DYMEX v2

Insect Tutorial

Table of Contents

An Overview of DYMEX

DYMEX is a computer package that enables interactive modelling of fluctuating populations of organisms in changing environments. The program provides the user with almost unlimited flexibility when building a model because the choice of the model's variables, functions and parameters, as well as its applications are made by the user who determines the level of complexity required. Similar flexibility is available when running the model applications because parameter values of the model can be altered within user-set boundaries in order to manipulate the model behaviour. Model refinement is therefore an iterative procedure.

It should be understood at the outset that DYMEX's applications are not limited to insects. DYMEX can be used to model the population dynamics of almost any species and to describe the environments in which organisms exist.

The DYMEX Package

Two separate programs are contained within the DYMEX package but the operation of each is complementary to the other. The DYMEX **Builder** is used to build the model, while the **Simulator** is used to run the model over a given period of time. The actual population model is a file, which always has the ending "gmd" (the "Model Description File"). This file is created by the Builder when a model is saved, and loaded into the Simulator when a model is opened for running. The user cannot edit the "gmd" file. Section 2 of the Simulator User's Guide gives a description of DYMEX, the associated files and structure of a DYMEX model. It should be read at this point to gain the maximum benefit from this tutorial. Note that this tutorial assumes that Version 2 of the DYMEX package is being used.

The Model Builder and its Modules

When DYMEX's **Builder** is used to build a model to describe a population's dynamics, it employs predetermined units called modules. The modules may be thought of as "building blocks" where each block has a set shape, which can be joined with other blocks to build a structure. The module concept differs from the building block concept in that the DYMEX modules can not only be joined in different ways; they can also pass information about the structure between each other. Like building blocks, some DYMEX modules can be used for a number of purposes, while others have a single purpose. Timer, Lifecycle, QueryUser, Evaporation, Meteorological Database (MetBase) and Event are some of the available module types in DYMEX.

The name of each type of DYMEX module suggests its purpose. For example, Timer performs the timekeeping for the model while Lifecycle models the organism's life cycle. Several modules allow the supply of external data to the model (for example, QueryUser and MetBase). Other modules such as Evaporation and Daylength are very specific as their name suggests. The Event module allows the user to introduce a management event into the model such as spraying or fire.

Building a Model

Before starting to build a model, all aspects of the species to be modelled should be considered together with the desired information outputs from the model. Some information about the organism's lifecycle may be unknown and estimates will then have to be used when choosing values for parameters which determine how the organism reproduces, matures or reacts to events. Familiarity with the module concept in DYMEX is of critical importance. If the user has limited understanding of the species' ecology, models may be expected to be equally limited in how well they simulate the actual changes in a species' abundance.

It is also important to know that a DYMEX model need not contain an organism. Although of limited use, models can be built which process only climatic or other similar variables. When these are run over a time period with access to a meteorological database, a variety of outputs can be produced including values for soil moisture, humidity, etc. which can be graphed or tabulated to provide information about the climatic conditions under which the organism's population is existing.

There are many ways to approach the building of a DYMEX model. One method is to start with a very simple model, which can be run, tested, and progressively refined by the addition of more modules. For example, a simple initial model could contain just two modules, a Timer and a Lifecycle. Lifecycle modules can simulate almost any aspect of species' life by setting up units within the Lifecycle module called "Lifestages". Each Lifestage represents one part of the life cycle. For example, a bee might have 7 lifestages of egg, larva1 (0-3 days), larva2 (4-12 days), pupa, juvenile worker and adult worker, and reproductive adults, if such a level of realism was required. However if the user was interested in a pathogen which attacked only adult bees, a Lifecycle module might contain only 3 lifestages: a "Juvenile", "Adult workers" and "Reproductive adults", with the "Juvenile" lifestage combining all previous lifestages up to adult. As in all aspects of constructing a model, the level of detail is determined by the final purpose of the model, and the available data.

Once the number of lifestages in the Lifecycle module is decided, a number of aspects must be considered: when and under what circumstances does the species develop; what are the conditions under which the species passes from one lifestage to the next; how is mortality to be modelled for each lifestage; what are the effects of climate; how and when does the species reproduce; what information is required from each lifestage? The list is bounded only by the degree of realism desired by the user and the time that can be allocated to the model's construction.

The complexity of information required by the Lifecycle module will almost certainly require additional modules to be added to the model in order to adequately form the required inputs to the Lifecycle module. For example, in order to model the effects of dryness on mortality of a species, DYMEX modules which refer to latitude (which in turn determines solar radiation intensity), rainfall and evaporation have to be linked to the function within the lifecycle which determines the final mortality rate. In many ways, the Lifecycle module can be thought of as the core of the model with peripheral modules acting as driving inputs. The Timer module keeps the whole system in step.

The Simulator

The Simulator is the vehicle that takes the model created using the Model Builder and allows the user to perform experiments on it. Each of these experiments is termed a simulation run. The RUN command starts the simulation, using default parameter values established while building the model, unless new values are set prior to the run. The Simulator controls the nature of the presentation of the results.

The Simulator can provide graphical and tabular outputs of the model variables over a period of time. The variables that can be output include all the module outputs as well as many aspects of the lifestage populations of the organism being studied. With these, the user can, for example, determine the best strategies to control a pest population. Provided the user has built this aspect into the model, the Simulator is able to adjust model parameters within pre-set limits and the user is thereby able to run the model for different settings of its parameter values.

If an Event module (such as spraying/dipping/ploughing etc.) has been built into the model, DYMEX offers another option in the form of a "Run Sequence". Here, the Simulator can be instructed to run the model successively over a single time period and step the event at a pre-set regular interval through that period. For example, suppose the period was a year and the regular interval was a week. The Simulator would then process 52 runs over the year's time period and in each the event would be placed one week further into the year. Using such a sequence the user can determine from graphical or tabular outputs where the event caused most destruction to the pest population. Similarly, "sensitivity" analyses, in which selected parameters are varied systematically over a defined range, can be performed automatically. This provides extra information to the ecologist to help determine where effort should be expended in collecting additional data for model improvement.

1 An Introduction to Modelling with DYMEX

Important
This set of tutorials assumes that the user has no previous knowledge of the DYMEX package. It is designed to be followed sequentially. Do not skip any section, as vital information will be missed.

1.1 The DYMEX Programs

DYMEX comes in two parts: the **Builder** and the **Simulator**. The **Builder** is used to build a model, while the **Simulator** applies the model's set of functions and parameters and simulates possible outcomes. This introduction will discuss the procedures involved in assembling such a model. DYMEX models can be extremely complex, however this introductory tutorial uses a greatly simplified hypothetical species so that the resulting model is reduced to the absolute minimum. This illustrates the basic procedures involved in operating DYMEX, and also shows how additional functionality can be added to this simple model in order to increase its realism. The complexity of any model created using the DYMEX **Builder** is determined by the user within the bounds of the **Builder**'s capabilities. The model's accuracy and reliability are measured by how well the model represents the real system. *Always remember that DYMEX is modelling a population, not an individual organism.*

1.2 Modelling a hypothetical Species

1.2.1 The "Pseudo-aphid'

The characteristics of the hypothetical species used in this tutorial were suggested by rose aphids and for convenience, the species is referred to as a "Pseudo-aphid" in all further discussions:

> Pseudo-aphids have an adult stage, which they reach 10 days after birth. The adults live for 20 days (i.e. from birth to death is 30 days total). Since the pseudo-aphids are parthenogenetic, all offspring are female and there is no egg stage. The adults begin to produce their 5 progeny after 8 days in the adult stage. The juveniles are produced at a rate which, in a small population, would be equivalent to two births on day 1, four births on day 2, six births on day 3, and so on. The population is free of predators and disease, so all pseudo-aphids only die after 20 days as an adult. Food is assumed to be abundant and has no effect on the population other than to allow it to increase.

1.2.2 Model Attributes

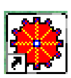
Even with such a simple life cycle, there are a number of attributes that need to be described in DYMEX. From the description of the life cycle, a DYMEX model will have to include the following:

The number of life cycle stages
The type of each life cycle stage (eg. juvenile, adult)
The length of time an average individual spends in each life cycle stage
Conditions which affect the organism's transformation from one stage to another
The timing of reproduction
The number of potential offspring
The pattern of production of offspring (eg batches, in a single birth, etc.)
Mortality and when and how it occurs; and finally
Output of results from the model.

Throughout the following tutorial, the user should refer to this list as the model is developed.

1.3 Using DYMEX to build the Model

1.3.1 Starting



Either open the “Start” menu and select “Model Builder” or if it is on the desktop, select the Model Builder icon (as shown here) in order to open the Model Builder program. From this point, keystrokes will be given as complete sequences after either explanations or discussions.

The main DYMEX window contains a blank screen and a menu bar containing various options. The following procedure will allow the user to produce a new model.

1. From the menu bar, select **“File”** to produce a drop-down menu
2. Select **“New Model”** from the drop down menu.

Once this selection is made, the **“Model Details”** window should appear (Figure 1-1). This window allows the user to insert details about the model and it should always be at least partially completed at this stage. The name of the new model, the name of the builder and the version number should be inserted; details on the model’s construction can be added immediately or added later as the model is developed. Note that since Version 2 of DYMEX is being used, the Version 1 option is unavailable (it will be enabled only for models created using Version 1). Whether or not the **“Model Details”** window appears automatically when a new file is created or a previously saved file is opened is determined by a setting in the **“Options”** menu. If it does not appear automatically, the user can always open the **“Model Details”** window by selecting “**Model”** from the main menu bar and then **“Details”** from the drop-down menu. If the user decides not to enter any details at this stage, then the **“OK”** button is selected from the window and the Model Window appears. The only difference will be that no information is provided to any future users of the model.

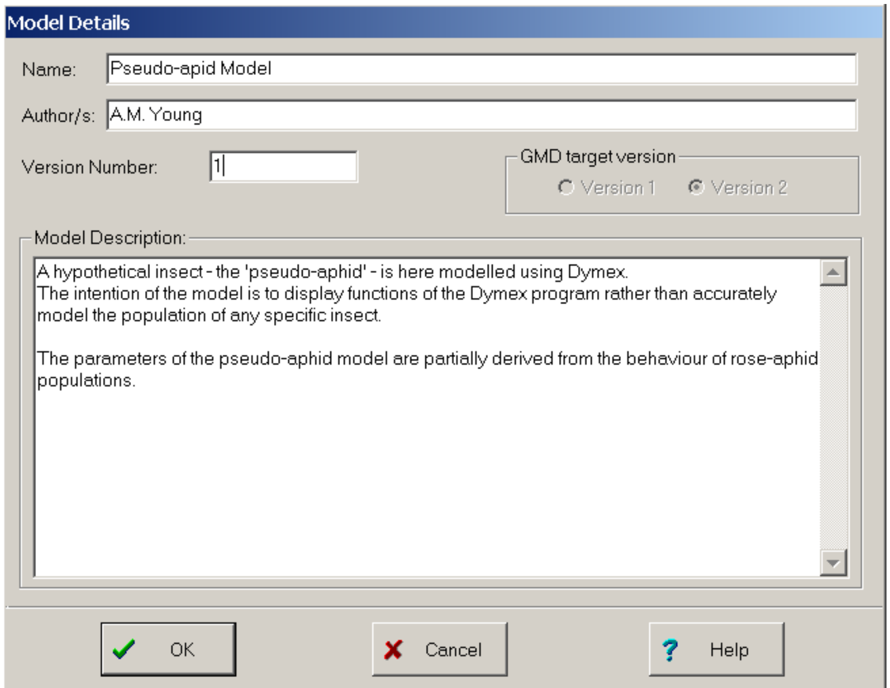


Figure 1-1 Model Details Window

3. Enter a model description and other information in the **"Model Details"** window and select **"OK"** on completion.

Once the above selection is made, the **"Model Components"** window appears (Figure 1-2). Currently, it will have no name other than the default "Document1" because it has not yet been saved under any specific model name. The **"Model"** window lists the modules used in the model currently under construction. Since all models *must* have a **"Timer"** module, this module is already present by default.

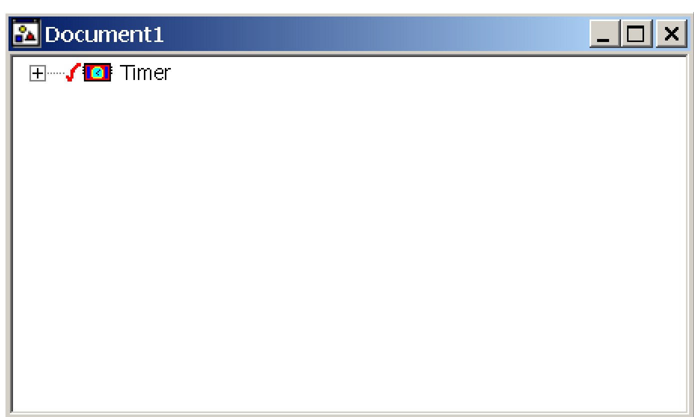


Figure 1-2 The DYMEX Model Components Window

The icon (representing a clock) indicates that the line of information represents the Timer module in the model while the red tick shows that the module already is sufficiently complete to allow a model to be run, although it may be altered by the user if a different time step or other settings are required. At the start of the line is a small "+" button. If this is selected, the module can be opened as a tree diagram to display its components (figure 1.3). By "double clicking" on any of these components, their relevant windows can be opened for data insertion. The **"Timer"** module can also be opened if the module name is "double clicked" with the mouse and the module's settings are then available for editing.

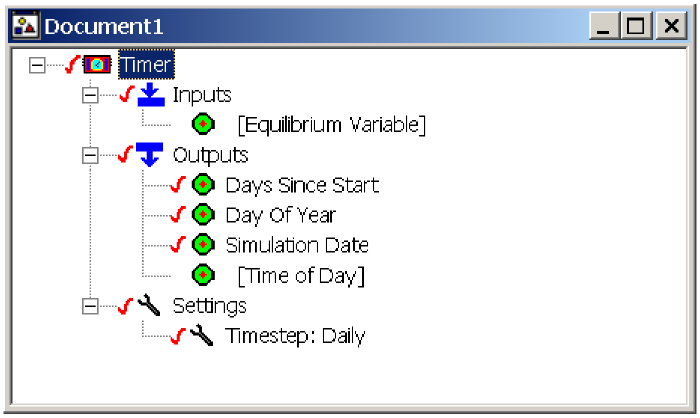


Figure 1-3 Timer module with 'Component Tree' opened

1.3.2 Building the Model

The Timer module is already in place, however it must be set up to operate with the required timestep. The timestep used should be somewhat smaller than the shortest anticipated duration of any lifestage. A weekly or monthly step may be suited to some organisms with long lifespans (for example, trees), but for many common insects, a daily step is the most appropriate. DYMEX will operate using either daily, weekly or monthly time steps as required. For very short-lived organisms, the daily timestep may be divided into even smaller segments. An advantage of using a longer timestep is that the model will run much faster in the Simulator.

1. "Double click" on the **"Timer"** text in the **"Model"** window to open the **"Timer"** dialog box

The "Timer" module cannot be given any user-defined name and this is indicated by the fact that its module name is "greyed out".

2. Select the **"Settings"** button to obtain the **"Timer Settings"** selection box
3. Check that the **Model Timestep** is set to 1 day and the **No of segments per day** is also 1.
4. Select **"OK"** and return to the **"Timer"** dialog box
5. Select the **"Outputs"** button to open the **"Outputs [Timer]"** selection box

The output variables available from the Timer module are shown in the listbox. In order to be able to display graphs on a time axis in the Simulator, at least one of these must be selected for model output. The **"Days Since Start"** variable, which reports the elapsed days since the simulation commenced, is sufficient for the puposes of this simple model.

6. In the list box, **"Days Since Start"** should be highlighted – if it is not, place the cursor on this choice and click once - it should then appear highlighted
7. Click once on the **"Select"** button - the symbol **"+>"** will appear in front of the text of **"Days Since Start"**
8. Select **"OK"** and return to the **"Timer"** window
9. Select **"OK"** and return to the **"Model"** window.

Once back in the **"Model"** window, open the **"Timer"** module's component tree (by clicking on the "+" in front of its name), and then similarly open the **Outputs** branch. If all steps have been completed correctly, there will now be a red tick in front of **"Days Since Start"**.

The next requirement for this simple model is to add the Lifecycle module that represents the Psedo-aphid's ecological characteristics.

1. From the menu bar, select "**Model**"
2. From the drop-down menu, select **"Add Module"**
3. From **"Create Module of Type ?"** list box, make sure that the Standard button is selected, and select **"Lifecycle"** from the list of modules
4. Select **"OK"**

The **"Lifecycle"** window will now be opened automatically with a **"Lifestage"** panel displayed (Figure 1-4). If the model and lifestage windows are kept minimised, the user can move back and forth between them using the standard Windows® method of clicking on the exposed part of the window required. If the two windows are set to maximised, the required window can be obtained by using the menu bar command of **"Window"** followed by selection of the required window from the drop down menu.

Important: When a model is loaded in the Builder, the model window is always loaded by default. To obtain the lifecycle window from this situation, double click on the line that represents the Lifecycle module. Once it is opened, the windows can be opened or changed as noted above. Any module in the model window can be opened by double clicking on its text.

NOTE: ***To delete a module that has been accidentally created***, return to the "Model" window and the unwanted module will be shown in the list of modules present. Click on the unwanted module so that it is highlighted. Next, select "Model" from the main menu bar and obtain the drop-down menu; select "Delete Module" and follow any required steps. When completed, the unwanted module will disappear from the listing.

At this point the lifecycle module can be given a name:

1. With the Lifecycle window active, select "**Lifecycle**" from the menu bar
2. From the drop-down menu, select **"Properties"**
3. Type the name ***"Pseudo-aphid"*** into the **"Name"** edit box
4. Select **"OK"**

1.3.3 Constructing the Lifecycle

The window now represents a **"Lifecycle"** and contains a **"Lifestage"** panel (Figure 1-4).

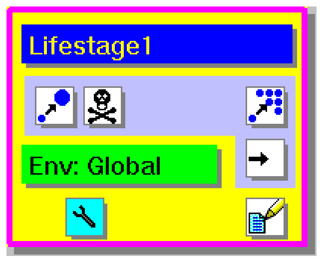


Figure 1-4 The Lifestage Panel

This panel represents one lifestage of the species being modelled. A number of environmental factors, (eg. temperature, moisture, predators, availability of food, diseases, etc.) influence the species' rate of development, survival and reproduction. DYMEX can be set to simulate these processes by using the button icons in the Lifestage panel, each of which controls some aspect of the lifestage (The function of each button icon is described below). When more than one lifestage is present, the currently selected lifestage can always be identified by its broad pink outline.

Each lifestage panel defines a particular stage (eg. egg, pupa, adult, etc.) together with its environment and attributes. The number of stages is set by the user. For the pseudo-aphid, two stages are required (juvenile and adult), however several stages could be used depending upon the detail required by the model or the life cycle of the organism. Many insects may require a number of lifestages depending upon the instars that are present. For example, a model of a tree fern would probably include stages corresponding to spore, gametophyte, gamete and sporophyte. However, it is conceivable that a user may wish to condense two of these three stages into one for a particular model and might end up with the three stages of spore, gametophyte-gamete and sporophyte. Exactly how the model is constructed depends completely on the requirements and applications of the user.

The lifestage can now be given a name. (The green lifestage button marked **"Env: Global"** is only required for more complex simulations and can be ignored for this tutorial.)

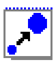
1. Click on the blue button labelled **"Lifestage1"** to open the "Lifestage Name" dialog box
2. Type **"Juvenile"** into the "Name" edit box
3. Exit from the dialog box by selecting **"OK"**.

1.3.4 Lifestage Attribute Buttons

These buttons (located in the **Lifestage** panel) permit the user to open dialog boxes in order to select variables or functions, name parameters and enter values for constants or variables. When entering variable names into edit boxes, always choose descriptive names that allow easy recognition of the variable's application in the model.

Lifestage Outputs

When opened with this button, the "Lifestage Outputs" dialog box allows the user to select which variables will be used as outputs from that lifestage. Some of these output variables could be used as input to processes elsewhere. After the model is run in the Simulator, these outputs may be tabulated, graphed or written to a file.

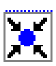


Development

The "Development" dialog box is opened with this button. It enables the user to select the functions and parameters controlling lifestage development and aging (i.e., the rate of accumulation of "Physiological Age").

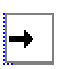
Mortality

The "Mortality" dialog box is opened with this button. The user is able to select the functions and parameters controlling the lifestage mortality rate.



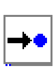
User-defined Cohort Properties

Note that this button appears on the lifecycle icon only after at least one new cohort property has been correctly defined. Within a model, certain cohort properties (such as chronological age, physiological age, fecundity, etc.) are pre-set. Often these pre-set cohort properties are insufficient, and the user may need to create new cohort properties, such as stress, size or sex ratio, for the particular organism being modelled. This button allows the user to define the processes that control the values of these properties.



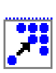
Next Stage

The **"Next Stage"** button adds a further lifestage. Once it has completed this operation, it changes and becomes a **"Stage Transfer"** button (see below) and consequently, only the last lifestage in a lifecycle will still have an operational **"Next Stage"** button. To remove an unwanted lifestage, select its panel, then choose **"Lifecycle"** from the window's menu bar, and finally select **"Delete Stage"** from the drop-down menu.



Stage Transfer

This button opens the "Transfer Function" dialog box in order to define/create or modify the transfer process which governs how an organism moves from one lifestage to the next. It is derived from the **"Next Stage"** button (see above).



Reproduction

The **"Reproduction"** dialog box is opened with this button. Two processes, "Fecundity" and "Progeny Production" (as well as their associated parameters) can be selected by the user.

Other Lifestage Properties

This button opens the "Lifestage Properties" dialog, which gives access to various properties that may be used in more advanced models. One of these properties is the "Resource Variable", which is used as the divisor by the program when it is calculating density.

1.3.5 Completing the Lifecycle

Since the pseudo-aphid only occurs as either a juvenile or an adult, two life cycle stages are needed in the model.

1. Select the **"Next Stage"** button
2. Name the new lifestage **"Adult"** as described in Section 1.3.3.

Reproduction for the Adult stage is now modelled.

1. Select the **"Reproduction"** button
2. In the **"Adult Reproduction"** dialog box use the **"Destination Stage"** scroll button (▾) to find and select **"Juvenile"**.
3. Select **"OK"**.

A line ending in an arrow is now present which links the reproductive stage back to the juvenile stage, and the whole structure should now resemble Figure 1.5. The arrows on the diagram define the direction of the flow of individuals within the life cycle.

(Note that the Juvenile lifestage's **"Next Stage"** button has become a **"Stage Transfer"** button.)

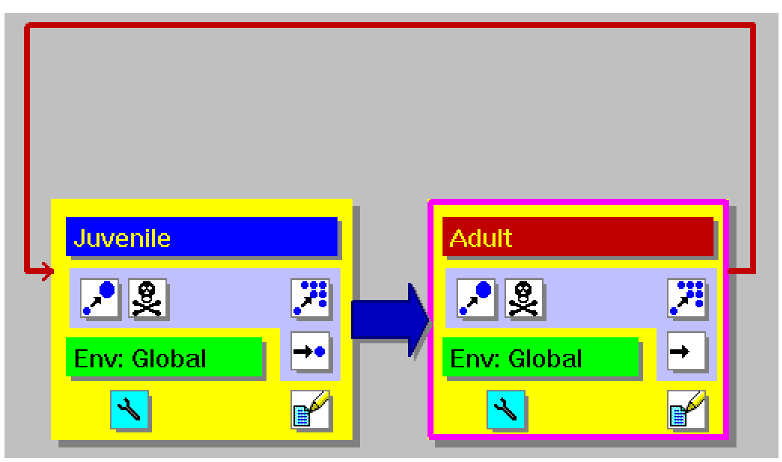


Figure 1-5 The completed life cycle structure for the pseudo-aphid

1.3.6 Setting Lifestage Processes, Functions and Parameters

Once the structure of the life cycle has been defined, the physiological and ecological processes that define conditions under which individuals develop, die, and reproduce need to be specified. The lifestage buttons are used to select and define these relationships. Because the pseudo-aphid has such a simple life cycle, not all of the buttons and their operations are needed at this stage.

1.3.7 Completing the Juvenile Stage

In this initial form of the pseudo-aphid model, Juveniles have no development requirements, reproduction or death, so the **"Development"**, **"Reproduction"** and **"Mortality"** buttons are ignored for this lifestage. The **"Lifestage Outputs"** button is required to produce an output from the stage and the **"Stage Transfer"** button is used to set the conditions under which the pseudo-aphids become adults. The extreme simplicity of the model means there are very few output variables to consider.

1. Select **"Lifestage Outputs"** button to obtain the **"Juvenile Outputs"** dialog box
2. In the **"Module Output Variables"** list box, highlight **"Total Number",** then click on the **"Select"** button
3. Select the **"Rename"** button and type in a suitable name (eg. "Total Number of Juveniles"
4. Select **"OK"** until back at the **"Lifecycle"** window.

The red tick that now appears on the "**Lifestage Outputs"** button indicates that an output variable has been successfully selected for use by the model. **Always** give variables names that are easily recognisable and distinct from all others. Since each lifestage has the same default names for its output variables, they will be rejected by the Simulator unless the user inserts new names for each variable used in the model.

The **"Stage Transfer"** button is used to modify the process under which the pseudo-aphids change from Juveniles into Adults. When variables and functions for any "rates of change" lifestage process are required, DYMEX uses a standard dialog box (Figure 1-6). The top line contains the name of the process and a button ("Description") that can be used to provide a comment for the process. The large panel labelled "Components" will list the process components. The "Cut", "Copy" and "Paste" buttons can be used to copy an existing process component, and then insert (paste) it into another process. Components can be parameters, functions or even other processes and a new component is added by clicking on either the "Parameter", "Function" or "Process" button. An existing component is changed by highlighting it and the clicking on the "Edit Component" button. "Delete Component" has the same effect as "Cut", except that the component is removed completely (it is not available for "pasting"). The "Change" button in the "Combination Rule" panel is used only when two or more Process Components are used to define the process.

1. Select the **"Stage Transfer"** button.

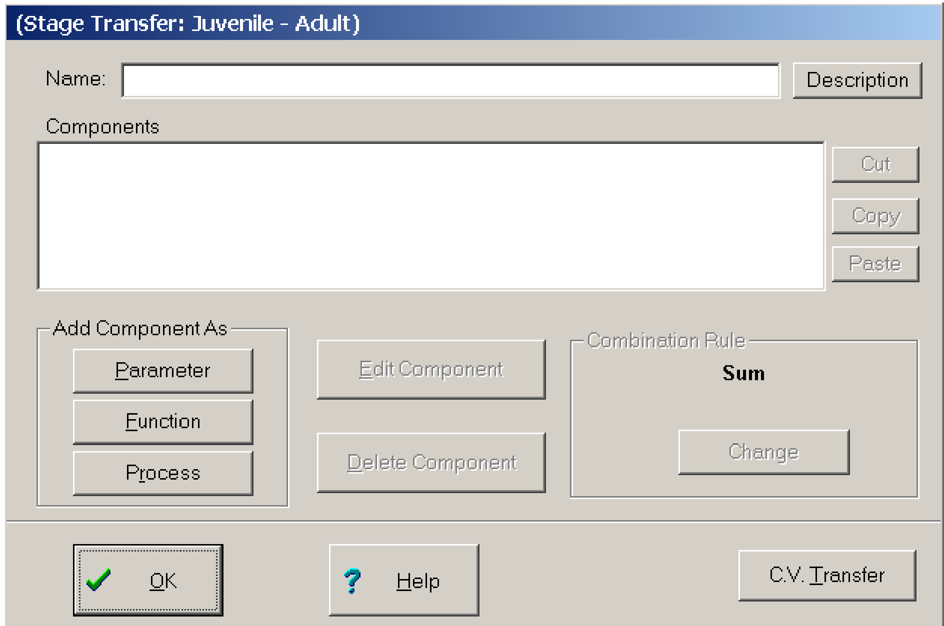


Figure 1-6 The "Juvenile – Transfer" dialog box

The **"Juvenile – Transfer"** dialog box is now used to select the functions and variables that define stage transfer. All Juvenile pseudo-aphids become adults 10 days after birth which implies the stage transfer independent variable is chronological age; since all pseudo-aphids become adults simultaneously, a step function is indicated.

2. The stage transition being represented here is the juvenile to adult "moult", so type "Moult" into the process **"Name"** box.
3. Select the **"Function"** button to add a process component that is a function and to obtain its dialog box (Figure 1-7).

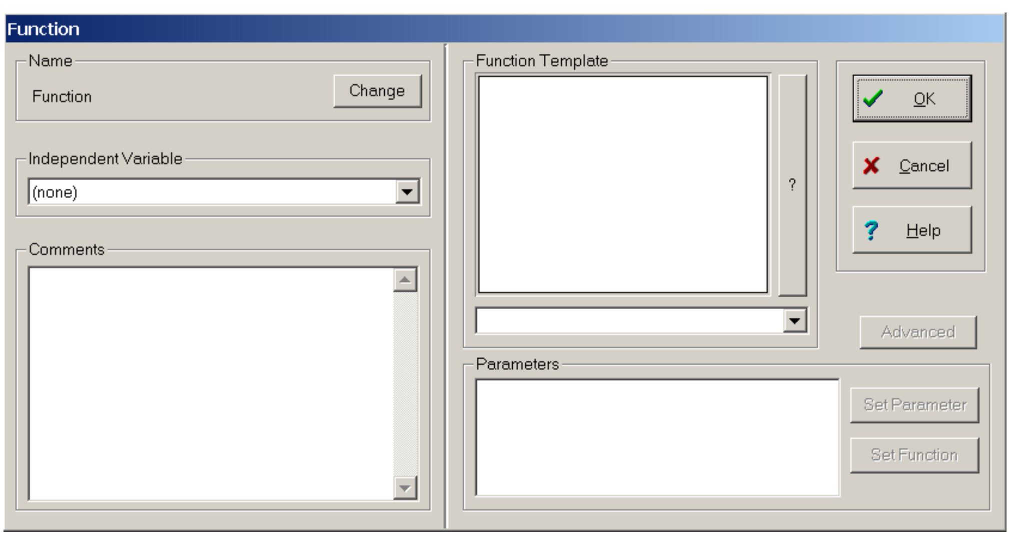


Figure 1-7 The "Function" dialog box

The **"Function"** dialog box (Figure 1-7) is used throughout the Builder to specify process components that are functions, and will be used here to set the stage transfer (moulting) function and its associated variables. Under the white screen on the right (in the **"Function Template"** panel) is a button that permits selection from the library of mathematical functions contained in DYMEX. When a function is selected, its shape is illustrated on the screen. The **"Parameters"** panel (at the lower right) is used to set default and limiting values for the parameters of the function. (The **"Set Function"** button even allows another function to be substituted for a parameter, though this feature will not be used in the simple Pseudo-aphid model). A **"Comments"** edit box can be used to record pertinent information such as a reference to the source of the data used to estimate parameter values.

4. Using the function scroll button (at the bottom right of the **"Function Template"** panel), select **"Step"**
5. Select the "Independent Variable" scroll box button and then select **"Chronological Age"** from the list that appears.

*A "Step" function will now be illustrated in the screen and "Chronological Age" will have appeared in the "Name" box. The name of the function should be suitably re-defined. The "Step" function, which defines the "Juvenile to Adult Stage Transfer", requires two parameters: the Chronological Age at which the juveniles become adults and the proportion of juveniles which actually become adults during that time step. These two parameters are shown in the **"Parameters"** panel. Note that information that describes the selected function in detail (including its equation) is available by clicking on the long, narrow button labelled **"?"** in the **"Function Template"** panel.*

6. Click on the **"Change"** button and suitably name the function (eg. "Juvenile to Adult Transfer Function")
7. Select the first parameter in the **Parameters** panel (**"p1: Threshold"**) and click the **"Edit Parameter"** button.

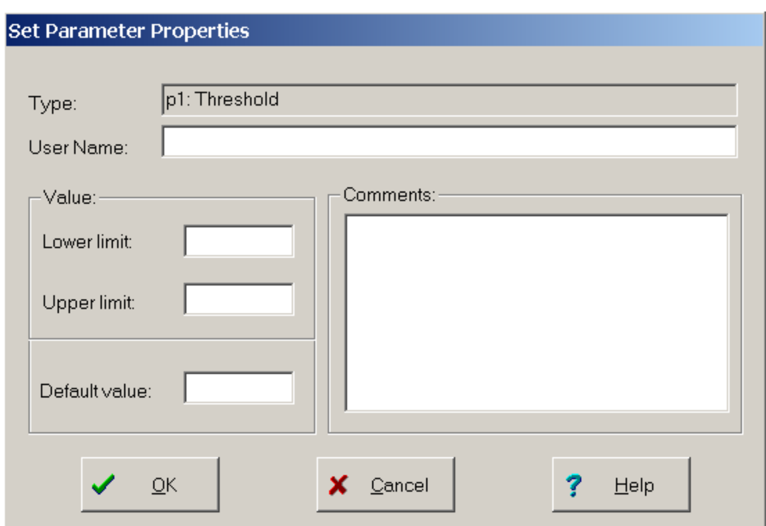


Figure 1-8 The "Set Parameter Properties" dialog box

The **"Edit Parameter"** button opens the "Set Parameter Properties" dialog box (Figure 1-8). The selected parameter, "p1: Threshold" is shown in the **"Type"** box. An edit box (**"User Name"**) allows a user-defined name to be inserted for the parameter and this should always be done as otherwise the user may confuse parameters with the same names. The numbers that define the value of the parameter are entered in three edit boxes titled "Lower limit", "Upper limit" and "Default value". For the pseudo-aphid, the default Threshold value is 10; the lower and upper limits define the range over which the parameter can be varied while the model is in DYMEX's Simulator. If no limits are set, the Threshold value of 10 days can be varied to any value; if upper and lower limits are both set equal to 10, the parameter cannot be varied at all. With different limits (eg 5 and 12), the Threshold value is restricted to that range. A comment edit box is provided for explanatory remarks.

8. Select **"Lower limit"** edit box, type in the value 5
9. Select **"Upper limit"** edit box, type in the value 15
10. Select **"Default value"** edit box, type in the value 10
11. Select **"User Name"** edit box, delete "Threshold" and type in a suitable name (eg. "Juvenile to Adult Threshold")
12. Type in comments if required by selecting **"Comments"**
13. Click **OK** to return to the Function dialog

The second parameter to be set (***"p2: Step Height"***) is the proportion of juveniles that become adults every day after the threshold of 10 days is reached. DYMEX uses a decimal fraction to indicate this proportion; a value of 1 indicates all members of the population become adults simultaneously.

14. From the **"Parameters"** list box, select **"p2: Step Height"**, and click on the **"Edit Parameter"** button to open the **Set Parameter Properties** dialog
15. Select in turn each limit edit box and type in the value 1
16. Select the **"Default value"** edit box, type in the value 1
17. Select the **"User-defined Name"** edit box, delete **"Step Height"** and type in a suitable name (eg. "Proportion of juveniles becoming adults")
18. Type in comments if required by selecting **"Comments"**
19. Select **"OK"** in the dialog boxes as necessary and return to the lifecycle window.

The "Stage Transfer" button will now have a red tick to indicate its parameters are set and this completes the settings for the Juvenile lifestage.

1.3.8 Completing the Adult Stage

The simplicity of this present pseudo-aphid model means that the two buttons **"Development"** and **"Stage Transfer"** can be ignored for the Adult stage; the former because the model uses "Chronological Age" to determine maturation, and the latter because there is no "next stage". Data output from the stage is specified by selecting the **"Lifestage Outputs"** button.

1. Select the **"Lifestage Outputs"** button to obtain the **"Adult Outputs"** dialog box
2. In the "**Module Output Variables"** list box, highlight **"Total Number",** then click on the **"Select"** button
3. Select **"Rename"** button and type in suitable name (eg. "Total Number of Adults"
4. Select **"OK"** until back at the **"Lifecycle"** window.

The "Adult Outputs" button will now have a red tick displayed indicating that its operations have been set correctly.

As the pseudo-aphids all need to die as adults, a **"Mortality"** process needs to be set. There are three types of Mortality process: "Continuous", "Establishment" and "Exit". Continuous mortality operates throughout the duration of the lifestage and is the most common type of mortality. However, certain species pass through life cycle stages where considerable mortality occurs when the organism tries to gain a "foothold" in its new stage (eg. ticks attaching to an animal - a large proportion are rejected during their initial attempts to attach). Organisms such as this will require an "Establishment" Mortality process (which acts only during the move to the new stage) in addition to, or instead of, the "Continuous" Mortality process. The "Exit" mortality acts only on those individuals that are leaving the lifestage to transfer to the next stage. The pseudo-aphid requires only the "Continuous" process. Since the adults all die at the end of a fixed time period, their mortality-inducing variable is "Chronological Age", with the effect being defined by a "Step" function. The threshold value (i.e., the age of the adults at death) will be 20 and the constant value will be 1 (i.e, they all die on the 20th day). Note that the "Chronological Age" is a measure of the length of time an individual has stayed **within that stage**. The pseudo-aphids are dying at the end of a 30-day life span. However, since "Chronological Age" in the Adult stage refers only to the length of time an individual has spent in that stage, 20 days is the correct value of the parameter.

1. Select the **"Mortality"** button on the **"Adult"** lifestage
2. In the **"Adult Mortality"** dialog box, select the **"Continuous"** button
3. In the **"Mortality (Adult)"** dialog box, give the process a name such as "Continuous Adult Mortality"
4. Click on the **"Function"** button to obtain the "Function" dialog box
5. Using the Function Template scroll button, select **"Step"**
6. From the **"Independent Variable"** list box, select **"Chronological Age"**
7. Select the **"Change"** button and suitably name the function (eg. "Age-related mortality")
8. In the **"Parameters"** panel, select the first parameter **("p1: Threshold")** and click on the **"Edit Parameter"** button to obtain the "Set Parameter Properties" dialog box
9. Select the lower and upper limit edit boxes and type 20 in each
10. Select the **"Default value"** edit box and enter the value of 20

11. Select the **“User Name”** edit box, delete **“Threshold”** and type in a suitable name (eg. “Age at Adult Death”)
12. Type in comments if required by selecting **“Comments”**
13. Click **“OK”** to return to the Function dialog, select the second parameter (**“p2: Step Height”**), and click on the **“Set Parameter”** button
14. Select the **“Default value”** edit box and enter the value of 1
15. Select the lower and upper limit edit boxes and enter 1 in each
16. Select the **“User Name”** edit box, delete **“Step Height”** and type in a suitable name (eg. “Proportion of adults dying”)
17. Type in comments if required by selecting **“Comments”**
18. Select **“OK”** as necessary to exit and return to the life cycle window.
19. Save the model by selecting **“File”** on the main menu bar, followed by **“Save”** from the drop-down menu.

Note again how a red tick is displayed on the “Mortality” button once the dialog boxes are all closed, indicating that all functions and parameters are correctly set.

1.4 Progeny Production in the Pseudo-aphid

1.4.1 Introduction

The starting point for setting **Reproductive parameters** is the “Reproduction” dialog box (Figure 1-9). Three components determining reproduction can be set: **Fecundity(E)**, **Fecundity(R)** and **Progeny Production**. Fecundity is the total number of possible offspring that can be produced per individual. The first two components set this number in slightly different ways. **Fecundity(E)** is an “establishment” process, acting only when individuals first enter the adult stage. This is often the most appropriate way to set the fecundity for an insect – i.e., a maximum number of potential offspring (which may depend on how well the insect has fed in the juvenile stages). **Fecundity(R)** is a “continuous” process and can be used to “recharge” the fecundity at intervals. This is useful in plant models, where, for example, a tree will produce fruit every year, with the number of fruit produced depending on the soil moisture or size of the tree. Progeny Production determines the rate at which the progeny are produced over time. Note that the Fecundity does not take into account that half the population consists of males, and normally this will need to be modelled by adjusting the value of Fecundity by the sex ratio. The Pseudo-aphid of our example, however, is parthenogenetic (only females are present) and this is not a consideration.

For the pseudo-aphid model, only the **Fecundity(E)** process is used, and a constant value of 5 was chosen. Fecundity will usually vary with environmental factors, but in this model of the pseudo-aphid, Fecundity will not alter. “Progeny Production Rate” defines the rate at which the offspring are produced. For example, some species produce all their progeny at once; others produce batches of offspring over short periods of time separated by intervals, while others may steadily increase production of offspring over a period of time and then decrease the production rate gradually or rapidly back to zero.

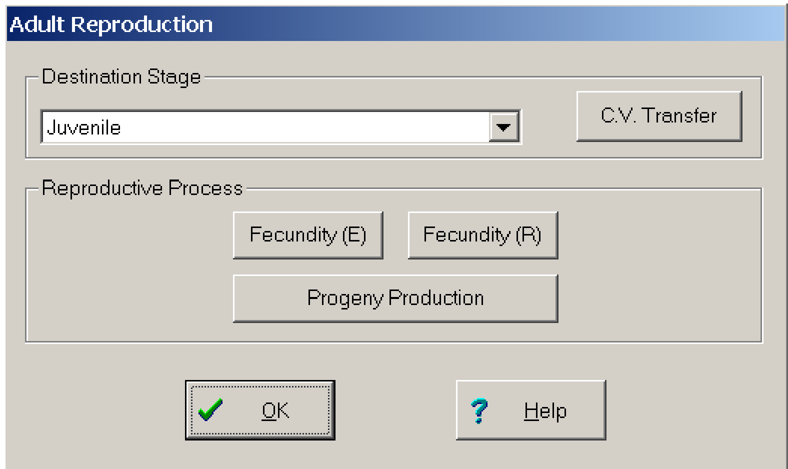


Figure 1-9 The "Reproduction" dialog box

1.4.2 Setting Fecundity

1. Select the **"Reproduction"** button on the Adult lifestage
2. Select the **"Fecundity (E)"** button to obtain the **"Fecundity (Adult) - Establishment"** dialog box
3. Type in ***"Fecundity"*** as the **"Name"** of the process

*The **"Fecundity"** dialog box has the standard process format. Nothing in the pseudo-aphid model affects Fecundity so it is set to a constant value of 5 progeny per adult female aphid.*

4. Select the **"Parameter"** button to obtain the **"Set Parameter Properties"** dialog box
5. Select the **"Lower limit"** edit box, enter the value 1
6. Select the **"Upper limit"** edit box, enter the value 8
7. Select the **"Default value"** edit box, enter the value 5
8. Select the **"User Name"** edit box and enter a suitable name (eg. "Pseudo-aphid Fecundity")
9. Select **"OK"** until the "Adult Reproduction" window is reached.

1.4.3 Setting Progeny Production

In order to present the use of different functions to the reader, the pseudo-aphid was defined as having a birth rate equivalent to a linear function with a slope of 0.2 starting on day 8. This means that at 8 days, an adult female with a fecundity of 5 potential offspring will commence reproduction. After a further day (the 9th day of her adult lifestage) she will produce 0.2 progeny; after 2 days (the 10th day of her adult lifestage) she will produce 0.4 progeny and so on. The progeny production reaches a peak at day 14 with 1.2 individuals and on day 15 the remaining progeny of 0.8 individuals are produced (Figure 1-10).

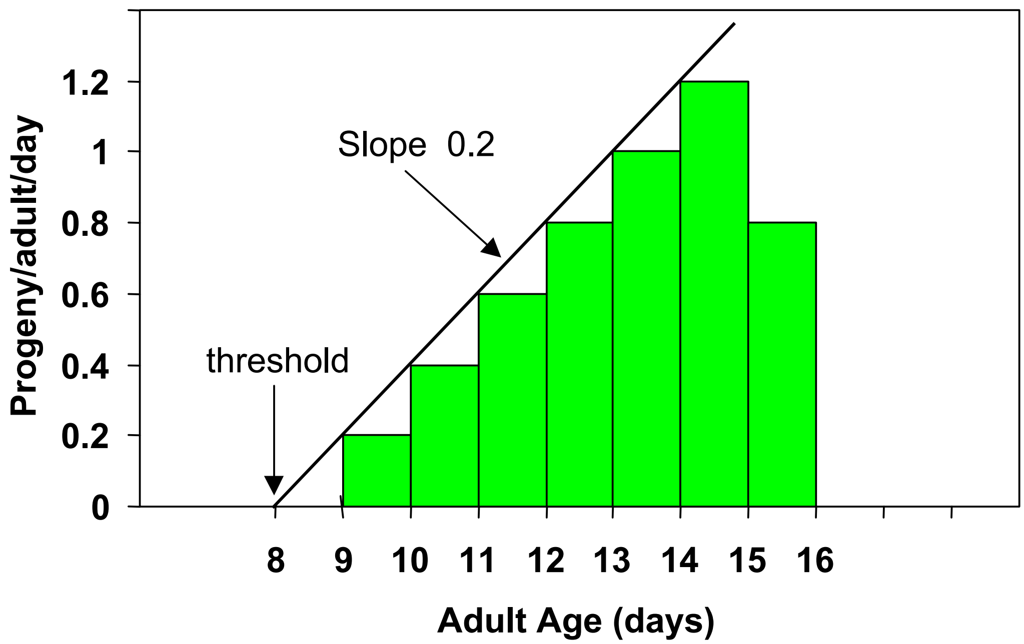


Figure 1-10 The Progeny Production Function

While the fractional number of progeny do not apply to an individual pseudo-aphid, they are readily applicable to populations. An important point to remember is that although the pseudo-aphid's progeny production graph shows that the actual fecundity is exactly equal to the potential fecundity, shown by the area under the curve (shaded), this is in general not so. It is so only because of the deliberately limited nature of this hypothetical organism. The effects of Mortality invariably reduce the actual fecundity.

1. Select the **"Progeny Production"** button in the **"Reproduction"** dialog box to obtain the **"Adult-Progeny Production"** selection box
2. Provide a name for the process (for example, "Progeny Production rate")
3. Select **"Function"**
4. Select the **"Change"** button and suitably name the function (eg. "Rate of Progeny Production")
5. Using the function scroll button, select **"Linear above Threshold"**
6. Using the **"Independent Variable"** list box, select **"Chronological Age"**
7. Select the first parameter (**"p1: Threshold"**) in the **"Parameters"** panel, then click on **"Edit Parameter"** to obtain the **"Set Parameter Properties"** dialog box
8. Select the **"Lower limit"** edit box, enter the value 4
9. Select the **"Upper limit"** edit box, enter the value 12
10. Select the **"Default value"** edit box, enter the value 8
11. Select the **"User Name"** edit box, delete "Threshold" and enter a suitable name (eg. "Commencement of offspring production")

12. Enter comments if required
13. Click **“OK”** to return to the **“Function”** dialog box
14. In the **“Parameter”** panel, select **“p2: Slope”** and click the **“Edit Parameter”** button
15. Select in turn each of the limit edit boxes and enter the value 0.2 for each
16. Select the **“Default value”** edit box and enter the value 0.2
17. Select the **“User Name”** edit box, delete the “Slope” and enter a suitable name (eg. “Rate of offspring production”)
18. Enter comments if required
19. Select **“OK”** as required and exit to the **“Lifecycle”** window
20. Save the model by selecting “File” from the main menu followed by “Save As” from the drop-down menu and then insert a suitable name for the model.

Assuming all is correct, there will now be 5 red ticks on the **“Lifecycle”** diagram; one each on the “Lifestage Output” and “Lifestage Transfer” buttons of the Juvenile stage and one each on the “Lifestage Output”, “Mortality” and “Reproduction” buttons of the Adult stage.

If the Model window is examined, the lifecycle module should now exhibit a red tick beside its cube icon indicating that sufficient of its variables have correctly entered values to permit a simulation run to be conducted. While in the “Model Components” window, it is worth examining the model by expanding its tree diagram (The tree diagram can be opened fully by selecting **“Model”** from the main menu bar followed by **“Expand all branches”**. The same menu item can be used later to **“Close all Branches”**.). Eventually, the ends of each branch will be the parameter values that have been set during the procedures just covered. If the parameter values require alteration, the Lifecycle window can be obtained by double clicking on the terminal text values of the tree diagram and the values are then altered in the usual way.

This completes the formation of the Pseudo-aphid model. The user may now wish to use the Simulator to examine the model immediately. Before exiting from the Model Builder, read the first paragraph of the Simulator tutorial.

1.5 Using DYMEX to Run the Model

1.5.1 Starting the Simulator

The DYMEX **Simulator** can be started from the desktop either by selecting the desktop icon, or by opening the **“Start”** button for programs.

The Simulator can provide the user with hints on operation and also has a button bar that has a number of short cut buttons for frequently used operations. The **“Hints”** dialog box can be turned on or off during any operating session by selecting **“Preferences”** from the main menu bar followed by **“Show Hints”** in the drop down menu: a tick will appear beside **“Show Hints”** while the **“Hints”** dialog box is present on the screen. The **“Hints”** dialog box can be turned off permanently by setting user preferences - see the next Section.

1.5.2 User Preferences

The "Operating Preferences" selection box allows personal preferences to be set for the operating conditions of the Simulator. Until the user is more familiar with DYMEX, the default settings (Figure 1-1) are likely to prove acceptable. However, for low-resolution screens, a maximized window could be useful.

1. Select **"Preferences"** from the menu bar
2. Select **"Operating..."** from the drop-down menu
3. In the **"Operating Preferences"** selection box, check that the options are set to the defaults shown in Figure 1-11
4. Select **"OK"**.

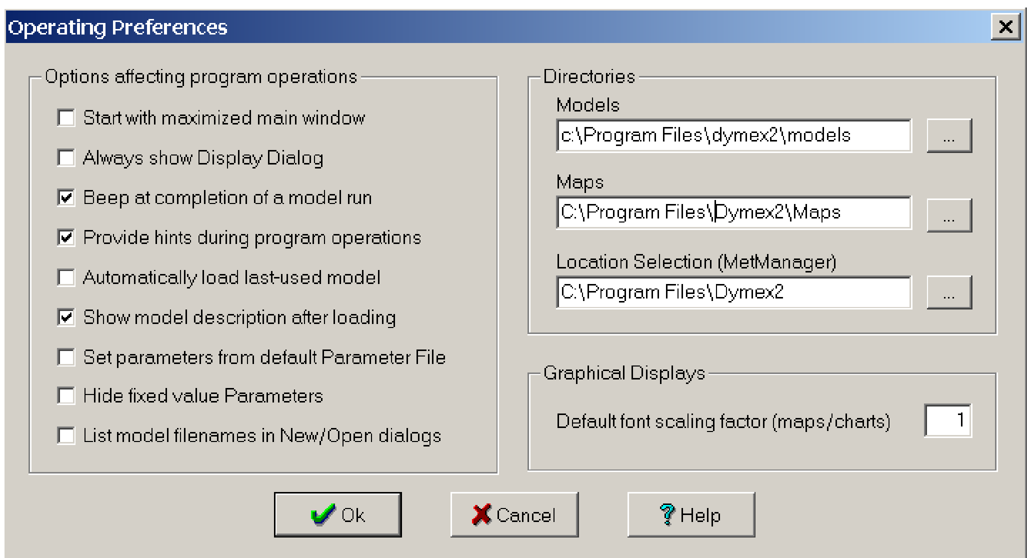


Figure 1-11 The "Operating Preferences" dialog

1.5.3 Loading Files

The model that was saved in the Builder is a Model Description (gmd) file and contains a representation of the model's structure. The Simulator does not load these files directly. Instead it loads a ***Simulation File*** (.dxs or .ini), which contains settings for a simulation run using the model. Each model can have many Simulation Files (for example, if a particular model is needed to run at 5 locations, a separate Simulation File with the appropriate settings could be used for each location). Since this is a new model, no Simulation File exists for it yet. Therefore one must be created and this is done using the **New** option on the **File** menu (also accessible via the button bar's icon). This opens the **New Simulation** dialog (Figure 1-12). It lists all of the models in the model directories in the panel at the left.

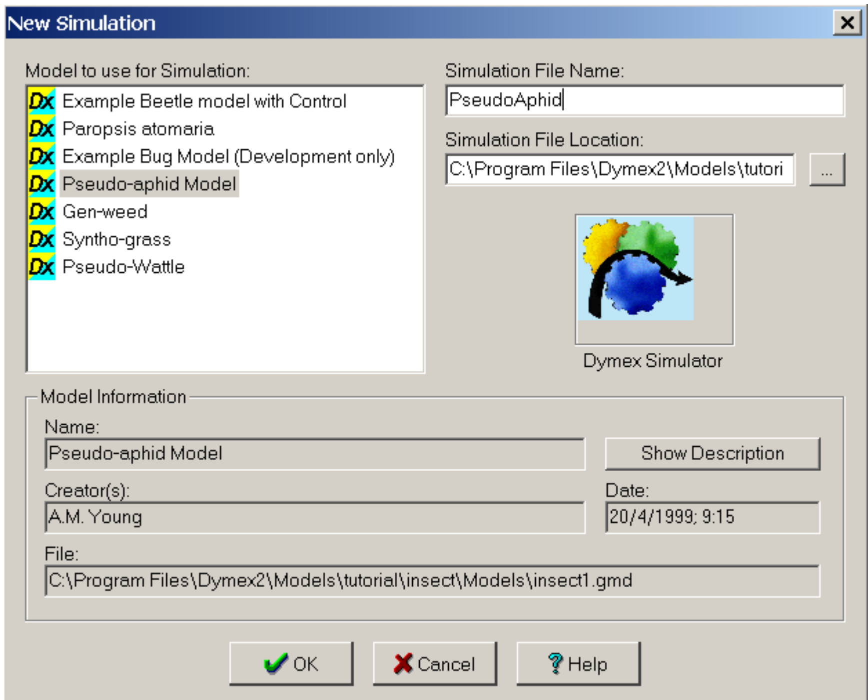


Figure 1-12 The "New Simulation" dialog

To create a new ***Simulation File***:

1. Select the Pseudo-aphid Model by clicking on it in the **"Model to use for Simulation"** panel
2. Verify that the correct model is selected by looking at the fields in the **"Model Information"** panel
3. Type in the name of the new Simulation File (for example, ***"PseudoAphid"***) in the **"Simulation File Name"** edit box
4. Click on **"OK"** to create the new Simulation File and exit the dialog.

DYMEX now loads and checks the model "gmd-file" for errors; any problems found are reported as error messages. During the loading of the pseudo-aphid file for the first time, the Simulator will also report that a Parameter ("gmp") File is missing and will be created.

While operating the Simulator, the user can alter parameter values within the ranges set by the limiting values that were incorporated into the model while it was being built in the Model Builder. The Simulator is prohibited from altering the master "gmd-file" and so during the loading of a file, the program makes a working copy of the model's parameters (the "Parameter" file) which can then be altered as the user requires. The Simulation File is a record of the user's personal settings for the Simulator. A Simulation File can be saved at any time using the "Save" or "Save As" options on the **File** menu.

If at any time, the "gmd"-file is copied to another computer or placed in a different directory, it is useful to copy the Simulation and Parameter files also. If they are not copied across, all user settings will have to be re-entered before the model will run in the new location.

From this point on, the tutorial assumes that the pseudo-aphid model has been correctly built and a new Simulation File for it is loaded in the Simulator. In the rest of this document, the "Simulation File and its corresponding model" will often be referred to as the "model".

Once a Simulation File is successfully loaded, the Simulator window changes: additional items appear on the main menu, more buttons on the button bar are activated, the Simulator status is shown on the bottom window bar, and the "Model Components" window appears (Figure 1-13).

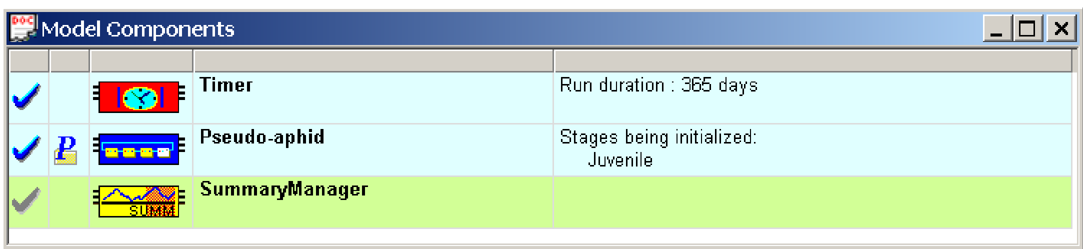


Figure 1-13 The Simulator's "Model Components" window

The **"Model Components"** window indicates that a model is loaded in the Simulator and the window lists the number and type of modules present. A tick must be present before every module symbol before the model can be run. Any modules without such a tick need to be initialized. Figure 1-13 was produced by the Simulator after loading the pseudo-aphid model: it indicates that both the "Timer" and "Lifecycle" modules are correctly constructed and initialised, so that the model is ready to run. Note, however, that the Lifecycle module may require further initialization. If the model is run at it stands, the Total Numbers output for both lifestages will be zero at all times as no individuals have been specified to 'seed' the population. The Parameter icon (P) in front of the Lifecycle module symbol indicates that this module contains user-settable parameters. Note also the special "SummaryManager" module, which is present in every DYMEX model. It is not used in our very simple Pseudo-aphid model and can be ignored, but is important in more complex models. If the "Close" button ✕ at the top right corner is clicked, the model file will be closed and must be re-loaded if further work is required.

1.5.4 Module Initial Settings

Assuming all modules are correctly constructed (i.e., the model loaded without errors and a Model Components window is shown), initial settings and parameter values can be entered or re-set for each module. There are two ways of selecting the initialization dialog boxes: either the menu bar by selecting "Initialization" or from the "Model Components" window. Each method opens the same series of dialog and edit boxes.

Each module icon within the "Model Components" window acts as a button to open either selection box or a dialog box and module settings can then be made.

1.5.4.1 The Timer Module

1. Select the **"Timer"** module icon
2. From the drop-down menu, select the **"Initialize Module"** button
3. Check the **"Simulation Duration"** selection box is set to the defaults shown in Figure 1-14
4. Select **"OK"**.

Since the run will simply start from a "day 1" and proceed for one year (365 days), there is no need to alter any settings in the Timer module.

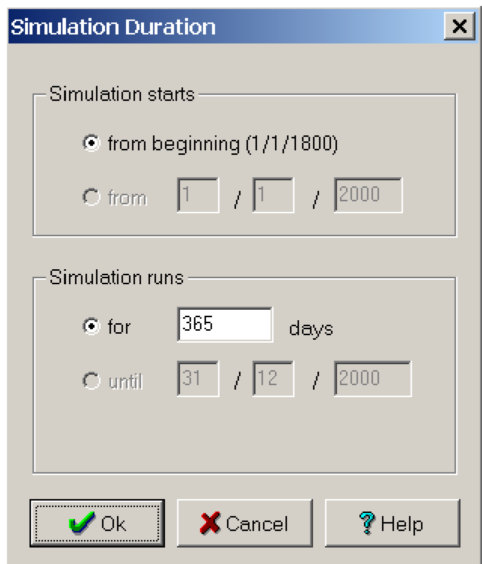


Figure 1-14 The "Simulation Duration" dialog box

1.5.4.2 The Lifecycle Module

1. Select the **"Lifecycle"** module icon
2. From drop-down menu select **"Initialize Module"**

The "Initialize Lifestage Numbers" dialog box (Figure 1-15) allows the user to add individuals to any lifestage to "seed" the simulation. Inspection will show that the Juvenile stage is selected by default but that no initialization settings are present. DYMEX will still run the pseudo-aphid model, however it will not produce pseudo-aphid population results because there are no individuals within the model (This can sometimes be useful in more complex models to test the rest of the system in the absence of a population). However, for the current tutorial, a single 'seed' individual will be added to the Juvenile lifestage.

3. Select the **"New"** button in the **"Initialize Lifestage Numbers"** selection box to open the **"Edit Lifestage Initialization Set"** dialog box (Figure 1-16).

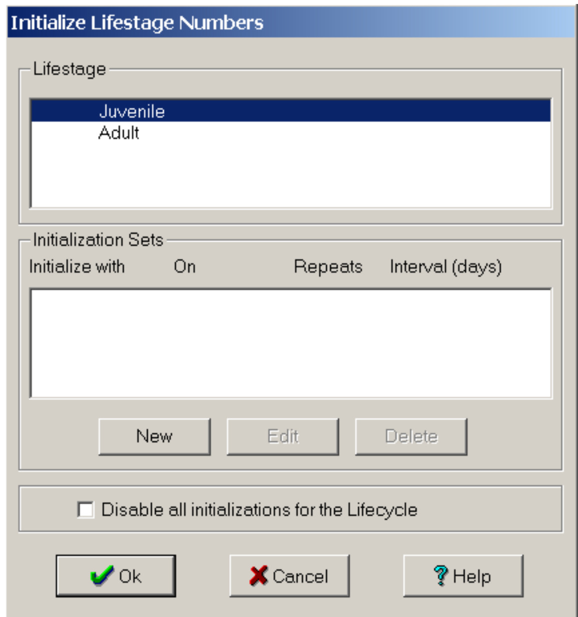


Figure 1-15 The "Initialize Lifestage Numbers" dialog box

4. Select the **"Add Individuals"** text entry area and enter the value 1
5. Select **"OK"**

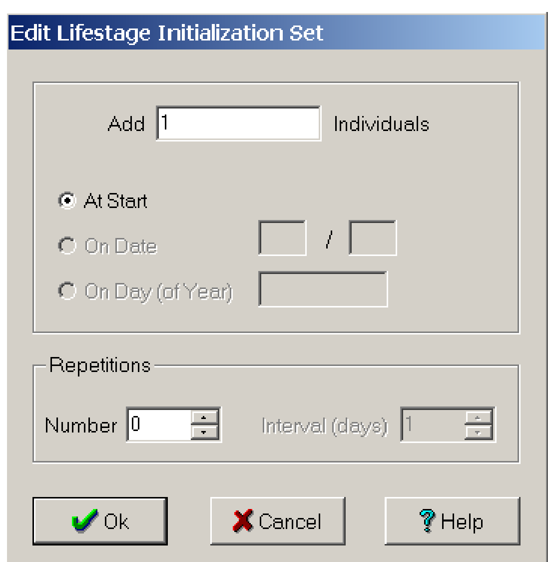


Figure 1-16 The "Edit Lifestage Initialization Set" dialog box

This returns the user to the "Initialize Lifestage Numbers" selection box which will now show that the Juvenile lifestage has been initialized with one individual at the start and there are to be no repeats.

6. Select **"OK"** to return to the main simulator window.

The Lifecycle module will now have an altered message indicating that the juvenile lifestage has been initialized.

The structure of the pseudo-aphid lifecycle can be shown using the "Lifecycle" module icon.

1. Select the **"Lifecycle"** module icon
2. From the drop-down menu select **"Show Lifecycle Diagram"**

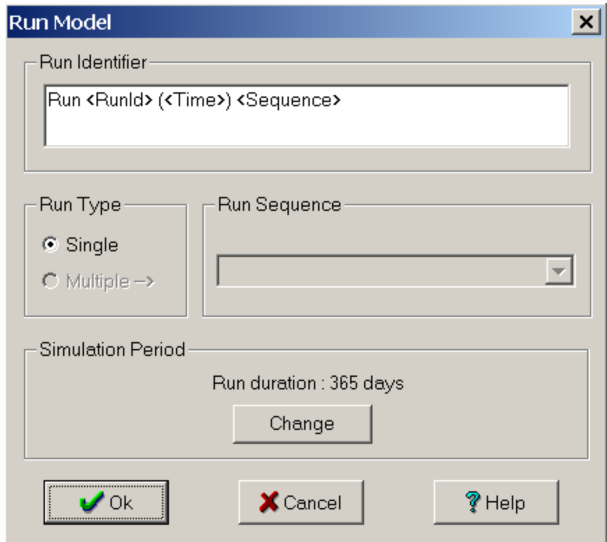
A diagram will appear showing the lifecycle of the pseudo-aphid. Like the "Model Components" window (Figure 1-2*), the lifestages of the lifecycle can be used as button icons in the Simulator to examine the model structure.*

1. Select the **"Juvenile"** lifestage icon
2. Examine and then close the window
3. Select the **"Adult"** lifestage icon
4. Examine and then close the window
5. Close the Lifecycle diagram window.

1.5.5 Running the Model

 DYMEX allows two methods of starting a model run, using either the menu or the button bars. From the menu bar, "Execution" produces a drop-down menu containing "Run", or the "Lightning Flash" button on the button bar produces the same result.

1. Select the "Run" button from the button bar to produce the **"Run Model"** dialog box.



.

Figure 1-17 The "Run Model" dialog box

The "Run Model" dialog box (Figure 1-17) allows the run conditions to be set. Within the dialog box, the "Run Identifier" edit box allows the insertion of a suitable title for the run. The title can contain special phrases (macros) that can automatically insert information about the initialization options used and the start date of the model, but these options will not be used for this version of the pseudo-aphid model. Only a single run will be used for the pseudo-aphid and therefore the "Run Type" and "Run Sequence" edit boxes can be ignored. The "Change" button permits alteration of the duration of the run time. For the pseudo-aphid it has a default value of 365 days.

1. Select **"OK"**

The model will now run and a"Running Model" window (Figure 1-18) will appear briefly to indicate the progress of the Simulator. The dialog shows a "progress" bar indicating the current state of the run. It also shows whether the **Trace** options is turned on and the number of ***Cohorts*** in the simulation at any time (see the ***Simulator User's Guide*** for more information on these).

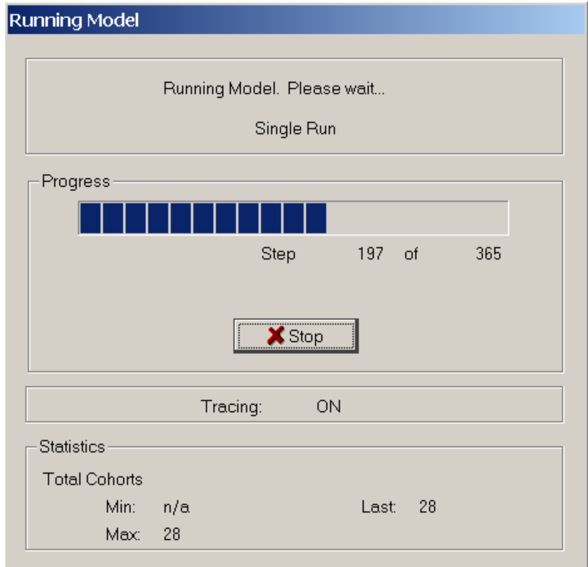


Figure 1-18 The "Running Model" window

On completion of the run, a "Run" window will appear (Figure 1-19) which summarises information on the run. For the pseudo-aphid, this is very limited due to the simplicity of the model, however the "Run" window summary contains more information for more complex models.

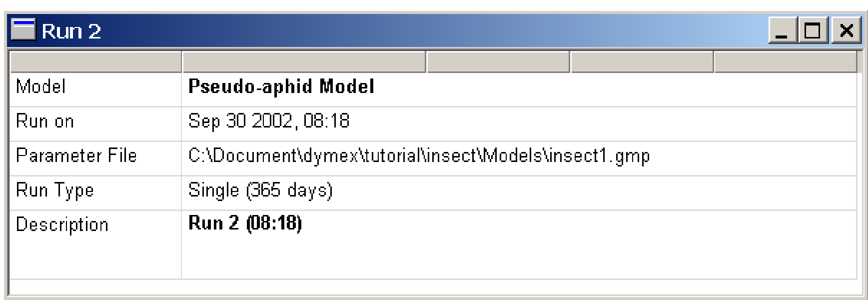


Figure 1-19 The "Run" window

1.6 Producing Model Outputs

On completion of a run, DYMEX outputs may be obtained using either the menu bar or the button bar. If the menu bar is used, select "Results" and then choose from the drop-down menu. The button bar offers the same processes with a single selection. Since a model has now been run, two more buttons on the button bar are activated:

 - Charts output button.

 - Tables output button.

These buttons allow the model output to be presented in either chart or tabular form respectively. They open a series of dialog and edit boxes that permit selection of the variables to be presented and the user can define the format of the presentation.

1.6.1 Creating a Table Display

1. Select button

The **“Table Specification”** selection box is now open (Figure 1-20) and for the pseudo-aphid there are four **“Model Variables”** in the list box. **“Days Since Start”** counts the number of days since the commencement of the run and *for this model*, it is equivalent to **“Step”** which counts the number of timesteps since the simulation started. For the pseudo-aphid model the step is equal to one day, but other models may use weekly timesteps. The remaining variables were selected in the Lifecycle window of the pseudo-aphid model during its construction in the Model Builder. Any combination of the variables can be selected for presentation in a table.

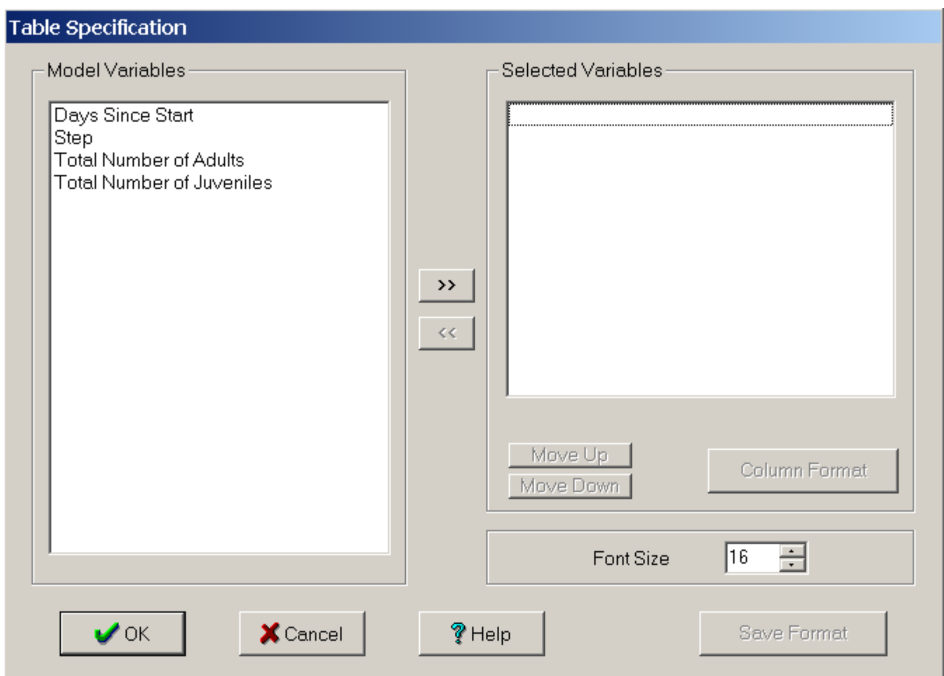


Figure 1-20 The “Table Specification” dialog box

2. From the “Available Variables” edit box, select **“Days Since Start”**
3. Select the **“>>”** button
4. Repeat 1 & 2 above for the remaining variables.

Variables can be removed from the **“Selected Variables”** list box by highlighting them and then clicking on the **“<<”** button. The procedure used above will select all variables for the output table.

DYMEX can change the format any selected variable. In Figure 1-20, the **Format** button is greyed out, but with the selection of any of the selected variables, the Format button becomes active and opens the **"Column Format"** window (Figure 1.20) which allows the selected column's variable to be displayed in any suitable format. Once a table format is fully defined, it can be saved using the **"Save Format"** option in the **"Table Specification"** dialog box (Figure 1-20). The **"Save Format"** option opens a standard dialog box in which the name of the format can be entered for future use.

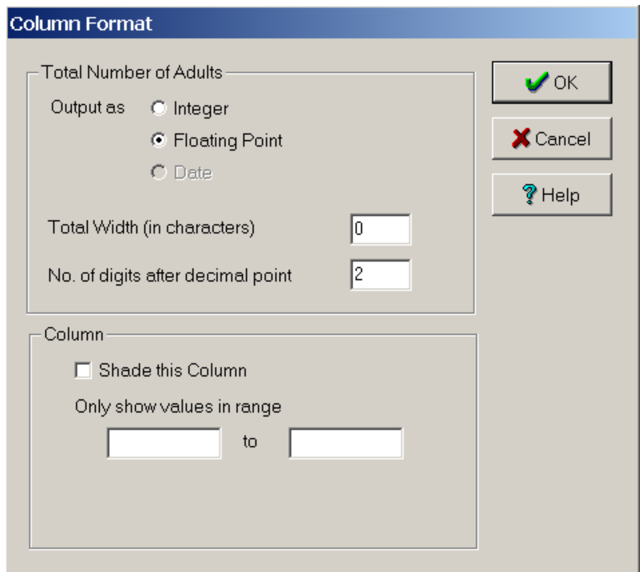


Figure 1-21 The "Column Format" dialog

5. Select **"OK"** to produce the output table (Figure 1-22).

DYMEX constructs condensed headings for each of the columns of the table. The data can be printed in full by selecting the **"Print"** button on the button bar. The **"Print Preview"** button (or menu selection) can be used to examine how the printed table will fit onto the printer pages. If it is too wide to fit on a page, columns can be narrowed by clicking on the line between the column headings (so that a ↔ icon becomes visible) and "dragging" the mouse to change the column width. Alternatively, the font size of the table items can be reduced in the "Table Format" dialog (Figure 1-20). Selected parts of the table can also be printed separately, by first marking the required area followed by **"Print Selection"** from a drop-down menu (see below).

Several options are available from a drop-down menu that is obtained by placing the cursor in a data column and "double-clicking" the left mouse button. These options are: examination of the variable description, changing the column's format, saving of the table data in a separate file, and production of a "Quick Graph".

1. Place the cursor in the **"TotNumofJuv"** (Total Number of Juveniles) column and double click the left mouse button
2. From the drop-down menu, select **"Variable Description"**

3. After inspecting the **"Details of Variable"** list box, select **"OK"**

The **"Details of Variable"** dialog box summarises information about the particular variable of the selected column and the procedure is available for any column. Note that the **Description** field in the dialog may not contain any information if this has not been provided as a comment when the model was built.

The **"Quick Graph"** procedure is a simple method of creating a chart of a single model variable. It automatically uses the "Days since Start" variable for the X-axis.

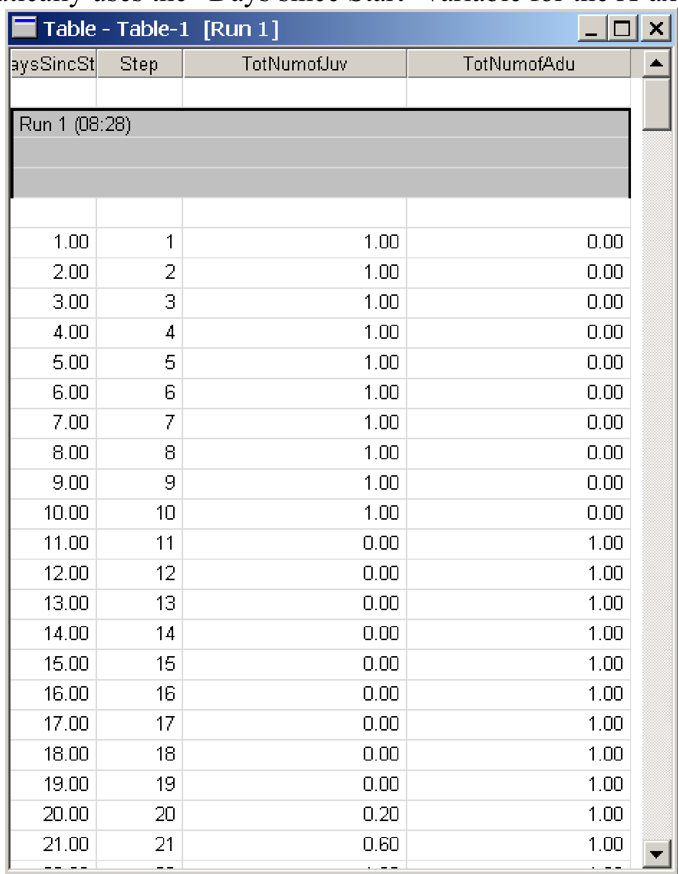


Table - Table-1 [Run 1]

Run 1 (08:28)

aysSincSt	Step	TotNumofJuv	TotNumofAdu
1.00	1	1.00	0.00
2.00	2	1.00	0.00
3.00	3	1.00	0.00
4.00	4	1.00	0.00
5.00	5	1.00	0.00
6.00	6	1.00	0.00
7.00	7	1.00	0.00
8.00	8	1.00	0.00
9.00	9	1.00	0.00
10.00	10	1.00	0.00
11.00	11	0.00	1.00
12.00	12	0.00	1.00
13.00	13	0.00	1.00
14.00	14	0.00	1.00
15.00	15	0.00	1.00
16.00	16	0.00	1.00
17.00	17	0.00	1.00
18.00	18	0.00	1.00
19.00	19	0.00	1.00
20.00	20	0.20	1.00
21.00	21	0.60	1.00

Figure 1-22 Table display for the Pseudo-aphid run (365 day run)

1. Select the **"Total Number of Juveniles"** (**'TotNumOfJuv'**) column by double clicking
2. From the drop-down menu, select **"Quick Graph"**

The resulting graph shows the population of Juveniles against time. Due to the absence of any limiting factors, the population is increasing more or less exponentially (Figure 1-23).

A similar procedure can be used for graphing the other columns.

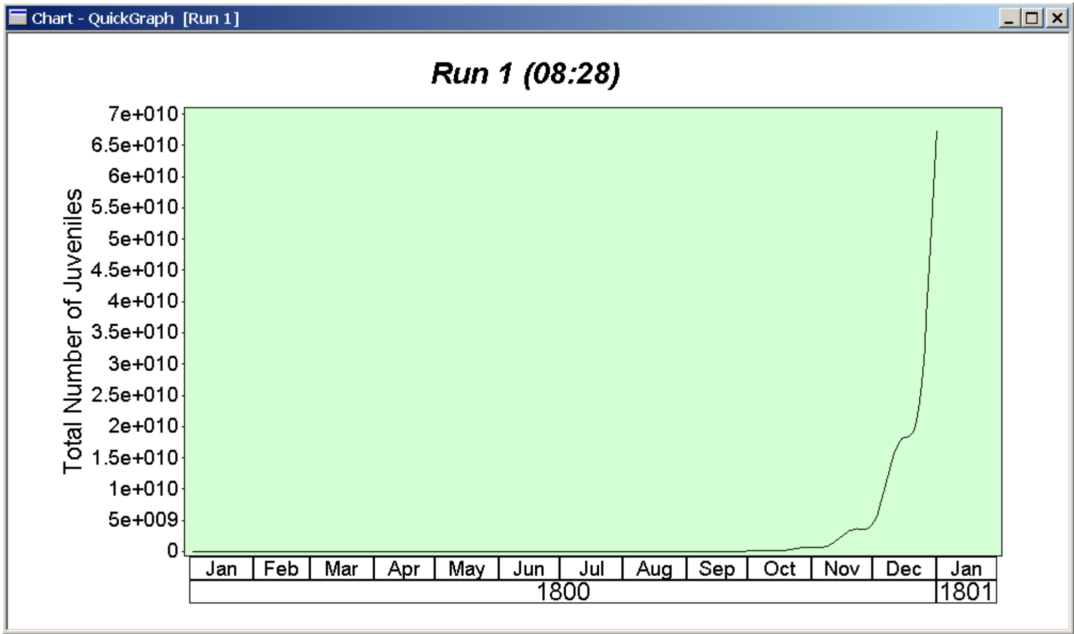


Figure 1-23 "Quick graph" of Total Number of Juveniles against time

While in the table mode, an "area selection" mode is also available. If the cursor is clicked on a table cell, it can now be used to mark/select an area of the table by the standard Windows technique of holding the left mouse button down while dragging the cursor across the required area. When this is done the area is highlighted and the right hand mouse button can then be used to open a drop down menu while the cursor remains in the highlighted area.

The drop-down menu contains the following options, which are explained here for use as required:

Copy Selection	Places a copy of the selected area into the clipboard. The copy can then be accessed by the usual "Paste" command of the wordprocessor, spreadsheet or other program in use.
Page Setup	Allows setting of some properties for the printed table.
Print Selection	Sends the highlighted area to the printer. The dialog box that appears refers **strictly** to the highlighted area.
Print Table	Prints the whole table.

This completes the introduction to the table displays and the user should now close the table and the Run by using the standard Windows procedure of clicking on the top right Window "X"-button of the Run window. This will close the "Run" window and any table and graph windows created from that run. Of course, graph or table windows can be closed individually by the same procedure. All that should remain open at this point should be the "Model Components" window with its two icons.

1.6.2 Creating a Chart of the Results

Since this simple model has no population regulating mechanisms built in (this will be added in later tutorials), populations are increasing geometrically. To get a better picture of what is happening to the population, it is useful to run the model for a shorter period.
Starting with the **"Model Components"** window:

1. Click once on the **"Timer"** icon and then from the drop down menu select **"Initialize Module"**
2. In the **"Simulation Duration"** selection box (see Figure 1-14) change 365 days to 65 days
3. Select **"OK"**
4. Select **"Run"**.
5. When the run has completed and the Run Window is being displayed, select the button to give the **"Chart Specification"** dialog box (Figure 1-24)

The **"Chart Specification"** dialog determines which variables will be used for chart formation, the format of the charts including their axis labels and other chart options such as whether the output values will be plotted on natural or logarithmic scales and whether a legend is to be shown. An option to save frequently used formats is also provided. With two output variables, either a single combined chart output panel could be used, or two separate output panels might be employed.

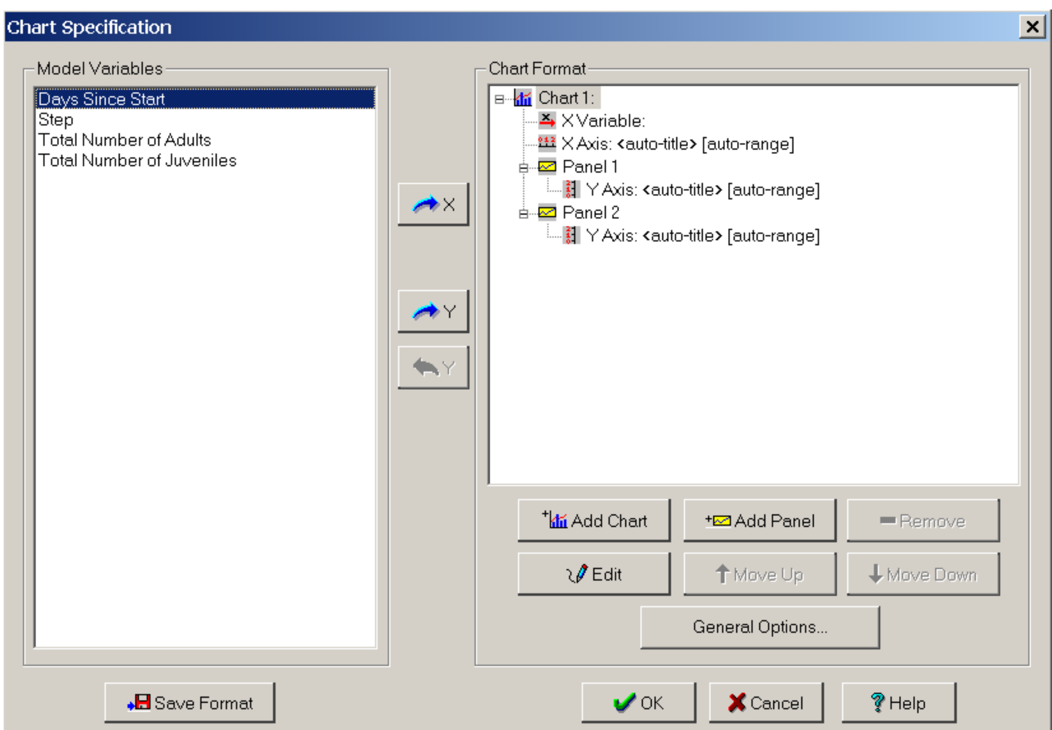
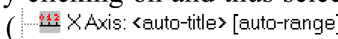


Figure 1-24 The "Chart Specification" dialog box

For this initial display, a single chart with one panel containing both variables on a common X-axis will be used.

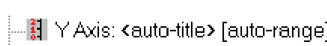
1. From the **"Model Variables"** list, select the **"Step"** variable and click on the "**X-variable transfer button"** 

The user could now alter various X-axis properties by clicking on and thus selecting the line representing the X-axis in the **Chart Format** box (

), and then clicking the **"Edit"** button. However, we will leave the X-axis in its default state for the moment. The Model Variable **"Total Number of Adults"** will already be selected by default and in order to insert it into the graph, the user needs only click on the transfer button.

2. From the **"Model Variables"** list, select the **"Total Number of Adults"** variable
3. Select the **"Y-variable transfer button"** 

Once this button has been selected, a "Series Format" dialog will appear, which allows the series representing the variable to be specified. Available options allow for a line, point or bar graph and a selection of colours. For this graph, the default choices are adequate and the dialog box can be closed by clicking on **"OK"**. The **"Total Number of Adults"** variable will appear in the Chart Format box as Series 1 of Panel 1. In addition, a "reverse" transfer button directly underneath the "Y-variable transfer button" will also now activate and allow errors to be corrected if the wrong variable is transferred. For this tutorial, both the Juvenile and Adult populations will be plotted on the same chart panel, to allow detailed comparison. This is achieved by keeping the "Panel 1" selected while adding the second variable. If instead, "Panel 2" is selected in the "Chart Format" box before the second variable is transferred to the chart, then the Juvenile and Adult graphs will be drawn in separate panels that share the same x-axis.

4. Select **"Total Number of Juveniles"** from the **"Model Variables"** list box by clicking on it
5. Select the **"Y-Variable transfer button"**
6. Click on "OK" in the "Series Format" dialog to accept the defaults and move **"Total Number of Juveniles"** into the **"Chart Format"** box as *Series 2* of *Panel 1*.
7. Click on the Panel 2 line in the **"Chart Format"** box and click on the **"Remove"** button.
8. Select the Y-axis line (

) in the **"Chart Format"** box and then select the **"Edit"** button
9. Change the **Title** to non-automatic by clicking on the small button next to the edit box. This will make the edit box active, and the Y-Axis title can be changed to "**Aphid Population"**
10. Select **"OK"** to return to the Chart Specification dialog
11. Select the **"Save Format"** button and save the display under a suitable name such as "Aphid Population").
12. Select **"OK"** and the chart will now display. It should be similar to Figure 1-25.

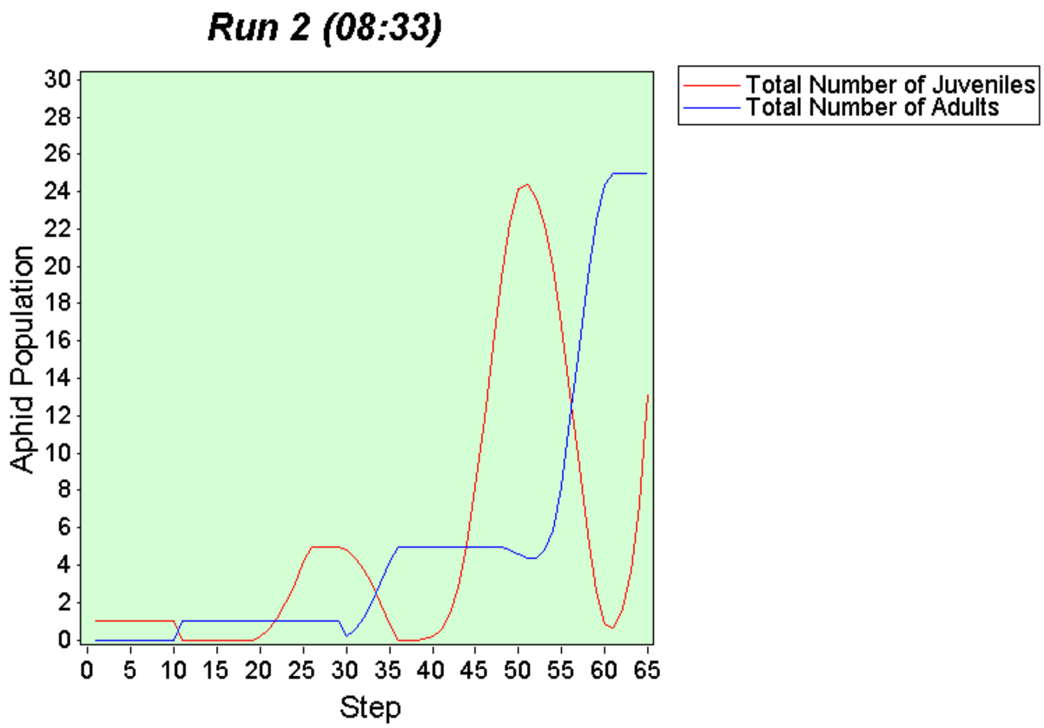


Figure 1-25 Pseudo-aphid Population Chart

Note how initially the population consists of the single individual used to seed the simulation. At the 10th timestep, this Juvenile becomes an adult. The adult produces new Juveniles from the 20th timestep onwards. The adult dies at timestep 30, while second generation adults start entering the population at that time.

The two variables can be plotted in separate panels by placing the second variable into "Panel 2" (Figure 1-26). If completely separate graphs are required where a different X variable is needed for each Y-axis, then the "Add Chart" button can be used to add a second chart to the display.

All charts can be printed either by selecting the button bar's Print button or **"File"** from the main menu bar followed by **"Print"** or **"Print Preview"** from the drop down menu.

Figure 1-26 Pseudo-aphid Population chart using 2 panels

Tutorial 1 - Summary

Timer

Inputs: none
Output: Days since Start

Settings: Timestep 1 day

Lifecycle

Juvenile

Transfer to adult function (Step)
Driving variable: Chronological Age
Transfer threshold: 10 days
Prop.juveniles transferring: 1

Output:
Total number

Adults

Continuous mortality function (step)
Driving variable: Chronological age
Threshold: 20 days
Prop. adults dying: 1

Reproduction
Fecundity:
Constant: 5 juveniles per adult
Progeny Production (Linear above Threshold)
Driving variable: Chronological Age
Threshold: 8 days
Slope: 0.2

Output:
Total number

2 Temperature-dependent Development

2.1 Chronological and Physiological Age

In the first tutorial, the pseudo-aphid model used chronological age to determine the sequence of development and reproduction. Chronological age is unsatisfactory as the only controlling influence on an insect's lifecycle because the insect's physiological development (and by implication its reproduction and mortality) is often independent of chronological age and is generally controlled to a large extent by temperature. This tutorial presents an example of how DYMEX can model the pseudo-aphid's development from juvenile to adult when the lifecycle development is based on physiological age and the rate of that development becomes temperature dependent. Since physiological age now enters all further tutorial models, it is defined below:

> Physiological age measures the state of development of an individual with its units generally stated as a proportion (or percentage) of completed development. As an example, the birth of the pseudo-aphid could be scaled to 0 and its transition to adult scaled as 1. Since insect development is generally temperature dependent, accumulation of physiological age is usually non-uniform.

2.2 Changing the Model

2.2.1 The "Pseudo-aphid" and Temperature

Since the pseudo-aphid's lifecycle is 'well known from published papers', the effects of temperature on development are available and are presented in table form (Table 2-1).

Temperature (°C)	*No. of days to develop to adult*
10	No development
15	20
20	12
25	8
30	5

Table 2-1 Temperature effects on juvenile to adult development in the Pseudo-aphid

The model used in the first tutorial remains essentially intact, but the transition from juvenile to adult (currently determined by Chronological Age) now becomes dependent upon Physiological Age, which in turn is dependent upon temperature. All juveniles still become adults when they reach the required Physiological Age. To preserve simplicity of the model, the Pseudo-aphids are maintained under temperature controlled incubator conditions.

The results of Table 2-1 can be amended to display rate of development per day. This is done by calculating the reciprocal of the number of days taken to develop to adult (which assumes that the value "1' represents the physiological age of an adult). For example, suppose an organism takes 50 days to develop from egg to adult; its rate of development per day would therefore be 0.02 (ie. 0.02 * 50 = 1). Table 2.2 shows the results for the pseudo-aphid.

Temperature (°C)	*No. of days to develop to adult*	*Rate of development per day*
10	No development	0
15	20	0.05
20	12	0.08
25	8	0.125
30	5	0.2

Table 2-2 Temperature effects on the rate of Juvenile to Adult development in the Pseudo-aphid

These results can now be transposed into graphical format (Figure 2-1).

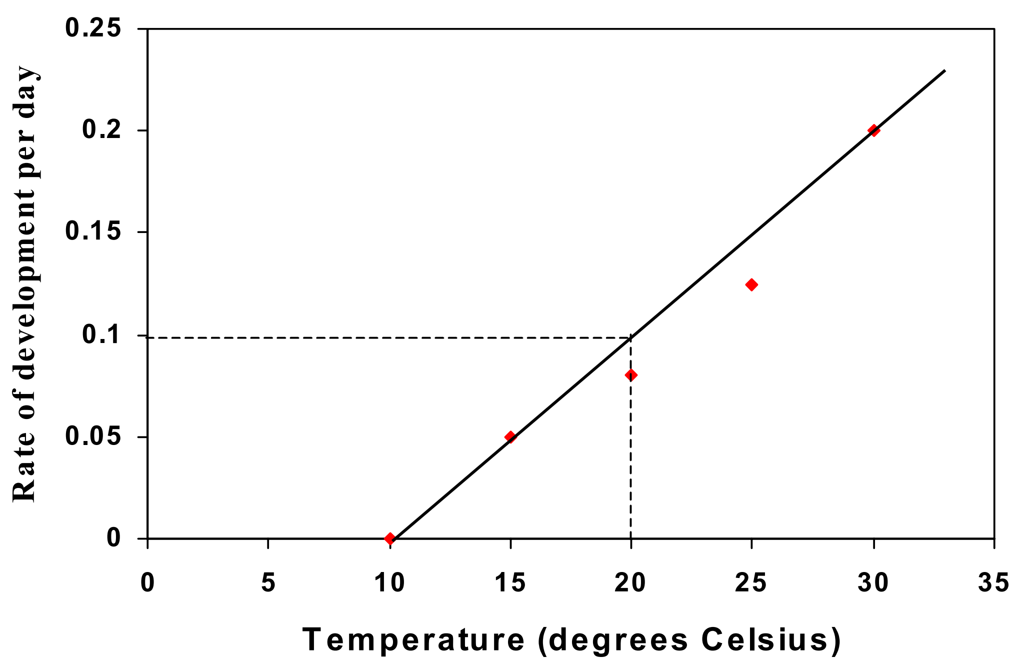


Figure 2-1 Rate of development of juvenile Pseudo-aphids

An inspection of Figure 2-1 shows that the development threshold is 10°C, while the slope of the graph is quite close to 0.01. These values will be used for this tutorial.

2.2.1 Making the changes in the Builder

Start the DYMEX **Builder** and open the Pseudo-aphid model. The "Model Components" window will appear and display two modules, **"Timer"** and **"Lifecycle"**. An additional module will be added to control temperature and these temperatures will be user-defined and constant during any particular simulation run (set by the incubator). DYMEX provides a type of module ("QueryUser") suited for this situation, in which a constant value can be provided by the user.

1. Select **"Model"**
2. From the drop-down menu, select **"Add Module"**
3. From the **"Create Module of Type ?"** list box select **"QueryUser"**. Make sure that the **"Standard"** button at the top left of the dialog is selected
4. Select **"OK"** to obtain the **"Query User"** module window
5. In the **"Module Name"** text entry box delete **"QueryUser1"** and enter **"Temperature"**
6. Select the **"Outputs"** button to obtain the **"Output Variables"** dialog box.

The range of temperatures under which the pseudo-aphid can be incubated will now set. The values entered into DYMEX will be strictly a decision of the user, but for this tutorial, a suitable range might be 5-35° C with 20° C selected as the default value.

7. Select the **"New"** button and the name **"Temperature Variable1"** will appear in the **"Module Output Variables"** list box
8. Click on the **"Select"** button and **"+>"** will appear beside the variable name to indicate it is selected as an output variable
9. Select **"Rename"** and in the **"Rename Variable"** edit box, change the variable name to **"Incubator Temperature"**
10. Select and set **"Minimum allowed value"** to 5, **"Maximum allowed value"** to 35 and **"Default value"** to 20
11. A description of this variable can be entered into the **"Description"** box. Such a description might be "*The constant temperature at which the pseudo-aphids are being held*"
12. A short name for the variable (for example, "*Temperature*") can be provided in the **"Mnemonic"** panel.
13. Select **"OK"** as required to return to the **"Model"** window.
14. Save the model.

The "Model Components" window now shows a new module called **"Temperature"**. If the **"Temperature"** module tree is opened, a branch labelled **"Outputs"** will appear, which in turn has a sub-branch labelled **"Incubator Temperature"**.

The pseudo-aphid model now has a range of temperatures within which it can be run, however the juvenile lifecycle module must be changed to allow those temperatures to influence the lifecycle appropriately. “Chronological Age” must be altered to “Physiological Age” and the functions, variables and parameters modified to reflect the data of Table 2-2. The adult lifestage is not subject to any of these changes which all occur within the “Development” and “Stage transfer” processes of the juvenile lifestage.

The “Stage Transfer” process already contains a step function based on chronological age. This function will be retained as all juvenile pseudo-aphids still transfer to the adult lifestage the moment they reach adult Physiological Age. The required modifications are therefore: first, to alter Chronological Age to Physiological Age and second, to change the “step width” parameter (which represents the age of a juvenile at completion of development) to 1.

1. Double click on the **“Pseudoaphid”** module and obtain the **“Lifecycle”** window;
2. Select the **“Juvenile”** lifestage
3. Select the **“Stage Transfer”** button to obtain the **“Juvenile Transfer”** dialog

The “Process Components” list box will already have “Juvenile to Adult Transfer Function” highlighted and selected.

4. Select **“Edit Component”** to obtain the **“Function”** edit window
5. Select the **“Independent Variable”** scroll button
6. From the scroll list, select **“Physiological Age”**
7. In the **“Parameters”** panel, select the first parameter and click on the **“Edit Parameter”** button to obtain the **“Set Parameter Properties”** edit window
8. Set the default and both limits to the value 1
9. Select **“OK”** as required to return to the **“Lifecycle”** window.

The Juvenile lifestage will now be modified to reflect the data of Table 2-2. This is done by adding a “Development” process (which affects the Physiological Age). A “Linear above Threshold” function will be chosen, and its threshold and slope set at 10° C and 0.01, in accordance with Figure 2-1.

10. Select the **“Development”** button to obtain the **“Development (Juvenile)”** selection window
11. Name the process “***Development***”
12. Select the **“Function”** button (to add a Function component) and obtain the **“Function”** window
13. From the scroll box, select **“Linear above Threshold”** as the **Function Template**
14. Set the **“Independent Variable”** to **“Incubator Temperature”**

15. Select **“Change”** in the **“Name”** edit box
16. In the **“Process Factor Name”** text edit box enter a suitable name (e.g. **“Temperature-dependent Development”**)
17. Select **“OK”**
18. In the **“Parameters”** panel select the **“*p1: Threshold*”** parameter and click on **“Edit Parameter”** to open the **“Set Parameter Properties”** dialog
19. Set the threshold default value to 10 and the lower and upper limits to 5 and 35 respectively
20. Rename the parameter **“Threshold for Physiological Development”**
21. Select **“OK”** to return to the **“Function”** dialog
22. In the **“Parameters”** panel select the **“*p2: Slope*”** parameter and click on **“Edit Parameter”** to open the **“Set Parameter Properties”** dialog
23. Set the upper and lower limits and default to 0.01
24. Change the **“User Name”** to **“Rate of Physiological Development”**
25. Select **“OK”** as required to close all dialogs
26. Save the file.

There will now be a tick on the “Development” button of the Juvenile lifestage.

2.2.2 Module Ordering in the Model

When the DYMEX Simulator loads a model, it displays the model structure by means of a series of module icons. The present pseudo-aphid model has only three modules and until now, there were few occasions where any problems with module sequence could arise. With increasing model complexity, the number of modules present rises rapidly and it is important to be aware that when the model is run in the Simulator, **the program processes the modules in the order in which they appear in the Model Components window**.

While this may not seem significant, it has important implications for model processing within the Simulator and if the order of two modules is altered, quite different outputs can be produced. DYMEX always sets the **“Timer”** module as the first module to be included in a new model, it is also the first to be processed during the run of any model and its position cannot be changed by the user. The relative positions of all other modules can be changed. By default, the processing sequence of the model is set to the order in which the modules were added to the model, however the module processing sequence can be changed by using the **“Sort Order”** facility which can only be accessed from within the Model Builder. Each module (except the Timer) is given a number (its “Sort Order”), and modules are ordered in increasing Sort Order values. The actual Sort Order value of a module is shown in the “Model Components” window in the Builder, in square brackets after the module name.

For all modules except Lifecycle the Sort Order is changed from the module's main dialog. The Lifecycle module sort order can be changed by opening the Lifecycle window, selecting **"Lifecycle"** from the main menu bar and then **"Properties"** from the drop down menu to give the Lifecycle Properties dialog box.

For the Pseudo-aphid model, since the "Incubator Temperature" is used as input to the lifecycle, the module that produces that variable should appear before the lifecycle module. In the current situation, it does not actually matter which module comes first, as the QueryUser module is initialized with the required value before a run and then its output value does not change any more during a run. However, it is good practice to consider the order in which operations must be performed in the model as part of the model design. To move the "Temperature" model to its desired position, merely replace its current Sort Order number (20) by one that is smaller than the Lifecycle module's (10) – 5 would be suitable.

From this point onwards, all the tutorials will assume that the sort order has been amended if necessary so that the Lifecycle module has been placed last in the list of modules.

2.3 Running the Improved Model

2.3.1 Loading the Model

Loading and running the temperature modified Pseudo-aphid file is identical to the procedures already described. An important difference with this new model is that the incubator temperature can be altered to examine the resulting pseudo-aphid population increases.

1. Start the DYMEX Simulator
2. Open the pseudo-aphid model, either by selecting the name of the Simulation File from the "recently-used files" list (just above the **"Exit"** item in the **File** menu, or by clicking on **"Open"** in the **File** menu.

If the latter method is used, the **"Open Simulation"** dialog will appear. This is very similar to the **"New Simulation"** dialog (Figure 1-12), but instead of supplying a name for a new Simulation file, the name of an existing Simulation File is selected from the **"Simulation Files"** list after a model is selected from the **"Models"** list.

While the pseudo-aphid file is loading, DYMEX will recognise that the parameters and model structures differ from the previously run model whose settings are stored under the same name. Recall that the DYMEX Simulator creates a Parameter File ("gmp"-file) when a model is first loaded, and in subsequent loads of the same model attempts to use that Parameter File to set its parameter values. Since the model has now changed, DYMEX will report that the original Parameter File is no longer correct for the new model and the program requests permission to update the parameter file with the model changes (using the default values specified in the Builder). When this request occurs, select **"Yes"** from the query box; the Simulator will proceed to complete further operations on the model and finally display the loaded model in the "Model Components" window (Figure 2-2).

2.3.2 Initializing and Running the Model

A new "Temperature" module will be present in the "Model Components" window (Figure 2-2). Although it is not absolutely necessary for this model at this early stage, it is worth checking that all the modules have been initialized and set to the required values. This can be done either by using the menu bar, or by using the three module buttons of the "Model Components" window to open their "Initialize Module" windows. If the user ever omits to initialize a module that requires initialization, DYMEX will report the problem to the user and will not allow a simulation run to start until the module has been initialized.

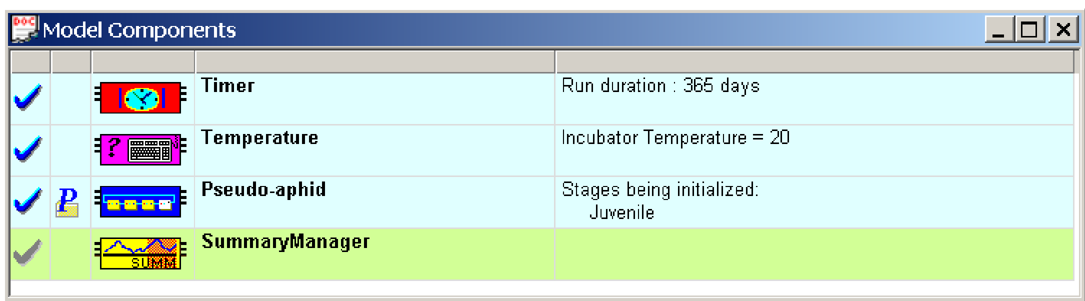


Figure 2-2 The Model Components window

1. Click on the "**Timer"** module to obtain the drop-down menu
2. Using the **"Initialize Module"** option of the drop down menu, ensure that the **"Timer"** module is to a run length of 365 days and that the simulation starts "**from beginning"** and then exit back to the "**Module Components"** window
3. Click on the "**Temperature"** icon
4. Select "**Initialize Module"** from the drop down menu
5. Check that the "**Incubator Temperature"** is set to a default of 20° C (this can also be verified from the **"Model Components"** window
6. Select "**OK"** as necessary and return to the "**Model Components"** window
7. Select the "**Lifecycle"** module icon
8. From the drop down menu, select "**Initialize Module"**
9. Select the "**Lifestage"** scroll button
10. Ensure that the "**Juvenile"** lifestage is initialized with 1 individual at the start
11. Return to the **"Model Components"** window

12. Run the model for a period of 365 days (Select).

2.3.3 Producing Charts of Results

As in Tutorial 1, tables of any selected variables can be produced, examined and "Quick Graphs" produced from any variable in the table. It is left to the reader to explore those aspects, only the graphical displays will be dealt with here.

1. Select
2. Select "**Step**" from the **"Model Variables"** list and click on to set the X-axis variable
3. Select "**Total Number of Adults"** and click on to set the "Y-axis" variable. Click on **"OK"** when the **"Series Format"** dialog appears to select the default settings
4. Click on the **"Panel 2"** line in the **"Chart Format"** box, and then select the **"Remove"** button (since only a single panel is needed for this chart)
5. Click on the **"General Options"** button to open the **"General Chart Options"** window
6. Unselect the **"Show"** item in the **"Legend"** panel (by clicking on it), as we do not want a legend for this single-variable chart
7. Select "**OK"** to return to the **"Chart Specification"** dialog
8. We will want to use this chart format again, so click on the **"Save Format"** button to save the format. In the **"Save chart format"** dialog, type in ***"Adult Population"*** as the name of the chart, and select **"OK"**
9. Select **"OK"** again to draw the chart

The chart that is produced (Figure 2-3) shows the by now familiar, nearly exponential curve. Even though Juvenile development has been made responsive to temperature, this temperature is a constant and the model has stayed essentially the same. Close all windows, including the **"Run"** window, and return to the "**Model Components"** window.

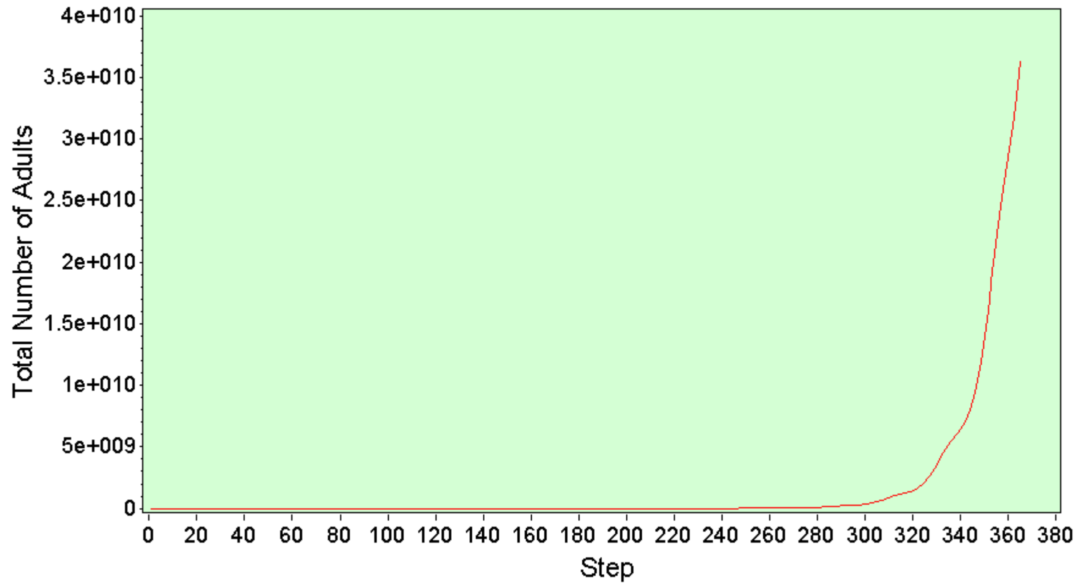


Figure 2-3 Adult population at a temperature of 20°C

1. Re-open the **"Initialize Module"** dialog for the "**Temperature"** module
2. Re-set the current value (20° C) to 30° C using either the slider or by directly entering the value into the edit box
3. Close the dialog box by clicking **"Ok"**, and re-run the model
4. Create a chart of the Adult population. Note that the previously created chart format ("***Adult Population***") now appears as a selection on a drop-down menu when the Chart button ( ) is clicked, thus saving all the work of setting up the format

The chart produced for 30° C (not shown here) is almost perfectly exponential as the pseudo-aphid population virtually "explodes" under these "ideal" conditions, where Juveniles develop to Adults in only 5 days.

1. Close all chart windows **including the "Run" window** and return to the **"Model Components"** window
2. Reset the Incubator Temperature to 12° C
3. Re-run the model
4. Create a chart of the Adult population as before.

The 12° C model's chart (Figure 2-4) is very different as the low temperature prevents a steady adult population increase. Because juveniles take a much longer time to develop, there are periods when all the previous generation's adults have died before the new generation of juveniles become adults. Thus the graph shows a distinct separation of generations. The increase in the population is still exponential, but takes place over a much greater period of time.

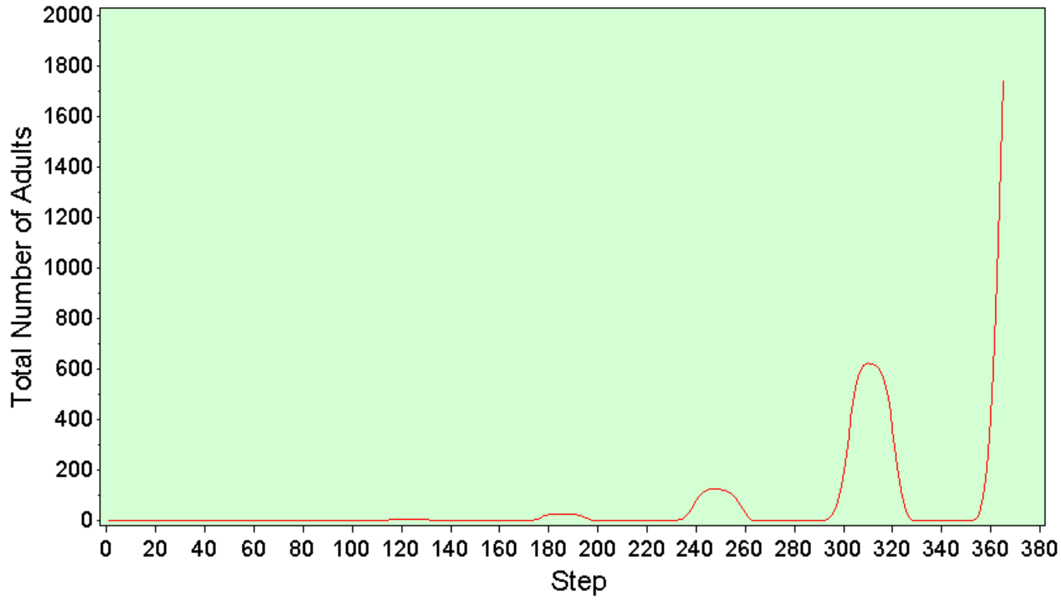


Figure 2-4 Adult population at a temperature of 12°C

Setting the "Incubator Temperature" to 10° C or less will cause the population to effectively die out, as the single Juvenile used to initialize the population will never develop to the Adult stage.

Note that in the graph shown in Figure 2-3, where the population is increasing rapidly, the numbers at the end of the year are so large that any information from the first 200 days of the graph are lost. This can happen commonly, even in more realistic models than the Pseudo-aphid one, where populations fluctuations are large (several orders of magnitude between the smallest and largest). These populations may be better displayed on a chart that uses a "logarithmic" scale for the population axis. The procedure for creating such a chart is the same as that detailed for the population charts in Tutorial 1, with the simple addition of selecting the Y-axis line (Y Axis: <auto-title> [auto-range]) in the "Chart Format" box and then clicking on "Edit" to open the **"Tick-axis Format"** dialog. When the **"Logarithmic"** checkbox (in the "Scale" panel) is then selected, the chart will be drawn with that axis using a logarithmic scale. Note that logarithmic axes cannot be set for a "Quick Graph". A logarithmic axis equivalent of Figure 2-3 is shown in Figure 2-5.

We have changed the Simulation File by defining a new Chart Format (*"Adult Population"*). When closing the Component Window, the Simulator will now ask whether the changed Simulation File should be saved. It is important to answer "Yes" to this request – if the Simulation File is not saved at this point, the newly created Chart Format will not be retained.

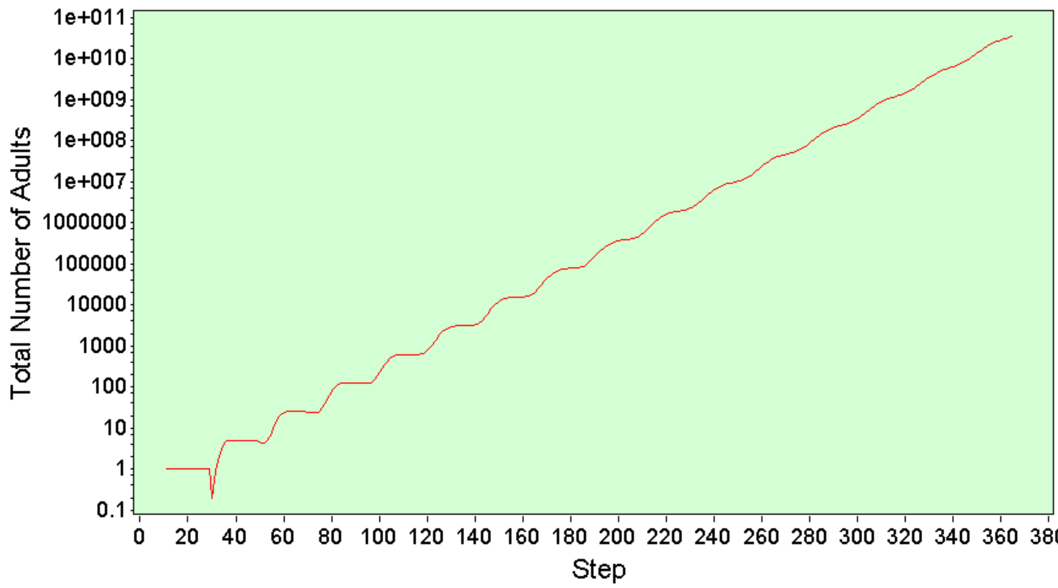


Figure 2-5 Logarithmic chart of the adult population at 20°C

Tutorial 2 - Summary

Timer

See ***Tutorial 1***

QueryUser (Temperature)

Lifecycle

Juvenile

Transfer to adult function (Step)

Driving variable: Physiological Age

Transfer threshold: 1

Prop.juveniles transferring: 1

Juvenile development function (Linear above Threshold)

Driving Variable: Incubator Temperature

Threshold: 10

Slope: 0.01

Output:

Total number

Adults

See ***Tutorial 1***

3 Changing to Field Temperatures

The previous tutorial introduced the concept of "Physiological Age" and the effects of temperature on the development of the Pseudo-aphid. If a model is to simulate field conditions, it must use field temperatures. Tutorial 3 shows how DYMEX is able to utilize meteorological data for different geographical locations to simulate populations of the insect at those locations. The tutorial also shows how an "Expression" module can be used to combine selected variables mathematically to obtain a new variable.

3.1 Modifying the Model

3.1.1 Adding Temperature from a data file

In Tutorial 2, the Pseudo-aphid model used a "QueryUser" module to obtain a constant temperature value for each run, resembling the fixed temperatures obtainable from an incubator. Since the "QueryUser" module is capable of providing only a single, user-defined value per simulation run, it must be replaced by a module capable of transferring field temperatures (which vary daily) into the simulation. In addition, the pseudo-aphid model should now operate with calendar dates since meteorological data is date referenced. This is achieved by selecting another "Timer" output variable (the "Simulation Date") for use within the model.

1. Open the DYMEX Builder program
2. Load the pseudo-aphid file
3. Open the **"Timer"** module by double clicking the text
4. Select the **"Outputs"** button
5. From the **"Module Output Variables"** scroll list, select **"Simulation Date"** so that it is highlighted
6. Click on the **"Select"** button (the **"+>"** symbol will appear) to add that variable to the module's outputs
7. Select **"OK"** as necessary to return to the **"Model"** window
8. Using the menu bar, delete the **"Temperature"** module by selecting **"Model"** and then **"Delete Module"** from the drop-down menu

Once the "**Delete Which Module ?"** selection box is open, highlight the "**Temperature"** module and then follow the delete options that are presented by the Model Builder program. Once the "**Temperature"** module is deleted, the module disappears from the **"Model Components"** window. Note that the red tick in front of the "**Lifecycle"** module also disappears. This is because the driving variable for the Juvenile development process has been removed by the removal of the **"Temperature"** module. The next procedure adds the meteorological database module (although it does **not** inform DYMEX how to read the information from a given file).

9. From the menu bar select **“Model”**
10. From the drop-down menu select **“Add Module”**
11. Making sure that the **“Standard”** button is selected, select “**MetBase”** from the list box
12. Select **“Ok”**

The user should re-name the Metbase module using the the text edit box at the top of the Metbase module window using a name such as **“Meteorological Data”** to indicate its function. The Module window contains a series of buttons, with a description of the function of each button displayed beside it. At this point, the “Sort Order” should also be changed to, say, 5 to move the module to near the beginning of the list of modules.

13. Select the “**Inputs”** button to produce the **“Inputs”** link window (Figure 3-1)

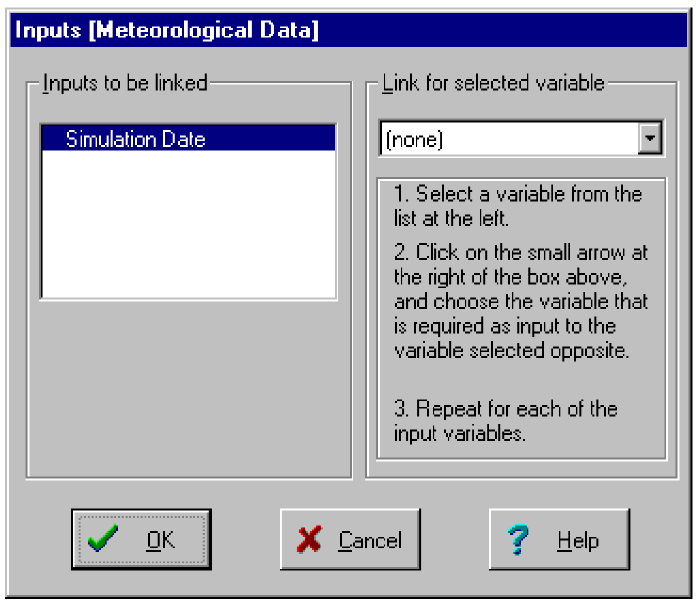


Figure 3-1 The “Meteorological Data” module “Inputs” link dialog

The “**Inputs to be linked”** list box displays the Metbase module’s single input (“Simulation Date”). On the right hand side of the dialog is the “**Link for selected variable”** scroll box - it will be currently displaying a default of “**none”**. This means that at the moment the module’s input is not set to receive information from another module.

14. Using the “**Link for selected variable”** scroll box, select “**Simulation Date”**

Both boxes will now display “**Simulation Date”** in highlighted form and the “**Inputs to be linked”** box will display **“->“** in front of the text “**Simulation Date”** to indicate that the input has been linked to another module’s output.

Understanding the linking of module outputs to module inputs is crucial to understanding DYMEX models – the user is referred to Section 2 of the **Builder** User's Guide for a more detailed explanation.

15. Select **"OK"** to return to the module window
16. Select the "**Outputs"** button to produce the **"Outputs"** dialog
17. With **"Minimum Temperature"** highlighted, click on the **"Select"** button

Once "**Minimum Temperature"** is highlighted and selected, previously greyed-out areas of the dialog become available and the user is able to set range limits on the data that will be read into the model. Since temperatures will be read from a data file, the user may feel that setting their minimum and maximum values is irrelevant. However, such settings can be useful as a check on incorrect or unusual values of data. If a value being read into the model falls outside the set range it will be reported as a possible error. When setting the range, consider whether the model is to be used for other locations where the temperature range may vary from the original location. There is little point in re-naming the temperature variables in this model.

18. Set **"Minimum allowed value"** to -10
19. Set **"Maximum allowed value"** to 25
20. Set the **"Mnemonic"** to ***"Tmin"***
21. Highlight **"Maximum Temperature"**
22. Click on **"Select"** button
23. Set the minimum to 10 and the maximum to 45 (both temperatures will now have "**+>**" in front to indicate they are selected for output)
24. Set the **"Mnemonic"** to ***"Tmax"***
25. Exit to the **"Model"** window
26. Save the model.

3.1.2 Using an "Expression" module to average temperatures

To provide the Simulator with an average daily temperature, the minimum and maximum temperatures are combined using a DYMEX module that can calculate mathematical averages, the "Expression" module.

1. Using the menu bar, commence the procedure to add a module
2. From the "**Create Module of Type?"** dialog's list box, select the module **"Expression"**
3. Re-name the module to **"Average Daily Temperature"**
4. Select the "**Inputs"** button to obtain the **"Inputs"** dialog (Figure 3-2)

When the "Expression" module's "**Inputs"** dialog is first opened, there are clearly **no** inputs present in the "**Inputs to be linked"** list box. The **"Expression"** module provides several mathematical operations and the exact number of inputs required will depend on just what the user wants to do with the module. For average daily temperature, two inputs are required (and need to be created) because two variables (minimum daily temperature and maximum daily temperature) are used to create a daily average. The "**Inputs to be linked"** box allows the user first, to create the inputs that are needed to evaluate the expression and second, to link these internal inputs to the required variables from the model (i.e., minimum and maximum daily temperatures). The two inputs are created for the module by selecting the "**Add Extra Input"** button. The created inputs are called sequentially "Variable1", "Variable2",

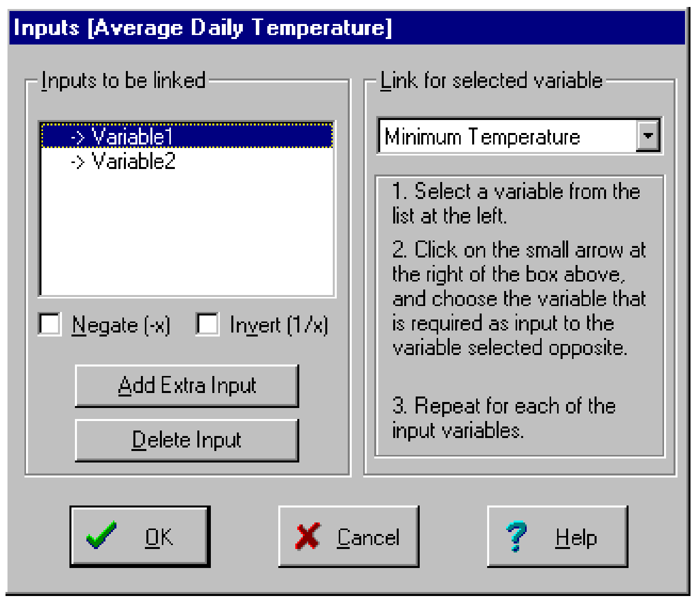


Figure 3-2 "Inputs" dialog for the "Average Daily Temperature" module. Input 1 (Variable1) is shown linked to the Minimum Temperature

5. Select **"Add Extra Input"** twice to produce two input variables in the "**Inputs to be linked"** list box

Once this is done, two inputs, "**Variable1"** and **"Variable2"** will appear in the list box.

6. Select (by highlighting) **"Variable1"** - the scroll box under **"Link for selected variable"** will now become active
7. Click on the scroll button of the **"Link for selected variable"** list box
8. Select **"Minimum Temperature"**
9. Select by highlighting **"Variable2"**
10. Using the same procedure of steps 7 & 8, link **"Variable2"** to **"Maximum Temperature"**

Check that both Variable1 and Variable2 have been correctly linked to the required external variables by selecting each in turn - the scroll box variable should change to the linked external variable in each case.

11. Close by selecting **"Ok"**
12. Select **"Outputs"** to obtain the **"Outputs"** edit window - the single output variable will already be highlighted
13. Click on the **"Select"** button
14. Select "**Rename"**
15. Alter the variable name to **"Average Daily Temperature"** and click **"OK"**
16. **Provide a "Description"** ("The average of the daily minimum and maximum temperatures") and a **"Mnemonic"** ("TAverage")
17. Select **"OK"** to return to the "Expression" module window
18. Select the **"Settings"** button to obtain the **"Select Expression Operation"** dialog box
19. Select **"Average"** from the list of available operations and then click **"Ok"** to return to the module window
20. Select the **"Sort Order"** window and type in a value that will place the module between the "Meteorological Data" and "Lifecycle" modules.
21. Close the module dialog to return to the "Model Components" window.

The "**Average Daily Temperature"** module icon should now have a red tick in front of it to indicate that it is correctly set up. The only module still requiring completion is the "**Lifecycle"** module. Because the previous "Temperature" module was deleted, its output variable, "Incubator Temperature", is not available as a driving variable any more. A new driving variable must be re-inserted into the Juvenile Development function.

22. Double-click on the "Lifecycle" module to open the Lifecycle window
23. Select the "**Development"** button in the Juvenile lifecycle
24. Select "**Edit Component"**
25. Open the **"Independent Variable"** scroll box and select "**Average Daily Temperature**" and close all dialog windows
26. Click on the "**Module"** window - the "Lifecycle" module will now show a red tick to indicate its internal processes are complete

Since meteorological data fluctuates seasonally, it is useful to add the "Mean Development Time" as a second output from the Juvenile stage. Because juvenile development is driven by temperature, the development times will vary with the seasons.

27. Click on the "Lifecycle" window to return to the lifecycle
28. Click on the "Lifestage Outputs" button in the "Juvenile" lifestage

29. Select **"Development"** and re-name **"Mean Juvenile Development Time"**

The "Development" lifestage output reports the time taken for the "Physiological Age" to reach a pre-set value (by default, this value is 1). This is precisely how we have defined Juvenile development.

30. Select **"Ok"** as necessary to return to the "Lifecycle" window
31. Save model and exit the program

3.2 Initializing the Model

3.2.1 The New Model Components Window

Once the file is loaded into the DYMEX Simulator, a "Model Components" window appears showing the modules present (Figure 3-3). This is different from earlier forms of the model in that ticks of different colours appear and there is no tick beside either the "Timer" or the "Meteorological Database" modules. The colour of the ticks indicates whether user initialization may occur: a blue tick indicates that there are settings of the module that the user can change in the Simulator, a grey tick indicates that there are no user-adjustable settings. If a tick is absent, user initialization of the module settings is <u>required</u> before the Simulator will run the model.

The "Model Components" window shows that "Lifecycle" initialization settings can be changed, the "Average Daily Temperature" module cannot be altered and both the "Meteorological Database" and the "Timer" modules require user initialization.

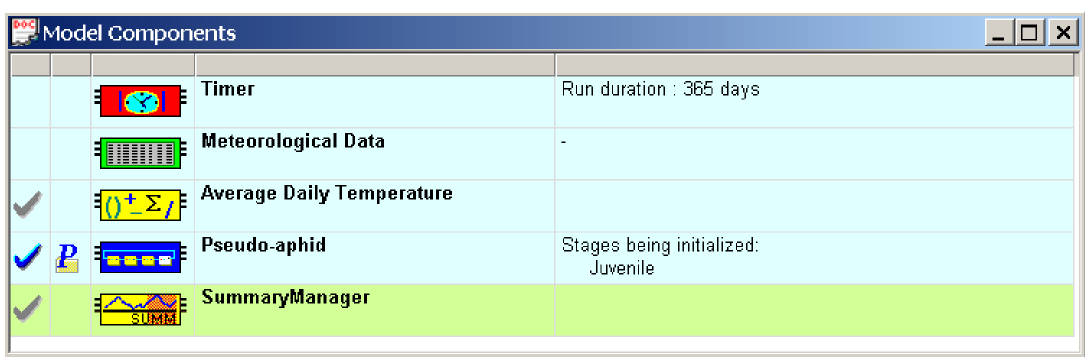


Figure 3-3 The "Model Components" window

Using the Meteorological Data module, the Simulator can read meteorological information from a file with almost any format as long as two conditions are met:

1. **Each line of the file must have exactly the same format; and**
2. **The user must know precisely the format of the file.**

A typical meteorological file contains information on temperature, rainfall, humidity, air pressure, evaporation, etc. and may be daily, weekly, monthly etc. The exact format depends upon the circumstances under which the file was built and DYMEX must be “told” the precise format in order to find the necessary information.

3.2.2 A Meteorological File

The Amberley meteorological file (Figure 3-4) will be used in the pseudo-aphid model. Only the first few lines of the file are shown. The file is in a “fixed-field”, columnar format. Columns are counted from the left and blank columns must be included in the count. For the lines of data, column one is currently blank but is marked by the capital A at the start of the file's data location line, column two contains the numbers 1-6, column three is blank, column four contains 1's, etc. The tutorial user ***must*** be familiar with the file structure before progressing further.

```
A     Amberley Airport   Lat: -27.6  Long: 152.7  Elev:   25
       TMIN TMAX RAIN           RH9           RH3  DLNG EVAP
 1 165 18.0 30.0  0.  5.0 16.0 73.  17.3 17.0 46.  13.85  5.6 AMB
 2 165 17.0 32.0  0.  2.9 18.0 84.  21.1 17.0 41.  13.85  5.5 AMB
 3 165 19.0 33.0  0.  1.8 21.0 91.  22.2 18.0 41.  13.84  5.6 AMB
 4 165 19.0 31.0  0.  5.3 17.0 73.  19.2 17.0 43.  13.83  6.1 AMB
 5 165 19.0 31.0  0.  5.3 17.0 73.  19.2 17.0 43.  13.82  6.1 AMB
 6 165 19.0 32.0  0.  4.6 18.0 77.  21.1 17.0 41.  13.81  6.3 AMB
 7 165 21.0 29.0  0.  6.5 17.0 69.  16.4 16.0 45.  13.80  6.4 AMB
 8 165 17.0 29.0  0.  7.7 11.0 56.  20.8 10.0 31.  13.79  7.4 AMB
 9 165 15.0 29.0  0.  4.7 13.0 70.  20.8 10.0 31.  13.78  6.4 AMB
10 165 16.0 35.0  1.  5.6 15.0 70.  31.6 12.0 25.  13.77  8.1 AMB
11 165 21.0 39.0 11. 10.8 16.0 56.  44.9  7.0 14.  13.75 12.7 AMB
12 165 19.0 35.0  0. 13.0  8.0 38.  38.2 -2.0  9.  13.74 12.4 AMB
```

Figure 3-4 The “Amberley.dat” meteorological data file

Structures in the file on which DYMEX will require information for this tutorial are:

Lines 1 and 2 of the file are information “headers”.

The first 6 columns of the file are date information with format “ddmmyy”. (Notice that some of these columns are blank at first but will be filled when either double digit days or months are reached.)

Columns 8-11 contain the daily minimum temperature.

Columns 13-16 contain the daily maximum temperature.

The remainder of the file contains various other data and can be ignored for this tutorial.

3.2.3 Initializing the Model

The first thing to do is to initialize the "Meterological Data" module to specify that the Amberley data should be used, and to instruct the module how to read the necessary data from the file.

1. Select the "**Meteorological Data"** icon in the "**Model Components"** window and then select "**Initialize Module"** from the drop down menu

This will open the "**Data Files"** selection window which will allow the user to find, open and format the meteorological file that is required for a field temperature run. The selection window also allows the user to specify how the file data has been formatted (either fixed width columns or comma delimited data sets). The default selection of fixed width columns is the appropriate selection for the Amberley file (**Amberley.dat)**

2. Select **"Browse"** and scan the files/directories until **Amberley.dat** is located (it should be in a sub-directory called **"MetData"** included within the DYMEX directory
3. Select **Amberley.dat** and return to the **"DataFiles"** dialog box
4. Select the **"Format"** button to produce the "**File Format"** dialog (Figure 3-5).

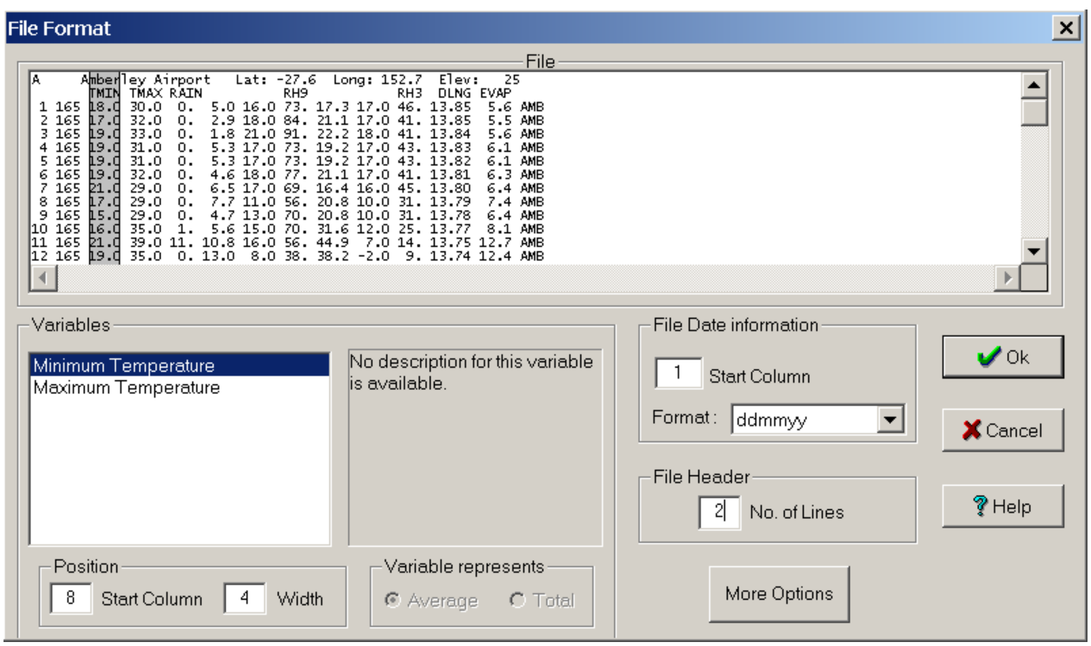


Figure 3-5 The "File Format" dialog with "Amberley.dat"

The "File Format" dialog (Figure 3-5) allows the user to specify the format of the selected meteorological data file so that DYMEX can interpret the file correctly. Note that this needs to be done only once for any particular file – the settings are than stored in the 'Simulation File' for later use. For this tutorial the "**More Options"** button, which enlarges the dialog to provide more advanced options**,** is not required.

Header Lines

Data files sometimes have introductory lines that contain information other than required data and the file reader must be set so it will ignore these lines. Inspection of the file contents (see Figure 3-4) shows that only the first two lines contain header material.

1. From the window panel labelled "**File Header"** select the "**No. of Lines"** edit box
2. Enter the value 2.

Date Information

So that DYMEX can synchronize reading of the file correctly with other modules, the date format must be defined. DYMEX has a list of standard date formats from which the user can select the one corresponding to the data file. Again, from examining the file, it can be seen that file dates start in column 1 and are in the form "ddmmyy".

3. In the window panel marked **"File Date Information"** examine the "**Start Column"** text entry box and ensure its value is set to 1
4. Examine the "**Format"** scroll box. If the format is not set to "**ddmmyy"**, then scroll down the list of formats until it is found and select it

Temperature Information

The DYMEX Simulator must "know" where to look for the required temperature information in the file. This is done for each variable in the "Variables" list by using the standard windows "select, drag and mark" routine with the mouse. Place the cursor at the start of the desired area, hold down the left-hand button and slide the mouse until the desired area is highlighted and then release the button.

1. Assuming the **"Minimum Temperature"** is selected by default, place the cursor just before the start of the 18 (this will be just beneath and in front of the "T" of the word TMIN in the preceding heading line)
2. With the left hand mouse button held down, slide the mouse to the right until 18.0 is highlighted
3. Release the mouse button - the selected area of the file will remain highlighted and will extend to the full height of each selected column
4. The **"Start"** column edit box in the "Position" area of the window should now show 8 while the **"Width"** box should now show 4
5. Select **"Maximum Temperature"**

6. Repeat steps 1-4 with the maximum temperature values which start at 30 (under the “T” of TMAX)
7. The **“Position”** edit boxes should now show 13 and 4
8. Select “**Ok”** as required to return to the **“Model Components”** window.

The “Meteorological Database” icon will now have a tick beside it. The final task is to initialize the “Timer” module.

1. Select the “**Timer”** icon to produce the drop down menu
2. Select “**Initialize Module”**
3. In the “**Simulation Duration”** selection window and in the **“Simulation Starts”** panel, select the “**from”** button (It should already contain the date 1/1/1965, as this is when the Amberley file begins)
4. Verify that the run length is set to 365 days
5. Select “**OK”** and return to the “**Model”** window

A tick beside the “Timer” module now indicates that it is initialized and that the model can be run.

3.3 Running the Model

Run the model using the usual procedures. On completion of the run, the usual charts and tables may be displayed and it will be found that the population is able to exponentially increase as occurred in previous runs.

A new output variable was added to the lifecycle when the model was modified – the “Mean Juvenile Development Time”. To create a chart showing this variable plotted on a time axis:

1. Open the **“Chart Specification”** window by clicking on the chart icon and then selecting **“New..”** from the drop-down menu
2. Select **“Days since Start”** for the X-variable and **“Mean Juvenile Development Time”** for the Panel 1 Y-variable
3. Remove Panel 2
4. Select the **“General Options”** button and untick the **“Show”** button in the Legend box
5. Click on the **“Save Format”** button, and save the chart format with the name “Juvenile Development” (it will be used again later)
6. Select **“OK”** to display the chart.

Examination of “Mean Juvenile Development Time” chart (Figure 3-6) shows that the pseudo-aphid development time is greatly increased during the winter (when temperatures are low). Note that all development times are given in integral number of days. This is due to the model having a daily timestep and the development time is the number of days taken by individuals that were born on a particular day to reach the adult stage.

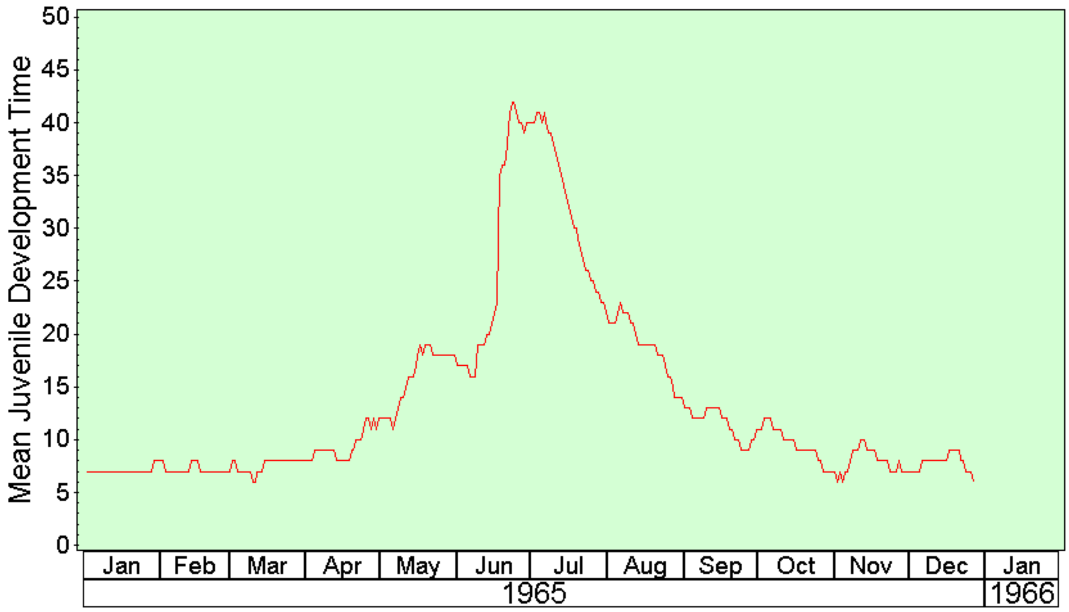


Figure 3-6 A chart of "Mean Juvenile Development Time" for 1965

DYMEX's charting procedures allow the scale of any chart to be altered. This becomes very useful when multiple chart displays are to be produced. Unless instructed otherwise, DYMEX will use a default scaling that the program considers is most suitable for the range of the user's data for each of the chart outputs, and sometimes this may not suit the user's purpose.

Chart properties such as the scale may be changed while the chart is being specified initially, or after it has been drawn. As an example of changing the scale of an existing graph, use the following procedure:

1. With the previous chart window selected, click on the **"Chart"** item in the menu bar and select **"Edit Format"** from the drop-down menu
2. In the "Chart Format" box, select and highlight the "Y-axis" line (below **"Panel 1"**) by clicking on it
3. Select the **"Edit"** button to open the **"Tick-axis Format"** dialog
4. From the **"Scale"** panel select the **"Manual"** button
5. Select the **"Manual"** button
6. Change the "Minimum value" from **0** to **10**
7. Change the **"Number of divisions"** from **10** to **4** and deselect the **"Show minor tick marks"** option
8. Select **"Ok"** as necessary to produce the chart, which should be similar to Figure 3-7.

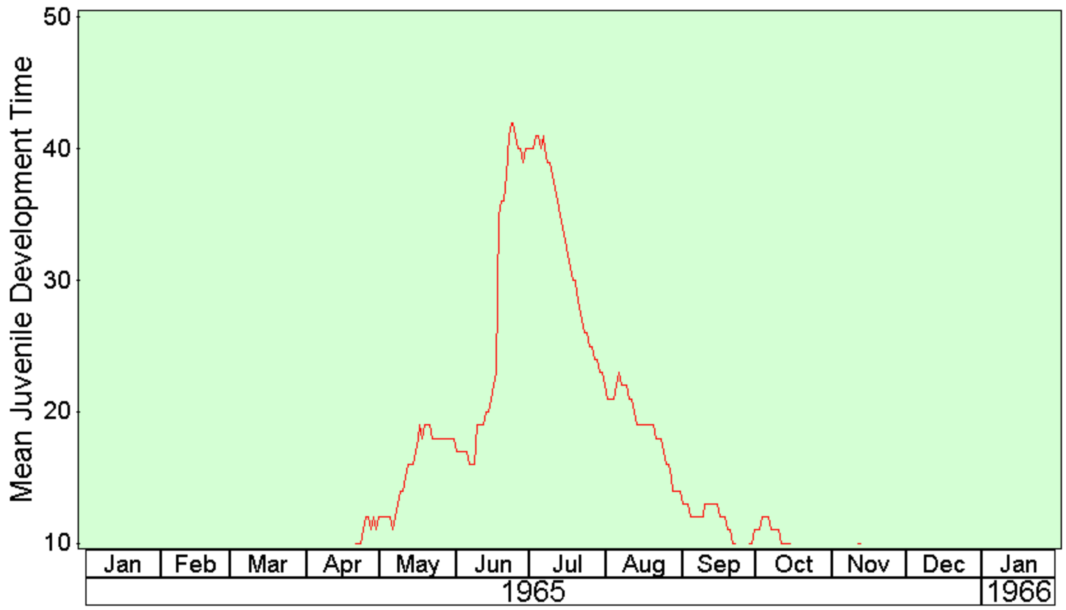


Figure 3-7 Juvenile development time with user-defined scaling

The selected settings have truncated the data display so that only the values within the limits set by the user are present. The major interval has been changed to a 4-unit step and minor ticks are now omitted. Note that manual X-axis scaling in "Common X-axis" charts is not available unless either "Show Year" or "Show Months" are selected.

Tutorial 3 - Summary

Timer

Inputs: none
Outputs: Days since Start
Simulation Date
Settings: Timestep 1 day

MetBase (Meteorological Data)

Input: Simulation Date
Outputs: Minimum Daily Temperature
Maximum Daily Temperature

Expression (Average Daily Temperature)

Inputs: Minimum Daily Temperature
Maximum Daily Temperature
Output: Average Daily Temperature

Lifecycle

Juvenile

Transfer to adult function (Step)
Driving variable: Physiological Age
Transfer threshold: 1
Prop.juveniles transferring: 1
Juvenile development function (Linear above Threshold)
Driving Variable: Average Daily Temperature
Threshold: 10
Slope: 0.01
Output:
Total number
Juvenile Development Time

Adults

See ***Tutorial 1***

4 Degree Days

4.1 Introduction

In the previous tutorial, development of the Pseudo-aphid was linked to temperatures, with actual field temperatures used as input. An average temperature was calculated from the daily minimum and maximum, and this was applied to the development process to drive the increase in Physiological Age. This approach is unsatisfactory in that it does not fully reflect the actual conditions the organisms experienced in the field. A simple example will make this clear: Minimum and maximum temperature values of 8°C and 12°C give an average temperature of 10°C, but so do 0°C and 20°. In the model of Tutorial 3, both these would give the same increase in development, yet to the insect in the field, they are likely to have quite different effects. The latter temperature regime would result in a substantial part of the day well above the developmental threshold of 10°C, and is likely to result in more development than the former regime. This concept (termed the 'day-degree' concept) is described and applied to the model in this tutorial.

4.2 The 'Degree Day' Concept

The pseudo-aphid has been given a development threshold of 10°C, which implies that once the temperature rises above that threshold, development proceeds. Suppose that the pseudo-aphid was existing under ideal conditions in an incubator, where the temperature was maintained at a steady 23°C. This situation is illustrated in Figure 4.1, where the area between the developmental threshold and the actual temperature is shaded. The shaded area represents the number of 'degree-days' available to the aphid's development. The name 'degree-days' is derived from the fact that the size of the area is obtained by multiplying the height of the shaded rectangle (in <u>degrees</u>) by the width (a number of <u>days</u>). Each day has a temperature that is 13°C above the threshold of 10°C, thus the number of degree days accumulated over a 3-day period would be 13 x 3 = 39.

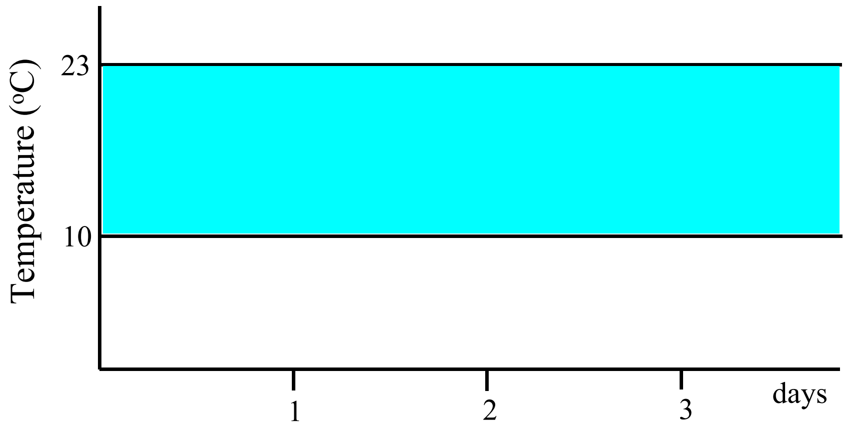


Figure 4-1 Accumulation of "Day-degrees" at a constant temperature

Suppose now that the temperature is different each day, but constant within the day. Three successive days, with temperatures of 15°C, 6°and 12°C are shown in Figure 4.2. Again, the number of degree-days that have accumulated is calculated by summing the area under the temperature curve (and above the threshold temperature line). In this case, it is 5 + 0 + 2 = 7 degree days. The temperature on the second day is below the developmental threshold, and its contribution to the sum is zero. Note that the average temperature for the three days is 11°C, which would yield only 3 degree-days over the 3 days if it were applied directly.

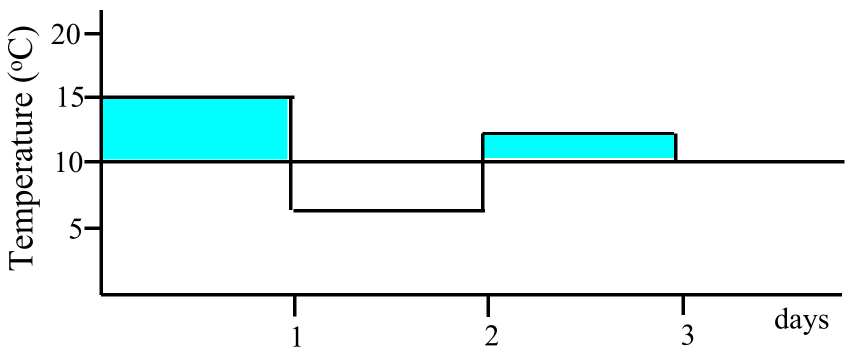


Figure 4-2 Accumulation at day-degrees at different temperatures

An important point to note is that determining accumulated degree-days in this way is exactly equivalent to specifying development in terms of "Physiological Age", with "Physiological Age" a function of temperature using the "Linear above Threshold" function. This is easily seen by comparing Figure 4.2 with the graph below (Figure 4.3), where the slope of the function is set to 1.

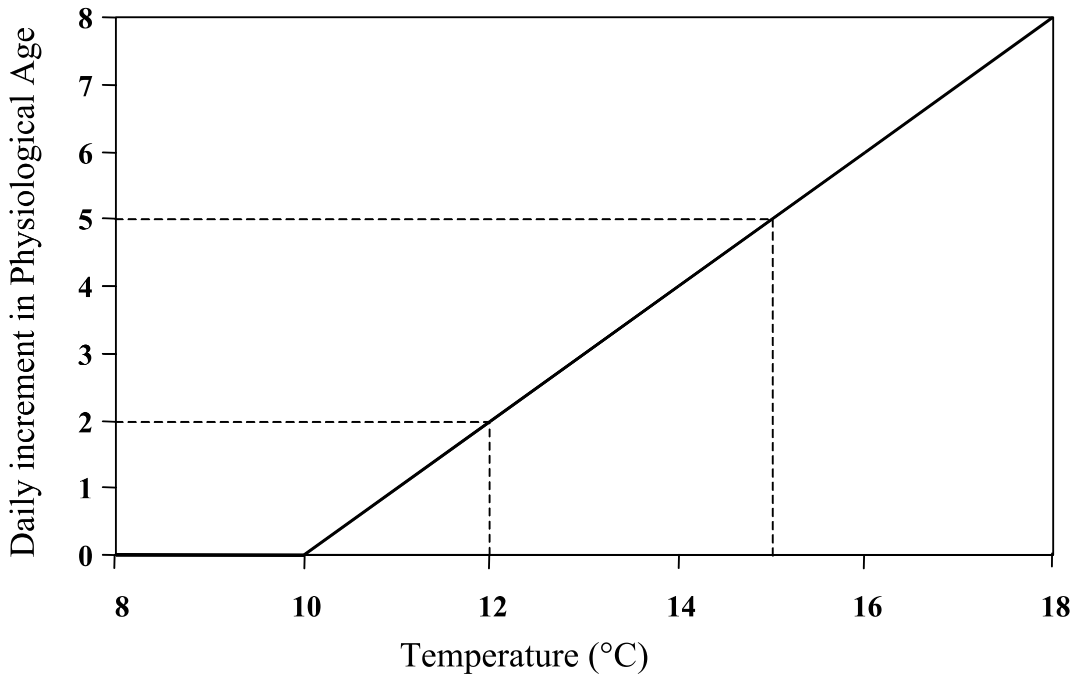


Figure 4-3 Daily increment in Physiological Age with "Linear above Threshold" function

4.2.1 Calculating Degree Days Using the Circadian Cycle

When using actual field temperatures, the situation is more complex, as the temperature varies between the day's minimum and maximum in some cyclical way each day. DYMEX uses the daily maximum and minimum temperatures of the meteorological database to fix the "crest" and "trough" limits of a sinusoidal, circadian cycle of temperatures (Figure 4-4). To calculate the number of degree-days, the same procedure as in the above examples is used. The sine curve is divided into 12 segments (each of 2 hours). A rectangle (of width 2 hours) is created for each of the 12 segments, with the base on the threshold temperature and the midpoint of its top edge on the sine curve (Figure 4-4). The area of this rectangle is used as the degree-days accumulated for those 2 hours, and all 12 areas are added to obtain the total degree-days for the whole day. The sinusoidal approximation should be adequate for most purposes that DYMEX is used for, and will be used for the Pseudo-aphid model. However, for more specialized applications, two other types of curves are available in DYMEX that approximate the daily cycle in different ways (see the *Builder User's Guide* for more information).

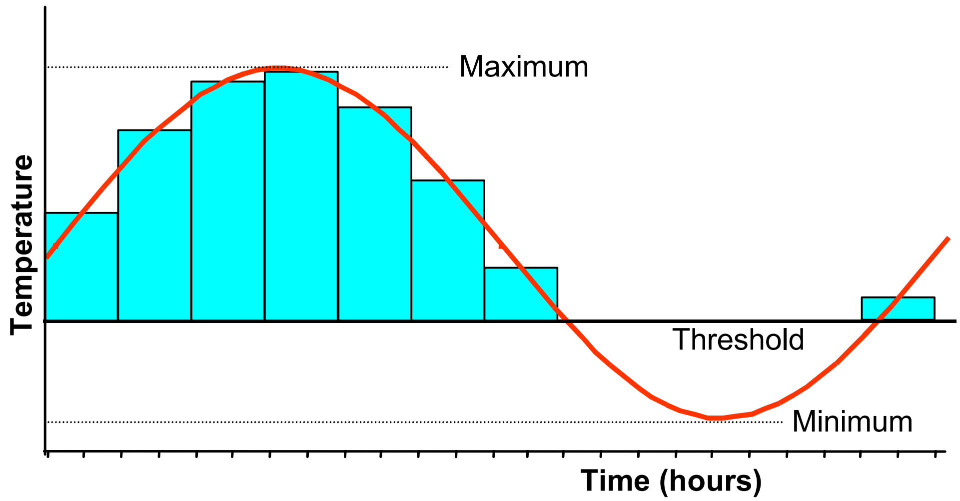


Figure 4-4 Circadian cycle between daily maximum and minimum temperatures, and the 2-hourly degree-day summation

The current Pseudo-aphid model already implicitly uses degree-days for determining development (since the "Linear above Threshold" function is used to determine the accumulation of Physiological Age). The model will now be modified so that a circadian curve of temperature is used for driving the development.

4.3 Modifying the Model

1. Open the Model Builder and load the pseudo-aphid model
2. Delete the **"Average Daily Temperature"** module
3. Select **"Add Module"** from the drop down menu
4. Select **"Circadian"** from the **"Create Module of Type ?"** list window
5. Rename the Circadian module to **"Daily Temperature Cycle"**
6. Type a value of 20 into the **"Sort Order"** edit box, so that the new module replaces the deleted module at the same position
7. Click on the **"Inputs"** button
8. From **"Inputs to be Linked"** list box, select **"Daily Minimum Value"**
9. From **"Link for Selected Variable"** scroll box, select **"Minimum Temperature"**
10. Repeat step 7 for "**Daily Maximum Value"** and link it to **"Maximum Temperature"** (**Note:** the variable **"Daylength"** is not linked for this model – it is only required if a non-sinusoidal cycle is used)
11. Select **"Ok"**

12. Check **“Output”** is set to **“Daily Cycle”** and then click on **“Select”** to give **“+>”** beside the set variable
13. Rename the output **“Daily Temperature Cycle”** and then select **“OK”**
14. Provide a **“Mnemonic”** for the output (“TDaily”)
15. Select **“OK”** as necessary to exit to the **“Model”** window

With the circadian cycle set, the remaining modification takes place in the Juvenile Development process, which must be changed to use the **“Daily Temperature Cycle”**.

16. Select the **“Lifecycle”** module for editing by double-clicking on it in the “Model” window
17. Select the **“Development”** button in the Juvenile lifestage

The “Juvenile – Development” dialog box will have been previously set to use “Average Daily Temperature”. This needs to be altered.

18. Select the **“Edit Component”** button
19. Select the **“Independent Variable”** list box scroll button
20. Select **“Daily Temperature Cycle”**
21. Select **“Ok”** as necessary to return to the **“Model”** window

4.4 Running the Model and displaying the results

When this altered model is run in the Simulator, only small changes will be noticed in the population charts. There are, however, changes visible in the “Juvenile Development” output chart (This chart format was saved in the last tutorial with the name “Juvenile Development”, and should show up as a choice in a popup menu when the “Chart” button or menu item is selected). A major effect is that the time that juveniles take to develop during the winter months is reduced by up to 25% (compare Figure 3-6 and Figure 4-5), while the development times in summer stay approximately the same. The reason for this was explained in Section 4.2, but an example may help to underline this. Assume that on a particular day in winter, the minimum and maximum temperatures are 2°C and 18°C, respectively, giving an average temperature (as used for calculating Figure 3-6) of 10°C. Since the development threshold is set to 10°C, no development would take place with that average temperature. However, using the more sophisticated model in the current tutorial, the interpolated sine curve would result in a significant part of the day above the threshold temperature and a corresponding amount of development would occur.

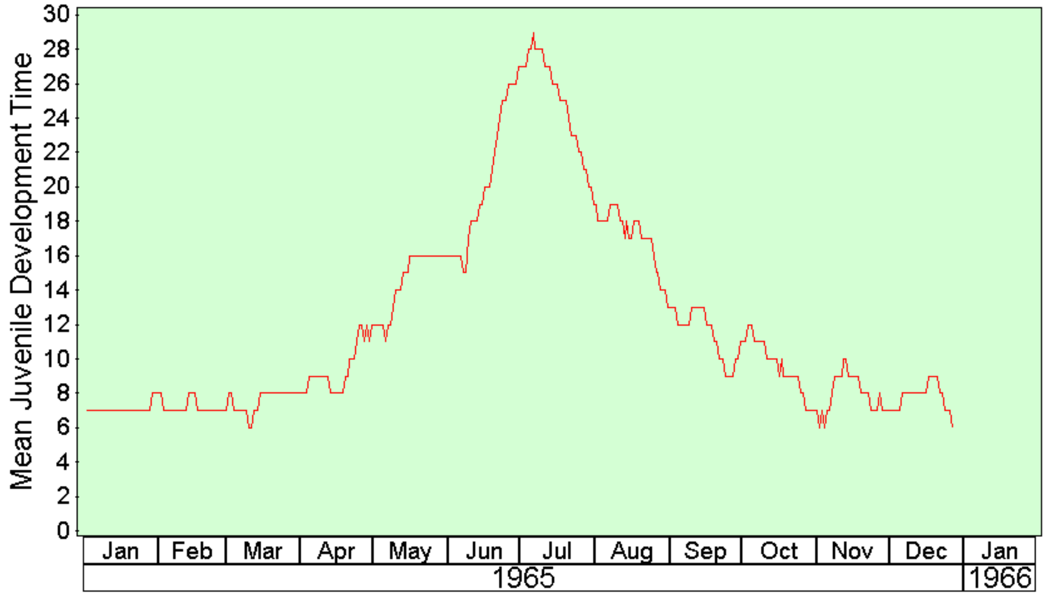


Figure 4-5 Mean Juvenile Development Time using a daily temperature cycle

Tutorial 4 - Summary

Timer

See ***Tutorial 3***

MetBase (Meteorological Data)

See ***Tutorial 3***

Circadian (Daily Temperature Cycle)

Inputs: Minimum Daily Temperature
Maximum Daily Temperature
Output: Daily Temperature Cycle

Lifecycle

Juvenile

Transfer to adult function (Step)
Driving variable: Physiological Age
Transfer threshold: 1
Prop.juveniles transferring: 1
Juvenile development function (Linear above Threshold)
Driving Variable: Daily Temperature Cycle
Threshold: 10
Slope: 0.01
Output:
Total number
Juvenile Development Time

Adults

See ***Tutorial 1***

5 Mortality

5.1 Introduction

In the current pseudo-aphid model, the only source of mortality is in the Adult stage, where adults die of "old age" (i.e., after they reach a specified Chronological Age). DYMEX permits mortality to be built into each lifestage and mortality can be linked to any suitable user selected variable(s). In the field, aphids suffer mortality for a variety of reasons, predators, humidity, disease, temperature, etc. and all these can be modelled by DYMEX. In this tutorial, temperature-driven mortality will be added to the pseudo-aphid model. Aphids are affected adversely by both excessively high and low temperatures. It will be assumed that each lifestage of the pseudo-aphid suffers a linear mortality in response to unfavourably high/low temperatures and that this will be the only additional cause of mortality apart from the currently existing adult death due to age.

5.2 The Temperature-based Mortality

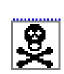
Assume that both the juveniles and adults have the same sensitivity to extremes of cold and heat. Further assume that we have data that indicates that mortality starts to occur when temperatures drop below 4°C and increases by 0.1 for each degree temperature drop below that value (i.e., at 3°C, there is 10% mortality per timestep, at 2°C, there is 20%, etc). The data also shows that mortality occurs above 34°C, and increase by 0.05 for every degree above that temperature (so that, for example, 20% of individuals die each timestep at 38°C due to heat stress). These responses will now be incorporated into the model.

Note: When providing slope values for negative slopes, (eg. linear below threshold functions) always ensure that a negative sign precedes the slope value. (eg. -0.03).

5.3 Modifying the Model

With the pseudo-aphid model loaded in the Model Builder, complete the following steps:

1. Select the **"Lifecycle"** module for editing
2. Select the **"Juvenile"** lifestage **"Mortality"** button



3. In the **"Juvenile Mortality"** list box, select **"Continuous"** as this mortality acts at <u>all</u> times when the temperature is unfavourable

4. In the **"Mortality (Juvenile)"** dialog box (Figure 5.1), type in "*Juvenile Mortality*" into the **"Name"** edit box
5. Select **"Function"** to obtain the **"Function"** dialog
6. Select **"Linear below Threshold"** from the **"Function"** scroll list
7. Select the **"Independent Variable"** scroll button

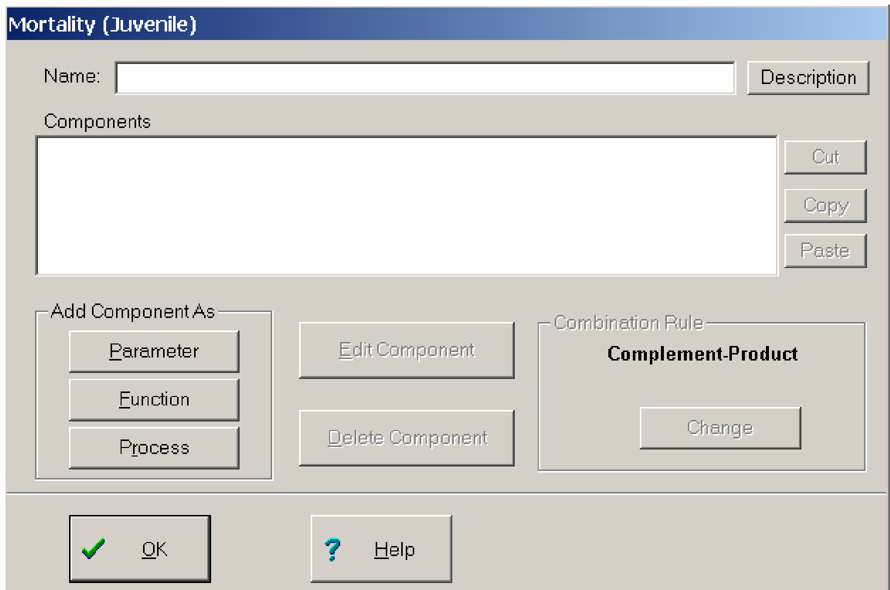


Figure 5-1 The "Juvenile – Continuous Mortality" dialog box

8. From the list, select **"Minimum Temperature"**
9. Insert any desired comments in the edit box and suitably name the function (eg. "Cold Mortality")
10. In the **"Parameters"** panel select **"p1: Threshold"** and click on the **"Edit Parameter"** button
11. Enter 4 as default, and allow users to change the parameter within the range –2 to 10 (i.e., enter -2 and 10 for the lower limit and upper limit values, respectively)
12. Enter a suitable name for the **"User Name",** eg. "Cold Threshold"
13. Insert suitable comments if required, and select **"OK"**

Note: It good practice to use a standard format for the names of model parameters as otherwise they can become rather confusing when listed.

14. In the **"Parameters"** select **"p2: Slope"** and click on **"Edit Parameter"**
15. Enter -0.1 as the default value, and -0.2 and 0 for the lower, and upper limits, respectively

Note: Note that the "Linear below Threshold" function has a negative slope, and the 0.1 increase in mortality per degree translates to a slope of –0.1.

16. Insert suitable comments if required

17. Enter a suitable name for the **“User Name”**, eg. “Cold Slope
18. Select **“OK”** as necessary to return to the **“Mortality (Juvenile)”** dialog box
19. Re-select the **“Function”** button to add the high-temperature mortality component
20. Select **“Linear above Threshold”** as the function
21. In the **“Function”** dialog box, select **“Maximum Temperature”** as the **“Independent Variable”**
22. Insert parameter values, names (“Heat Threshold” and “Heat Slope”) and comments for the high temperature values as outlined in steps 10-17 using the values from Section 5.2. Choose lower and upper limits for the parameters that allow the model user reasonable flexibility to adjust the parameters in the Simulator (perhaps 30-38 for the threshold, and 0-0.2 for the slope)
23. Return to the **“Mortality (Juvenile)”** dialog box

An important step must now be completed. Inspection of the “Process Components” list box shows that two separate functions (Maximum Temperature and Minimum Temperature) contribute to the Mortality processes. These two functions must be combined and inspection of the buttons on the right of the “Mortality (Juvenile)” dialog box shows that the **“Combination Rule”** panel now shows “Complement product” as the selected method of combining the process factors, and the **“Change”** button is now active. Although there are a number of ways in which functions can be combined in DYMEX, mortality processes generally use the **Complement product** rule, which will now be described.

5.3.1 Combining Mortality Functions With the Complement product Rule

DYMEX provides several ways of combining process components to give a final value for the process rate. The components are always evaluated first, after which the combination rule is applied to the individual component’s values to give the process rate.

All mortality component rates are expressed as values from 0 to 1. If a particular process factor (for example, the high-temperature component) evaluates to a 0 during a particular timestep, then no mortality occurs due to that component during the timestep. Similarly, a value of 1 indicates that all individuals in the affected cohort will die during that timestep due to the that component.

The **Complement product** combination rule is the appropriate choice for mortality processes, since, when combining mortalities from different sources, we are actually doing *survival* (1-mortality) calculations. An example will illustrate this best. Assume in the Pseudo-aphid model during a particular timestep, minimum and maximum temperatures of 15°C and 38°C are encountered. From the relationships described in Section 5.2, this gives mortality components of 0 and 0.2 for low and high temperatures, respectively. That means that, taken on their own, the low-temperature component would cause no mortality at all, while the high temperature component would cause 0.2.

Combining these using the Complement product rule would yield a total mortality of 0.2, as below:

$$1-(1-0)\times(1-0.2)=0.2$$

If we had used the **Product** combination rule, the total mortality from these factors would be 0 – obviously wrong:

$$0\times 0.2=0$$

Note that the **Sum** rule would seem to give the right answer here also. This is only so, however, because one of the components is 0. If the two components were 0.5 and 0.5, respectively, the Sum rule would give a mortality of 1.0 (i.e., all die), which is wrong. The Complement product rule would still give the correct mortality of 0.75.

24. Verify that the **"Complement-product"** is shown as the combination rule
25. If this is not so, click on the **"Change"** button, and select that combination rule from the list in the **"Combination Rule"** dialog
26. Select **"OK"** as necessary to return to the **"Lifecycle"** window

5.3.2 Modifying the Adult stage mortality process

When the Adult mortality is being altered, Chronological Age is left unchanged as an adult mortality component, and the low and high temperature components are added as the 2nd and 3rd components. This does not cause any problems, the **Complement product** combination rule correctly combines all three factors together to obtain an overall mortality. To avoid having to repeat the lengthy procedure of steps 2-25 above, the **"Copy"** and **"Paste"** buttons can be used to copy the factors from the Juvenile stage to the Adult stage.

1. Select the **"Juvenile"** lifestage **"Mortality"** button, and then the **"Continuous"** button to open the **"Mortality (Juvenile)"** dialog
2. Make sure the "*Cold Mortality*" factor is selected in the **"Components"** list, then click on the **"Copy"** button
3. Select **"OK"** to return to the Lifecycle window
4. Select the **"Adult"** lifestage **"Mortality"** button, and then the **"Continuous"** button to open the **"Mortality (Adult)"** dialog
5. Click on the **"Paste"** button – the component will appear in the **"Components"** list
6. Select **"OK"** to return to the Lifecycle window
7. Repeat steps 1-5, substituting the "*Heat Mortality*" factor for "*Cold Mortality*"
8. Click on the **"Change"** button and make sure that the **"Complement-product"** Combination Rule is selected

9. Click **“OK”** as required to return to the Lifecycle window
10. Save the file

If all steps are successful, the “Mortality” button in each stage diagram will display a red tick.

5.4 Running the modified Model

The model is loaded into the Simulator exactly as previously described. This time, however, we will initialize the model with 1000 adults rather than the 1 juvenile used in previous tutorials. To make that change:

1. Select the **“Lifecycle”** module icon
2. From drop-down menu select **“Initialize Module”**
3. In the lower list-box of the **“Initialize Lifestage Numbers”** dialog, click on the only entry (specifying 1 individual added), and click on the **“Delete”** button to remove it
4. Select the Adult stage in the upper list-box
5. Select the **“New”** button in the **“Initialize Lifestage Numbers”** selection box to open the **“Edit Lifestage Initialization Set”** dialog box (Figure 1-16).
6. Select the **“Add Individuals”** text entry area and enter the value 1000
7. Select **“Ok”**as required to return to **“Model Components”** window

The model should now be run for 365 days. A chart output of both Juvenile and Adult populations should produce the results of Figure 5-2. (Note that the legend in a chart can be moved anywhere by clicking on it with the mouse and then, with the mouse button down, dragging it to the required spot.)

Figure 5-2 shows that for the first year the pseudo-aphid population is kept in check to some extent by temperature extremes, so that there is population growth under good conditions, while population growth is reduced or even negative (leading to reduced population size) under adverse conditions.

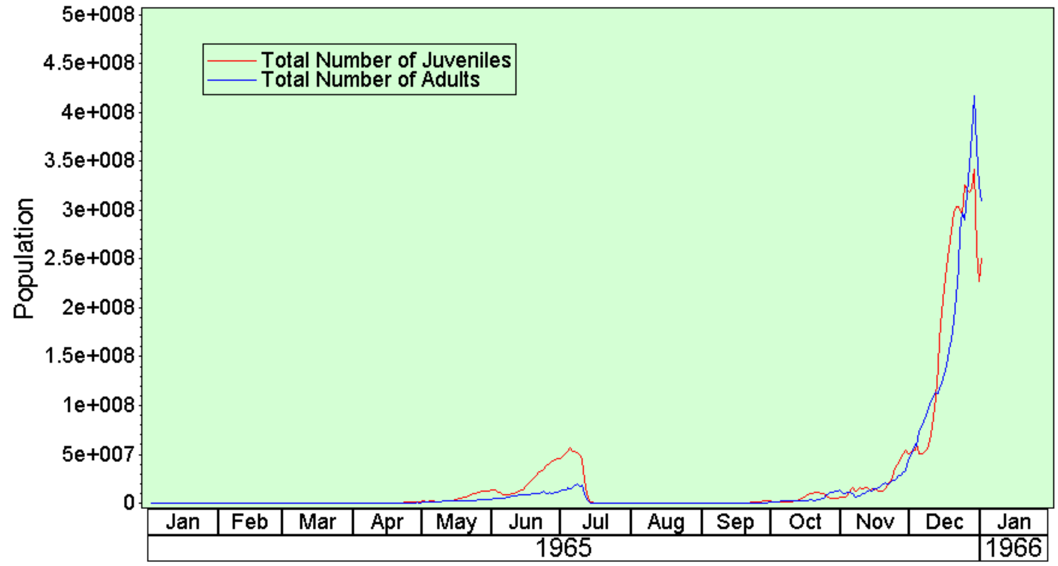


Figure 5-2 Juvenile and Adult Pseudo-aphid populations with temperature-based mortality

The delicate balance of this model is readily seen if the user experiments with some alterations to the parameters before new runs of the model. Pseudo-aphid model parameters are changed in the Simulator by clicking on the Parameter icon (P) of the lifecycle module. (or, alternatively, selecting **"Model Parameters…"** from the **"Initialization"** menu). This displays the "Model Parameters" dialog (Figure 5-3), from which any of the parameters may be altered within the range set in the Builder. The large panel on the left displays a "tree"-diagram of the model (somewhat similar to that shown in the Builder's Model window). Each module that contains parameters is shown (with a ■ icon). The '+' box in front of each icon can be clicked on to show more details such as the lifestages (▲) in a lifecycle module, and the processes (●) within each lifestage. Each parameter (▬) is a terminal branch on the tree. The right side of the dialog describes the selected node of the tree in more detail, including (if applicable) a graphic of its function. A description of the selected node will be present if one was provided when the model was built. A particular parameter's value is changed by finding it in the tree an selecting it, and then setting the new value in the "Value" box (or moving the slider to the correct position). Note that changes are made only to the Parameter (gmp) file, the Model Description (gmd) file is not affected. A convenient way to return to the "default" values is to merely delete the model's "gmp" file (But make sure not to delete the "gmd" file by mistake!).

Note: Lifecycle parameters may also be changed by displaying the Lifecycle diagram (use the **"Show Lifecycle Diagram"** menu item on the Lifecycle icon's popup menu to show the lifecycle diagram), clicking first on the required stage, and then on the graphic of the function that contains the parameter. The parameter is adjusted by selecting its name in the listbox, and using the slider or edit box in the resulting dialog.

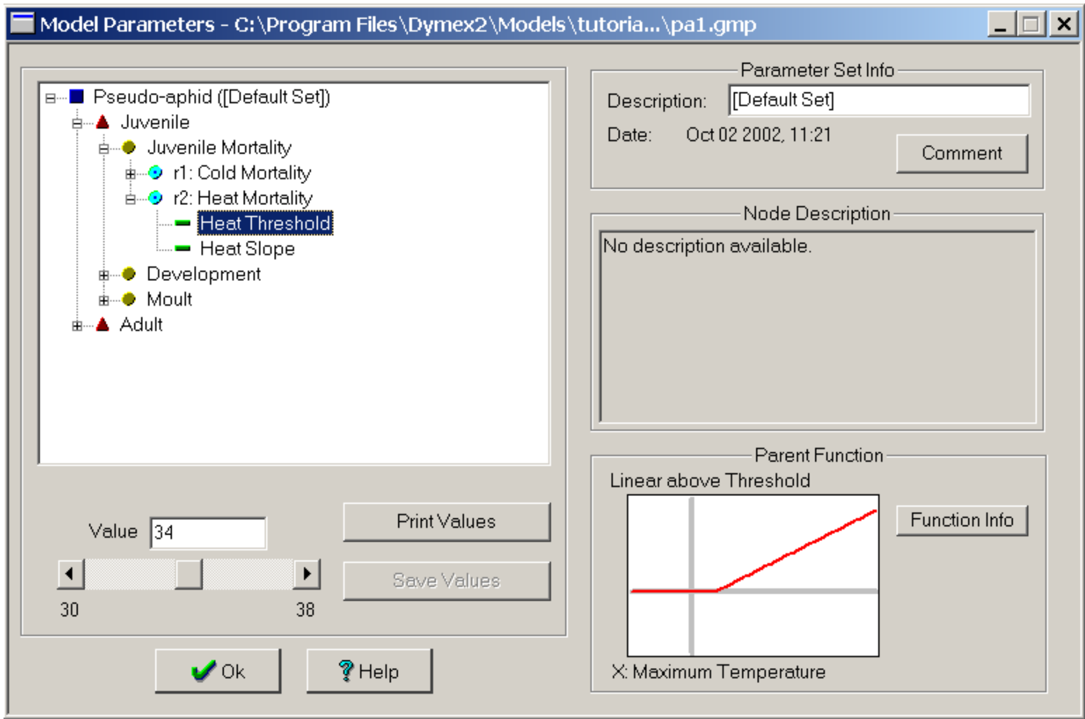


Figure 5-3 The "Model Parameters" dialog

If the low temperature threshold for mortality is increased to its maximum of 10°C, the pseudo-aphid cannot establish itself and dies out. The user might like to experiment briefly with other alterations of thresholds and slopes.

The low July and August temperatures at Amberley suppress pseudo-aphid populations during the first year. Mackay on the tropical Australian coast has more suitable temperatures. Use of climatic data reflecting Mackay's temperatures should produce very rapid pseudo-aphid population increases. Alternatively, colder conditions such as those experienced in southern Australia will most likely lead to population extinction within the first winter.

In any event, the user will be aware that temperature dependent mortality by itself is insufficient to stabilize an isolated population. The mechanism that is used to produce stable populations in the model is explored in the next tutorial.

Tutorial 5 - Summary

Timer

See ***Tutorial 3***

MetBase (Meteorological Data)

See ***Tutorial 3***

Circadian (Daily Temperature Cycle)

See ***Tutorial 4***

Lifecycle

Juvenile

Transfer to adult function (Step)

See ***Tutorial 4***

Juvenile development function (Linear above Threshold)

See ***Tutorial 4***

Low Temperature Mortality (Continuous, Linear below Threshold)

Driving Variable: Minimum Temperature
Threshold: 4.0
Slope: -0.1

High Temperature Mortality (Continuous, Linear above Threshold)

Driving Variable: Maximum Temperature
Threshold: 34.0
Slope: 0.05

Output:

See ***Tutorial 4***

Adults

Age-based mortality function (Step)

See ***Tutorial 1***

Low Temperature Mortality (Continuous, Linear below Threshold)

Driving Variable: Minimum Temperature
Threshold: 4.0
Slope: -0.1

High Temperature Mortality (Continuous, Linear above Threshold)

Driving Variable: Maximum Temperature
Threshold: 34.0
Slope: 0.05

Reproduction

See ***Tutorial 1***

Output:

See ***Tutorial 1***

6 Density-dependent Mortality

6.1 Introduction

Two variables have so far been used to control the pseudo-aphid's mortality: Chronological Age in the adults and temperature in both stages. In Tutorial 5, the effect of these mortality factors was explored. It was evident that population stability was not achieved – populations either increasing forever in suitable locations or becoming extinct in unfavourable locations. Much of this is due to the fact that an isolated population is being modelled and in fact populations will often die out in the field, to be re-established by immigrants later. However, many factors that cause mortality in populations are linked in some way to the density of the population, generally increasing the proportional mortality as the population density increases. These mortality factors (termed *density-dependent*) do have the potential to stabilize isolated populations.

6.2 Population Density and Mortality

Population density is measured as the number of individuals per unit area. If for a simple case, the living area available for the population was fixed (eg. the number of pseudo-aphids present in a lucerne crop), population density would simply be a reflection of total population. This tutorial will assume such a situation so that an increase in population will automatically mean an increase in population density.

Exactly how an increase in population density affects the population growth rate depends on the individual species' characteristics, but there are generally a few main causes of increased mortality rates in a larger population: crowding, food depletion, predation by attracted predators and increase of disease through rapid transmission of pathogens.

Density dependent mortality is a stabilising influence on a population because it has a negative feedback effect. Put simply, the more the population grows, the greater will the rate of mortality become and the ensuing deaths will therefore reduce the population growth rate. As the population decreases, the mortality rate decreases also and the population growth rate increases once more. The population therefore tends to be cyclical with numbers sometimes exceeding the ability of the environment to sustain it but never quite falling away to extinction.

This is an area of modelling where data is often absent, or if available, difficult to interpret. It is usually evident that a population is fluctuating between some limits, but the causes are usually due to many factors and difficult to establish experimentally. When modelling the population, it may be necessary to lump many into one single mortality function, and "guess" at the function shape and parameter values, so that the "model" population predicts the known field population. In that way, the model forms a hypothesis, which will stand or fall when tested with further data. As with all hypotheses, the model will lead to a more directed data collection effort and thus a better utilization of available resources. The improvement in understanding may then allow the previous, composite function to be split into component parts that represent the actual processes occurring better.

6.3 Defining the Mortality Rate

When a population is small, it can be assumed that mortality due to population density effects will be very low. For this tutorial, it will be proposed that no mortality due to population density effects occurs until a minimum of 1000 individuals is present. (This might be the organism's population on a rose bush.) The proposition further suggests that as the population increases, density dependent mortality rate increases in direct proportion until once the population reaches 100,000 individuals, the mortality rate becomes 1. This mathematical structure suggests a "Linear above Threshold" function. If the data is plotted (Figure 6-1) the slope can be calculated and is approximately equal to 0.000001. For this tutorial, it is assumed that the population on the x-axis is the population of the stage whose mortality rate is being calculated (i.e., adults affect adult density-dependent mortality and juveniles affect juvenile mortality). This is obviously not realistic, and an exercise suggested at the end of the tutorial will propose a better model.

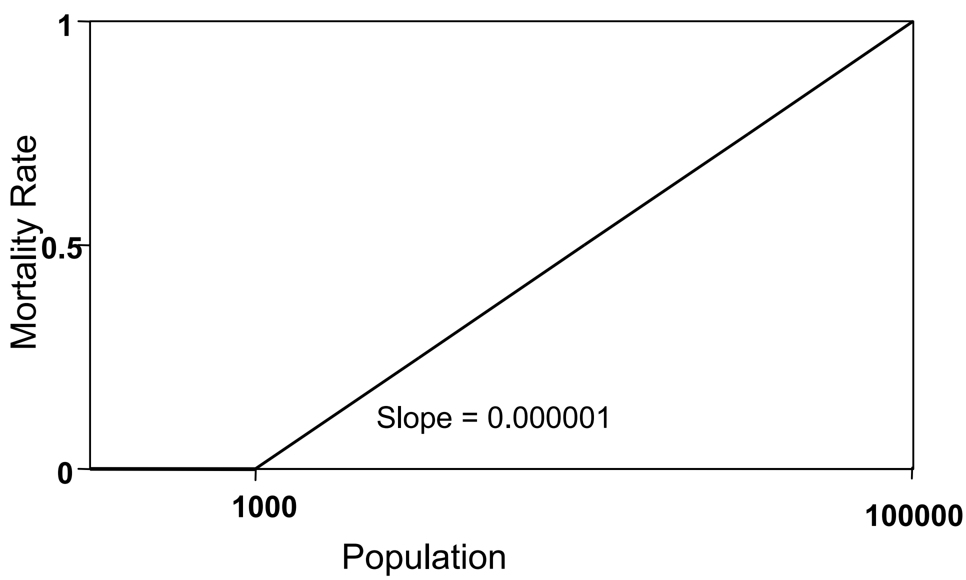


Figure 6-1 Density-dependent mortality function

6.4 Modifying the Model

1. With the **"Lifecycle"** window of the model open in the Model Builder, select the Adult lifestage **"Mortality"** button
2. Select the **"Continuous"** button

3. In the **“Mortality (Adult)”** selection box, select the **“Function”** button

Steps 4-11 below add the density-dependent mortality component. An additional change that is made to the model lessens the effect of Low-Temperature Mortality in both lifestages. The previous “slope” setting of -0.1 was deliberately exaggerated in order to ensure that the effects of Minimum Temperature Mortality were seen. A reduced slope parameter value of -0.05 will be substituted, which ensures the simulated pseudo-aphid population will not become extinct with the temperatures experienced at Amberley. This alteration occurs in steps 12-16.

4. From the **“Function”** scroll box, select **“Linear Above Threshold”**
5. Select **“Change”** and name the function **“Density Dependent Mortality”**
6. Select **“Total number of Adults”** as the **“Independent Variable”**
7. Add any desired comments necessary
8. In the **“Parameters”** panel, select **“p1: Threshold”** and then select the **“Edit Parameter”** button
9. In the **“Set Parameter Properties”** dialog box, do the following:
    (a) Give the threshold a suitable user name (eg, “Density Threshold”)
    (b) Add any desired comments
    (c) Set the default to 1000
    (d) Set the lower and upper limits to 500 and 5000
10. In the **“Parameters”** panel, select **“p2: Slope”** and then select the **“Edit Parameter”** button
11. In the **“Set Parameter Properties”** dialog box, do the following:
    (a) Give the slope a suitable user name
    (b) Add any necessary comments
    (c) Set all default and limit values to 0.000001
12. Return to the **“Mortality (Adult)”** dialog
13. Verify that the **“Complement-Product**” is selected as combination rule
14. Open the **“Cold Mortality”** function for editing
15. Select the **“p2: Cold Slope”** parameter and alter its default setting to -0.05
16. Select **“Ok”** as necessary to return to the **“Lifecycle”** window
17. Repeat the slope alteration for Juvenile **“Cold Mortality”**
18. Exit to the **“Model”** window and save the model

6.5 Running the modified Model

By now, many of the operations in DYMEX should be familiar to the user and some detail will be omitted. Make sure the model is still initialized with 1000 adults. Select “Run” and change the simulation’s run duration to 10 years (3650 days). When the run has completed, create a chart of “Total No of Juveniles”, selecting “Show Years” for the X-axis, to give the results shown in Figure 6-2.

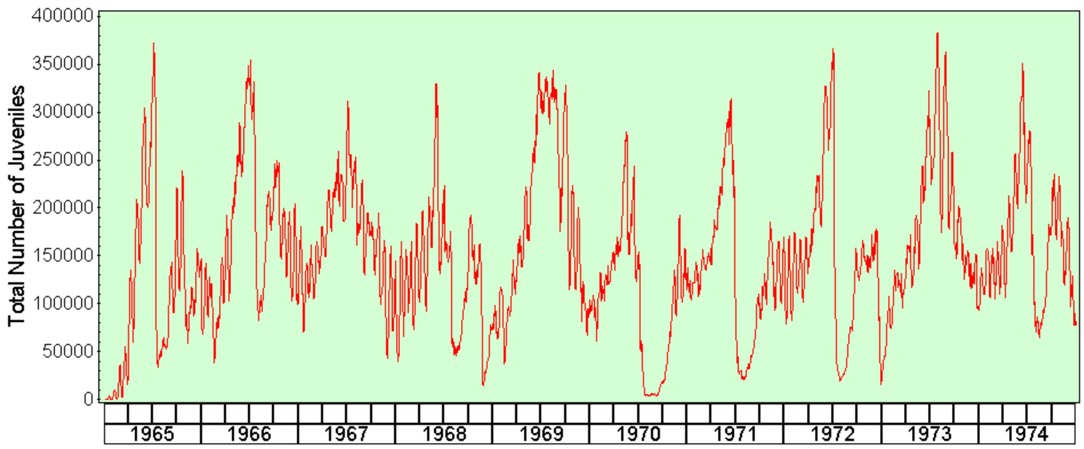


Figure 6-2 Juvenile pseudo-aphid population with density-dependent mortality

The most obvious difference in this graph from the previous graphs is that the population is now stable. It varies over quite a wide range, but there is certainly no more "exponential" increase with time. The juvenile and adult numbers are cyclical and show population increases during favourable intervals with decreases during those intervals that are not. Of interest are the very low numbers during the sixth year (1970) where the population is nearly destroyed. To look in more detail at just what is happening in that year, the model will now be re-run for a two year period (1969-1970), and a chart including both aphid numbers and temperatures will be produced.

1. Close the current "Run" window, returning to the "Model Components" window
2. Select **"Run"** and in the **"Run Model"** dialog, click on **"Change"**
3. Change the run settings to start the simulation on 1/1/1969, and run until 31/12/1970.
4. Click **"Ok"** as required to start the simulation
5. When the run has finished, select **"Chart"** then **"New.."** from the drop-down menu
6. For the X-axis, select both **"Days Since Start"**
7. For the Y-axis variables in Panel 1, select both "Total Number of Adults" and "Total Number of Juveniles"
8. Select the Y Axis for Panel 1 in the **"Chart Format"** box and click on **"Edit"**
9. Make the **"Title"** text the string "***Population Size***" and select **"OK"**
10. Click on **"Panel 2"** and select both "*Maximum Temperature*" and "*Minimum Temperature*" for the Y-axis
11. Select the Y Axis for Panel 2 in the **"Chart Format"** box and click on **"Edit"**
12. Make the **"Title"** text the string "***Temperature***"
13. Click on the **"Manual"** button in the **"Scale"** panel

14. Set the **"Minimum Value" and "Maximum Value"** to 0 and 40, respectively
15. Set the **"No of Major Divisions"** to 4 and the **"No of Minor Divisions"** to 10
16. Select **"OK"**
17. Click on **"Panel 2"** and then on the **"Move Up"** button to exchange the two panels (so that the **"Temperature"** panel is on top).
18. Save the chart format as "***Population and Temperature***" for later use
19. Click on **"OK"** to display the chart

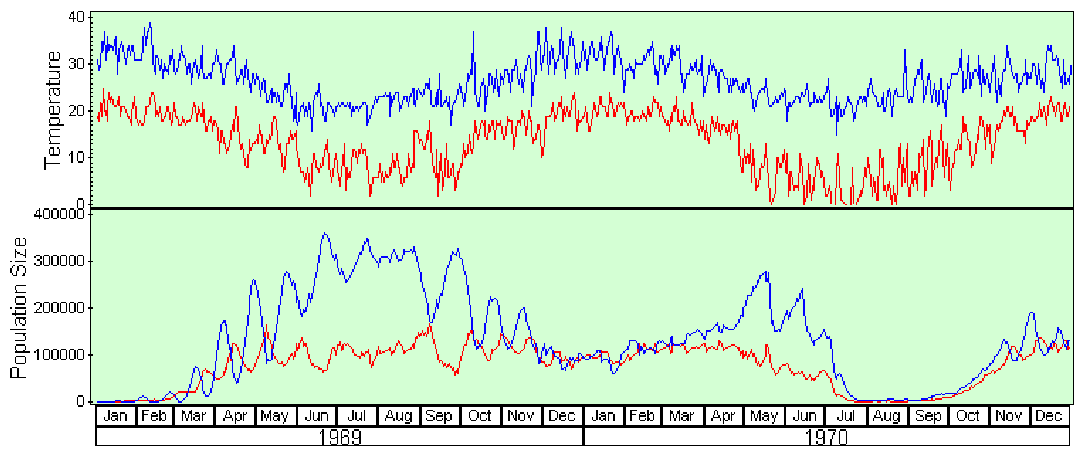


Figure 6-3 Pseudo-aphid population chart with temperatures (legend not shown)

From this chart (Figure 6-3), the cause of the population "crash" in 1970 is now more easily explained. The winter was a particularly cold one, with minimum temperatures below 4°C for a substantial proportion of the time. It is quite clear from the graph that the modelled population was very close to extinction that year.

The density-dependent mortality must now be added to the juvenile lifestage using the same parameter values as for the adult lifestage, with the "Total Number of Juveniles" as the independent variable. For the procedure, repeat steps 1-18 (omitting 14-17) from Section 6.4, substituting "Juvenile" for "Adult". You may, alternatively, wish to use the **"Copy"**/**"Paste"** buttons to save some work in this procedure (but do not forget to change the independent variable in the function to "Total Number of Juveniles"). The new model should then be saved, as it will be extended in subsequent tutorials.

An obvious modification to the model would be to drive each density dependent mortality function from the total number of individuals in the population (rather than adults for the adult stage and juveniles for the juvenile stage). The steps required to make this modification are only given in outline here, the details are left as an optional exercise for the reader:

1. Add a new model module (“Expression”) that sums the juveniles and adults to produce a “Total No of Aphids” variable.
2. Replace the independent variable in each density-dependent function with the new variable.

Other modifications also suggest themselves: e.g. do similar numbers of juveniles and adults have the same effects, so should the numbers of one population be “weighted” before the summing. These suggestions are left to the user to explore.

Tutorial 6 - Summary

Timer
See ***Tutorial 3***

MetBase (Meteorological Data)
See ***Tutorial 3***

Circadian (Daily Temperature Cycle)
See ***Tutorial 4***

Lifecycle
Juvenile
Transfer to adult function (Step)
See ***Tutorial 4***
Juvenile development function (Linear above Threshold)
See ***Tutorial 4***
Low Temperature Mortality (Continuous, Linear below Threshold)
Driving Variable: Minimum Temperature
Threshold: 4.0
Slope: -0.05
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
Driving Variable: Total Number of Juveniles
Threshold: 1000
Slope: 0.000001
Output:
See ***Tutorial 4***
Adults
Age-based mortality function (Step)
See ***Tutorial 1***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 5***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
Driving Variable: Total Number of Adults
Threshold: 1000
Slope: 0.000001

Reproduction
See ***Tutorial 1***

Output:
See ***Tutorial 1***

7 Dryness Dependent Mortality

7.1 Introduction

All organisms are sensitive to the absence of water, and the pseudo-aphid is no exception. Survival can be adversely affected if it is too dry, and in this tutorial a first attempt is made to model this sensitivity to dryness using rainfall. In the next tutorial, a better way of modelling the effects of inadequate rainfall will be demonstrated.

7.2 Modelling Dryness Dependent Mortality

As rainfall decreases, the soil dries out to produce plant stress and in this in turn affects pseudo-aphid mortality. In this tutorial, this will be modelled very simply, by assuming that there is a particular level of rainfall each day that is required for no mortality due to dryness to occur. If insufficient rain falls during any one day, mortality on that day is linearly dependent on how far the rainfall is below that threshold. The relationship is illustrated in Figure 7-1. A Threshold of 10 mm\day is chosen – rainfall above that level produces no mortality. The slope of the function is selected so that 25 consecutive days without rainfall would produce an accumulated mortality of about 0.5.
Hence,

$$(1-m)^{25} = 0.5, \quad where \quad m = mortality/day$$
$$i.e., \quad 25 \times \ln(1-m) = \ln(0.5)$$
$$i.e., \quad m = 1 - e^{\frac{\ln(0.5)}{25}} \approx .027$$

Using .025, which is close enough for our purpose, fixes the slope at –0.0025 (the "-" sign is required as the slope is negative, with decreasing rain increasing the mortality rate).

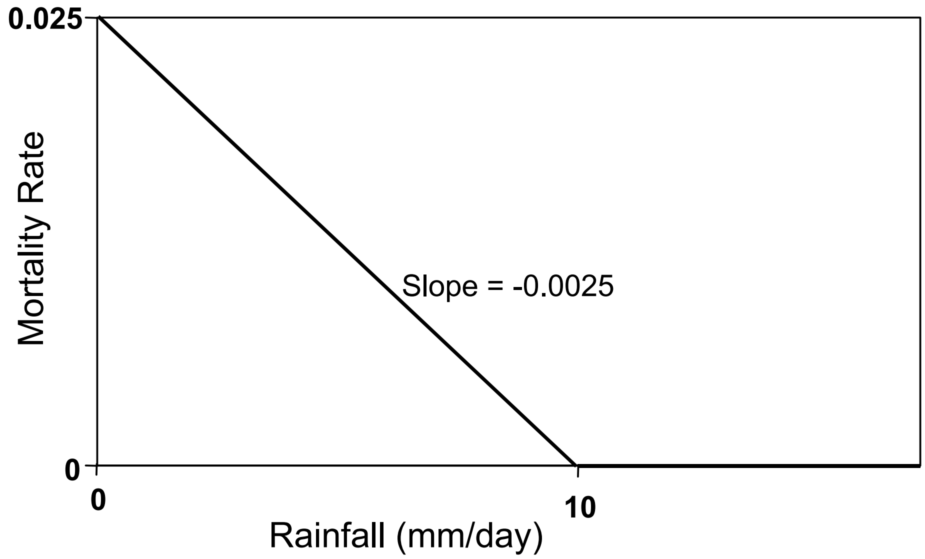


Figure 7-1 Mortality rate as a function of Rainfall

7.3 Modifying the Model

Currently the model does not have a rainfall variable that can be used for the mortality process. However, the "Amberley.dat" file that is being as the source of temperatures also contains a column of rainfall data. The first step, therefore, is to add a rainfall output to the "Meteorological Data" module.

1. Open the DYMEX Builder and load the Pseudo-aphid model
2. Select the **"Meteorological Data"** module for editing
3. Select **"Outputs"** in the "**Meteorological Data"** dialog
4. Select **"Rainfall"** as an output (the characters **"+>"** will appear in front of the name)
5. Return to the **"Model"** window

The new "Dryness Dependent Mortality" components can now be added to both the Juvenile and Adult continuous mortality.

6. Select the **"Lifecycle"** for editing
7. For the **"Juvenile"** lifestage, click on the **"Mortality"** button
8. Click on the **"Continuous"** button, and then select the **"Function"** button in the **"Juvenile Mortality"** dialog to open the **"Function"** dialog
9. Select **"Linear below Threshold"** as the function
10. Click on **"Change"** and set the name to **"Dryness Mortality"**
11. Select **"Rainfall"** as the **"Independent Variable"**
12. Set the **"p1: Threshold"** parameter as follows:
    (a) Use 0, 10, 20 as the lower limit, default and upper limit, respectively

   (b) Type in a suitable name for the parameter (eg "Dryness Threshold")
   (c) Enter a comment or description if desired
13. Set the **"p2: Slope"** parameter as follows:
   (a) Use –0.01, -0.0025, -0.0005 as the lower limit, default and upper limit, respectively
   (b) Type in a suitable name for the parameter (eg "Dryness Threshold")
   (c) Enter a comment or description if desired
14. Click on **"OK"** as often as necessary to return to the "Lifecycle" window.
15. Repeat steps 7-14 for the **"Adult"** lifestage (or use **"Copy"** and **"Paste"**)
16. Save the model.

7.4 Running the modified Model

Start the Simulator and load the pseudo-aphid file. Before any runs are commenced, the model's **"Meteorological Data"** module must be re-initialized to enable it to read the rainfall column of the Amberley data file.

1. In the **"Model Components"** window, click on the **"Meteorological Data"** moduleto obtain the drop down menu
2. Click on **"Initialize Module"** to obtain the **"Data Files"** dialog, where the "Amberley.dat" file will already be selected
3. Click on the "Format" button to obtain the "File Formats" dialog
4. Select the **"Rainfall"** variable from the **"Variables"** list box - it will then be highlighted
5. Place the cursor on the position just under the "i" of Airport in the heading line of the file and press (and keep held down) the left mouse button while 'dragging' the mouse towards the right until 4 columns are highlighted.
6. Verify that the "Start Column" box shows 17, and the "Width" box shows 4
7. Select **"Ok"** to return to the **"Model Components"** window
8. Select **"Lifecycle"** followed by **"Initialize Module"** from the drop down menu to check that the population is 'seeded' with 1000 adults
9. Return to the **"Model Components"** window and run the model for a two year period (730 days) starting on 1/1/1969

To enable the results to be more easily interpreted, the "Population and Temperature" chart format that was saved in the last tutorial will be modified to add a 3rd panel to the chart, with the "Rainfall" variable plotted in that panel.

10. From the "Run" window, select "Chart" (via the button or menu), and select "Population and Temperature" from the popup menu.

This will result in the "Population and Temperature" chart being displayed. To add a "Rainfall" panel:

11. Click on the **"Chart"** item in the menu bar and then select **"Edit Format"** from the drop-down menu
12. Click on the **"Add Panel"** button to create a 3rd panel
13. Select the **"Rainfall"** variable from the **"Model Variables"** list and double-click on it or press the ">Y" button to make it the panel's Y-variable
14. In the **"Series Format"** dialog, select **"Bars"** and de-select **"Include in legend"**
15. Click on **"Save Format"** to save this format, using the name ***"Population and Met. Data"***
16. Click **"OK"** to display the chart.

The chart should appear the same as Figure 7-2. Comparison with the chart from the previous tutorial (Figure 6-3) shows that the peak population size has been reduced by about 25%. The June 1969 peak in particular is lower than the May 1969 peak (i.e., the population has decreased between those generations), whereas the opposite is true for Figure 6-3. Examination of the rainfall lines shows the period between those peaks was very dry.

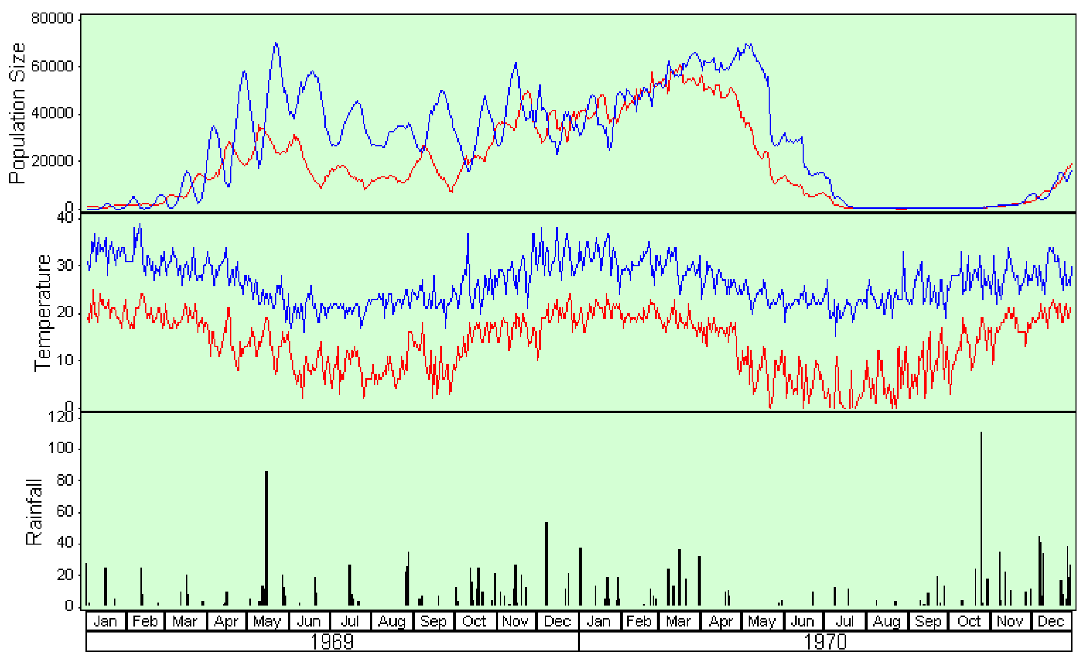


Figure 7-2 Pseudo-aphid population with temperature and rainfall

The sensitivity of the model to the dryness mortality parameters can be examined by re-running the model with the dryness dependent mortality "slopes" for both

Juveniles and Adults set to -0.008. This causes the simulated population to die out completely.

One aspect relating to rainfall that could be modelled is the effect of too much rainfall. For aphid populations in the field, excessive rainfall can cause serious losses either by washing them off the host plant, or by causing physical damage through raindrop impact or by drowning. Excess rainfall mortality can be easily added in a similar way to the rainfall deficiency modelled in this tutorial, but this will be left as an exercise for the reader.

Rainfall is generally not a good driving variable for moisture-related effects on populations. What is important is not the amount of rain that falls on a particular day, but the accumulated effects of rainfall and evaporation. In the Pseudo-aphid's situation, for example, what is important to the insect is a healthy plant. The host plant's moisture condition will in turn be driven by water availability from the soil. A better method is to consider rainfall as a contributor to soil moisture, which can combine the effects of both rainfall and evaporation. This will be the subject of the next tutorial.

Tutorial 7 - Summary

Timer
See ***Tutorial 3***

MetBase (Meteorological Data)
Input: Simulation Date
Outputs: Minimum Daily Temperature
Maximum Daily Temperature
Rainfall

Circadian (Daily Temperature Cycle)
See ***Tutorial 4***

Lifecycle
Juvenile
Transfer to adult function (Step)
See ***Tutorial 4***
Juvenile development function (Linear above Threshold)
See ***Tutorial 4***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 6***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
Driving Variable: Rainfall
Threshold: 10
Slope: -0.0025
Output:
See ***Tutorial 4***
Adults
Age-based mortality function (Step)
See ***Tutorial 1***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 5***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
Driving Variable: Rainfall
Threshold: 10
Slope: -0.0025

Reproduction
See ***Tutorial 1***

Output:
See ***Tutorial 1***

8 Dryness Dependent Mortality using Soil Moisture

8.1 Introduction

Tutorial 7 implied that rainfall by itself may not be the best variable to use in modelling dryness dependent mortality. This is due to the fact that rainfall is rather erratic in nature, but its effects on insects are generally not direct. The soil and plant environment tends to integrate the effects of rainfall over time. In addition, there are the effects of atmospheric moisture (measured as relative humidity or saturation deficit) which may be more appropriate drivers of insect mortality.

Soil moisture is usually the dominant factor determining microclimatic conditions and the moisture content of vegetation. For an insect that lives in close contact with the plant (such as the Pseudo-aphid), soil moisture alone can provide a very useful indicator of the moisture conditions the insect is experiencing. DYMEX contains a module that is designed to simulate the changes in soil moisture with given rainfall and evaporation. The module is described in detail in the Builder User's Guide, which should be referred to at this point. This tutorial will illustrate the use of this module within the Pseudo-aphid model.

8.2 The Soil Moisture Module

The **"Soil Moisture"** module provides an output variable (Soil Moisture) that reflects the simulated current state of the soil's moisture content. The output is scaled between 0 and 1, with a value of 0 indicating a completely dry soil, and 1 denoting a saturated soil. The module requires two variables as input, rainfall and "class A" pan evaporation.

The parameters required for the soil moisture module are: 'Soil Moisture Capacity', 'Evapotranspiration Coefficient' and 'Drainage Rate'. The Soil Moisture Capacity is the maximum water holding capacity of the soil and is normally between 50 and 200 mm. In practice, a setting of about 100-150 mm is typical of many soils. Sandy soils would have a low capacity while its value would be higher for clay soils. The Evapotranspiration Coefficient sets the loss of water from the soil through evaporation and transpiration from plants compared with an equal area of water surface; a value of 0.8 is generally adequate. The Drainage Rate sets the loss of soil water from causes other than evapotranspiration, such as deep drainage. For most purposes, it should be set to 0.

Two variables form the main inputs to the "Soil Moisture" module. These are rainfall and evaporation. Rainfall can be obtained from the "Meteorological

Data" module through the procedures already described for temperature. If a file of evaporation data is available, that variable could be read from the file in a similar way. If such data is not available, DYMEX provides a module that will calculate the evaporation from several inputs. The latter approach will be used in this tutorial.

8.3 Obtaining Evaporation for the Soil Moisture module

The "Evaporation" module requires 5 inputs: minimum temperature, maximum temperature, 9am relative humidity, 3pm relative humidity and daylength. All of daylength are obtained from the meteorological data file using the "Meteorological Data" module and only require that that module be altered so that it is able to read the additional data columns from the file. The other input to the "Evaporation" module is daylength, the number of hours between sunrise and sunset. Again, DYMEX has a built-in module that can calculate this from other variables. The required inputs are: the day of the year (which is available from the "Timer" module) and the latitude which can be user-supplied using a "QueryUser" module. The best way to envisage the process is to examine a schematic diagram (Figure 1-1).

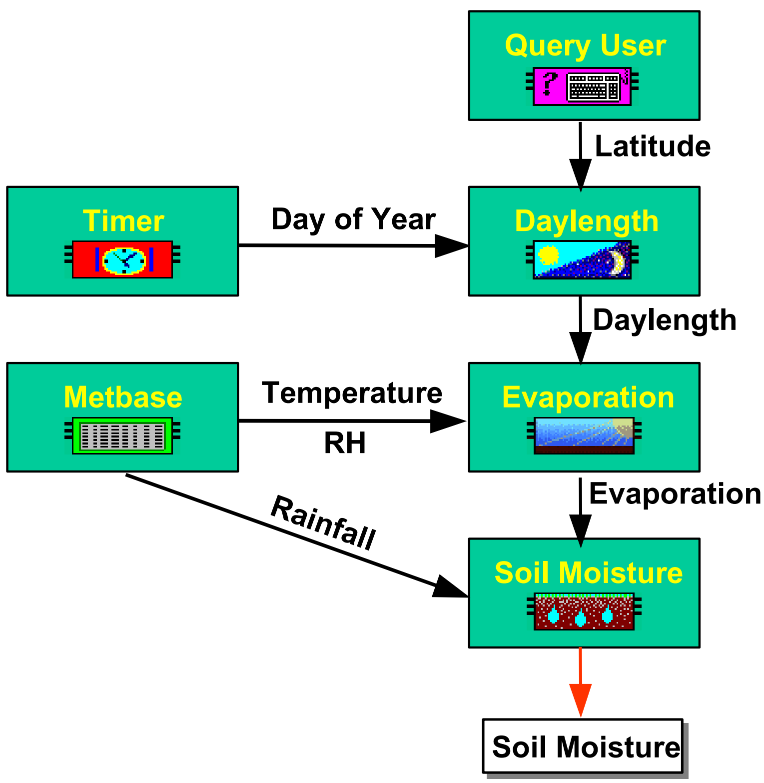


Figure 8-1 Modules involved in calculating the "Soil Moisture" variable

8.4 Modifying the Model

Note: The user's knowledge of already covered DYMEX procedures is assumed. Individual key strokes are only given where new procedures are required.

(1) Adding the module that will provide the "Latitude" value

1. Load the Pseudo-aphid file into the Builder program
2. Select the **"Add Module"** procedure
3. Select a **"QueryUser"** module
4. Rename the module to "Latitude"
5. Click on the **"Outputs"** button, add one output variable, select it and rename it to "Latitude"
6. Set the lower and upper limits to –90 and 90, respectively *(These limit values allow the latitude of any world location to be specified in the Simulator).* Do not specify a default value to force the user to set a value in the Simulator.
7. Set the "Sort Order" of this module to (say) 1, to place it immediately after the Timer.
8. Return to the "Model" window

(2) Modifying the "Timer" module to provide the "Day of Year" output

1. Select the **"Timer"** module for editing and ensure that all 3 output variables (**"Days since Start", "Day of Year" and "Simulation Date"** are selected as output variables
2. Return to the **"Model"** window

(3) Adding and connecting the "Daylength" module

1. Add **"Daylength"** as a new module and name it "Daylength"
2. With the **"Daylength"** edit window open, select the **"Inputs"** button
3. Select as inputs **"Latitude"** and **"Day of Year"** in turn, and link them to the variables of the same name from the right hand list box
4. Return to the Module window
5. Select the **"Outputs"** button
6. Select the output variable from the module and call it "**Daylength"**
7. Set the "Sort Order" of this module to (say) 2, to place it immediately after the "Latitude" module.
8. Return to the **"Model"** window

(5) Add the relative humidity variables to "Meteorological Data" outputs

1. Select the **"Meteorological Data"** module for editing and ensure that **all** variables except Evaporation {temperature (max/min), relative humidity (9am/3pm) and rainfall} are selected as outputs
2. Return to the **"Model"** window

(6) Add the "Evaporation" module

1. Add a new module of type **"Evaporation"** and obtain its editing window
2. Name the module "Evaporation"

3. Link each of its inputs (**Maximum Temperature, Minimum Temperature, 9am Relative Humidity, 3pm Relative Humidity** and **Daylength**) to the appropriate variable
4. Select the **"Output"** button and select **"Evaporation"** to set it as the output variable
5. Set the "Sort Order" of this module to place it immediately after the "Meteorological Data" module
6. Exit back to the **"Model"** window

(7) Add the "Soil Moisture" module

1. Select the **"Soil Moisture (1-layer)"** module as a new module, and name it "Soil Moisture"
2. Link each of its inputs (**Rainfall and Evaporation**) to the appropriate variable
3. Select the **"Output"** button and select **"Soil Moisture"** to set it as the output variable

4. Set the **"Sort Order"** of this module to place it immediately after the "Evaporation" module
5. Click on the **"Factors"** button in the Module window

There are three soil moisture "factors" (the 3 parameters discussed in Section 8.2), but all will be specified as "Parameters" so that no functions need be selected.

6. Select **"Soil Moisture Capacity"** and then click on the **"Set Parameter"** button
7. Set the default to 100 and the lower and upper limits to 50 and 150, respectively
8. Select "**Evapotranspiration Coefficient"** and the click on the **"Set Parameter"** button
9. Enter the values of 0.8, 1.5 and 0.3 for the default, upper and lower limits, respectively
10. Select "**Drainage Rate"** and the click on the **"Set Parameter"** button
11. Set the default and limits to 0
12. Exit to the **"Model"** window

There will now be 8 modules shown in the "Model" window. The user is reminded that the small boxes containing the "+" symbol in allow the corresponding component diagram to be expanded to show more detail.

The **"Lifecycle"** module presently uses rainfall as the input variable that controls dryness dependent mortality. This will now be changed so that soil moisture levels control this mortality, but the "Linear below Threshold" relationship will be retained. The parameters will be chosen so that dryness dependent mortality commences to have an effect once the soil moisture level drops to 0.4, and 10 days at a soil moisture level of 0 will produce an accumulated mortality of 0.5.

Using the same method for calculating the slope of the function as used in the rainfall case (Section 7.2), gives a value of about –0.168 for the slope.

1. In the "Lifecycle" window, select the Juvenile stage's **"Mortality"** button and then the **"Continuous"** button
2. Select the **"Dryness Mortality"** process component
3. Change the Independent Variable to **"Soil Moisture"**
4. For the **"p1: Dryness Threshold"** parameter, change the default to 0.4 and the lower and upper limits to 0 and 1, respectively
5. For the **"p2: Dryness Slope"** parameter, set the default to –0.168 and the lower and upper limits to –0.4 and 0, respectively
6. Amend or add any desired comments
7. Exit to the **"Lifecycle"** window and then **repeat the process for the Adult lifestage**
8. Exit and save the lifecycle.

8.5 Running the modified Model

1. Load the Simulator and open the pseudo-aphid file - the **"Model Components"** window should resemble Figure 8-2.

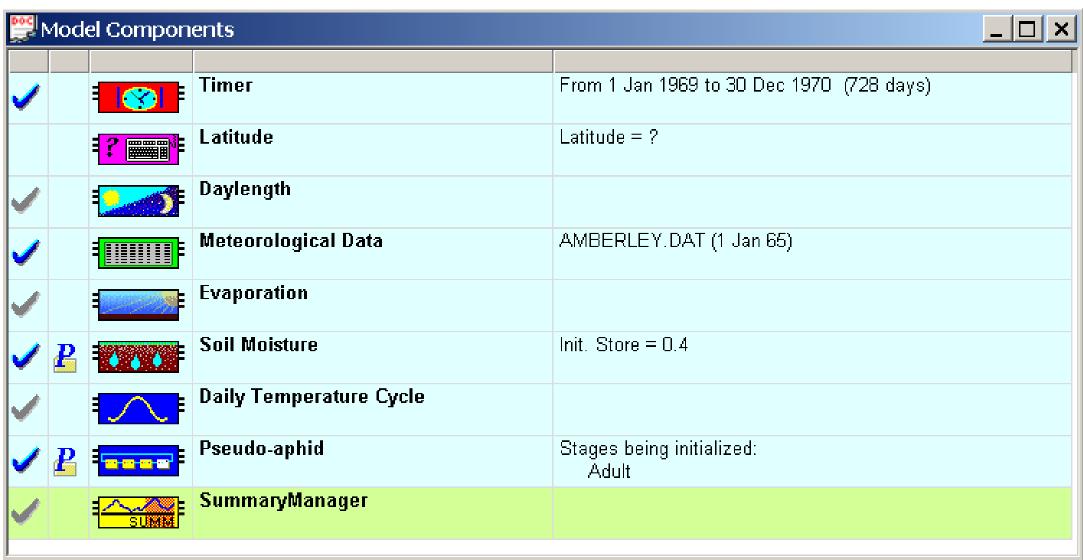


Figure 8-2 The Pseudo-aphid model incorporating Soil Moisture

2. Initialize the **"Latitude"** module by setting the latitude to –27.6 (the latitude of Amberley, with the '-' sign indicating that it is south of the equator)
3. Initialize the **"Meteorological Data"** module by setting the column parameters for 9am (column 31, width 4) and 3pm (column 45, width 4) relative humidity

4. Initialize the **"Soil Moisture"** module by setting the initial store to 0.5. This number is the soil moisture value at the start of the simulation.
5. Run the model for two years from 1/1/1969 and create a chart consisting of 4 panels, as detailed below:
   - Panel 1: "Total Numbers of Juveniles" and "Total Number of Adults"
   - Panel 2: "Maximum Temperature" and "Minimum Temperature"
   - Panel 3: "Rainfall"
   - Panel 4: "Soil Moisture"

*Hint: Click on **Chart**, and select the **"Population and Met. Data"** chart saved in the previous tutorial. Then select "Edit Chart" from the "Chart" menu and in the resulting "Chart Specification" dialog, add the "Soil Moisture" panel. Save the new chart format using the same name.*

When this is done, the results should be similar to Figure 8-3. Note in particular the soil moisture changes, and how they are affected by rainfall. The soil moisture content increases immediately after a rainfall event and then drops slowly as evaporation and transpiration deplete the soil moisture store. Evaporation is higher in summer and the rate of depletion is therefore larger. Note also how the soil moisture approaches 0 asymptotically.

The user may wish to experiment with various settings of this model to become familiar with the "behaviour" soil moisture model. In particular, the "Soil Moisture Capacity" and "Evapotranspiration Coefficient" parameters could be changed to simulate sandy and clay soils with little or heavy vegetation cover.

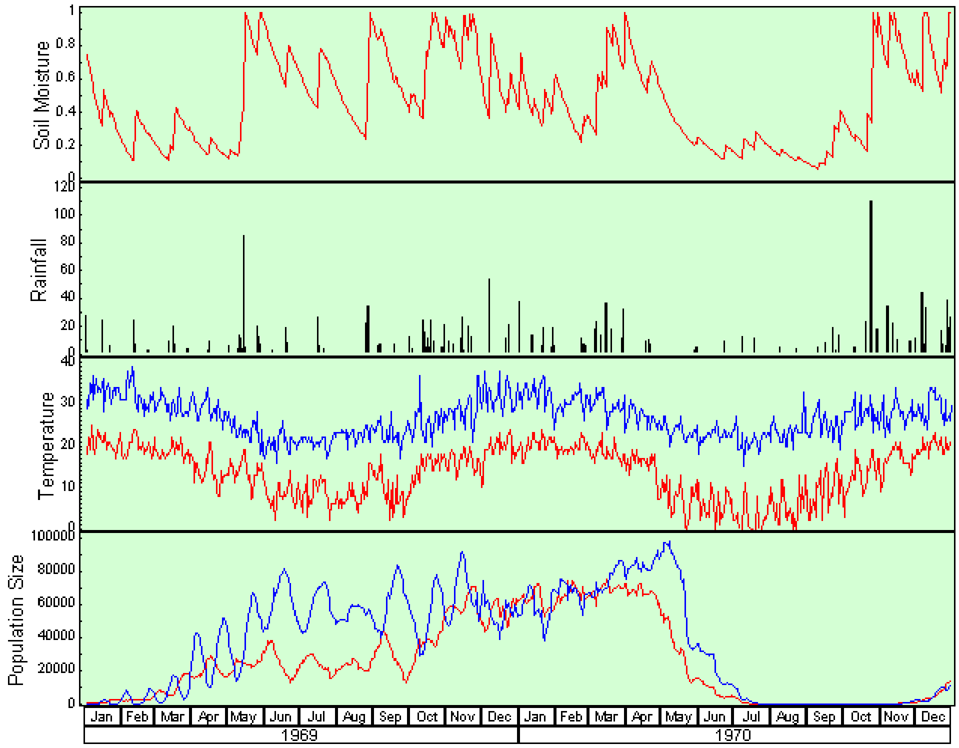


Figure 8-3 Pseudo-aphid population with temperature, rainfall and soil moisture

Tutorial 8 - Summary

Timer

Inputs:	none
Outputs:	Days since Start
	Day of Year
	Simulation Date
Settings:	Timestep 1 day

QueryUser (Latitude)

Outputs:	Latitude (Range: -90 to 90)

Daylength

Inputs:	Latitude
	Day of Year
Output:	Daylength

MetBase (Meteorological Data)

Input:	Simulation Date
Outputs:	Minimum Daily Temperature
	Maximum Daily Temperature
	Rainfall
	9am Relative Humidity
	3pm Relative Humidity

Evaporation

Inputs:	Minimum Daily Temperature
	Maximum Daily Temperature
	9am Relative Humidity
	3pm Relative Humidity
	Daylength

Soil Moisture

Inputs:	Rainfall
	Evaporation
Output:	Soil Moisture

Circadian (Daily Temperature Cycle)

See ***Tutorial 4***

Tutorial 8 – Summary (continued)

Lifecycle

Juvenile

Transfer to adult function (Step)
See ***Tutorial 4***
Juvenile development function (Linear above Threshold)
See ***Tutorial 4***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 6***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
Driving Variable: Soil Moisture
Threshold: 0.4
Slope: -0.168
Output:
See ***Tutorial 4***

Adults

Age-based mortality function (Step)
See ***Tutorial 1***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 5***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
Driving Variable: Soil Moisture
Threshold: 0.4
Slope: -0.168

Reproduction
See ***Tutorial 1***

Output:
See ***Tutorial 1***

9 Adding an 'Event' Module

9.1 Introduction

Mortality in the Pseudo-aphid model is now affected by several variables that are dependent upon either climatic conditions or intrinsic properties of the population (i.e., population size).

Human induced mortality, as caused by management actions, can also be added to the model. Assuming that the Pseudo-aphid is destructive on its host plant and that the host plant is economically useful, an agriculturalist's problem resolves itself into either reducing population numbers to acceptable levels or eradicating the organism completely. There are many ways that this may be done, from chemical spraying to perhaps introducing a biological control such as a parasite. Though DYMEX provides facilities to model most of the available options, in this tutorial only the effect of a chemical spray is considered.

Management options in DYMEX are modelled using an "Event" module. The Builder User's Guide describes this module in considerable detail. In brief, the "Event" module is able to simulate the occurrence of actions that are specified to occur on particular dates, or in response to given conditions. An event could be either a natural occurrence or human induced. Examples of events are the application of a spray, a fire, ploughing, heavy rainfall, or sudden loss of food. How DYMEX is used to model an event depends completely on how the user wishes to apply it.

9.2 The "Event" Module

The Event module has several inputs and two outputs. For the Pseudo-aphid model only a single output will be used, and this output reflects the effect of the event (Figure 9-1). Two of the inputs to the module are the by now familiar "Timer" module variables: "Day of Year" and "Simulation Date". The "Event" module uses these inputs to synchronize the occurrence of events with the rest of the model. The "Threshold" input is an optional input that is required only if some model variable is used to trigger the occurrence of events. For example, to simulate the effect of a farmer spraying aphids only when he "sees" them, the "Total number of Aphids" could be linked to that input so that spraying takes place whenever that variable exceeds, say, 100. For this tutorial, the "Threshold" input will not be used and spraying will be scheduled to occur on a calendar basis.

The output of the module is the "effect" of the event. Normally, this is 0 until an event action is triggered (for example, the date of a scheduled pesticide application occurs), when the output takes on the value determined by its parameter or function.

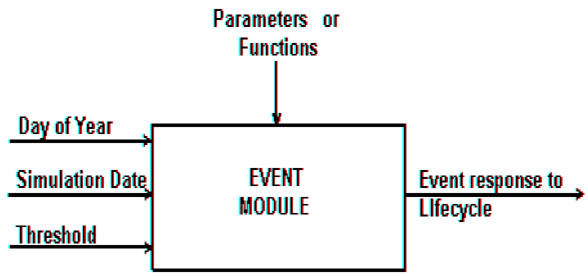


Figure 9-1 The "Event" module

Since, for this model, the event will be the administration of a pyrethrin spray, the parameter or function will determine the mortality rate for the pseudo-aphid. Users will appreciate that a spray application and its consequent effects are determined by quite complex interactions. Among these are spray concentration, wind-drift, tolerance of the population, lessening effects over time, etc. To model these in detail, a series of additional functions or modules may be needed in addition to the "Event" module. For simplicity, the Pseudo-aphid will be considered to suffer a 90% mortality on initial application of the spray followed by an exponential decay of the spray effects over the following 2 days. The output from the "Spray" Event module will then be used to drive mortality in both the juvenile and adult stages of the aphid. Figure 9-2 shows the pre-defined DYMEX function "Exponential Decay" which will be used to simulate the decay in effectiveness of the insecticide. The function includes a parameter that specifies a time after which the exponential decay starts (the "Duration of maximum effectiveness"). For this example, it will be set to 0 – i.e., the insecticide is assumed to start losing effectiveness immediately.

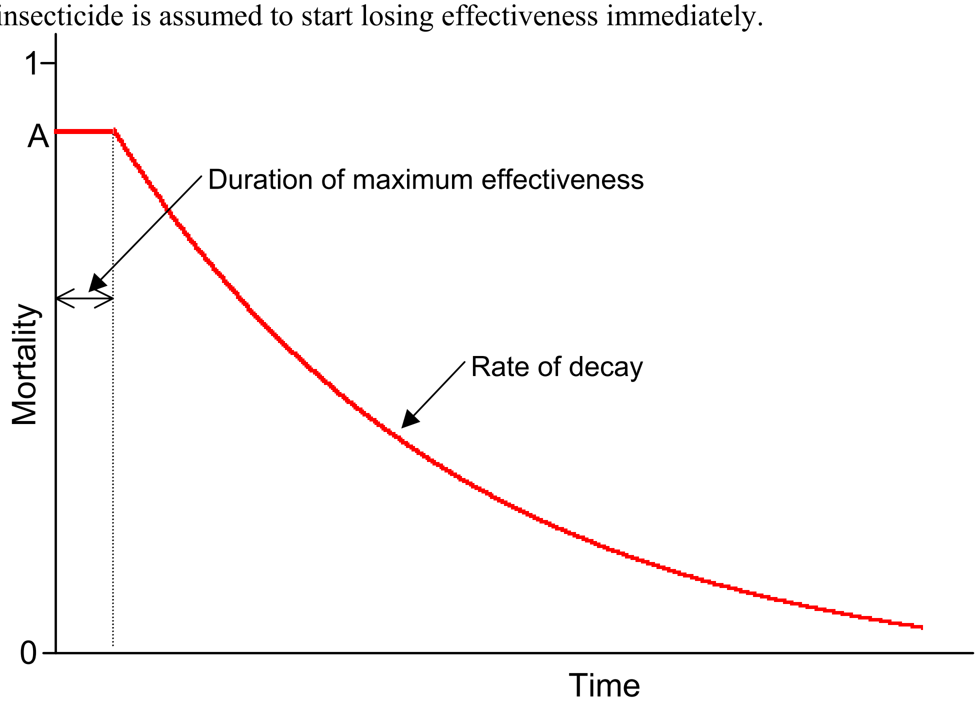


Figure 9-2 Mortality effects for the pyrethrin spray

The rest of the curve is an exponential decay, described by the equation

$$\mathbf{y = Ae^{-kT}}$$

In this equation, **T** is the time (in days) after insecticide application, **y** is the mortality, **A** is the maximum mortality (on the day of application) and **k** determines how quickly the effectiveness decays. As indicated earlier, the parameter **A** will be set to 0.9 (90% mortality). An exponential decay curve approaches, but never quite reaches, 0. In order to calculate the parameter **k**, it is assumed that after 2 days have elapsed, the spray-induced effects on the pseudo-aphids will be 5% or 0.05 mortality. If these values are substituted into the equation, we have:

$$\mathbf{0.05 = 0.9e^{-k.2}}$$

Dividing both sides by 0.9, and taking their logarithms produces the result:

$$\mathbf{\ln 0.0556 = -2k}$$

Which in turn produces the equation:

$$\mathbf{-2.89 = -2k}$$

Therefore:

k = 1.45 (approximately)

9.3 Modifying the Model

Two changes need to be made to the model to add the new functionality. Firstly, and event module needs to be added and set up to produce an effect as defined in the previous section whenever required. Secondly, alterations are required to the Lifecycle module so that it reacts to the Event module. A new mortality component will be added that depends on the “Event” module’s effect variable.

1. Load the Model Builder and open the Pseudo-aphid file
2. Select **“Model”** from the menu-bar and add an **“Event”** module (make sure to use an **Event** module and not one of its varieties, such as **Event1**)
3. Re-name the module to “Insecticide Treatment”
4. Change its “Sort Order” so that it is placed just before the Lifecycle module
5. Select the **“Inputs”** button and obtain the link window (Figure 9-3)
6. Link the input **“Day of Year”** with the variable of the same name and then repeat this for **“Simulation Date”**

The operations of Step 7 ensure that the “Insecticide Treatment” module is correctly synchronized with the “Timer” module when the model is run in the

Simulator. The other two variables do not need to be linked, as they will not be used in the tutorial.

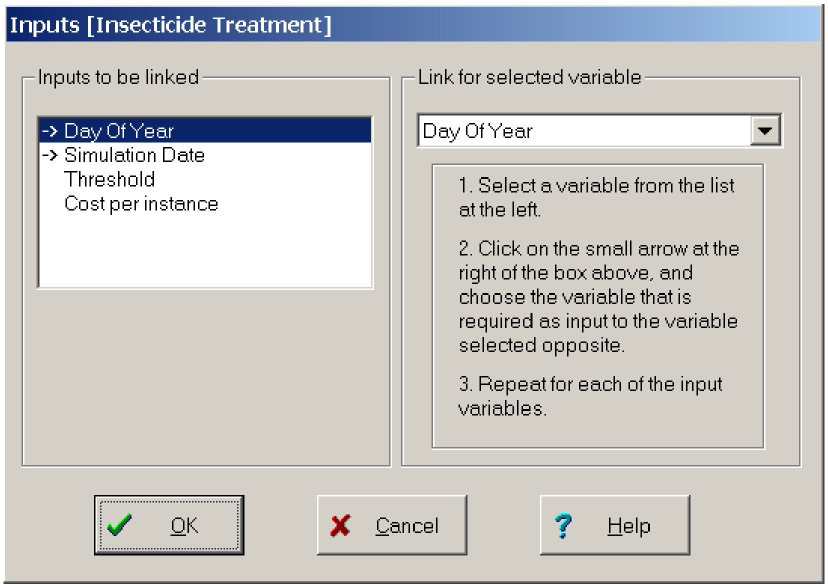


Figure 9-3 Event "Inputs" window

7. Return to the **"Event"** module window
8. Click on the **"Outputs"** button to obtain the **"Output Variables"** dialog box
9. Select **"Event Variable" (+>)** and rename it **"Spray Application"**
10. Return to the "**Event"** module window
11. Select the **"Factors"** button to obtain the **"Factors"** window (Figure 9-4)

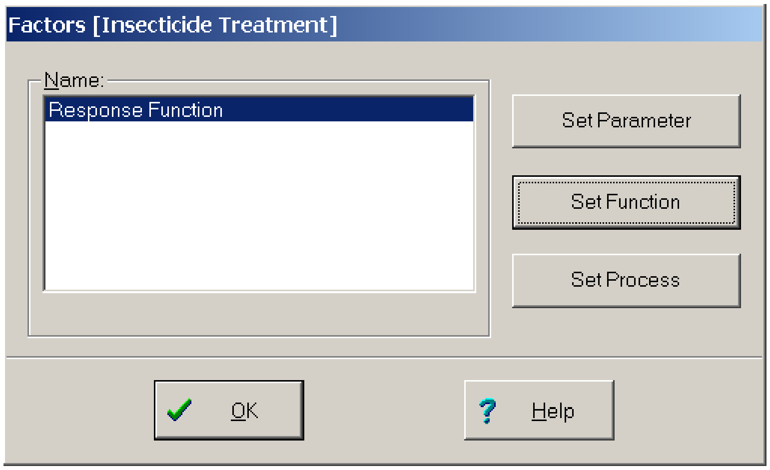


Figure 9-4 The "Factors" window for the Event module

12. Select the **“Set Function”** button to obtain the **“Function”** edit window
13. Select **“Exponential Decay”** as the function
14. Set the **“Independent Variable”** to **“Days since Event”**

The variable “Days since Event” is an internal variable of the “Event” module, and at any time contains the number of days that have elapsed since the last time an event action occurred.

15. Select the **“Parameters”** button
16. For the parameter **“p1: Threshold”**, set the lower and upper limits, as well as the default, to 0. This parameter is the “Duration of maximum effectiveness” shown in Figure 9-2 and should be renamed as such
17. For the parameter **“p2: Decay Constant”**, set the lower and upper limits to 0.5 and 2, respectively, and the default to 1.45. Rename it to “Insecticide Decay Constant”
18. The 3rd parameter, **“p3: Scaling Factor”**, is the maximum mortality (**A** in Figure 9-2). Set the values to 0.5, 1.0 and 0.9 for the lower limit, upper limit and default, respectively. Rename to “Maximum Effectiveness”
19. Return to the “**Event”** module window
20. Click on the **“Settings”** button obtain the “Event” settings dialog (Figure 9-5)

As noted earlier, an “exponential decay” function never actually reaches 0. In this dialog, a definite cutoff of 4 days will be specified for the insecticide effect, i.e., after 4 days, the effect of the insecticide treatment will be 0. We need also to specify what happens if treatments are applied so close together that their effect curves overlap. In the case of our insecticide, the effects will combine so that the total effect is

Total Effect = 1 – (1 – Effect1) × (1 – Effect2) …

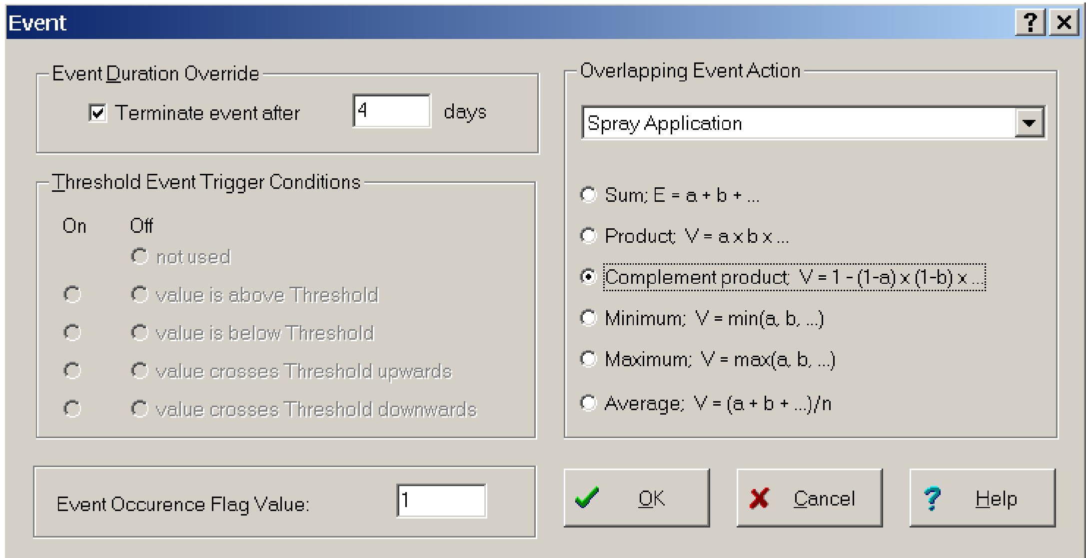


Figure 9-5 The "Event" settings dialog

21. Select the **"Event Duration Override"** checkbox, and enter 4 into the edit box, to terminate the event after 4 days.
22. Make sure **"Complement product"** is selected in the **"Overlapping Event Action"** panel
23. Exit to the **"Model"** window and save the model.

The Lifecycle module must now be altered to use the "Spray Application" variable from the Event module to cause mortality. The sequence of steps detailed below must be completed for both the juvenile and adult lifestages in the module. The "Direct" function that is used to couple the "Spray Application" variable to the mortality does not have any parameters (it is the function y = x).

1. Open the **"Lifecycle"** module for editing
2. Select the Adult lifestage **"Mortality"** button
3. Open the **"Continuous Adult Mortality"** dialog window
4. Select the **"Function"** button to add a new mortality factors
5. In the **"Function"** dialog box select **"Direct"** as the function template
6. Select **"Spray Application"** as the independent variable
7. Return to the **"Lifecycle"** window
8. Repeat steps 2-7 for the "**Juvenile"** lifestage
9. Save the model.

9.4 Running the modified Model

1. Start the Simulator and load the Pseudo-aphid model
2. In the "**Model Components"** window, select the "**Insecticide Treatment"** module

3. Open the **"Initialize Module"** from the drop-down menu

The Event Calendar dialog lists the schedule of Events for the simulation run. Currently the "Event Schedule (triggers)" list is empty, indicating that no insecticide treatments are yet specified.

4. In the **"Event Calendar"** window select the **"Add Date"** button to open the **"Specify Event date/s"** dialog (Figure 9-6)
5. Use the "**Specify Event Date(s)"** edit box to set 14 March as the spraying date (it will show the date and that there are to be no repeats)
6. Exit to the "**Model"** window and run the model for 2 years. Use the same starting date as in the previous tutorials (1/1/1969)

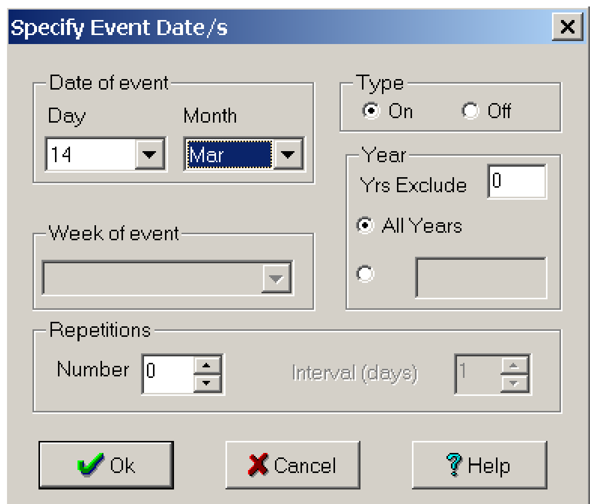


Figure 9-6 The "Specify Event Date/s" dialog

Create a chart of the model output as follows (and save its format as "Population and Spray" for subsequent tutorials):

- Panel 1: "Total Numbers of Juveniles" and "Total Number of Adults"
- Panel 2: "Spray Application"

The resulting chart (Figure 9-7) shows clearly the effect of the insecticide treatments on both the juvenile and adult populations. The output of the "Insecticide Treatment" module is shown in the top panel. Comparison of the population numbers with the untreated numbers of Figure 8-2 shows that the single treatment per year is making little difference to the aphid population. The reader may wish to experiment with adding more treatments to the "treatment calendar", or changing the effectiveness and duration of each treatments.

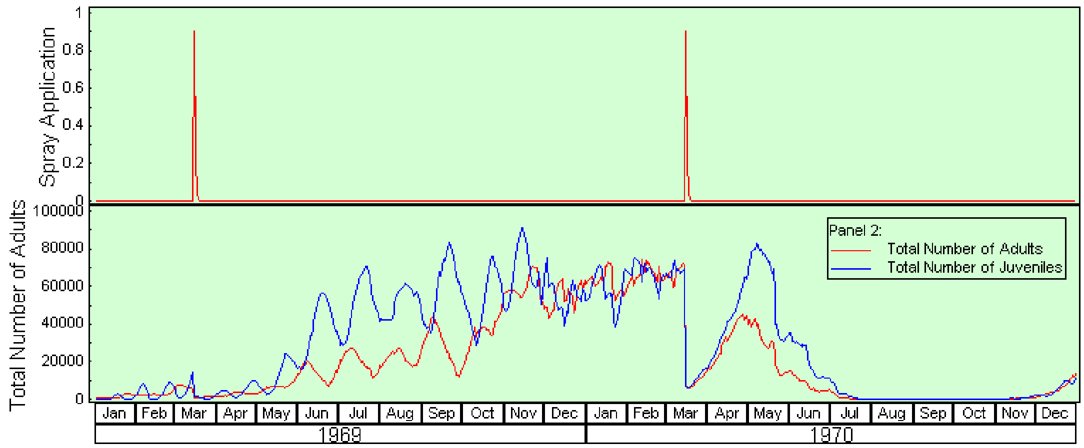


Figure 9-7 Chart showing aphid population with one insecticide treatment per year

Tutorial 9 - Summary

Timer
See ***Tutorial 8***

QueryUser (Latitude)
See ***Tutorial 8***

Daylength
See ***Tutorial 8***

MetBase (Meteorological Data)
See ***Tutorial 8***

Evaporation
See ***Tutorial 8***

Soil Moisture
See ***Tutorial 8***

Circadian (Daily Temperature Cycle)
See ***Tutorial 4***

Event (Insecticide Treatment)
Inputs: Day of Year
Simulation Date
Output: Spray Application
Factor:
Function: Exponential Decay
Independent Variable: Days since Event
Threshold: 0
Decay Constant: -1.45
Scaling Factor: 0.90

Tutorial 9 – Summary (continued)

Lifecycle

Juvenile

Transfer to adult function (Step)
See ***Tutorial 4***
Juvenile development function (Linear above Threshold)
See ***Tutorial 4***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 6***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
See ***Tutorial 8***
Insecticide Application Mortality (Continuous, Direct)
Driving Variable: Spray Application
Output:
See ***Tutorial 4***

Adults

Age-based mortality function (Step)
See ***Tutorial 1***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 5***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
See ***Tutorial 8***
Insecticide Application Mortality (Continuous, Direct)
Driving Variable: Spray Application

Reproduction
See ***Tutorial 1***

Output:
See ***Tutorial 1***

10 Making the Spraying 'Event' Cost Effective

10.1 The "Event" Threshold

The previous tutorial used a particular date (or day of the year) as the trigger for the spraying event. The date was chosen by looking at the graphical display of the pseudo-aphid numbers and selecting a time when the numbers appeared to be rapidly increasing. This represents an approach denoted as "strategic" spraying, where the decisions on when to apply control methods are made in advance, based on a knowledge of the species' biology and other factors (such as cost and availability of labour).

This strategic approach is useful, but has some shortcomings in practice. The spray will be completely wasted if the pest is not present in the area at the time of spray application or the spray application may occur at a particular part of the insect's lifecycle where the spray is largely ineffective. Alternatively, the spray may kill the insect pest, but the damage to the crop may already be complete or a very mild winter may have so altered flowering, fruit formation and insect attack that all dates are completely out of step with the normal season. Even carefully controlled, large scale agriculture (for example, cotton or fruit production) cannot use set dates for spraying because the crop may alter its development in accordance with the seasonal climate. In practice, the farmer/horticulturalist would generally use a more "tactical" approach to pest control, where decisions on spraying are always made after examination of the state of the crop and its pests.

If we consider the field practices of an actual gardener or horticulturalist, many spray applications are made when the population of the insect pest is sufficiently great to suggest that "something had better be done or damage to the plants will result". If roses are the target plant and aphids are the pests, application of a spray is usually made when the gardener/horticulturalist sees that the stems of the rose buds are crusted with insects. In this case, the threshold for spraying is visual recognition that the level of aphids exceeds a threshold. In DYMEX, this is achieved by linking the triggering of the treatment event to a variable that reflect the pest population size. This will now be done for the pseudo-aphid.

10.2 Linking the Adult Population and Event Threshold

1. Start the DYMEX Builder and load the Pseudo-aphid model
2. Open the **"Insecticide Treatment"** module
3. Select the **"Input"** button to open the **"Inputs"** link dialog
4. Link **"Threshold"** with **"Total number of Adults"**
5. Return to the **"Insecticide Treatment"** module window

6. Click on **"Settings"** and verify that the **"value is above Threshold"** option is selected as the "**<u>On</u>**" **"Threshold Event Trigger Condition"** (the **"<u>Off</u>"** condition should be set to **"not used"**
7. Exit back to the **"Model"** window and save the model

10.3 Creating a Run Summary for comparison of results

In previous tutorials, the model has been run and tables or charts have been compared from one run to the next as a basis for judging differences in population size. For evaluating the effectiveness of different control options, a better method is required. This is provided in DYMEX by allowing the user to specify variables whose values will be summarized over the period of a run or over the last year, as selected. For example, the total number of aphids could be calculated over the whole of a simulation, and this could be a rough measure of the damage done to the host plant by that pest. The aim of optimization would then be to minimize that total with the fewest treatments.

Run summaries are set up in the Builder program, and are stored in the '.gmd' file so that they are available for all subsequent runs. To create a run summary that will report the <u>average</u> values of each of the output variables "Total Number of Juveniles" and "Total Number of Adults" over the full run:

1. From the **"Model"** menu, select **"Summary Variables"** to obtain the **"Summary Variables"** dialog box
2. Select the "Add" button to add a new summary variable.
3. From the "Variable to summarize" drop-down box select **"Total Number of Adults"**
4. Type **"*Average Adults*"** as the summary variable's name into the **"Name" box**
5. Select **"Average"** as the **"Summary Variable Statistic"**
6. Select the **"Whole Run"** as the **"Summary Period"**
7. Supply an optional description for the variable (for example, "*The average number of adult bugs present per day over the whole simulation period*")
8. Highlight **"Total Number of Juveniles"** and then select the **"Add Variable to Summary List"** button
9. Place the cursor on the **"Total Number of Adults"** item in the **"Selected Variables"** box, and click the left mouse button once
10. Select **"OK"** to leave the dialog
11. Repeat steps 2-10 for **"Total Number of Juveniles"**, calling the summary variable **"*Average Juveniles*"**
12. Click on **"OK"** to return to the **"Model"** window

10.4 Running the "Threshold Spraying" Model

To obtain a basis for comparison for later sections of this tutorial, the model will be run initially with no insecticide treatment. A few other changes will also be made to module settings in the Simulator and then used for the rest of the current tutorial, as follows. The "Lifecycle" initialization will be changed to simulate immigration of individuals from outside into the population. Since the adult stage is the winged stage, immigration will be simulated by introducing 10 individuals into the population every week and the "Timer" initialization will be changed so that a run starts "at the beginning" (1/1/1965), and has a duration of 1460 days (4 years).

1. From the **"Model Components"** window, select the "Lifecycle" module and then **"Initialize Module"** from the popup menu
2. Click on the **"Adult"** stage in the top panel, and then on the initialization details line shown in the bottom panel
3. Select **"Delete"** to remove the initialization set from the lifestage
4. Click on **"New"** to open the **"Edit Lifestage Initialization Set"** dialog
5. Enter 10 as the number of individuals to add at the start, and, in the **"Repetitions"** panel, specify 208 as the **"Number"** and 7 as the **"Interval"** (i.e 10 individuals will be introduced weekly for 4 years)
6. Return to the **"Model Components"** window
7. Select the "Timer" module, and its **"Initialize Module"** menu item to set the run to start on 1/1/1965 and have a duration of 4 years (1460 days)
8. Select the "Insecticide Treatment" module's **"Initialize Module"** menu item to remove any treatments that may currently be set

The model simulation can now be started in the usual way. When completed, note that the Run window that appears contains the requested Run Summary, giving the average juvenile and adult population over the 4 years (Figure 10-1).

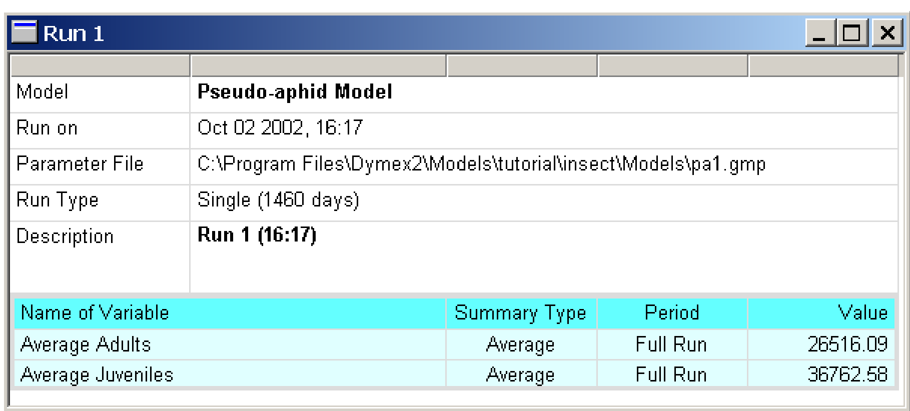


Figure 10-1 Run window showing "Run Summary"

Click on the "Chart" button or menu item and select the "Population and Spray" chart from the popup menu (the format was saved in the previous tutorial). The chart should appear as in Figure 10-2.

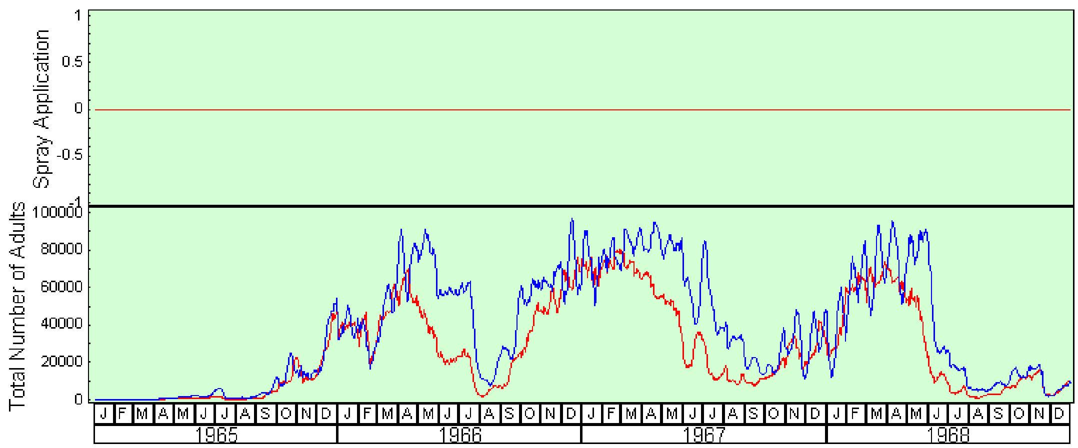


Figure 10-2 Aphid population over 4 years with no treatment

Assuming the visual threshold for spraying is used and that this is linked to the numbers of adult insects present, the only extra information required is the number of insects that will provide the visual trigger. Actual counts of rose aphids, suggests that their presence is very obvious to the gardener if about 30-60 mature aphids are present on any one stem. If it is assumed that a normal, healthy rose bush will have at least 10 bud/flower/new shoot stems, the number of pseudo-aphids on the rose bush that will cause the farmer to spray can be arbitrarily set at 500. To set this condition in DYMEX, proceed as follows:

1. In the Simulator, select the **"Insecticide Treatment"** module followed by **"Initialize Module"** to open the **"Event Calendar"** window
2. Select the **"Set Threshold"** button to open the **"Specify Threshold Value"** dialog box
3. Set the threshold value to <u>500</u>, the minimum days between events to <u>7</u> (this means that the farmer cannot spray at intervals less than 7 days) and leave repetitions at 0
4. Exit to the **"Model Components"** window;
5. Run the model for as before, and produce a "Population and Spray" chart

The first things to note are the average juvenile and adult populations, as shown in the "Run" window. These are now around **<u>275</u>** and **<u>174</u>**, respectively – a large reduction from the equivalent numbers of the untreated population (Figure 10-1).

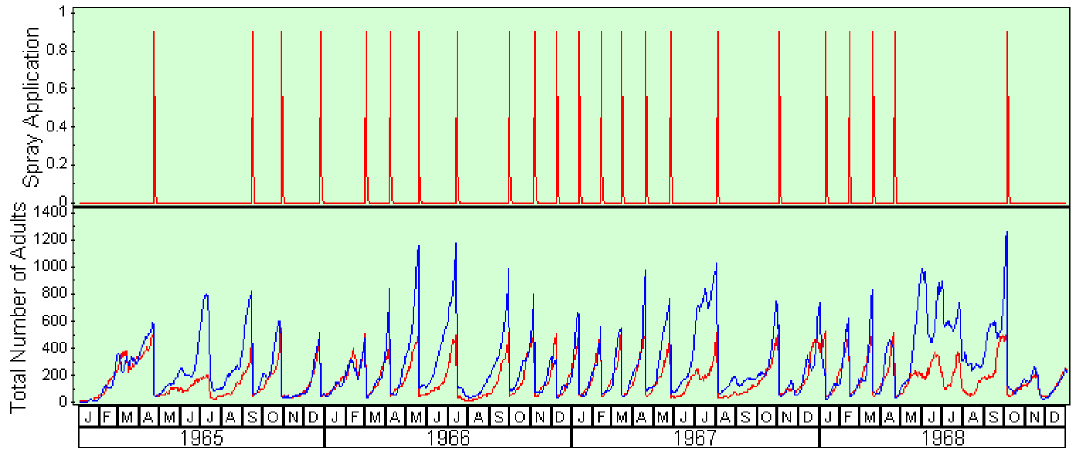


Figure 10-3 Aphid population over 4 years with threshold spraying

The graph confirms these numbers, and shows that this was achieved with a total of 23 sprays (Figure 10-3). The pseudo-aphid population is now relatively well controlled, but the situation may still be unacceptable for several reasons.

- The delay between when populations thresholds are detected and the spray (one day in the model), may lead to unacceptable levels of damage
- There may be unnecessary work for the spray user
- The cost of the spray may be prohibitive
- Overuse of any spray is not desirable environmentally.

The results of Figure 10-3 lead directly into a consideration of how to best utilize a spray treatment in order to obtain results comparable to the control of threshold sprays, but with mimimal use of the spray and therefore as little cost as possible to the user. DYMEX contains a facility (a "Run Sequence") that can help in providing this information.

10.5 Setting up a "Run Sequence"

A "Run Sequence" is a series of runs over the same period, but where some variable, parameter or other condition is varied in a systematic way. At the end of this series of runs, the "Run Summary" variables for each run in the series are available for tabulation or charting. In order to control the pseudo-aphid population, insecticide spraying has to be carried out, however the timing of the spray can be important for it to be most effective. Since pseudo-aphid populations rise and fall with the seasons (Figure 10-2), a simple way to find out where a single application of spray would do most good would be to provide a series of treatments, and monitoring the effect of changing the starting time of these treatments. For example, the farmer may want to try 5 sprays, spaced 1 week apart. The treatment regime could be started on 1/1 each year, then 8/1, etc. This would entail 52 runs of the program with the user then comparing the effects of each starting time on the average population. The operation could be done manually by the user, however DYMEX is able to do this automatically using a "Run Sequence".

To proceed, the exact "Run Sequence" must first be defined:

1. Load the Simulator and open the Pseudo-aphid file

*Note the drop-down selection buttons (**"Quick Selector"** buttons) that have appeared in the "Component Window" for the "Meteorological Data" and "Insecticide Treatment" modules. Once summary variables have been defined, these buttons are present for all modules that can be used to define **Run Sequences**.*

2. Click on the Insecticide Treatment Quick Selector button and select the **"(Add/edit/remove list item)"** choice from the resulting menu to open the **"Sequences (Insecticide Treatment)"** dialog
3. Click on **"New"** to obtain the **"Event Sequence – Insecticide Treatment"** dialog box
4. Name the sequence **"5 Treatments @ 1 wk intervals"**
5. From the **"Sequence Variation Type"** panel, select **"Vary Starting Date, fixed No. of Actions"**
6. Each spray event consists of 5 treatments, so set **"No of actions in group"** to 5 in **the "Base Event Definition"** panel
7. Set the **"Action Spacing (in days)"** to 7
8. Set the **"Starting Day-of-Year"** to 1 for the **"Base Event"** (i.e., the first event in the sequence
9. In the **"Run Sequence Definition"** panel, set the increment in starting day between runs in the sequence to 7 (1 week). This tells the program to shift the event forward by seven days for each run in the sequence.
10. Set the **"No. of runs in sequence"** to 52 to cover the whole year.
11. Set **"Initial Years to exclude"** to 0
12. Exit back to the **"Model Components"** window

Note that row of the Component Window that represents the **Insecticide Treatment** module has changed to a brown colour, and the name of the Sequence is now shown as the setting for that module.

10.6 Optimising treatments using a "Run Sequence"

The "Run Sequence" created in the previous section can now be used to automatically run the model multiple times.

1. At the "Model Components" window, select **"Run"** to obtain the **"Run Model"** window

Note that a previously unavailable selection is now available for use: "Run Type".

2. Note that the **"Multiple"** button in the **"Run Type"** panel of the dialog is automatically selected along with the "**5 Treatments @ 1 wk intervals"** Run Sequence
3. Ensure the "**Simulation Period"** is set to run for 4 years starting on 1/1/1965
4. Click on **"Ok"** to start the run.

WARNING: The run sequence now has to process 52 runs of 4 years each. This will take some time, depending on you computer.

At the completion of this series of runs, the "Run" window appears as usual, this time without a summary. The "Run Summary" variables (one set per run) can be examined using either a table or chart. The chart shown in Figure 10-4 shows the average juvenile and adult numbers for each run, with the starting week of treatments shown on the X-axis. (The chart was created using "Run" as the X-axis variable, with 1 panel, and manual scaling and labels for the X- and Y-axes.)

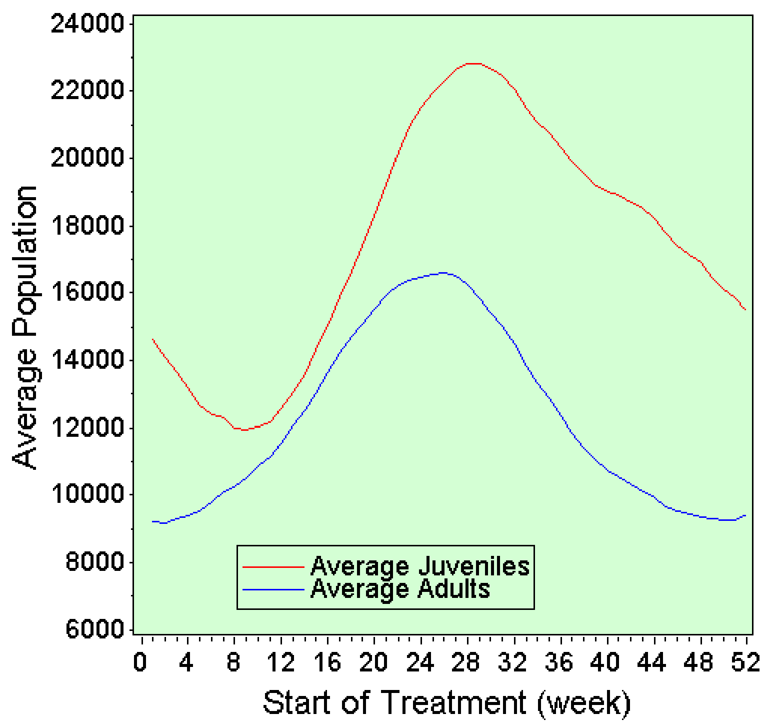


Figure 10-4 Summary of results from a Run Sequence using 5 treatments at 1 week interval

There are several points that can be deduced from the chart:

- None of these 5-spray treatments (totalling 20 sprays over the 4 years) produce a result that is as effective as the "threshold" treatment (Section 10.4)
- Winter (roughly weeks 20-35) is a particularly bad time to apply the treatments. This might be expected, as an examination of Figure 10-2 shows that fewer aphids are present at that time.

- The best time to treat is different depending on which is the target stage. If adults are doing most of damage to the plant, mid-Summer is the most favourable time to spray, while if the damage is done by juveniles, about week 10 (mid-March) is optimal.

Some of these results may seem obvious, but others may be difficult to obtain without the use of such a model.

The results of a “Multiple” run can also be presented in a table. Select the “Table” button or menu-item when the “Run” window is open, and select the variables “Run”, “Total Number of Juveniles” and “Total Number of Adults” for inclusion in the table (in that order). The first part of the resulting table is shown in Figure 10-5. Each line in the table relates to a separate run within the “Run Sequence”. For each run, the shaded area on the left shows the days on which treatment took place (for example, in run 1, insecticide spraying occurred on days 1, 8, 15, 22 and 29 of each year). Individual runs within the Sequence can be examined in detail by double-clicking on the row representing the required run, and selecting either “Show Detail Table” or “Show Detail Chart” from the drop-down menu. All of the formats that have been saved for single run tables and charts will be available on a further drop-down menu. For example, double-clicking on the row of results that starts spraying on day 36, selecting “Show Detail Chart”, and then selecting the “Population and Spray” chart, will result in the chart shown in Figure 10-6. Note that DYMEX actually reruns the simulation for the selected row to obtain the detailed results.

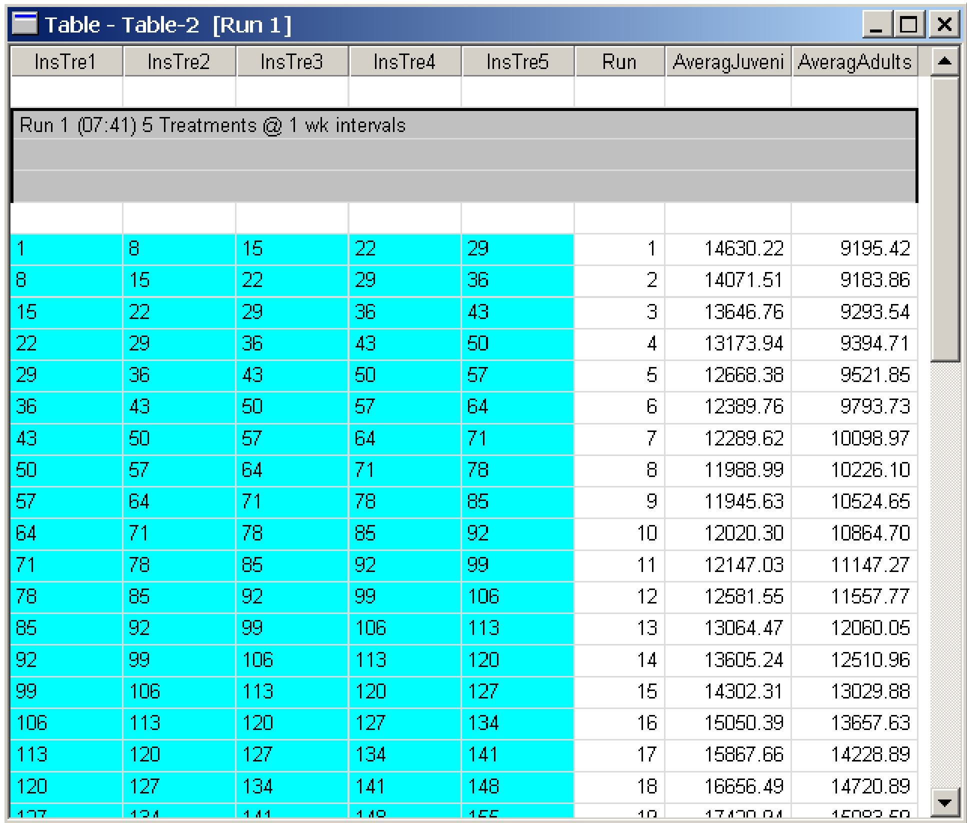


Table - Table-2 [Run 1]

Run 1 (07:41) 5 Treatments @ 1 wk intervals

InsTre1	InsTre2	InsTre3	InsTre4	InsTre5	Run	AveragJuveni	AveragAdults
1	8	15	22	29	1	14630.22	9195.42
8	15	22	29	36	2	14071.51	9183.86
15	22	29	36	43	3	13646.76	9293.54
22	29	36	43	50	4	13173.94	9394.71
29	36	43	50	57	5	12668.38	9521.85
36	43	50	57	64	6	12389.76	9793.73
43	50	57	64	71	7	12289.62	10098.97
50	57	64	71	78	8	11988.99	10226.10
57	64	71	78	85	9	11945.63	10524.65
64	71	78	85	92	10	12020.30	10864.70
71	78	85	92	99	11	12147.03	11147.27
78	85	92	99	106	12	12581.55	11557.77
85	92	99	106	113	13	13064.47	12060.05
92	99	106	113	120	14	13605.24	12510.96
99	106	113	120	127	15	14302.31	13029.88
106	113	120	127	134	16	15050.39	13657.63
113	120	127	134	141	17	15867.66	14228.89
120	127	134	141	148	18	16656.49	14720.89
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Figure 10-5 Table of results from a Run Sequence using 5 treatments at 1 week interval

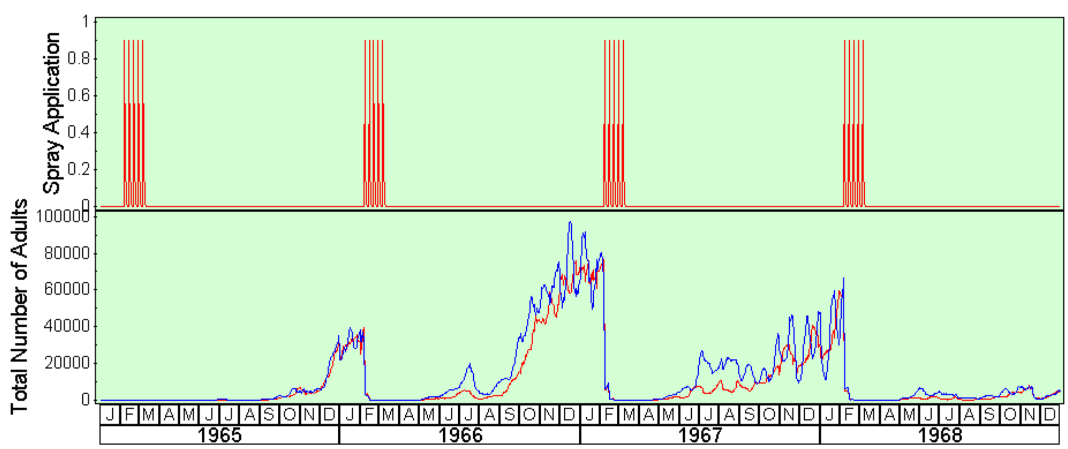


Figure 10-6 Detailed results from a using 5 treatments at 1 week interval, starting on day 36

Tutorial 10 - Summary

Timer
See ***Tutorial 8***
QueryUser (Latitude)
See ***Tutorial 8***
Daylength
See ***Tutorial 8***
MetBase (Meteorological Data)
See ***Tutorial 8***
Evaporation
See ***Tutorial 8***
Soil Moisture
See ***Tutorial 8***
Circadian (Daily Temperature Cycle)
See ***Tutorial 4***

Event (Insecticide Treatment)
Inputs: Day of Year
Simulation Date
Total Number of Adults
Output: Spray Application
Factor:
See ***Tutorial 9***
Lifecycle
Juvenile
Transfer to adult function (Step)
See ***Tutorial 4***
Juvenile development function (Linear above Threshold)
See ***Tutorial 4***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 6***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
See ***Tutorial 8***
Insecticide Application Mortality (Continuous, Direct)
See ***Tutorial 9***
Output:
See ***Tutorial 4***
Adults
Age-based mortality function (Step)
See ***Tutorial 1***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 5***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
See ***Tutorial 8***
Insecticide Application Mortality (Continuous, Direct)
See ***Tutorial 9***

Reproduction
See ***Tutorial 1***

Output:
See ***Tutorial 1***

11 Adding a new "Cohort Property"

DYMEX does not model the fate of individual organisms. Individuals are grouped into assemblages termed *Cohorts*, where each cohort consists of a number of individuals that belong to the same lifestage, occupy the same spatial unit, and share the same properties, like the time (day/week) they entered a stage. Cohorts are the basic units that are modelled in a DYMEX lifecycle. An example of a cohort would be all the juveniles born on a particular day during the simulation. All the individuals within a cohort experience the same conditions during the course of a simulation.

By now, the user of these tutorials should have a fairly good understanding of the use of cohorts in a DYMEX Lifecycle module. The Builder User's Guide (Section 6.2) has a detailed explanation of cohorts and Cohort Properties, while the Simulator User's Guide (Section 5.1) presents a worked example of the dynamics of cohorts during a simulation. This material will not be repeated here, and the reader should study these sections to ensure they have a thorough understanding of how cohorts and "Cohort Properties" are used in DYMEX.

This tutorial examines the DYMEX procedures that are used when a Cohort Property is required which is not contained within the default set of Cohort Properties present in the Model Builder.

11.1 Cohort Properties

Cohorts have a number of properties (referred to as either *Cohort Properties* or *Cohort Variables*) that are shared by all the individuals in the cohort. These properties is are:

- Chronological Age
- Number
- Density
- Physiological Age
- Residual Fecundity

Number contains the current number of individuals in the cohort, while **Physiological Age** contains their current state of development. There is no user-adjustable process associated with the **Chronological Age** property, which always reflects the number of timesteps since the cohort was created. The **Residual Fecundity** Cohort Property is only present in reproductive stages.

For many models, these properties are not enough to describe the state of the cohorts in a lifecycle. For example, when modelling a plant, it may be required to model the Size of the plant (in terms of stem diameter or leaf area). At other times, "stress" experienced by an individual in one stage may affect some process in the next stage. DYMEX provides the user with the ability to define Cohort Properties that can be used for tasks such as these. Up to 32 of these "user-defined" Cohort Properties may be defined for any particular lifecycle.

11.2 'Stress" in the Pseudo-aphid

Like all insects, the Pseudo-aphid has an optimum temperature range inside which, if other environmental conditions are also suitable, a colony of Pseudo-aphids will produce maximum population growth. Data shows that if juvenile aphids are exposed to temperatures outside this range, they will have a lower fecundity when they become adults (see the first two columns in Table 11-1). This effect is in addition to the direct temperature-induced mortality already accounted for in the model. At the extreme temperatures of –10°C and 40°C, a single day's exposure reduces the fecundity to 0. Exposure to temperatures from 5°C to 15°C for any length of time produces no reduction in fecundity.

Temperature (°C)	No. of days before fecundity = 0 (d)	1/d
-10	**1**	1
-5	**2**	0.5
0	**4**	0.25
5	**-**	0
10	**-**	0
15	**-**	0
20	**34**	0.0294
25	**8**	0.125
30	**4**	0.25
35	**2**	0.5
40	**1**	1

Table 11-1 Number of days that juvenile Pseudo-aphids can be exposed to constant temperatures before adult fecundity becomes 0

Note that the situation here requires that an effect that is *experienced* by the Juvenile stage actually has its *effect* in another stage, the Adults. This situation can be handled by a Cohort Property. However, none of the pre-defined Cohort Properties are adequate to handle the situation, and a new property needs to be defined.

One way of handling this situation (and as in usually the case in modelling, there is more than one way) is to define a Cohort Property that could be called *Stress.* Any periods of unfavourable temperatures experienced by the juvenile aphids then act to increase their level of *Stress* from an initial value of 0. When the aphids moult to the adult stage, the value of *Stress* that they have accumulated is transferred with them, and is then used to set the fecundity for the adult cohort.

To follow this through, two relationships need to be defined. The first is the relationship between the temperatures experienced by the juveniles, and the corresponding *Stress* increment. The second is the relationship between the cohort's accumulated stress and the resulting fecundity.

To tackle the first of these relationships, it is necessary to define exactly how the *Stress* property is used. For convenience, *Stress* can be scaled so that a value of 0 indicates that there will be no adverse effect on fecundity, while a value of 1 (or above) will reduce the fecundity to 0. It will also be assumed that at any time, the rate of accumulation of *Stress* is independent of any already accumulated *Stress*. For each constant temperature from Table 11-1, a third column can then be calculated (1/d), which is the *Stress* increment due to that temperature per day. When plotted on a graph, it is obvious that the increments are not linear functions of temperature (Figure 11-1). The dotted lines are straight lines fitted to the points on either side of the optimum temperatures, while the dashed lines fits a quadratic curve. It is clear that the quadratic curves are a better fit. This also makes sense biologically, as it is saying that 10°C outside the optimum for 1 day accumulates more *Stress* than 1°C outside the optimum for 10 days.

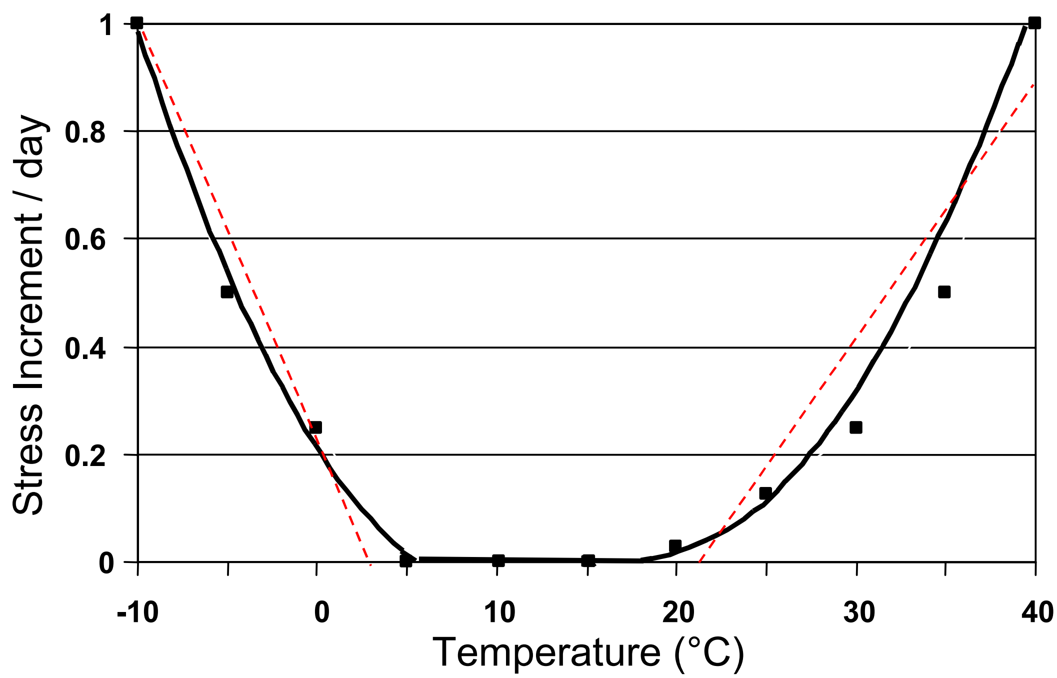


Figure 11-1 Graph of Stress increments per day against Temperature

The pre-defined DYMEX function "Double Quadratic" is exactly what is required to fit this data. This function is defined as follows:

$$f(x) = k_2 * (k_1 - x)^2, \quad \text{for } x < k_1$$
$$f(x) = 0, \quad \text{for } k_1 \le x \le k_3$$
$$f(x) = k_4 * (x - k_3)^2, \quad \text{for } x > k_3$$

From the graph, the two parameters k_1 and k_3 are estimated to be 5 and 17.5, respectively. Assuming that the curves pass through the points (-10, 1) and (40, 1), the other parameters can be calculated by substituting the values for these points into the equation:

Hence, $1 = k_2 * (5 – (-10))^2$
$1 = k_2 * 15^2$
$k_2 = 1/224 = 0.0044$ (approx)

and, $1 = k_4 * (40 – 17.5)^2$

$k4 = 1/(506.25) = 0.002$ (approx)

The second relationship, between a cohort's accumulated Stress and the resulting fecundity, can now be defined. Since the Stress accumulation function has taken care of the relationship inherent in the data of Table 11-1, the accumulated Stress can be applied directly and linearly to the fecundity. Thus, with no Stress accumulated, the fecundity will retain its current value of 5, and with a Stress of 1 or above, fecundity will be 0. The "Linear" function, with Y-intercept of 5 ("Maximum Fecundity") and slope of –5 is suitable in this case, because DYMEX ensures that fecundity cannot become negative even at values of Stress above 1.0 (Figure 11-2).

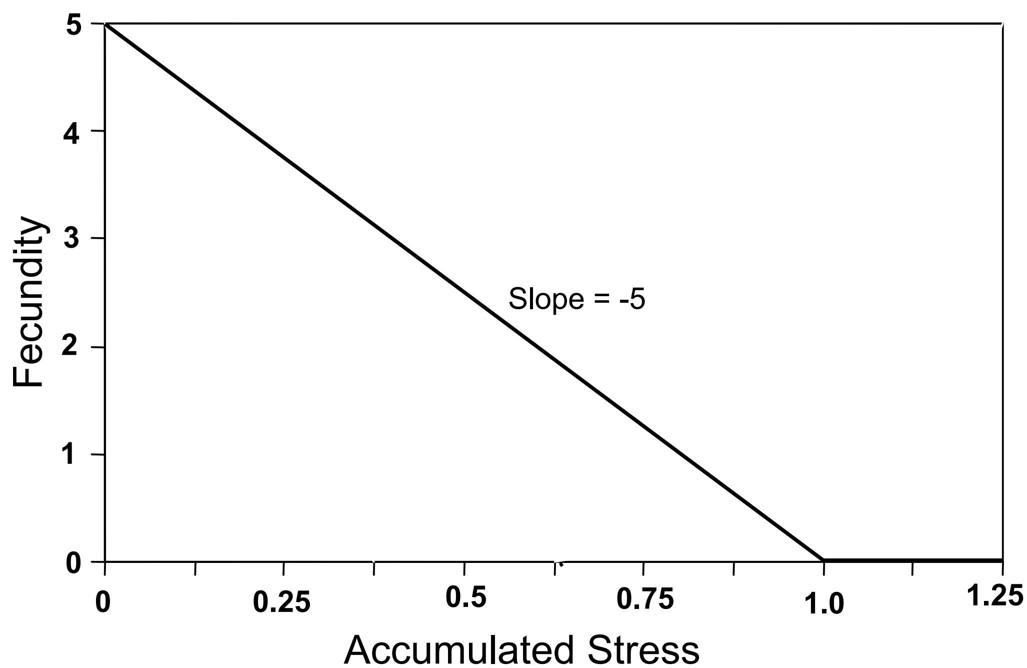


Figure 11-2 The Relationship of Fecundity to accumulated "Stress"

It is also important to note that the scope of the newly created cohort property must also be considered: it can be local or global. DYMEX modelling defines a cohort property to be **local** if the property variable is reset as the cohort passes from one lifestage to the next; the cohort property is defined to be **global** is the variable is carried over unchanged from one lifestage to the next. For this tutorial, the cohort property that will be created and applied will be temperature stress. In addition, the temperature stress that is produced in the juveniles will be assumed to decrease fecundity in the adult stage and so the effects of a new cohort property in the juvenile lifestage will be carried over to produce its results in the adult lifestage.

11.3 Modifying the Model

The modifications involve several steps. Firstly, the new Cohort Variable, Stress, needs to be created. The process that accumulates Stress in the Juvenile stage will then be specified, and finally, the "Fecundity" process in the Adult stage will be changed to be dependent on the Stress value carried through from the Juvenile stage.

1. Start the DYMEX Builder and load the pseudo-aphid model
2. Open the **"Lifecycle"** module for editing
3. Select **"Lifecycle"** from the main menu bar and then select "**User Defined Cohort Variables...."** from the drop down menu
4. In the **"User Defined Variables"** dialog box, select the **"Add"** button to obtain the **"Cohort Variable"** dialog box (Figure 11-3)

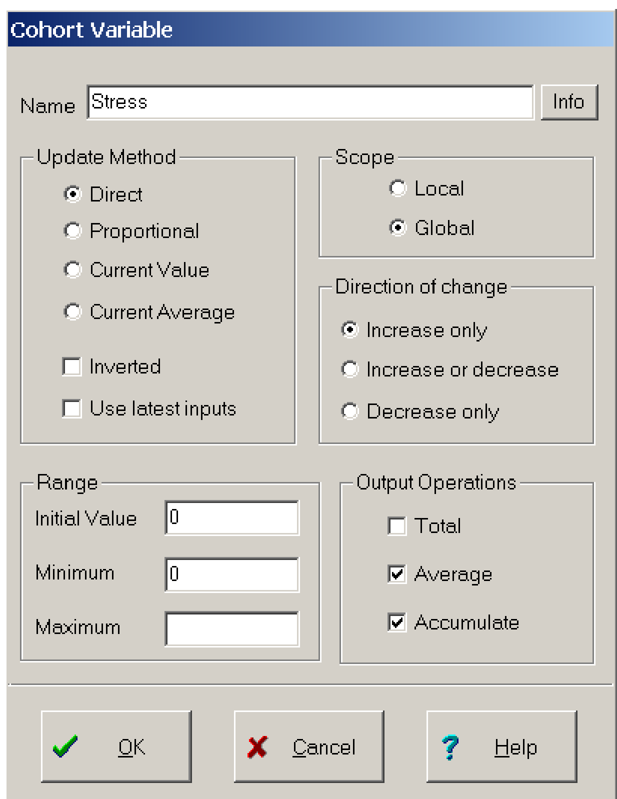


Figure 11-3 The "Cohort Variable" dialog box

5. Enter the name **"Stress"** in the **"Name"** edit box

6. Click on the **“Info”** button, and enter a description of the Cohort Variable (such as “*This variable is used to accumulate the effects of unfavourable temperatures in the Juvenile stage, and its final value is then used to reduce Fecundity in the Adult stage*”
7. In the **“Scope”** panel, ensure that the **“Global”** button is selected

A Cohort Property with “Global” *Scope* carries its value over into the next lifestage. Since values of “Global” Cohort Properties are not lost between lifestages, they must be reset at some point in the lifecycle. In the case of the Pseudo-aphid, this should occur after the Adult stage. If this is not done, each new Juvenile cohort would receive the Stress of its parent Adult cohort, and Stress would simply increase without limit. The reset is specified in the appropriate **Stage Transfer** or **Reproduction** dialog.

8. In the **“Update”** panel, ensure that **“Direct”** is selected.

“Direct” update means that the values of the Stress increment (r), calculated by the Stress process in the Juvenile stage, will simply be added to any existing Stress, i.e.,

$$\text{Stress}_n = \text{Stress}_{n-1} + r$$

9. For the **“Range”,** set the default and minimum values to 0, the maximum value should be left blank (in extreme climates, Stress can reach values well above 1)
10. In the **“Direction of Change”** panel, select **“Increase only”**, as a cohort’s Stress, as defined in this model, cannot ever decrease
11. The **“Output Operations”** settings must now be selected. Select only the **“Average”** and **“Accumulate”** options.

The “Output Operations” specify the way that this Cohort Property should be available for output. “Average” would make available the “average Stress” (over all cohorts), while “Accumulate” allows the actual stress accumulated by each cohort (and carried to the Adult stage) to be output. “Total” (the summed Stress over all cohorts) is not appropriate here as it has no biological meaning.

12. Select **“OK”** as necessary to return to the **“Lifecycle”** window.

Note that once a Cohort Property is added to a lifecycle, it is available for use in any stage of that lifecycle. Up to 32 of these user-defined Cohort Properties can be defined for a lifecycle. The fact that one or more user-defined Cohort Variables are in use in the lifecycle is indicated by the presence of a **“User-defined Cohort Property”** process button on each lifestage in the Lifecycle representation icon (Figure 11-4).

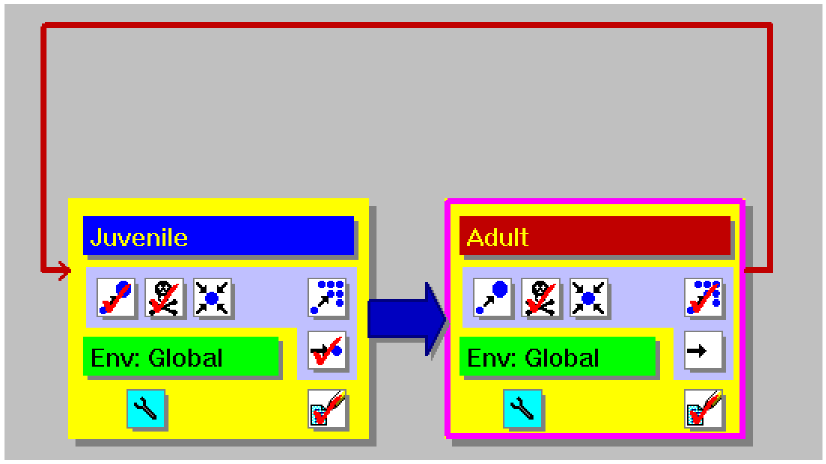


Figure 11-4 Lifestage representations with "User-defined Cohort Property" process

Since "Stress" is a "Global" Cohort Property, we must reset it somewhere in the lifecycle, as discussed above. To reset it after the Adult stage:

13. Click on the "Reproduction" icon in the "Adult" stage
14. Select the button labelled **"C.V. Transfer"** in the top right corner to open the **"Set Cohort Variable Transfer Action"** dialog
15. Select the "Stress" Cohort Property by clicking on it
16. Click on the **"Reset Value"** button at the bottom of the dialog
17. Click **"OK"** where required to return to the Lifecycle window

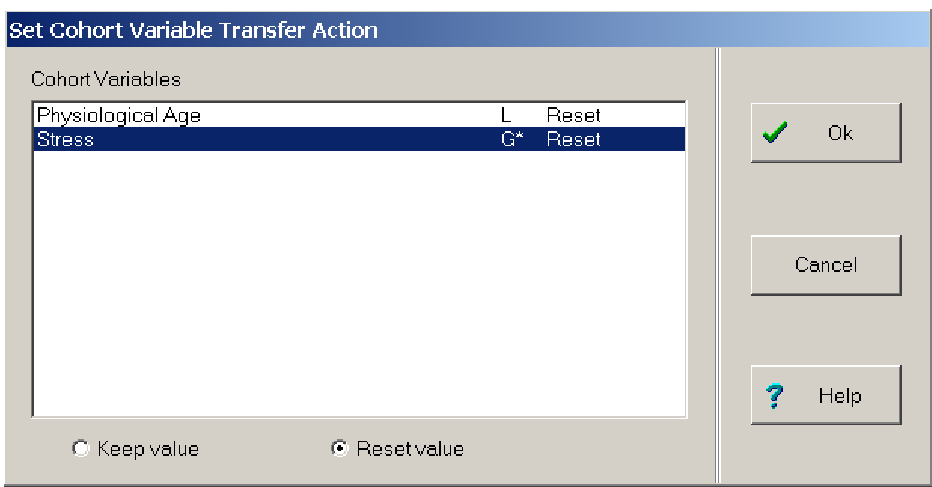


Figure 11-5 The "Set Cohort Variable Transfer Action" dialog box

The next procedure is to specify the Stress process in the Juvenile lifestage. This is done using this new **"User-defined Cohort Property"** button in the Juvenile stage.

1. Select the **“User-defined Cohort Property”** process button on the “Juvenile” lifestage to open the **“Select Process to Edit”** selection box
2. Select the “Continuous” button, as the stress will accumulate at any time that temperatures are unfavourable
3. In the “Stress” dialog, type in ***“Juvenile Stress”*** as the name of the process
4. Select the **“Function”** button, which will open its dialog
5. Select **“Double Quadratic”** as the function
6. Rename the function suitably (e.g. “Temperature-induced Stress ”)
7. Select **“Daily Temperature Cycle”** as the independent variable
8. Set the default value for **“p1: Lower Threshold”** to 5, and allow lower and upper limits of 0 and 10, respectively (rename the parameter if desired)
9. Set the default value of **“p2: Lower Slope”** to 0.0044, and allow lower and upper limits of 0.002 and 0.006, respectively (rename the parameter if desired)
10. Set the default value of **“p3: Upper Threshold”** to 17.5, and allow lower and upper limits of 15 and 25, respectively (rename the parameter if desired)
11. Set the default value of **“p4: Upper Slope”** to 0.002, and allow lower and upper limits of 0.001 and 0.004, respectively (rename the parameter if desired)
12. Exit back to the **“Lifecycle”** window by selecting **“Ok”** as necessary
13. Select the **“Juvenile Lifestage Outputs”** button and ensure that **“Average Stress”** is selected as an output
14. Save the model.

The Juvenile lifestage now has a process that will increment Juvenile Stress as a function of the daily temperatures. If the model is now run in the Simulator (1 year, starting on 1/1/1969), the resultant output for “Average Stress” is shown in Figure 11-6.

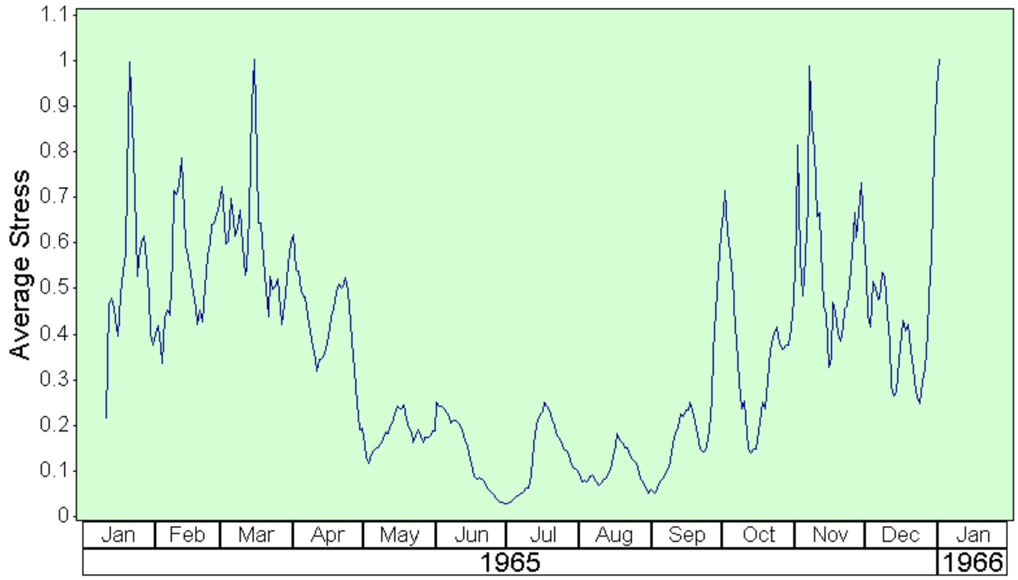


Figure 11-6 Average Stress in Juvenile Pseudo-aphids (Amberley, 1965)

The final procedure in modifying the model is to alter the fecundity process in the Adult lifestage so that it is affected by Juvenile Stress:

1. Select the Adult lifestage's **"Reproduction"** button to open the "**Adult Reproduction"** dialog
2. Click on the **"Fecundity (E)"** button
3. In the **"Fecundity (Adult)"** dialog's list box, delete the present parameter-based component
4. Click on the **"Function"** button to create a function-based component
5. Name the function **"Stress related Fecundity"**
6. Select **"Stress"** as the Independent Variable
7. Select **"Linear"** as the necessary function template
8. Set the **"p1: Y-axis intercept"** parameter default and limits to 5
9. Rename the parameter to "Maximum Potential Fecundity"
10. Set the default value of **"p2: Slope"** to -5, and allow lower and upper limits of 0 and -10, respectively
11. Rename the parameter to "Slope of Stress-Fecundity line"
12. Exit to the lifecycle window and then save the model.

11.4 Running the modified Model

When the model is opened, it should resemble

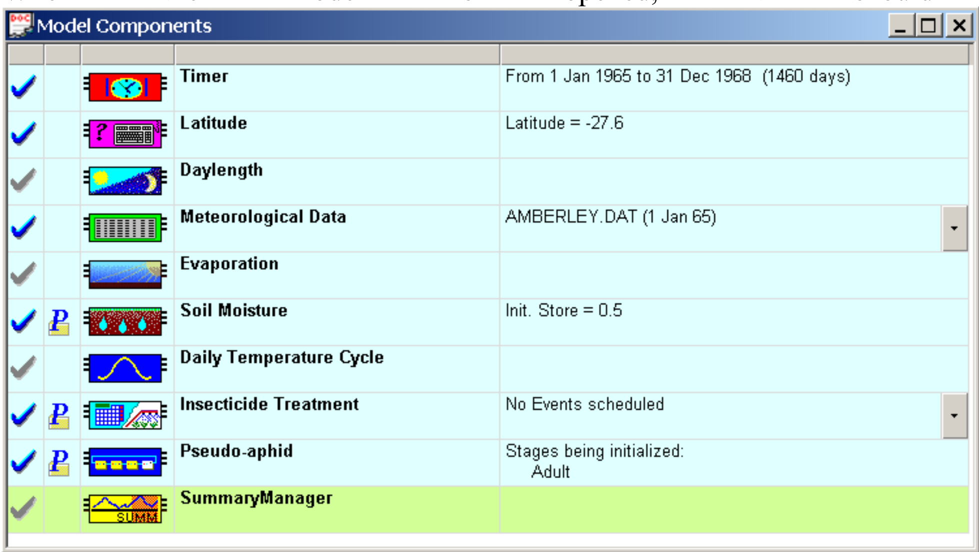


Figure 11-7 below.

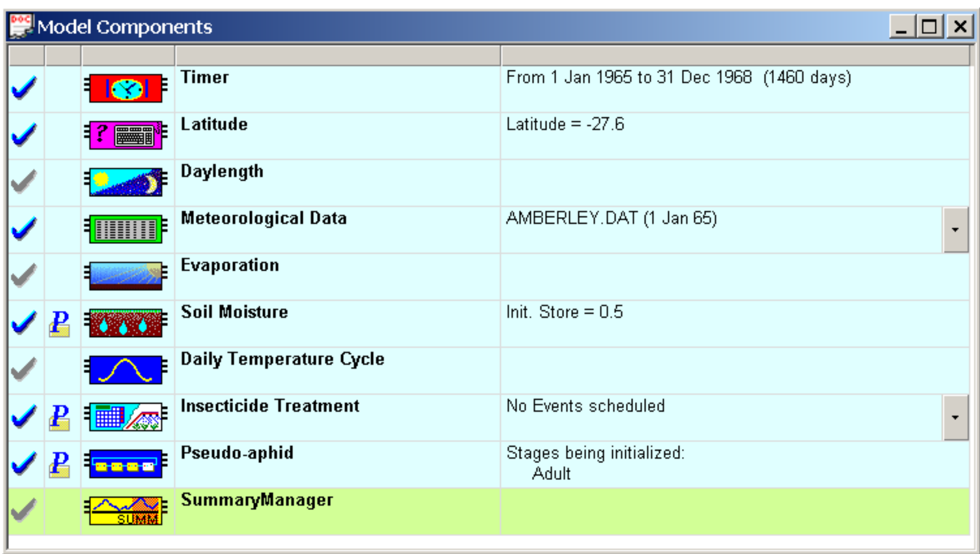


Figure 11-7 The current Pseudo-aphid “Model Component” window

Make sure that the initialization settings are as shown in the Figure, with Lifestage initialization the same as in the previous tutorial (10 “immigrants” per week for 208 weeks). When the model is then run, create a chart using the format “Population and Temperature” created in an earlier tutorial (Figure 11-8). With these new processes included, the Pseudo-aphid is now not able to survive in Amberley. Quite large populations are able to establish from immigrants for several favourable month (spring to early summer) in some years, but high temperatures in summer and low temperatures in winter prevent population build-up at other times.

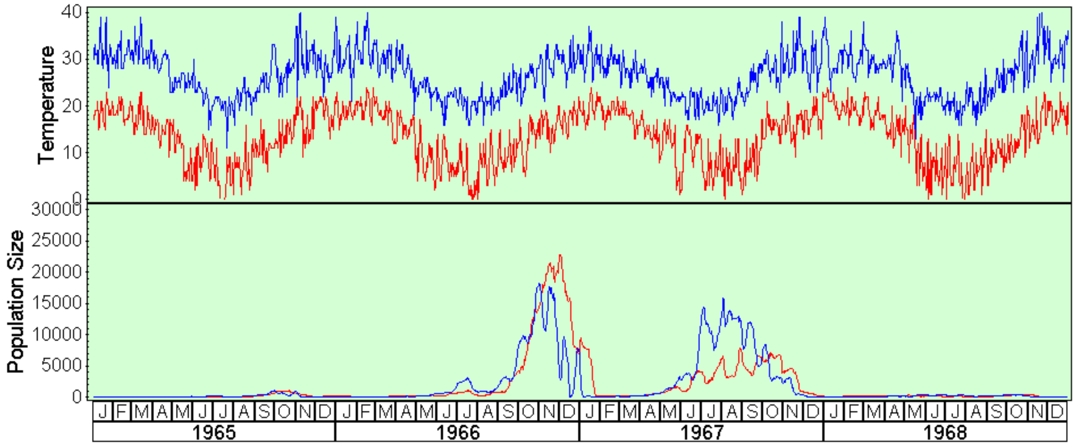


Figure 11-8 Pseudo-aphid population chart with stress-affected Fecundity

To help interpret the situation illustrated in Figure 11-8, the user can produce a graph that shows the population, average stress levels of juveniles and temperatures at any time. The details of producing this graph are left as an exercise, but the final graph should look similar to Figure 11-9. Note that stress levels are high (and therefore fecundity is low) throughout most of summer, whereas in earlier versions of the model rapid population growth occurred during that time.

As further exercises, the reader may wish to experiment with adjusting the parameter values that determine the accumulation of Stress.

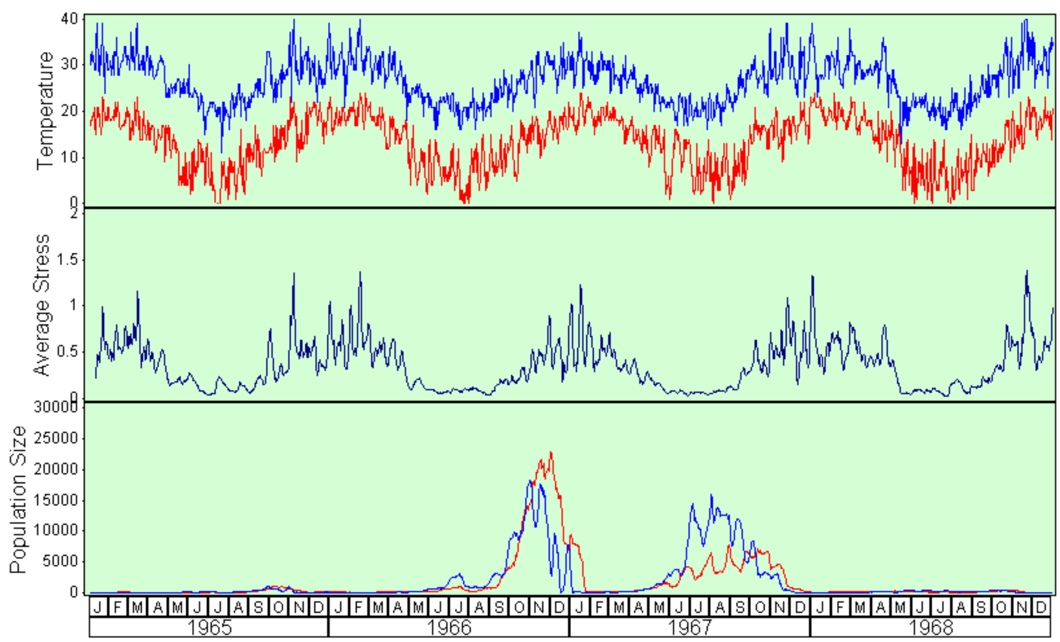


Figure 11-9 Graph of Pseudo-aphid population and “Stress”

Tutorial 11 - Summary

Timer
See ***Tutorial 8***

QueryUser (Latitude)
See ***Tutorial 8***

Daylength
See ***Tutorial 8***

MetBase (Meteorological Data)
See ***Tutorial 8***

Evaporation
See ***Tutorial 8***

Soil Moisture
See ***Tutorial 8***

Circadian (Daily Temperature Cycle)
See ***Tutorial 4***

Event (Insecticide Treatment)
See ***Tutorial 10***

Lifecycle
User-defined Cohort Variable: Stress

Scope:	Local
Update Method:	Direct
Direction of Change:	Increase Only
Range:	0-…; Initial Value, 0
Ouput Operations:	Average, Accumulate
Reset after Stage:	Adult

Tutorial 11 – Summary (continued)

Lifecycle (cont'd)

Juvenile

Transfer to adult function (Step)
See ***Tutorial 4***
Juvenile development function (Linear above Threshold)
See ***Tutorial 4***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 6***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
See ***Tutorial 8***
Insecticide Application Mortality (Continuous, Direct)
Driving Variable: Spray Application
Temperature-induced Stress (Double Quadratic)
Driving Variable: Daily Temperature Cycle
Lower Threshold: 5
Lower Slope: 0.0044
Upper Threshold: 17.5
Upper Slope: 0.002

Output:
Total number
Juvenile Development Time
Average Stress

Adults

Age-based mortality function (Step)
See ***Tutorial 3***
Low Temperature Mortality (Continuous, Linear below Threshold)
See ***Tutorial 5***
High Temperature Mortality (Continuous, Linear above Threshold)
See ***Tutorial 5***
Density-dependent Mortality (Continuous, Linear above Threshold)
See ***Tutorial 6***
Dryness-dependent Mortality (Continuous, Linear below Threshold)
See ***Tutorial 8***
Insecticide Application Mortality (Continuous, Direct)
Driving Variable: Spray Application

Reproduction
Reproduction
Fecundity (Linear)
Driving Variable: Stress
Y-axis intercept: 5
Slope: -5
Progeny Production
See ***Tutorial 1***

Output:
See ***Tutorial 3***