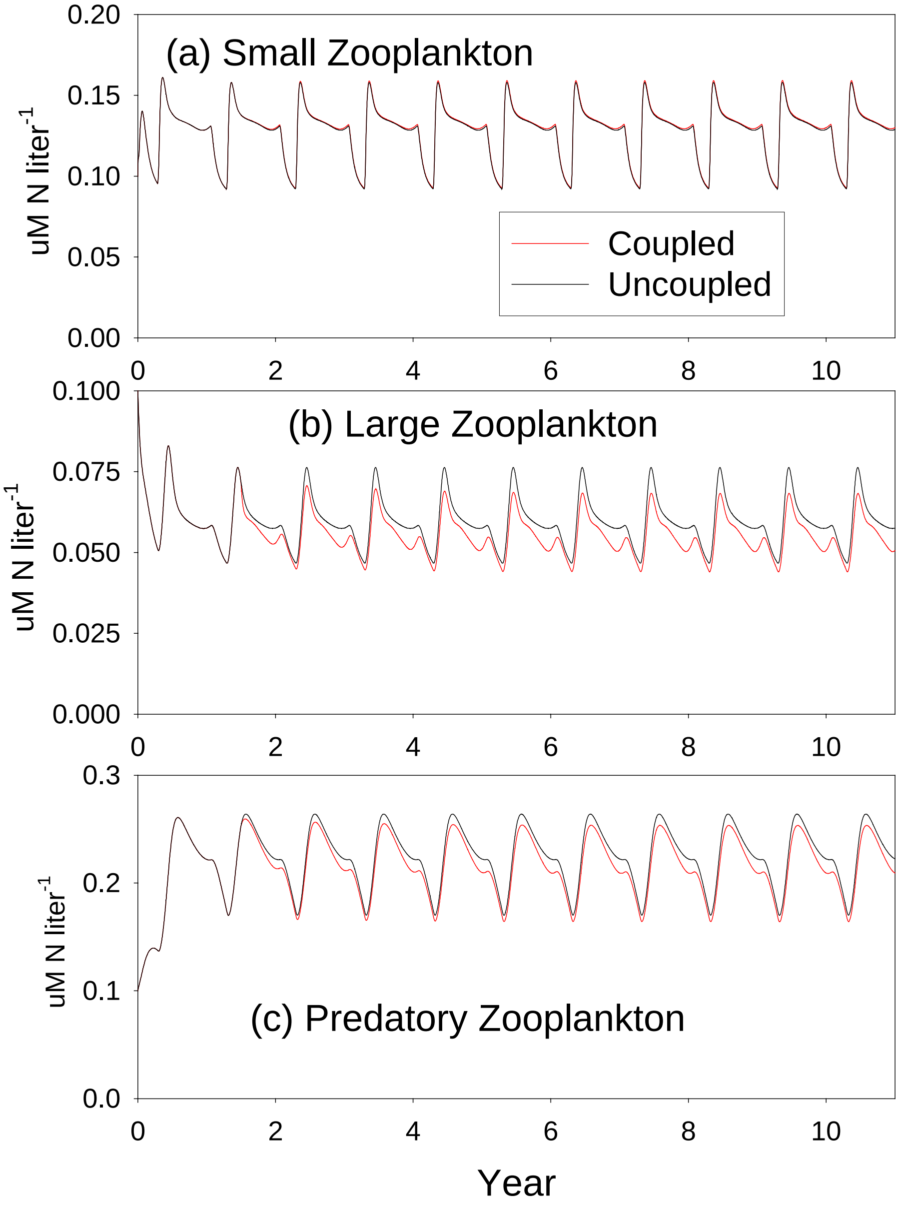

Fig. 4.4 Predicted daily zooplankton densities of (a) small zooplankton, (b) large zooplankton, and (c) predatory zooplankton from the 11-year coupled and uncoupled simulations of the NEMURO_Herring model.

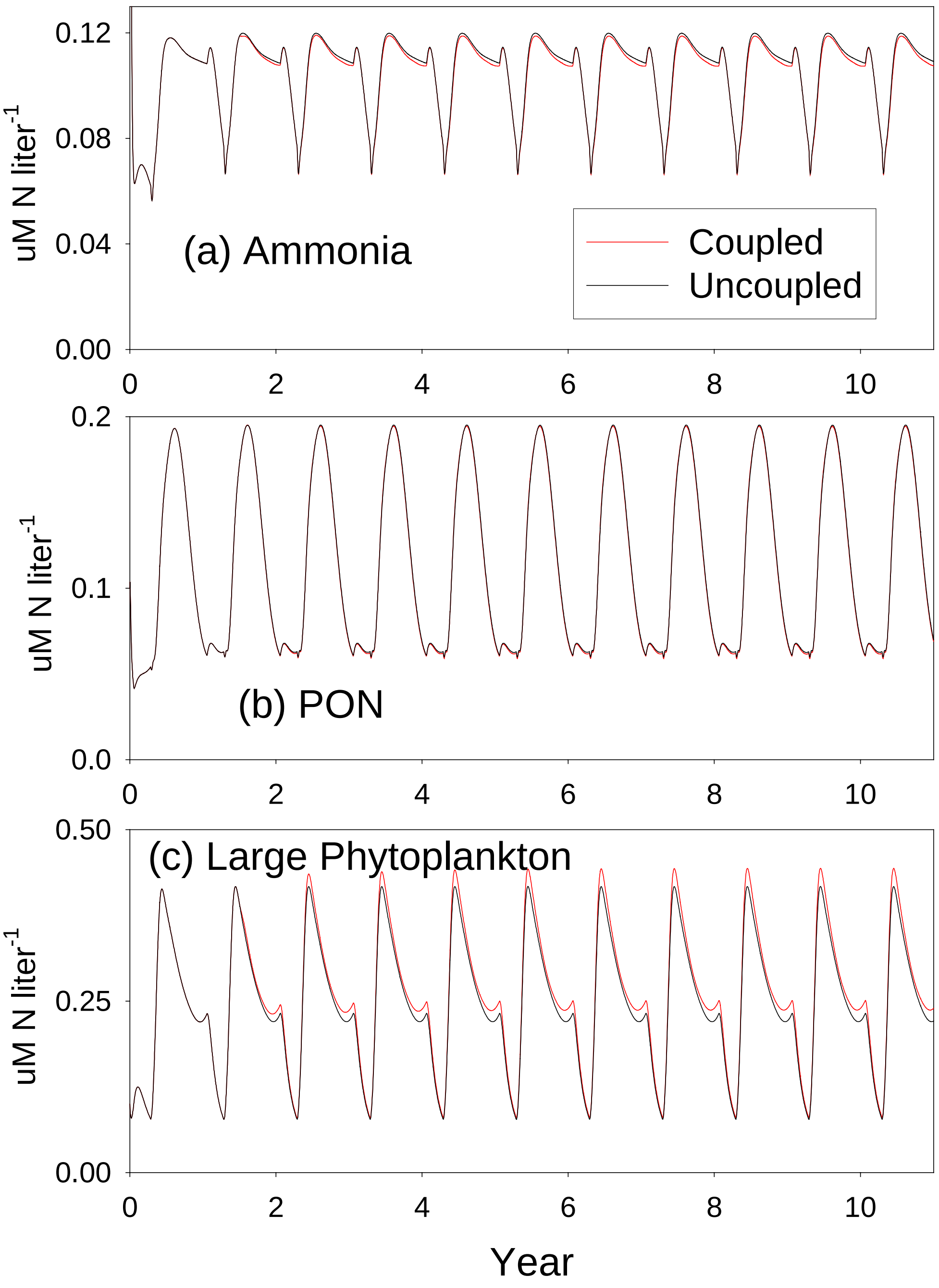

Fig. 4.5 Predicted daily concentrations of (a) ammonia, (b) PON, and (c) large phytoplankton from the 11-year coupled and uncoupled simulations of the NEMURO_Herring model.

4.4 Future plans

The NEMURO_Herring model is planned to be the basis for a manuscript for a special issue of the journal *Ecological Modelling*. The next steps needed to achieve this goal are to:

1. synthesize the field data on lower trophic level and herring dynamics from the Vancouver Island area;
2. continue the model calibration so that predicted dynamics better reproduce the known patterns of lower trophic level and herring dynamics in the Vancouver Island region;
3. expand the herring component to simulate individual herring within each age-class and include variation in prey encounter rates to allow for prediction of variation of size at age;
4. add a reproduction component to the herring bioenergetics; and
5. use the model to simulate the effects of environmental changes (*e.g.* climate change, regime shifts) on the food web (herring and lower trophic levels) under a variety of biological conditions (*e.g.* low versus high recruitment of herring to age-0).

5. References

Aita, M.N., Yamanaka, Y. and Kishi, M. J. 2003. On ontogenetic vertical migration of zooplankton in OB-GCM - Using NEMURO embedded in a General Circulation Model. Fish. Oceanogr. 12(4/5): 284-290.

Eslinger, D.L., Kashawai, M.B., Kishi, M.J., Megrey, B.A., Ware, D.M. and Werner, F.E. and 2000. MODEL Task Team Workshop Report: Final Report of the International Workshop to Develop a Prototype Lower Trophic Level Ecosystem Model for Comparison of Different Marine Ecosystems in the North Pacific. pp. 1-78. *In* Megrey, B.A., Taft, B.A. and Peterson, W.T. (Eds.) PICES-GLOBEC International Program on Climate Change and Carrying Capacity: Report of the 1999 MONITOR and REX Workshops, and the 2000 MODEL Workshop on Lower Trophic Level Modelling. PICES Scientific Report No. 15, 148 pp.

Fujii, M., Nojiri, Y., Yamanaka, Y. and Kishi, M.J. 2002. A one-dimensional ecosystem model applied to time series Station KNOT, Deep-Sea Res. Part II, 49, 5441-5461.

Ito, S., Kishi, M.J., Kurita, Y., Oozeki, Y., Yamanaka, Y., Megrey, B.A., and Werner, F.E. 2004. A fish bioenergetics model application to Pacific saury coupled with a lower trophic ecosystem model. Fish. Oceanogr., in press.

Ito, S., Kishi, M.J., Yamanaka, Y., and Fujii, M. 2002. Saury group report and model results. pp. 114-119. *In* Batchelder, H.P., McFarlane, G.A., Megrey, B.A., Mackas, D.L. and Peterson, W.T. (Eds.) PICES-GLOBEC International Program on Climate Change and Carrying Capacity: Report of the 2001 BASS/MODEL, MONITOR and REX Workshops, and the 2002 MODEL/REX Workshop. PICES Scientific Report No. 20. 176 pp.

Kishi., M.J., Megrey, B.A., Eslinger, D., Rose, K.A., Ware, D.M., Werner, F.E. 2001. Report of the 2001 PICES MODEL Task Team workshop on strategies for coupling higher and lower trophic level marine ecosystem models: status of the LTL model. GLOBEC Newsletter 7(1): 3-6.

Megrey, B.A., and Kishi, M.J. 2002. MODEL/REX workshop to develop a marine ecosystem model of the North Pacific Ocean including pelagic fishes. pp. 77-176. *In* Batchelder, H.P., McFarlane, G.A., Megrey, B.A., Mackas, D.L. and Peterson, W.T. (Eds.) PICES-GLOBEC International Program on Climate Change and Carrying Capacity: Report of the 2001 BASS/MODEL, MONITOR and REX Workshops, and the 2002 MODEL/REX Workshop. PICES Scientific Report No. 20. 176 pp.

Megrey, B.A., Kishi, M.J., Hay, D., Ito, S., Klumb, R.A., Rose, K. and Werner, F.E. 2002a. Report of the 2002 PICES MODEL/REX Task Team Workshop to Develop a Marine Ecosystem Model of the North Pacific Ocean Including Pelagic Fishes: NEMURO.FISH. GLOBEC Newsletter, Vol. 8, No. 2: 25-27.

Megrey, B.A., Kishi, M.J, Kashiwai, M., Ware, D.M., Eslinger, D.L., Werner, F.E. 2000. PICES Lower trophic level modeling workshop, Nemuro. PICES Press 8(2): 18-22.

Megrey, B.A., Rose, K.A, Werner, F.E., Klumb, R.A. and Hay, D.E. 2002b. A generalized fish bioenergetics/biomass model with an application to Pacific herring, pp. 80-88. *In* Batchelder, H.P., McFarlane, G.A., Megrey, B.A., Mackas, D.L. and Peterson, W.T. (Eds.) PICES-GLOBEC International Program on Climate Change and Carrying Capacity: Report of the 2001 BASS/MODEL, MONITOR and REX Workshops, and the 2002 MODEL/REX Workshop. PICES Scientific Report No. 20. 176 pp.

Miller, T.J., Crowder, L.B., Rice, J.A., and Marshall, E.A. 1988. Larval size and recruitment mechanisms in fishes: Toward a conceptual framework. Can. J. Fish. Aquat. Sci. 45: 1657-1670.

Rose, K.A., Rutherford, E.S., McDermot, D.S., Forney, J.L. and Mills, E.L. 1999. Individual-based model of yellow perch and walleye populations in Oneida Lake. Ecol. Monogr. 69(2): 127-154.

Thorton, K.W., and Lessem, A.S. 1978. A temperature algorithm for modifying biological rates. Trans. Am. Fish. Soc. 107: 284-287.

6. Appendices

6.1 Workshop participants

Canada

HAY, Douglas E.
Pacific Biological Station
Fisheries and Oceans Canada
3190 Hammond Bay Road
Nanaimo, BC, V9R 5K6
Canada
hayd@pac.dfo-mpo.gc.ca

Japan

AITA-Noguchi, Maki
Frontier Research System for Global Change
3173-25 Showa-machi, Kanazawa-ku
Yokohama, Kanagawa, 236-0001
Japan
macky@jamstec.go.jp

CHIBA, Sanae
Frontier Research System for Global Change
3173-25 Showa-machi, Kanazawa-ku
Yokohama, Kanagawa, 236-0001
Japan
chibas@jamstec.go.jp

FUJII, Masahiko
National Institute for Environmental Studies
16-2 Onogawa
Tsukuba, Ibaraki, 305-8506
Japan
fujii@ees.hokudai.ac.jp

HASHIOKA, Taketo
Hokkaido University
3-1-1 Minato-cho
Hakodate, Hokkaido, 041-8611
Japan
hashioka@ees.hokudai.ac.jp

ITO, Shin-ichi
Tohoku National Fisheries Research Institute
3-27-5 Shinhama-cho
Shiogama, Miyagi, 985-0001
Japan
goito@affrc.go.jp

KASHIWAI, Makoto
Hokkaido National Fisheries Research Institute
116 Katsurakoi,
Kushiro, Hokkaido, 085-0802
Japan
kashiwai@mua.biglobe.ne.jp

KATSUKAWA, Toshio
Ocean Research Institute
University of Tokyo
1-15-1 Minamidai, Nakano-ku
Tokyo, 164-8639
Japan
katukawa@ori.u-tokyo.ac.jp

KAWAMIYA, Michio
Frontier Research System for Global Change
3173-25 Showa-machi, Kanazawa-ku
Yokohama, Kanagawa, 236-0001
Japan
kawamiya@jamstec.go.jp

KISHI, Michio J.
Hokkaido University /FRSGC
3-1-1 Minato-cho
Hakodate, Hokkaido, 041-8611
Japan
kishi@salmon.fish.hokudai.ac.jp

MUKAI, Daiki
Hokkaido University
3-1-1 Minato-cho
Hakodate, Hokkaido, 041-8611
Japan
dmukai@sings.jp

ONITSUKA, Goh
Prefectural University of Kumamoto
3-1-100 Tsukide
Kumamoto, 862-8502
Japan
onizuka@pu-kumamoto.jp

SMITH, Lan
Frontier Research System for Global Change
3173-25 Showa-machi, Kanazawa-ku
Yokohama, Kanagawa, 236- 0001
Japan
lanimal@jamstec.go.jp

SUDA, Maki
National Research Institute of Fisheries Science
2-12-4 Fukuura, Kanazawa-ku
Yokohama, Kanagawa, 236-8648
Japan
msuda@fra.affrc.go.jp

TADOKORO, Kazuaki
Frontier Research System for Global Change
3173-25 Showa-machi, Kanazawaku
Yokohama, Kanagawa, 236-0001
Japan
denden@jamstec.go.jp

TIAN, Yongjun
National Research Institute of Fisheries Science
2-12-4 Fukuura, Kanazawa-ku
Yokohama, Kanagawa, 236-8648
Japan
yjtian@fra.affrc.go.jp

YAMANAKA, Yasuhiro
Graduate School of Environmental Earth Science
Hokkaido University
Kita 10, Nishi 5, Kita-ku
Sapporo, Hokkaido, 060-0810
Japan
galapen@ees.hokudai.ac.jp

YOSHIE, Naoki
Graduate School of Environmental Earth Science
Hokkaido University
Kita 10, Nishi 5, Kita-ku
Sapporo, Hokkaido, 060-0810
Japan
naoki@ees.hokudai.ac.jp

YOSHIMOTO, Sachie
Hokkaido University
3-1-1 Minato-cho
Hakodate, Hokkaido, 041-8611
Japan
naoki@ees.hokudai.ac.jp

Republic of Korea

YOO Sinjae
Korea Ocean Research and Development Institute
Ansan P.O. Box 29
Seoul, 425-600
Republic of Korea
E-mail: sjyoo@sari.kordi.re.kr

U.S.A.

MEGREY, Bernard A.
Alaska Fisheries Science Center
National Marine Fisheries Service, NOAA
7600 Sand Point Way NE
Seattle, WA, 98115-0070, U.S.A.
bern.megrey@noaa.gov

ROSE, Kenneth A.
Oceanography and Coastal Sciences
218 Wetland Resources Building
Louisiana State University
Baton Rouge, LA, 70803
U.S.A.
karose@lsu.edu

WERNER, Francisco E.
Department of Marine Science
University of North Carolina
Chapel Hill, NC, 27599-3300
U.S.A.
cisco@marine.unc.edu

CHAI, Fei
School of Marine Science
University of Maine
5471 Libby Hall
Orono, ME, 04469-5741
U.S.A.
fchai@maine.edu

6.2 Workshop schedule

<u>Monday March 3</u>

09:15-09:30	Facility introduction (Aita-Noguchi)
09:30-09:45	Introduction of participants
09:45-10:00	Goals of the workshop (Kishi)
10:00-10:15	Review of NEMURO (Kishi)
10:15-10:30	Review of NEMURO.FISH /Herring (Megrey)
10:30-11:00	Coffee break
11:00-11:15	Review of NEMURO.FISH/Saury (Ito)
11:15-11:45	MODEL website and SharePoint portal (Megrey)
11:45-13:00	Lunch
13:00-13:30	Coupling with population dynamics model (Katsukawa)
13:30-14:15	3-D model results (Noguchi-Aita)
14:15-15:00	Ongoing MODEL, NEXT and GLOBEC activities and future initiatives (Werner and Ito)
15:00-15:30	Coffee break
15:30-17:00	Discussion of Tuesday to Thursday agenda and charge for group breakouts (Werner and Ito)

<u>Tuesday March 4</u>

9:00-11:00	TG1 on a Population Dynamics Model: Build two-way model using population dynamics coupled model (Leader, Ito) TG2 on a Lagrangian Model: Explore explicit spatial dependence (Leaders, Kishi and Yamanaka)
11:00-12:00	Tour of FRSGC
12:00-13:00	Lunch
13:00-17:30	Continue working in teams
18:00-20:00	Dinner party in Chinatown

<u>Wednesday March 5</u>

09:00-10:00	Update by each group
10:00-11:45	Task Group work
11:45-13:00	Lunch
13:00-13:30	NEXT Discussion (Kashiwai)
13:30-17:00	TG1 on a Population Dynamics Model: Build two-way model using population dynamics coupled model (Leader, Ito) Age-structured model options (Rose) TG2 on a Lagrangian Model: Explore explicit spatial dependence (Leaders, Kishi and Yamanaka)
17:00-18:30	Future perspectives and closing remarks

6.3 Lagrangian model (FORTRAN program)

```
!*************************************************************************************
! 2003 PICES NEMURO Program written by Y.Yamanaka, S.Ito, M.J.Kishi, M.Fujii and T.Hashioka
!     For multi-particle with horizontal migration
!*************************************************************************************
!
      implicit none
      integer,parameter :: np =  1        ! np : number of particles
      integer,parameter :: nc = 16        ! nc : number of particles' attribute
      real(8)           :: ptc( np, nc )
      integer,parameter :: Latitude =  1, Longitude =  2, LifeDate =  3, &
                           ZooS     =  4, ZooL      =  5, ZooP     =  6, &
                           Temp     =  7, Number    =  8, KLength  =  9, Weight = 10, &
                           Uadv     = 11, Vadv      = 12, Umig     = 13, Vmig   = 14, &
                           TgradX   = 15, TgradY    = 16
!
      integer,parameter :: nx = 33, ny = 36               ! domain size
      integer           :: Land( nx, ny )                 ! Land Index( 1:Sea, 0:Land )
!
!     ..... Control for Time Integration .....
      character(19)     :: Cstart = '2000/02/01 00:00:00'  ! Starting date
      character(19)     :: Cend   = '2001/02/01 00:00:00'  ! Ending date
      character(19)     :: Cstep  = '0000/00/00 01:00:00'  ! Time step
      character(19)     :: Cmon   = '0000/00/01 00:00:00'  ! Monitor Interval
      character(19)     :: CTime
      real(8)           :: dt, TTime, Tbefore, Tmon
      integer           :: Iyr, Imon, Iday, Ihour, Imin, Isec, Impre, Impro
!
      real(8),parameter :: dx  = 100.0d5, dy = 100.0d5      ! grid size in 1deg = cm
      real(8),parameter :: d2s = 86400.0d0                  ! conversion of day ---> sec
!
!     ..... Parameters of fish bioenergetics & population dynamics model
      real(8)           :: wfsdot(np), nfsdot(np)
      real(8)           :: con1,con2,con3,resp,f,e,sda,egg
      real(8)           :: Nmort=0.15 / ( 365.0D0 * d2s )
      real(8)           :: Fmort=0.20 / ( 365.0D0 * d2s )
!
      character(20)     :: pfile
      real(8)           :: pxini(np),  pyini(np)
      integer           :: lt, nt, lp, lc
!
!     ..... statement function & def. type of functions .....
      real(8)           :: cd2tt, nd2tt
      character(19)     :: tt2cd
!
!     ****** initial set up *****
!
      TTime = cd2tt(Cstart)                       ! Starting Date
      CTime = TT2CD(TTime)                            ! present time (charactor form)
      dt    = cd2tt(Cstep) - cd2tt('0000/00/00 00:00:00')  ! Time Step (real8 form)
```

```
      Tmon  = cd2tt(Cmon)  - cd2tt('0000/00/00 00:00:00')  ! Monitor Interval (real8 form)
      nt = NINT( ( cd2tt(Cend) - cd2tt(Cstart) ) / dt )    ! Total Time Steps
!
      pfile = 'Japan_Particle00.csv'
      do lp = 1, np
        write(pfile(15:16),'(I2.2)') lp
        open( 40+lp, file=pfile, form='formatted' )
        write(40+lp,'(3A)')   &
        'Date, long, lat, age(day), ZS, ZL, ZP,', &
        ' temp, number, Knob length, weight, Uadv, Vadv,Umig, Vmig, TgradX, TgradY'
      end do
!
!     ***** initial values of particle *****
      do lp = 1, np
        pxini( lp ) = 10.5 * dx
        pyini( lp ) =  2.8 * dy
        ptc(lp,Longitude) = pxini( lp )
        ptc(lp,Latitude ) = pyini( lp )
        ptc(lp,LifeDate ) = 0.0
        ptc(lp,Number   ) = 1.0d-3
        ptc(lp,Weight   ) = 0.2d0
        ptc(lp,KLength  ) = 0.0
      end do
!
!       ...... Output of initial value ......
        do lp = 1, np
          write(40+lp,'(A,20(",", 1PE12.5),",",I5)') CTime, &
            ptc( lp, Longitude)/dx,  ptc ( lp, Latitude)/dy,&
          ( ptc( lp, lc), lc = 3, 16 )
        end do
!
!     ***** main loop *****
      do lt = 1, nt
        Tbefore = TTime                              ! one step before present time
        TTime   = TTime + dt                          ! present time (real8 form)
        CTime = TT2CD(TTime)                            ! present time (character form)
        call TT2ND(Iyr, Imon, Iday ,Ihour, Imin, Isec ,TTime)
!
!       ***** To get Temp, ZS, ZL, ZP, Uadv, Vadv, gradTx, gradTy at partile positions *****
        call ReadData( TTime, ptc, np, nc, Land )
!
!       ***** To calculate fish weight and number *****
        do lp = 1, np
          call Bioenergy(TTime, Tbefore, &
                    ptc(lp,ZooS), ptc(lp,ZooL), ptc(lp,ZooP), ptc(lp,Temp), &
                    con1, con2, con3, resp, f, e, sda, egg, &
                    ptc(lp,Weight),  ptc(lp,LifeDate) )
!         ..... use the ratio of calories/g of zoop (2580) to calories/g of fish (5533) .....
          wfsdot(lp)=(con1+con2+con3-resp-f-e-sda-egg) * ptc(lp,Weight) * 2580.0/5533.0/d2s
          nfsdot(lp)= - ( Nmort + Fmort ) * ptc( lp, Number )
        end do
```

```
!
!        ***** Horizontal migration *****
         call HMigration( TTime, ptc, np, nc, pxini, pyini, Land, nx, ny)
!
!        ******* time integration ********
         do lp = 1, np
           ptc(lp,Longitude) = ptc(lp,Longitude) + ( ptc(lp,Uadv) + ptc(lp,Umig) ) * dt
           ptc(lp,Latitude ) = ptc(lp,Latitude ) + ( ptc(lp,Vadv) + ptc(lp,Vmig) ) * dt
           ptc(lp,LifeDate ) = ptc(lp,LifeDate ) + dt / d2s
           ptc(lp,Number   ) = ptc(lp,Number   ) + dt * nfsdot(lp)
           ptc(lp,Weight   ) = ptc(lp,Weight   ) + dt * wfsdot(lp)
           ptc(lp,KLength  ) = max( ptc(lp,KLength), 6.13 * ptc(lp,Weight)**0.33 )
         end do
!
!        ***** Output for Monitor *****
!
         if ( int(TTime/Tmon).ne. int(Tbefore/Tmon) ) then
           do lp = 1, np
             write(40+lp,'(A,20(",", 1PE12.5),",",I5)') CTime, &
               ptc( lp, Longitude)/dx,  ptc ( lp, Latitude)/dy,&
             ( ptc( lp, lc), lc = 3, 16 )
           end do
         end if
       end do
!
       stop
       end
!*********************************************************************************
!      Horizontal Migration
!*********************************************************************************
       Subroutine HMigration( TTime, ptc, np, nc, pxini, pyini, Land, nx, ny)
!
       implicit none
       real(8),intent(IN)   :: TTime
       integer,intent(IN)   :: nx, ny, Land( nx, ny )
       integer,intent(IN)   ::      np, nc
       real(8),intent(INOUT):: ptc( np, nc )
       real(8),intent(IN)   :: pxini(np),  pyini(np)
       integer,parameter    :: Latitude =  1, Longitude =  2, LifeDate = 3, &
                               ZooS     =  4, ZooL      =  5, ZooP     = 6, &
                               Temp     =  7, Number    =  8, KLength  = 9, Weight = 10, &
                               Uadv     = 11, Vadv      = 12, Umig     = 13, Vmig  = 14, &
                               TgradX   = 15, TgradY    = 16
!
       integer          :: fishswimming              ! swimming manner index
       integer          :: lp, i, j
       integer          :: Iyr, Imon, Iday, Ihour, Imin, Isec, Impre, Impro
       real(8)          :: speed, ex, ey, ee
       real(8),parameter :: dx = 100.0d5, dy = 100.0d5      ! grid size in 1deg = cm
!
!      ...... Parameters of saury migration .....
```

```
      character(19)     :: CFst ='0000/02/01 00:00:00' ! Date to stop southward migration
      character(19)     :: CFnw ='0000/06/01 00:00:00' ! Date to move to colder region
      character(19)     :: CFsw ='0000/10/01 00:00:00' ! Date to migrate to original position
      real(8)           :: TFnw,TFsw,TFst
      integer, save     :: IyrT,ImonT,IdayT,IhourT,IminT,IsecT
      integer, save     :: IyrS,ImonS,IdayS,IhourS,IminS,IsecS
      integer, save     :: IyrN,ImonN,IdayN,IhourN,IminN,IsecN
!
      real(8)           :: cd2tt, nd2tt
      character(19)     :: tt2cd
!
      integer, save     :: First = 1
!     =================
      if ( First .eq. 1 ) then; First = 0
        TFnw = CD2TT( CFnw )
        TFsw = CD2TT( CFsw )
        TFst = CD2TT( CFst )
        call TT2ND(IyrS, ImonS, IdayS ,IhourS, IminS, IsecS ,TFsw )
        call TT2ND(IyrN, ImonN, IdayN ,IhourN, IminN, IsecN ,TFnw )
        call TT2ND(IyrT, ImonT, IdayT ,IhourT, IminT, IsecT ,TFst )
      end if
!     =================

      call TT2ND(Iyr, Imon, Iday ,Ihour, Imin, Isec ,TTime )
      TFst = ND2TT(Iyr, ImonT, IdayT ,IhourT, IminT, IsecT )
      TFsw = ND2TT(Iyr, ImonS, IdayS ,IhourS, IminS, IsecS )
      TFnw = ND2TT(Iyr, ImonN, IdayN ,IhourN, IminN, IsecN )
!
!     ***** definition of swimming direction *****
!     period between CFst and CFnw: no migration
!     period between CFnw and CFsw: moving to colder region
!     period between CFsw and CFst: moving to original position
!
      if ( TTime .lt. TFst ) then
        fishswimming = 2
      else if ( TTime .lt. TFnw ) then
        fishswimming = 0
      else if ( TTime .lt. TFsw ) then
        fishswimming = 1
      else
        fishswimming = 2
      end if
!
      do lp = 1, np
        i = int( ptc(lp,Longitude) / dx )
        j = int( ptc(lp,Latitude ) / dy )
        select case( fishswimming )
          case( 0 )    !..... No swimming .....
            ex = 0.0
            ey = 0.0
          case( 1 )    !..... Swimming to colder region (Northward) .....
```

```
            ex = - ptc(lp,TgradX) * Land(i-1,j  ) * Land(i+1,j  )
            ey = - ptc(lp,TgradY) * Land(i  ,j-1) * Land(i+1,j+1)
          case( 2 )    !..... Swimming to origin (Southward) .....
            ex = - ( ptc(lp,Longitude) - pxini(lp) ) * Land(i-1,j  ) * Land(i+1,j  )
            ey = - ( ptc(lp,Latitude ) - pyini(lp) ) * Land(i  ,j-1) * Land(i+1,j+1)
        end select
        ee = sqrt( ex*ex + ey*ey )
        if ( ee .eq. 0.0 ) then
          ex = 0.0
          ey = 0.0
        else
          ex = ex / ee
          ey = ey / ee
        end if
!
!       ***** Swimming Speed *****
        if ( ptc(lp, Temp) .gt. 12.0d0 ) then
          speed = 12.3d0 * ptc(lp,Weight)**0.33
        else
          speed =  2.0d0 * ptc(lp,Weight)**0.33
        endif
        ptc(lp,Umig) = speed * ex
        ptc(lp,Vmig) = speed * ey
      end do
!
      return
      end subroutine
!
!***********************************************************************************
!    Reading circulation field, Temperature and etc., and getting temp. at particles
!***********************************************************************************
      Subroutine ReadData( TTime, ptc, np, nc, Land )
!
      implicit none
      real(8),intent(IN)   :: TTime
      integer,intent(IN)   :: np, nc
      real(8),intent(INOUT):: ptc( np, nc )
      integer,parameter ::  swiming_layer = 4             ! depth of 15 to 20m
!
      integer,parameter    :: Latitude =  1, Longitude =  2, LifeDate = 3, &
                              ZooS     =  4, ZooL      =  5, ZooP     = 6, &
                              Temp     =  7, Number    =  8, KLength  = 9, Weight = 10, &
                              Uadv     = 11, Vadv      = 12, Umig     = 13, Vmig  = 14, &
                              TgradX   = 15, TgradY    = 16
      integer,parameter    :: nx = 33, ny = 36, nz = 20, nm = 12
      integer,intent(OUT)  :: Land( nx, ny )
!
      real(8)          :: Ru ( nx, ny, nz ), Rv ( nx, ny, nz ), Rt ( nx, ny, nz )
      real(8)          :: Rzs( nx, ny, nz ), Rzl( nx, ny, nz ), Rzp( nx, ny, nz )
!                     u,v: x-dir. and y-dir. of velocity, t: temperature (nm: month)
      real(8),save      :: u  ( nx, ny, nm ), v  ( nx, ny, nm ), t  ( nx, ny, nm )
```

```
!                        zs, zl, zp: concentration of ZooS, ZooL, ZooP (molN/l)
      real(8),save      :: zs ( nx, ny, nm ), zl ( nx, ny, nm ), zp ( nx, ny, nm )

      real(8)           :: x, y, Ue, Uw, Vn, Vs
      real(8)           :: Tsw,  Tse,  Tnw,  Tne, ZSsw, ZSse, ZSnw, ZSne
      real(8)           :: ZLsw, ZLse, ZLnw, ZLne,ZPsw, ZPse, ZPnw, ZPne
      real(8)           :: dTdxs, dTdxn, dTdyw, dTdye
!
      real(8)           :: mwt, Tmmid, Tmpre, Tmpro
      integer           :: Iyr, Imon, Iday, Ihour, Imin, Isec, Impre, Impro
      real(8),parameter :: dx  = 100.0d5, dy = 100.0d5     ! grid size in 1deg = cm
      real(8),parameter :: d2s = 86400.0d0                 ! day ---> sec
      integer           :: i, j, l, m
!
      real(8)           :: cd2tt, nd2tt
      character(19)     :: tt2cd
!
      integer, save     :: First = 1
!     ==================
      if ( First .eq. 1 ) then; First = 0
!        ***** Reading data of u, v *****
         open( 30, file='japan-u.dat',   form='formatted', access='sequential')
         open( 31, file='japan-v.dat',   form='formatted', access='sequential')
         open( 32, file='japan-t.dat',   form='formatted', access='sequential')
         open( 33, file='japan-zoos.dat',form='formatted', access='sequential')
         open( 34, file='japan-zool.dat',form='formatted', access='sequential')
         open( 35, file='japan-zoop.dat',form='formatted', access='sequential')
         do m = 1, nm
            read( 30, '(33F16.12)' ) Ru
            read( 31, '(33F16.12)' ) Rv
            do j = 1,ny
               do i = 1,nx
                  if ( Ru( i,j,swiming_layer ) .eq. -99.9999d0 ) then
!                    ..... grid in Land .....
                     u( i,j,m ) = 0.0
                     v( i,j,m ) = 0.0
                  else
                     u( i,j,m ) = Ru( i,j,swiming_layer )
                     v( i,j,m ) = Rv( i,j,swiming_layer )
                  end if
               end do
            end do
            read( 32, '(33F16.12)' ) Rt
            read( 33, '(33F16.12)' ) Rzs
            read( 34, '(33F16.12)' ) Rzl
            read( 35, '(33F16.12)' ) Rzp
            do j = 1,ny
               do i = 1,nx
                  if ( Rt( i,j,swiming_layer ) .eq. -99.9999d0 ) then
!                    ..... grid in Land .....
                     Land( i,j ) = 0
```

```
            t ( i,j,m ) = 0.0
            zs( i,j,m ) = 0.0
            zl( i,j,m ) = 0.0
            zp( i,j,m ) = 0.0
          else
            Land( i,j ) = 1
            t ( i,j,m ) = Rt ( i,j,swiming_layer )
            zs( i,j,m ) = Rzs( i,j,swiming_layer )
            zl( i,j,m ) = Rzl( i,j,swiming_layer )
            zp( i,j,m ) = Rzp( i,j,swiming_layer )
          end if
        end do
      end do
    end do
  close(30); close(31); close(32); close(33); close(34); close(35)
  end if
!   =================
!
!
    call TT2ND(Iyr, Imon, Iday ,Ihour, Imin, Isec ,TTime)
!   ..... to obtain 15th on the present month ......
    Tmmid = ND2TT(Iyr, Imon, 15 , 0, 0, 0 )
    if ( TTime .lt. Tmmid ) then
      Impro = Imon; Tmpro = Tmmid
!     ..... to obtain 15th on the previous month ......
      call TT2ND(Iyr, Imon, Iday ,Ihour, Imin, Isec ,TTime - 20.0*d2s )
      Impre = Imon
      Tmpre = ND2TT(Iyr, Imon, 15 , 0, 0, 0 )
    else
      Impre = Imon; Tmpre = Tmmid
!     ..... to obtain 15th on the proceeding month ......
      call TT2ND(Iyr, Imon, Iday ,Ihour, Imin, Isec ,TTime + 20.0*d2s )
      Impro = Imon
      Tmpro = ND2TT(Iyr, Imon, 15 , 0, 0, 0 )
    end if
    mwt = ( TTime - Tmpre) / ( Tmpro - Tmpre )
!
!   ***** Spatial interpolation into particles' point *****
    do l = 1, np
!     ..... u,v at particles .....
      i = int( ptc(l,Longitude) / dx ) + 1
      j = int( ptc(l,Latitude ) / dy ) + 1
      x =      ptc(l,Longitude) / dx   - i
      y =      ptc(l,Latitude ) / dy   - j
      Ue = ( u( i+1, j+1, Impre ) + u( i+1, j  , Impre ) ) * 0.5 * (1.0 - mwt )   &
        + ( u( i+1, j+1, Impro ) + u( i+1, j  , Impro ) ) * 0.5 *        mwt
      Uw = ( u( i  , j+1, Impre ) + u( i  , j  , Impre ) ) * 0.5 * (1.0 - mwt )   &
        + ( u( i  , j+1, Impro ) + u( i  , j  , Impro ) ) * 0.5 *        mwt
      Vn = ( v( i+1, j+1, Impre ) + v( i  , j+1, Impre ) ) * 0.5 * (1.0 - mwt )   &
        + ( v( i+1, j+1, Impro ) + v( i  , j+1, Impro ) ) * 0.5 *        mwt
      Vs = ( v( i+1, j  , Impre ) + v( i  , j  , Impre ) ) * 0.5 * (1.0 - mwt )   &
```

```
            + ( v( i+1, j  , Impro ) + v( i  , j  , Impro ) ) * 0.5 * mwt
           ptc(l,Uadv) = x * Ue + ( 1.0 - x ) * Uw
           ptc(l,Vadv) = y * Vn + ( 1.0 - y ) * Vs
!
!          ..... temperature, biomass of zooS, zooL and zooP at particles .....
           i = int( ptc(l,Longitude) / dx     - 0.5 )
           j = int( ptc(l,Latitude ) / dy     - 0.5 )
           x =      ptc(l,Longitude) / dx - i - 0.5
           y =      ptc(l,Latitude ) / dy - j - 0.5
           Tsw  = t (i  ,j  ,Impre) * (1.0 - mwt ) + t (i  ,j  ,Impro) * mwt
           Tse  = t (i+1,j  ,Impre) * (1.0 - mwt ) + t (i+1,j  ,Impro) * mwt
           Tnw  = t (i  ,j+1,Impre) * (1.0 - mwt ) + t (i  ,j+1,Impro) * mwt
           Tne  = t (i+1,j+1,Impre) * (1.0 - mwt ) + t (i+1,j+1,Impro) * mwt
           ZSsw = zs(i  ,j  ,Impre) * (1.0 - mwt ) + zs(i  ,j  ,Impro) * mwt
           ZSse = zs(i+1,j  ,Impre) * (1.0 - mwt ) + zs(i+1,j  ,Impro) * mwt
           ZSnw = zs(i  ,j+1,Impre) * (1.0 - mwt ) + zs(i  ,j+1,Impro) * mwt
           ZSne = zs(i+1,j+1,Impre) * (1.0 - mwt ) + zs(i+1,j+1,Impro) * mwt
           ZLsw = zl(i  ,j  ,Impre) * (1.0 - mwt ) + zl(i  ,j  ,Impro) * mwt
           ZLse = zl(i+1,j  ,Impre) * (1.0 - mwt ) + zl(i+1,j  ,Impro) * mwt
           ZLnw = zl(i  ,j+1,Impre) * (1.0 - mwt ) + zl(i  ,j+1,Impro) * mwt
           ZLne = zl(i+1,j+1,Impre) * (1.0 - mwt ) + zl(i+1,j+1,Impro) * mwt
           ZPsw = zp(i  ,j  ,Impre) * (1.0 - mwt ) + zp(i  ,j  ,Impro) * mwt
           ZPse = zp(i+1,j  ,Impre) * (1.0 - mwt ) + zp(i+1,j  ,Impro) * mwt
           ZPnw = zp(i  ,j+1,Impre) * (1.0 - mwt ) + zp(i  ,j+1,Impro) * mwt
           ZPne = zp(i+1,j+1,Impre) * (1.0 - mwt ) + zp(i+1,j+1,Impro) * mwt
           ptc(l,Temp) = y*x* Tne + y*(1.0-x)* Tnw + (1.0-y)*x* Tse + (1.0-y)*(1.0-x)* Tsw
           ptc(l,ZooS) = y*x*ZSne + y*(1.0-x)*ZSnw + (1.0-y)*x*ZSse + (1.0-y)*(1.0-x)*ZSsw
           ptc(l,ZooL) = y*x*ZLne + y*(1.0-x)*ZLnw + (1.0-y)*x*ZLse + (1.0-y)*(1.0-x)*ZLsw
           ptc(l,ZooP) = y*x*ZPne + y*(1.0-x)*ZPnw + (1.0-y)*x*ZPse + (1.0-y)*(1.0-x)*ZPsw
!          ..... gradient of temperature at particles .....
           dTdxs = ( Tse - Tsw ) / dx
           dTdxn = ( Tne - Tnw ) / dx
           dTdyw = ( Tnw - Tsw ) / dy
           dTdye = ( Tne - Tse ) / dy
           ptc(l,TgradX) = y*dTdxn + (1.0-y)*dTdxs
           ptc(l,TgradY) = x*dTdye + (1.0-x)*dTdyw
        end do
!
      return
      end subroutine
!*****************************************************************************************
      Subroutine Bioenergy(TTime, Tbefore, TZS, TZL, TZP, Temp, &
                 con1, con2, con3, resp, f, e, sda, egg, wfsi, Lifedate )
      implicit none
      real(8),intent(IN):: TTime, Tbefore, TZS, TZL, TZP, Temp
      real(8),intent(IN) ::  wfsi, Lifedate
      real(8),intent(OUT) :: con1, con2, con3, resp, f, e, sda, egg
!
      character(19)     :: CTime
      integer           :: Iyr, Imon, Iday, Ihour, Imin, Isec
      character(19)     :: CAge  ='0000/03/01 00:00:00'            ! Date of Aging 0->1
```

```
      character(19)     :: CAge2 ='0000/07/01 00:00:00'            ! Date of Aging 1->2
      real(8)           :: TAge, TAge2
      integer, save     :: IyrA, ImonA, IdayA ,IhourA, IminA, IsecA
      integer, save     :: IyrB, ImonB, IdayB ,IhourB, IminB, IsecB
      integer           :: iage                                ! age of saury
      integer           :: JJday
      real(8)           :: ZooP1, ZooP2, ZooP3
      real(8)           :: t1,t2,wtemp
      real(8)           :: vul(3), k(3)
      real(8),parameter :: d2s      = 86400.0d0    ! day ---> sec
!     ..... convert Nemuro zoop in MN/L to g ww/m3
!     ..... ConvUnit is conversion from molN/liter to g ww/m3
!     ..... 14ugN/uM * 1gdw/0.07gNdw * 1 gww/0.2 gdw * 1.0e3liters/m3
      real(8),parameter :: ConvUnit = 14.0  * (1.0/0.07)*(1.0/0.2) * 1.0e3
!
      real(8)           :: xk1, xk2, xk3, xk4, te1, te2, te3, te4
      real(8)           ::  tt5, t5, t4, tt7, t7, t6, gcta, gctb, gctemp, gcmax
      real(8)           :: u, cnum, c1, c2, c3, con
      real(8),parameter :: phalf = 0.10
      real(8)           :: wresp
!
      real(8)           :: cd2tt, nd2tt
!
      integer, save     :: First = 1
!     =================
      if ( First .eq. 1 ) then; First = 0
        TAge  = CD2TT( CAge  )
        call TT2ND(IyrA, ImonA, IdayA ,IhourA, IminA, IsecA ,TAge )
        TAge2  = CD2TT( CAge2  )
        call TT2ND(IyrB, ImonB, IdayB ,IhourB, IminB, IsecB ,TAge2 )
      end if
!     =================
!
      CALL TT2ND(Iyr, Imon, Iday ,Ihour, Imin, Isec ,TTime)
      JJday = 1 + ( TTime - ND2TT(Iyr  ,1,1,0,0,0) ) / d2s        ! Julian day
!
!     ***** Adjustment from NEMURO in MolN/l to FISH gww/m3 *****
      zoop1 = TZS * ConvUnit
      zoop2 = TZL * ConvUnit
      zoop3 = TZP * ConvUnit
      wtemp = max( 1.0d0, Temp )
!
!     ..... setting age of saury ......
      TAge  = ND2TT(Iyr, ImonA, IdayA ,IhourA, IminA, IsecA )
      TAge2 = ND2TT(Iyr, ImonB, IdayB ,IhourB, IminB, IsecB )
      if ( (TTime .lt. TAge) .and. (Lifedate .lt. 365.0)) then
        iage = 0
      else if ( (TTime .lt. TAge2) .and. (Lifedate .lt. 365.0)) then
        iage = 1
      else
        iage = 2
```

```
      end if
!
!     ..... set vulnerabilities and k values for 3 zoop groups .....
      if ( iage .eq. 0 ) then
        vul(1) =   1.0; vul(2) =  0.0; vul(3) =   0.0
        k  (1) = phalf; k  (2) = phalf; k (3) = phalf
      else if ( iage .eq. 1 ) then
        vul(1) =   1.0; vul(2) =  1.0; vul(3) =   0.0
        k  (1) = phalf; k  (2) = phalf; k (3) = phalf
      else
        vul(1) =   0.0; vul(2) =  1.0; vul(3) =   1.0
        k  (1) = phalf; k  (2) = phalf; k (3) = phalf
      endif
!
!     ..... weight affect on respiration .....
      wresp  = 0.0033 * wfsi**(-0.227)
!
      if (wtemp .le. 12.0 ) then
         u=2.0*wfsi**0.33*EXP(0.149*wtemp)
      else
         u=11.7*wfsi**0.33
      endif
      resp=wresp*EXP(0.0548*wtemp)*EXP(0.03*u)*5.258
!
!     --- Thornton and Lessem temperature effect
!     --- age dependent values
!     --- *******Arrhenius for age-0 he changed te4 from 25 to 23 degrees******
!
      if ( iage .eq. 0 ) then
         xk1 = 0.1;  xk2 = 0.98;  xk3 = 0.98;  xk4 = 0.5
         te1 = 5.0;  te2 = 20.0;  te3 = 26.0;  te4 = 30.0
      else if ( iage .eq. 1 ) then
         xk1 = 0.1;  xk2 = 0.98;  xk3 = 0.98;  xk4 = 0.5
         te1 = 5.0;  te2 = 16.0;  te3 = 20.0;  te4 = 30.0
      else if( iage .gt. 1 ) then
         xk1 = 0.1;  xk2 = 0.98;  xk3 = 0.98;  xk4 = 0.5
         te1 = 5.0;  te2 = 16.0;  te3 = 20.0;  te4 = 30.0
      endif
!
      tt5 = ( 1.0 / ( te2 - te1 ) )
      t5  = tt5 * log( xk2 * ( 1.0 - xk1 ) / ( (1.0-xk2) * xk1 ) )
      t4  = exp( t5 * ( wtemp - te1 ) )
!
      tt7 = 1.0 /( te4 - te3 )
      t7  = tt7 * log( xk3 * ( 1.0 - xk4 ) / ( (1.0-xk3) * xk4 ) )
      t6  = exp( t7 * ( te4 - wtemp ) )
!
      gcta  = ( xk1 * t4 ) / ( 1.0 + xk1 * ( t4 - 1.0 ) )
      gctb  =   xk4 * t6   / ( 1.0 + xk4 * ( t6 - 1.0 ) )
      gctemp= gcta * gctb
      gcmax = 0.6 * wfsi**(-0.256) * gctemp
```

```
!
!     --- multispecies functional response
!     --- usse either this or adjust little p
!
      cnum = zoop1 * vul(1)/k(1) + zoop2*vul(2)/k(2) +zoop3 * vul(3)/k(3)
      c1 = gcmax * zoop1 * vul(1) / k(1)
      c2 = gcmax * zoop2 * vul(2) / k(2)
      c3 = gcmax * zoop3 * vul(3) / k(3)
      con1 = c1 / ( 1.0 + cnum )
      con2 = c2 / ( 1.0 + cnum )
      con3 = c3 / ( 1.0 + cnum )
      con = con1 + con2 + con3
!
!     ..... egestion .....
      f = 0.16 * con
!
!     ..... excretion .....
      e = 0.1 * ( con - f )
!
!     ..... Specific Dynamic Action .....
!     *******Arrhenius age dependent SDA from 17.5% to 15% ****
      if ( iage .eq. 0 ) then
        sda = 0.15  * ( con - f )
      else
        sda = 0.175 * ( con - f )
      end if
!
!     ..... Swimming .....
      egg = 0.0
!
      return
      end subroutine
!*****************************************************************************
!* Utilities for Date Control written by Yasuhiro Yamanaka (galapen@ees.hokudai.ac.jp) *
!*****************************************************************************
!     exp. 1997/12/31 23:59:59 --> 6.223158719900000E+10
!     exp. 0000/01/01 00:00:00 --> 0.000000000000000E+00
!*****************************************************************************
      real(8) function CD2TT( Cdate )
!
      integer       :: Iyr, Imon, Iday , Ihour, Imin, Isec
      real(8)       :: ND2TT
      character(19) :: Cdate
!
      if ( len( Cdate ) .ne. 19 ) then
        write(*,*) '### Length of date is no good ###'
        stop
      end if
      read (Cdate( 1: 4),*)  Iyr
      read (Cdate( 6: 7),*)  Imon
      read (Cdate( 9:10),*)  Iday
```

```
      read (Cdate(12:13),*)  Ihour
      read (Cdate(15:16),*)  Imin
      read (Cdate(18:19),*)  Isec
!
      CD2TT = ND2TT(Iyr, Imon, Iday , Ihour, Imin, Isec)
!
      return
      end function
!**********************************************************************
!     exp. 6.223158719900000E+10 --> 1997/12/31 23:59:59
!**********************************************************************
      character(19) function TT2CD(tt)
!
      integer :: Iyr, Imon, Iday , Ihour, Imin, Isec
      real(8) :: tt
!
      call TT2ND( Iyr, Imon, Iday, Ihour, Imin, Isec , tt )
!
      write(TT2CD,'(I4.4,5(A,I2.2))') Iyr, '/', Imon, '/', Iday, &
                              ' ', Ihour, ':', Imin, ':', Isec
!
      return
      end function
!**********************************************************************
!     exp. 1997,12,31,23,59,59 --> 6.223158719900000E+10
!**********************************************************************
      real(8) function ND2TT(Iyr, Imon, Iday, Ihour, Imin, Isec)
!
      integer   :: IM2D(12,0:1) = &
         reshape( (/ 0,31,59,90,120,151,181,212,243,273,304,334,  &
                     0,31,60,91,121,152,182,213,244,274,305,335  /), (/12,2/) )
      integer   :: Iyr, Imon, Iday, Ihour, Imin, Isec
      integer   :: Iy4, Iy1, Ileap, Im, Itt
!
!
      Iy4 = 1461 * ( Iyr / 4 )
      Iy1 = 365 * mod( Iyr, 4 )
!
      if ( mod( Iyr, 4 ) .ne. 0 ) then
         Ileap = 0
      else
         Ileap = 1
      end if
      Im = IM2D( Imon, Ileap)
!
      Itt = Iy4 + Iy1 + Im + Iday - Ileap
!
      ND2TT = Ihour * 3600 + Imin * 60 + Isec
      ND2TT = ND2TT + Itt * 86400.0D0
!
      return
```

```
      end function
!************************************************************************
!     exp. 6.223158719900000E+10 --> 1997,12,31,23,59,59
!************************************************************************
      subroutine TT2ND(                                     &
              Iyr   , Imon  , Iday   , Ihour, Imin, Isec,   & !O & I
              tt    )
!
      integer :: Iyr, Imon, Iday , Ihour, Imin, Isec
      integer :: Itt, Iy, Iy4, Iyd, Iy1, Ileap, Imd, Im, Its
      integer   :: IM2D(12,0:1) = &
        reshape( (/ 0,31,59,90,120,151,181,212,243,273,304,334,  &
                0,31,60,91,121,152,182,213,244,274,305,335  /), (/12,2/) )
      integer :: IY2D(4) = (/0,366,731,1096/)
      real(8) :: tt, tt0, ND2TT
!
!     ..... ITT [day] .....
      Itt = 1 + tt / 86400.0D0
!
      Iy4   = (Itt-1) / 1461
      Iyd   = Itt - Iy4 * 1461
      do IY = 1, 4
        if ( IY2D(Iy) + 1 .le. Iyd ) then
          Iy1 = Iy
        end if
      end do
!
      Iyr   = Iy4 * 4 + Iy1 - 1
      if ( mod(Iyr,4) .ne. 0 ) then
        Ileap = 0
      else
        Ileap = 1
      end if
      IMD = IYD - IY2D(IY1)
!
      do IM = 1, 12
        if ( IM2D(IM,ILEAP)+1 .le. IMD ) then
          IMON = IM
        end if
      end do
      IDAY = IMD - IM2D(IMON,ILEAP)
!
      TT0 = ND2TT(IYR, IMON, IDAY ,0,0,0)
      ITS = nint(  TT - TT0 )
      Ihour = ITS / 3600
      Imin  = ( ITS - Ihour * 3600 ) / 60
      Isec  = ITS - Ihour * 3600 - Imin * 60
!
      return
      end subroutine
```