

## Simulation of heat and humidity budgets of Biosphere 2 without air conditioning<sup>1</sup>

Angela Nebot <sup>a</sup>, François E. Cellier <sup>b,\*</sup>, Francisco Mugica <sup>c</sup>

<sup>a</sup> *Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Jordi Girona Salgado 1–3, Barcelona 08034, Spain*

<sup>b</sup> *Department of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721, USA*

<sup>c</sup> *Depto. de Energía Nuclear, Instituto de Investigaciones Eléctricas, Reforma 113, Col. Palmira, 62001 Cuernavaca, Morelos, Mexico*

Received 15 July 1996; received in revised form 30 October 1997; accepted 29 November 1997

---

### Abstract

The main goal of this study was the development of a dynamic model that represents the thermal behavior of the complex ecological system of Biosphere 2, Nes. Tucson, AZ, USA. In this paper, a model that captures the thermal behavior of the ecological system in a non-controlled (i.e. passive) environment is presented. The bond graph methodology was used for modeling this highly complex system. The object-oriented nature of the bond graph approach enables the modeler to keep conceptually separated aspects of knowledge about the system's comportment isolated from each other. Thereby, the individual modeling entities remain small and manageable. This makes it easier for the modeler to properly debug and validate individual models. Uniform power-flow interfaces between all bond graph models ensure energy conservation at the connections between the individual models, and support the modeler in validating the interconnected bond graph model of the overall system. Although plausible simulation results are presented at the end of this paper, no true simulation verification could be made, because the real system has never, since its completion, been allowed to be operated in a purely passive mode, i.e. without its air handlers, as in fact, such an experiment would kill most of the biomes inside Biosphere 2. Yet, simulation runs of the passive system are meaningful for model validation purposes. The control systems that operate the air handlers reduce the sensitivity of the simulation output to modeling errors, and may, in fact, not only correct for Tucson's hot desert climate, but also

---

\* Corresponding author. Tel.: +1 520 6216192; fax: +1 520 6218076; e-mail: cellier@ece.arizona.edu

<sup>1</sup> A preliminary and much shorter version of this report was presented in Nebot et al. (1996).

for temperature deviations caused by an incorrect mathematical description of the system thermodynamics. © 1999 Elsevier Science B.V. All rights reserved.

*Keywords:* Closed ecological systems; Object-oriented modeling; Bond graphs; Thermodynamics; Energy balance modeling; Convection; Radiation; Evaporation; Condensation

---

## 1. Introduction

Biosphere 2 is a materially closed ecological system of high complexity built to deepen the understanding of ecological systems, to study the dynamics of closed ecologies, and to learn to control their behavior. The use of modeling and simulation is crucial to the achievement of these goals. Understanding a physical system is almost synonymous with possessing a model of its comportment.

A passive thermal behavior model of Biosphere 2 has been developed by means of bond graphs, a tool for the graphical representation of power flows through and energy balance in a system. Bond graphs offer an appealing means for dealing with complex physical systems, since they are object-oriented, i.e. they preserve the physical topology of the system in interconnections, and because, as a side effect of a consequent application of uniform power-flow interfaces between all bond graph models, they ensure that the energy conservation laws are never violated in model interconnections (Cellier, 1991; Cellier et al., 1995). The model presented in this paper is based on a previous research effort by Luttmann (1990).

The bond graph model has been coded in the Dymola language (Elmqvist, 1978), a commercially available object-oriented modeling environment designed for complex physical system modeling (Elmqvist, 1995; Dynasim, 1997). Dymola fully supports the reuse of models by means of libraries containing model classes from which individual models (objects) can be instantiated, and by means of class inheritance. Dymola can be used as a code generator for many different simulation software tools, such as ACSL (MGA, 1991), Desire (Korn, 1989), Simnon (Elmqvist, 1975; Elmqvist et al., 1990), and Simulink (Mathworks, 1992), as well as DSblock (Fortran) (Otter, 1992) and Dymosim (C) (Otter, 1996).

The simulation language code generated by Dymola from the bond graph model (ACSL was used in this research effort as the target language) contains only nine differential equations (one for each of the capacitors used in the model) but more than one hundred algebraic equations. The code is highly unreadable, and it would be rather difficult to make any modifications to the generated ACSL program manually. In fact, the ACSL code resembles strongly the Fortran code that had originally been developed by Luttmann (1990) in his Ph.D. dissertation. Although Luttmann's model had been developed for Biosphere 2, the code was never used by the engineers at Biosphere 2. The reason was simply that the code was totally unmanageable. After Luttmann had left, no-one at Biosphere 2 possessed the knowledge any longer that was necessary to make modifications to the code, i.e. the only thing that they could still do with the code was to run it as it was and look at the results that it generated.

Simulation is rarely the sole purpose of a model. A model is a means for organizing, in a systematic and unambiguous fashion, the knowledge that is available about a system under study. Yet, organization means structure. Luttmann's model, as represented by the FORTRAN code, was totally unstructured, and therefore, did not help at all with organizing available knowledge. It could only be used for simulation, and any minor modification that should be made on it was a major task, even for someone who knew the code very well.

The new bond-graph implementation of Luttmann's model, as presented in this paper, was designed for optimal readability, modularity, and flexibility. Clearly, acquiring a new 'language' is never easy. It requires a substantial effort to familiarize oneself with any new 'language,' and the bond-graph 'language' is no exception to the rule. To someone, who has never before seen a bond graph, the notation may look rather cryptic, and the natural reaction of especially experienced engineers is likely to be negative at first. Why should I learn a new 'language'? Haven't I been successful so far with using the 'languages' that I already know? Isn't it enough to rely on 'rock-solid' algebra as a means for expressing physics, and English as a vehicle for explaining the resulting mathematics?

Unfortunately, the answer is no. It is true that mathematics is the 'language' of physics and physicists. Yet, physicists are traditionally educated to investigate and understand physical objects. Consequently, they developed a 'language' that is ideally suited to describe such objects. Yet, the language of mathematics is not well suited for describing interconnections between objects. The only mathematical way of expressing interrelations between two equations is to reuse the same variable names in both of them. This leads to a flat amorphous description of a physical system. Yet, the organization of knowledge about a complex physical system lies precisely in the knowledge of the topology of its subsystems and their interconnections. This knowledge is thrown away in a description that is based on traditional algebra. The bond graph 'language' preserves topological knowledge, and it is precisely this property that makes the new model attractive. Once the bond graph is compiled down to simulation language code, there is no longer any difference between the new model and the one developed by Luttmann. In fact, the simulation results are exactly the same.

The Dymola (bond graph) code is highly modular with small individual modules, the functioning of which can be easily understood. It is hierarchically (topologically) structured, which makes the overall model easy to grasp and manipulate. This is essential, since Biosphere 2 is a research environment that allows for changes in the topology, and accommodates at all times a variety of different research objectives pursued by separate experiments.

## 2. The topology of the Biosphere 2 model

Fig. 1 shows the topology of the model used in this research effort. Represented in the model are the glass dome, called *cover* in the model, that holds the *internal atmosphere*. The atmosphere is considered homogeneous, i.e. the same temperature

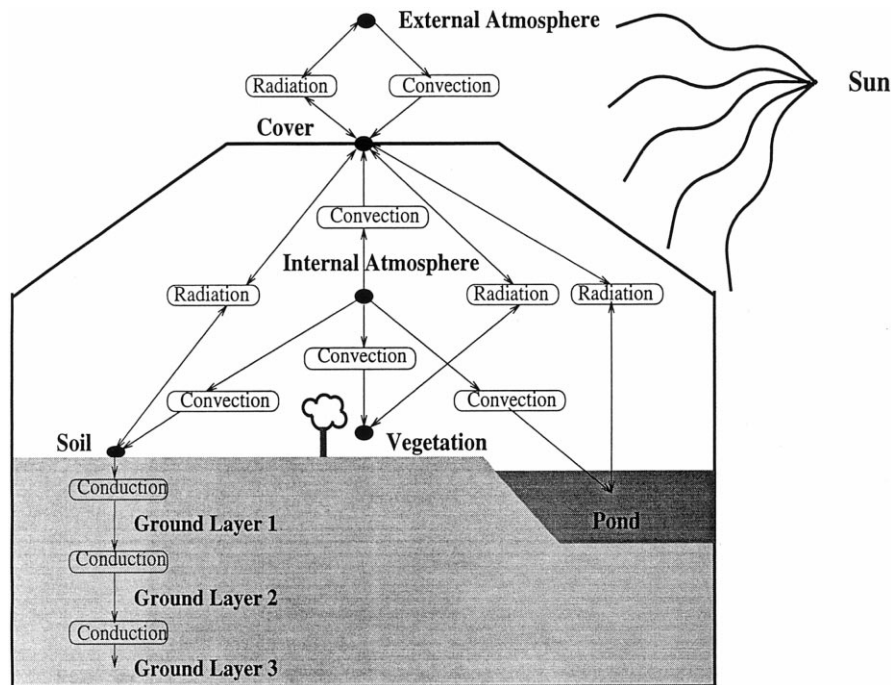


Fig. 1. Thermal interactions between entities inside Biosphere 2.

and humidity values are assumed to be present throughout the structure. Also Biosphere 2's ocean is represented, called the *pond* in the model. The top *soil* is shown that is partly covered by a layer of *vegetation*. Finally, three *ground layers* are also included in the model.

Each of the eight individual modeling units mentioned above is characterized by its own temperature value, and has the capacity to store thermic energy. The atmosphere furthermore contains humidity, i.e. it can store both sensible and latent heat. The primary mechanisms for energy exchange between the model entities are conduction, convection, radiation, evaporation, and condensation.

Conditions that exist outside the biome structure are characterized by three time-dependent processes: the *sky radiant temperature*, the *ambient temperature*, and the *solar energy input*. These three time-dependent functions are provided in a weather file available for the location of Tucson. For Biosphere 2, the data ought to be slightly corrected because of the higher altitude, but this has not yet been done.

The *cover* is a glass dome that hermetically encloses Biosphere 2 from the outside. The air flow across the dome is negligibly small. However, thermal energy is traded freely across the dome.

Heat is transferred across the cover by means of radiation and convection. It passes through the internal atmosphere, eventually reaching the other model

entities. Inside the dome, both *sensible* and *latent* heat convection play important roles. For this reason, it was decided to represent, in the model, the internal atmosphere through two separate modeling elements, one that captures the temperature (sensible heat convection), and another that represents the moisture (latent heat convection).

The *internal atmosphere* (air) inside Biosphere 2 is in direct contact with all the other model entities except the three ground layers below the soil surface. Both sensible and latent heat convection take place between the internal atmosphere and the soil surface, the vegetation canopy, the pond, and the cover.

Thermal processes that affect the *vegetation canopy* are convection and radiation. The vegetation canopy absorbs a part of the solar radiation. The amount of solar radiation absorbed is a function of the total leaf area and the thickness of the canopy.

The canopy emits thermal radiation to the cover, and sensible convection is produced due to air movement over the leaves. Latent heat convection (evapotranspiration) is produced from the interior of the leaves to the surrounding air. This diffusion process depends on air resistance and stomatal resistance. Stomatal resistance of a leaf relates to the degree of the opening of the leaf pores, and depends on available light and water.

The *soil surface* is considered an interface between the ground and the air. Heat (sensible heat) and moisture (latent heat) are convected from the soil surface area to the surrounding air. The former is a function of the air motion over the surface, whereas the latter depends on the resistance to vapor diffusion. Heat is transferred by conduction from this layer down to the ground layer below, and it is assumed that no evapotranspiration takes place in the ground layers.

The *ground* is divided into three layers to be able to model the thermal conduction between them. The thickness of the ground layers can be specified through thermal parameters. Heat losses at the boundaries between the soil inside Biosphere 2 and the ground surrounding it are neglected.

In the *pond*, the same thermal phenomena take place as in the vegetation and soil. A part of the solar radiation that reaches the pond is reflected, whereas the rest is absorbed by the pond. Thermal radiation is transferred between the pond and the cover, and evaporation takes place from the water surface to the surrounding air.

Each of the eight model entities has thermal capacitance. Thermal capacitances are thus associated with the cover, the pond, the three ground layers below the soil surface, the soil surface itself, the vegetation canopy, and the internal atmosphere. An additional capacitive storage element is associated with the internal atmosphere to reflect its capability of storing moisture.

The thermal state of Biosphere 2, as represented by the model, is determined at any time through a set of nine differential and 130 algebraic equations. They represent the energy balance for each of the entities inside Biosphere 2, including the moisture in the internal atmosphere. State variables are the temperature values of each entity, and the air humidity ratio in the internal atmosphere.

Any model always is a simplification of reality. It is the most important task of an experienced modeler to decide, which facets of reality must be represented in the

model, and which other effects are of second order small and can be safely left out of the model without making the model useless for its designated purpose.

The proposition that the entire atmosphere inside Biosphere 2 is at the same temperature and humidity levels is clearly incorrect. There are large variations of both temperature and humidity values among the different biomes inside Biosphere 2. Yet, the omission may be justified by the fact that the primary purpose of this model is the structuring and encapsulation of knowledge about the physical effects that govern the thermodynamics of Biosphere 2, and not, to be able to run meaningful simulations. In fact, if the modeling methodology is truly modular, as promised in the introduction, it should be a straightforward task at the end to instantiate several smaller Biosphere 2 models, all having similar structures but different temperature and humidity values, that are then interconnected by convective energy and mass flow models that describe the air exchange between neighboring biomes.

Similarly, neglecting the heat exchange that takes place between the soil and the ground outside the Biosphere 2 structure is a gross and hardly justifiable omission. Yet, adding such an interaction at a later point is straightforward. This interaction was left out of the model, because it hadn't been considered in the earlier Luttmann model either, and it was thought useful at this point, not to alter the physics of the Luttmann model at all, in order to be able to validate the new model by comparing its simulation results with those obtained by Luttmann.

### 3. The bond graph model

The bond graph model of the overall structure is shown in Fig. 2. Note that the structure of the global bond graph model has a direct mapping to the physical structure of the system.

Each entity is represented by a 0-junction. Capacitances attached to each of these junctions represent the heat or moisture storage of the entity. Flow sources attached to the cover, soil, vegetation, and pond represent solar input. Effort sources are used to model the ambient temperature and the effective sky radiant temperature. Note that the internal atmosphere is represented by two separate 0-junctions. The one to the left represents sensible heat storage, whereas the one to the right represents latent heat storage.

The radiation, convection, conduction, evaporation, and condensation phenomena that take place between the internal elements of the overall structure are represented as power flows between corresponding 0-junctions. Bond graph models representing these flows are instantiations of generic radiation, convection, conduction, and evapora-condensation models containing, where needed, additional specific equations for computing parameter values of the generic models.

Four different types of generic models representing the different types of thermal flows appear in the model of Fig. 2. These are not standard bond graph elements as they can be found throughout the bond graph literature. They are macro elements that are themselves composed of bond graphs making use of more basic

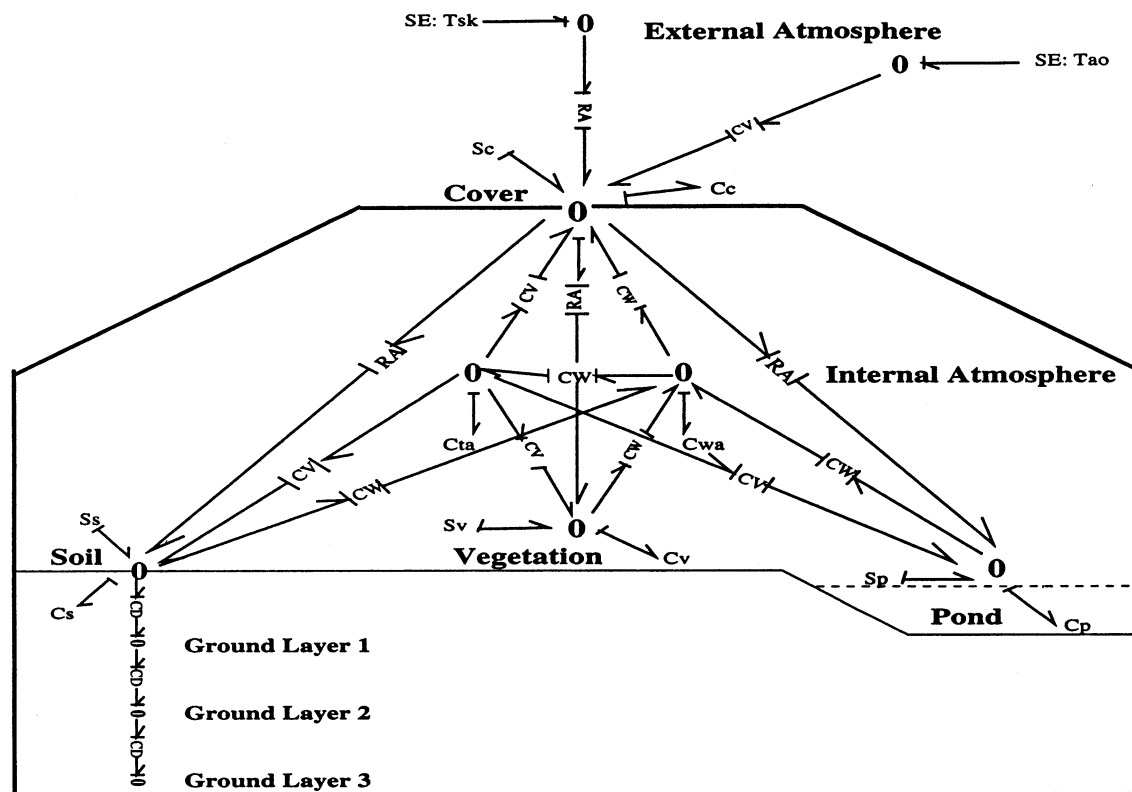


Fig. 2. Bond graph model for Biosphere 2. Bond graph not shown are as follows: RA = thermal radiation; CV = sensible nest convection; CD = thermal conduction; CW = latent/sensible nest conversion; Sc = solar absorbed by cover; Sv = solar absorbed by vegetation; Sv = solar absorbed by pond (ocean); Ss = solar absorbed by soil.

bond graph elements. **RA** is a bond graph generic model representing thermal radiation, **CV** is a bond graph generic model representing sensible heat convection, **CD** is a bond graph generic model representing thermal conduction. The **CW** element represents both evaporation and condensation. Those **CW** flows pointing at the 0-junction that represents latent heat storage in the internal atmosphere denote conversion of sensible to latent heat through evaporation, whereas the two **CW** flows that point away from that 0-junction symbolize conversion of latent to sensible heat by means of condensation.

#### 4. The Dymola model

The bond graph model shown in Fig. 2 captures the topological structure of the overall Biosphere 2 model. In this section, the individual modeling components, such as the **RA**, **CV**, **CD**, and **CW** model classes shall be analyzed in more detail.

A mixture between mathematical descriptions of the model equations and their representation in the Dymola model code has been chosen to describe the component models. The mathematical description has the advantage of being familiar to all readers, whereas the Dymola implementation has the advantage of being unambiguous, enabling interested readers to reproduce the results shown in this paper<sup>2</sup>.

##### 4.1. Solar input

It would have been possible to model the solar input topologically, showing a single flow source that represents the solar energy arriving at the dome, splitting the flow at the dome into three substreams, one representing the portion of the energy that is absorbed by the dome, a second showing the energy that gets refracted through the glass panels, and finally, a third portion representing the energy that is reflected back into space, and that is consequently lost to the model. Yet, the physics are simple enough to represent them directly in the model equations themselves, i.e. without preserving the topology. This simplifies the resulting bond graph, and makes Fig. 2 more readable.

Hence the bond graph model of Fig. 2 shows four separate flow sources, called **Sc**, **Sv**, **Sp**, and **Ss** showing the amount of solar input absorbed by the cover, the vegetation, the pond, and the soil, respectively.

Let  $so(t)$  denote the solar input per square meter arriving at Biosphere 2 at any given point of time. The total solar input to Biosphere is then:

$$\dot{Q}_{\text{tot}} = a_g \cdot so \quad (1)$$

where  $a_g$  denotes the exposed surface of Biosphere 2. The amount of solar energy refracted through the glass panel is:

<sup>2</sup> The full Dymola code described in this paper can be found on the world wide web at the URL [http://www.ece.arizona.edu/~cellier/eoeng\\_97.html](http://www.ece.arizona.edu/~cellier/eoeng_97.html)



$$\dot{Q}_{\text{refr}} = a_g \cdot so \cdot \tau \quad (2)$$

where  $\tau$  denotes the percentage of refracted heat flow. The amount of solar input reaching the pond is:

$$\dot{Q}_{\text{pond}} = a_g \cdot so \cdot \tau \cdot a_{\text{pix}} \quad (3)$$

where  $a_{\text{pix}}$  is the percentage of the Biosphere 2 surface covered by the pond. Finally:

$$\dot{Q}_{\text{pond,abs}} = P_p = a_g \cdot so \cdot \tau \cdot a_{\text{pix}} \cdot (1 - \text{refl}) \quad (4)$$

is the amount of solar input absorbed by the pond. The quantity  $\text{refl}$  is the percentage of solar input that is reflected at the surface of the pond.

The Dymola code contains a generic model that calculates the available solar input:

```

model class   SolarInput
  inherit     PhysPar
  main cut    A(e| - f)
  parameter   m = 11.247, a = 0
  local       so, p
               so = m*rad(Time) + a
               p = e*f
end

```

The variable  $so$  is calculated by interpolation in the input table  $rad$ , which is stored as a function of time. The parameter  $m$  is a conversion factor to express  $so$  in the desired units ( $\text{Wm}^{-2}$ ).

The *SolarInput* model is connected to a single bond that carries the power  $p$  away from it. The power is the product of the ‘effort’ (the temperature)  $e$  and the ‘flow’ (the entropy flow)  $f$ . The interface is defined in the main cut statement that declares the connector of the model to the outside world. The model class inherits another model class called *PhysPar* where all the physical parameters are being stored. In this way, all physical constants are available to the inheriting model whenever needed.

The specific solar input model for the pond takes the following form:

```

model class   Sp
  inherit     SolarInput
               p = ag*so*tau*apix*(1. - refl)
end

```

It inherits the *SolarInput* model, and with it indirectly also the *PhysPar* model. Therefore, the values of  $a_g$ ,  $\tau$ ,  $a_{\text{pix}}$ , and  $\text{refl}$  are known to the model.

The reader may think that there are two definitions provided for  $p$ , one in the **Sp** model, the other in the parent model *SolarInput*. However, this is a misinterpretation of the Dymola code. Dymola solves each equation for one of its variables, depending on the environment in which the equation is being used. Dymola computes  $p$  from the **Sp** model, because the only other variable in the equation is  $so$ , which has already been computed from the *SolarInput* model. The effort (temperature)  $e$  is already known, because it is determined by the capacitance **Cp** that is attached to the same 0-junction. Therefore, Dymola solves the equation:

$$p = e \cdot f \quad (5)$$

for the remaining unknown:

$$f = \frac{p}{e} \quad (6)$$

which is consistent with the notion that the solar input is, in fact, a flow source.

An interesting model is that describing the solar input to the cover. The amount of energy absorbed by the cover is:

$$\dot{Q}_{\text{refr}} = a_g \cdot so \cdot \alpha \quad (7)$$

where  $\alpha$  denotes the percentage of heat flow absorbed by the cover. However, this is only a part of the absorbed heat. The transmitted heat is:

$$\dot{Q}_{\text{refr}} = a_g \cdot so \cdot \tau \quad (8)$$

A portion of the transmitted heat,  $p_p$ , gets absorbed by the pond, other portions,  $p_v$  and  $p_s$  get absorbed by the vegetation and the soil. The remaining heat:

$$\dot{Q}_{\text{refl}} = a_g \cdot so \cdot \tau - p_v - p_s - p_p \quad (9)$$

has been reflected, and is now traveling outward again. It hits the cover from the inside, and again, a portion,  $\alpha$  is absorbed on the way out. Hence the total energy absorbed by the cover is:

$$\dot{Q}_{\text{cover}_{\text{abs}}} = a_g \cdot so \cdot \alpha + (a_g \cdot so \cdot \tau - p_v - p_s - p_p) \cdot \alpha \quad (10)$$

This equation is expressed in the Dymola model:

```

model class Sc
  inherit SolarInput
  terminal pv, ps, pp
    p = ag*so*alph + (ag*so*tau - pv - ps - pp)*alph
end

```

The **Sc** model needs knowledge of the percentage of heat absorbed by the pond, the vegetation, and the soil. This is implemented by another interface declaration. In bond graph notation, this corresponds to the declaration of an interface for three *activated bonds*, i.e. pure information flow connections. This concept is physically dubious. There is no such thing as ‘pure information flow’ in physics.

The problem has to do with the way, in which the topology was broken. Had the total solar input to Biosphere 2 been represented by a single source element followed by a number of heat flows showing the distribution of the solar input through the system, there would have been no need for activated bonds. This is precisely one of the problems of relying too much on mathematical equations and their manipulations, rather than sticking with the underlying physics of energy and mass conservation. Yet, the solution was chosen in order not to clutter up the bond graph of Fig. 2 with additional heat flows showing the delivery and distribution of solar energy to and through the system.

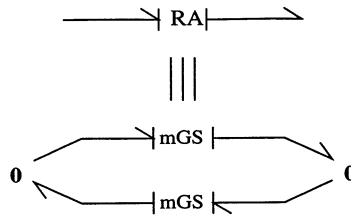


Fig. 3. Bond graph for thermal radiation (RA).

#### 4.2. Radiation

Net radiation from a body  $B_1$  at temperature  $T_1$  to a body  $B_2$  at temperature  $T_2$ , where  $T_1 > T_2$  can be described by the well-known Stefan–Boltzmann law:

$$\dot{Q}_{\text{rad}_{12}} = \sigma \cdot a \cdot (T_1^4 - T_2^4) \quad (11)$$

where  $\sigma$  is a constant describing the capability of a body to emit and/or absorb light, which depends on the body's color, and  $a$  is the radiating surface. Yet, radiation is a property of a body, and should not depend on two separate temperatures. Body  $B_1$  radiates, irrespectively of whether body  $B_2$  is around to receive the radiation or not.

The subtraction in the above equation is a consequence of merging two separate phenomena, the radiation of the two bodies  $B_1$  and  $B_2$ , into a single equation. For the purpose of this investigation, it was deemed more useful to keep the two phenomena separate. Fig. 3 shows two separate radiative flows, the top flow symbolizes the radiation from the left body (0-junction) to the right body, whereas the bottom flow shows the reverse radiation. Each of the **mGS** elements now implements an equation that exhibits only one of these flows, i.e., is described by the equation:

$$\dot{Q}_{\text{rad}_i} = \sigma \cdot a_i \cdot T_i^4 \quad (12)$$

where  $T_i$  is the temperature of the emitting body, and  $\dot{Q}_{\text{rad}_i}$  is the radiative heat flow away from that same body. Hence the **RA** element shown on Fig. 2 is, in fact, a macro element that contains a bond graph as shown in Fig. 3. The Dymola implementation of this bond graph is as follows:

```

model class   RAcp
  inherit     Two Port
  submodel    (mGScp) mGScp1, mGScp2
    connect    mGscp1 from A to B
    connect    mGScp2 from B to A
  end

```

Model class **RAcp** is the specialized radiation model describing the radiation from the cover to the pond. It instantiates two models called *mGScp1* and *mGScp2* of the specific model class **mGScp** that implements the equation for the radiative heat flow. *A* and *B* are two cuts that are inherited from the generic model class *Two Port*:

```

model class  TwoPort
inherit     PhysPar
cut        A(e1/f1), B(e2/-f2)
main cut    C[A, B]
main path   P < A - B >
end

```

Equivalent specialized model classes exist for the other radiative flows shown in Fig. 2.

Stefan–Boltzmann’s law describes *heat flow*, i.e. a variable of type power. For the purpose of a bond graph description, this equation should be modified to describe *entropy flow*. Since the heat flow is the product of the temperature and the entropy flow:

$$\dot{Q}_{\text{rad}_i} = T_i \cdot \dot{S}_{\text{rad}_i} \quad (13)$$

Stefan–Boltzmann’s law can also be written as:

$$\dot{S}_{\text{rad}_i} = \sigma \cdot a_i \cdot T_i^3 \quad (14)$$

or using the standard bond graph notation:

$$f_i = \sigma \cdot a_i \cdot e_i^3 \quad (15)$$

This is an algebraic relation between effort and flow. It can be represented by a conductance  $G$  as follows:

$$f_i = G_i \cdot e_i \quad (16)$$

and thence:

$$G_i = \sigma \cdot a_i \cdot e_i^2 \quad (17)$$

In Dymola, a generic radiation model, called **mGSr**, is encoded as:

```

model class  mGSr
inherit     Two Port
local       G, G0
              G*e1 = f1
              e1*f1 = e2*f2
              G = G0*sgma*(e1**2)
end

```

The first equation states Ohm’s law. The second equation states that the power flow in ( $e_1 \cdot f_1$ ) equals the power flow out ( $e_2 \cdot f_2$ ), and the third equation lists the Stefan–Boltzmann law in the form suitable for a bond graph description. The surface that is involved in the radiation,  $G_0$ , is specific for each radiative flow. For example, the specific **mGSep** model is encoded as follows:

```

model class  mGSep
inherit     mGSr
              G0 = ag*apix
end

```

It inherits the generic radiation model, and simply adds an equation specifying the missing surface value. Here:

$$G_0 = a_g \cdot a_{\text{pix}} \quad (18)$$

is the surface area of the pond.

#### 4.3. Conduction and convection

The basic conduction and convection equation describes the heat transport as a function of temperature gradient:

$$\dot{Q} = \frac{1}{\theta} \cdot \Delta T \quad (19)$$

where  $\theta$  is the thermal resistance of the transport medium. This can be rewritten in terms of entropy flow:

$$\dot{S} = \frac{1}{\theta \cdot T} \cdot \Delta T \quad (20)$$

and interpreting the above equation as a conductance:

$$\dot{S} = G \cdot \Delta T \quad (21)$$

it is found that:

$$G = \frac{1}{\theta \cdot T} \quad (22)$$

Contrary to radiation, conduction and convection are phenomena that describe the interplay between two bodies at different temperatures. The bond graph two-port model describing conductive and convective phenomena is shown in Fig. 4. The two 0-junctions denote the two interacting bodies at temperatures  $T_1$  and  $T_2$ . At the 1-junction, the temperature difference:

$$\Delta T = T_1 - T_2 \quad (23)$$

is computed. The specialized Dymola model implementing the bond graph of Fig. 4 for the convection layer above the pond is as follows:

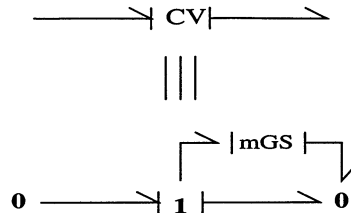


Fig. 4. Bond graph for sensible heat convection (CV).

```

model class   CVqp
inherit      TwoPort
submodel     (mGSqp) mGSqp1
submodel     (bond) B1, B2, B3
node         n1, n2
connect      B1      from A to n1
connect      B2      from n1 to B
connect      B3      from n1 to n2
connect      mGSqp1 from n2 to B
end

```

The generic convection model, called **mGSc**, takes the form:

```

model class   mGSc
inherit      TwoPort
local        G, G0
                $G \cdot e_1 = f_1$ 
                $e_1 \cdot f_1 = e_2 \cdot f_2$ 
                $G = G_0 / (e_1 + e_2)$ 
end

```

Variable  $e_1$  here denotes the effort at the input of the **mGS** element, thus,  $e_1 = \Delta T$ . Variable  $e_2$  denotes the effort variable at the output of the **mGS** element, thus,  $e_2 = T_2$ . Since the conductance should be divided by temperature  $T_1$ , the model contains in the denominator the sum of  $e_1$  and  $e_2$ :

$$e_1 + e_2 = \Delta T + T_2 = (T_1 - T_2) + T_2 = T_1 \quad (24)$$

The specialized convection model **mGSqp** describing the convection above the pond can be implemented as follows:

```

model class   mGSqp
inherit      mGSc
                $G_0 = a_g \cdot a_{\text{pix}} \cdot (b_1 + b_2 \cdot v)$ 
end

```

It inherits the generic model and adds an equation for the active surface. The term  $a_g \cdot a_{\text{pix}}$  calculates the true surface of the lake. The  $b_1$  term describes the conduction across the surface of the pond, and  $b_2 \cdot v$  describes the convection in the boundary layer above the pond, where  $v$  is the velocity of the air flow (Duffie and Beckman, 1980).

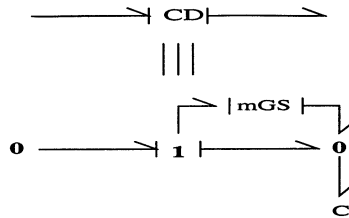


Fig. 5. Bond graph for thermal conduction (CD).

The conduction bond graph is shown in Fig. 5. It is similar to the convection bond graph, except that it doesn't correct the  $G_0$  equation for a transport term  $b_2 \cdot v$ , and it adds a capacitor to the second body. In this way, the conduction in the ground, as shown in Fig. 2, can be expressed by series connection of several conduction models without need of drawing a capacitor at each of the 0-junctions. The capacitor truly is there. It has simply been included in the conduction model.

#### 4.4. Evaporation and condensation

The generic evaporation model takes the following form:

```

model class   mGSwl
  inherit     TwoPort
  local       G, p, svp, hum
    svp = 0.61078*exp(17.269*(e1 - 273.15)/(e1 - 35.85))
    hum = 0.62198/((pao/svp) - 1.)
    p = G*(hum - e2)
    p = e1*f1
    p = e2*f2
end

```

This model describes the transformation of sensible into latent heat. The amount of converted heat is  $p$ . On the primary side, the usual bond graph variables of thermodynamics are used, where  $e_1$  is the temperature of the evaporating body, and  $f_1$  is the entropy flow consumed in the conversion.

At this point, it may be useful to remind the reader of the primary postulate of the bond graph methodology: each power flow can be expressed as a product of two adjugate variables, an *effort* and a *flow*. In thermodynamics, the power flow is a sensible heat flow. The effort variable was (arbitrarily) defined to be the temperature. Consequently, the flow variable had to denote 'sensible heat flow per temperature.' A new name was coined to denote this quantity. It was called 'entropy flow'.

The power on the secondary side of the evaporation model denotes latent heat flow. Arbitrarily, the effort variable was defined to be the relative humidity of the air, sometimes also referred to as the humidity ratio. Consequently, the flow variable had to represent 'latent heat flow per relative humidity'. The thermodynamics literature hasn't introduced a name for this quantity.

The conversion of sensible to latent heat is driven by the difference between the humidity of air at saturation level,  $hum$ , and the current humidity,  $e_2$  (Rohsenow et al., 1985). The humidity at saturation is computed from the saturation vapor pressure,  $svp$ , which in turn, is calculated according to Teten's law.

The calculations of the specific conductances  $G$  for the pond, the vegetation, and the soil proceed in similar ways to those for the convection model. For example, the evaporation from the pond is described as follows:

```

model class   CWqp
  inherit     mGSwl
    G = ag*apix*b1*lambda/(1.005 + 1.859*e2)
end

```

where  $\lambda$  is the heat evaporation coefficient of water, and the expression in the denominator describes the specific heat of moist air, which changes linearly with the humidity  $e_2$  (Ashrae, 1981).

The condensation model is basically the same as the evaporation model, except that here the input side represents latent heat, whereas the output side represents sensible heat:

```

model class   mGSw2
inherit      TwoPort
local        G, p, svp, hum
    svp = 0.61078*exp(17.269*(e2 - 273.15)/(e2 - 35.85))
    hum = 0.62198/((pao/svp) - 1.)
    p = G*(e1 - hum)
    p = e1*f1
    p = e2*f2
end

```

The generic condensation model is used to describe the condensation of water at the cover. The specific model describing this effect can be written as follows:

```

model class   CWqc
inherit      mGSw2
local        flag
    G = ac*b1*lambda*flag/(1.005 + 1.859*e1)
    flag = if e1 < hum then 1 else 0
end

```

Condensation on the glass panels only takes place when the temperature of the cover is below or at the dewpoint temperature of the air. In the model, these quantities have been converted to equivalent humidity values.  $a_c$  is the true surface of the dome, which is larger than  $a_g$ , the perceived area of the biosphere complex when looking at it from the sun.

The generic condensation model is also used to describe the effects of fogging up and 'rain fall'<sup>3</sup> within the dome. The corresponding specific model can be encoded as follows:

```

model class   CWq
inherit      mGSw2
local        flag
    G = vai*b1*lambda*flag/(1.005 + 1.859*e1)
    flag = if hum < e1 then 1 else 0
end

```

In this model,  $e_1$  represents the true humidity ratio of the air,  $W_{\text{air}}$ , whereas  $hum$  denotes the saturation humidity ratio,  $W_{\text{sat}}$ , at the current air temperature,  $T_{\text{air}}$ . Fogging up and 'rain fall' only take place when  $W_{\text{air}} > W_{\text{sat}}$ , i.e. when the air is saturated with humidity. The model is almost identical to the condensation model, except that the dome surface,  $a_c$ , is replaced by the dome volume,  $vai$ , since dew can be formed anywhere within the atmosphere.

<sup>3</sup> Fog has been observed quite frequently in Biosphere 2 during the closure period, especially after sun set. True rain cannot develop because the distance from the floor to the ceiling is too small for rain drops to form. However, dew does form as a consequence of oversaturated air when the temperature drops.



Dew, just like condensation on the glass panels, describes a phenomenon of convective transformation of latent to sensible heat. When dew drops are formed, energy is freed up, since liquid water is at a lower energy level than water vapor. This energy is absorbed by the atmosphere, which consequently heats up. In the bond graph of Fig. 2, 'rain fall' is shown as a convective **CW** flow emanating at the 0-junction that describes the moisture of the atmosphere, and ending at the 0-junction that describes its temperature.

This completes the description of the individual library modules. The specific modules that have not been listed here for space limitations are similar to the ones that have been provided. They were left out, because their inclusion would not have added any significant information to the paper. The entire program containing all the library modules can be found on the world wide web at the URL: [http://www.ece.arizona.edu/~cellier/ecoeng\\_97.html](http://www.ece.arizona.edu/~cellier/ecoeng_97.html).

Each of the modules is extremely short, and therefore easy to debug and maintain. More importantly, these models represent an excellent means for communicating knowledge about the physics behind the thermal comportment of Biosphere 2 to scientists and engineers interested in using the instrument.

A few of the models, such as the *SolarInput* model shown in this paper, refer to weather data files. The data needed are the *global horizontal radiation* and the *diffuse radiation* (both measured in *Btu/hr sqft*), the dry bulb temperature and wet bulb temperature (both measured in *F*), and the *wind speed* (measured in  $\text{m s}^{-1}$ ). The models referring to the data files then convert the temperatures from *F* to *K*, and *sqft* to  $\text{m}^2$ . Appropriate data can e.g. be found in Ashrae (1981). The data used in the simulation runs presented in this paper were compiled from NOAA records for the City of Tucson. Adjustments to the actual site (Oracle) should be made because of the somewhat higher altitude (lower temperatures, higher radiation), and different wind patterns, but this was not done in the study presented here.

#### 4.5. The overall model

The overall model is straightforward, but too long to be listed here. It simply invokes the individual library modules as submodels, and then connects them in accordance with Fig. 2. Each 0-junction is represented as a node in the main model. In addition, although this model is long, it is not necessary to generate it manually in Dymola. It is just as easy (and probably more user-friendly for debugging) to draw the bond graph of Fig. 2 using Dymodraw (Dynasim, 1995), and then generate the Dymola model automatically from the graphical representation.

### 5. Simulation results

In order to be able to simulate the Biosphere 2 model, initial conditions are needed for the nine state variables (the nine temperature values). Whereas some of those were available from measurements, for others, there were no direct measurement values available. In order to determine appropriate initial conditions for all

state variables, the following procedure was used. Starting from estimated values on January 1, 1995, a whole year was simulated, and the values of the state variables were determined at the end of the year. These values were then used as new initial values in a second simulation run. The final values were sufficiently independent of the chosen initial conditions to make an iteration unnecessary.

Fig. 6 shows the simulated inside air temperature as a function of time over the entire year. The high-frequency oscillation is the daily temperature variation. The low-frequency oscillation is the seasonal temperature variation. The temperature values were converted back from K to C for better readability. Notice that the lowest temperature measured during night hours in January is 7°C, whereas the highest temperatures measured during late June is 53°C. Clearly, these temperature variations would not be acceptable in reality, but they are plausible given the fact that the air conditioning system was not simulated.

Fig. 7 shows the simulated temperature variation of the water inside the pond (ocean). The daily variations are much smaller than for the air, but the seasonal variations are almost identical. Again, this would be totally unacceptable, given the fact that the pond is populated by tropical fish and a coral reef.

Fig. 8 shows the annual variation of the air humidity ratio within Biosphere 2. The humidity ratio basically follows the temperature variation. The results obtained are reasonable, but not accurate, because the plants have been modeled in a rather crude way up to now.

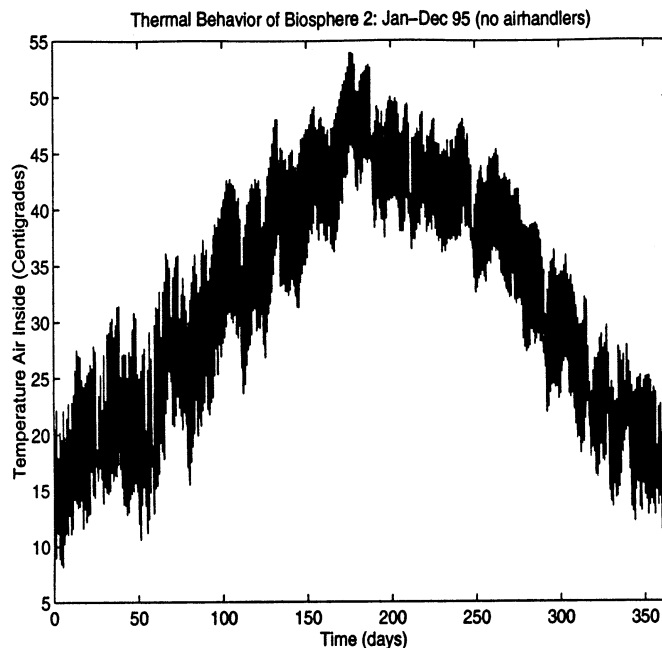


Fig. 6. Simulated annual variation of air temperature inside Biosphere 2 without air conditioning.

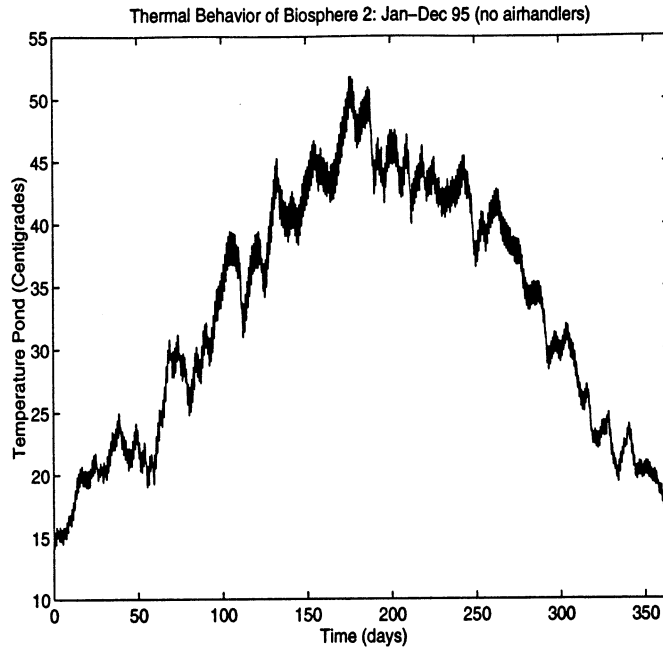


Fig. 7. Simulated annual variation of water temperature of the pond (ocean) without air conditioning.

Fig. 9a shows an excerpt of Fig. 6 denoting the simulated temperature variation during the first three days of June. The average temperature, without air conditioning, is  $\approx 43^{\circ}\text{C}$  with a temperature variation of  $\approx 10^{\circ}\text{C}$  between the daily high and the nightly low temperatures. The minima are reached at  $\approx 05:00$ , and the temperature rises quickly as soon as the sun comes up. It peaks at  $\approx 14:00$ . Fig. 9b shows the air temperature variation for the first three days in December 1995. The average temperature is now around  $22^{\circ}\text{C}$ . The daily temperature variation is still  $\approx 10^{\circ}\text{C}$ .

Fig. 10a shows the simulated air humidity ratio for the same three days in December. The curve generally follows that of the temperature variation (Fig. 9b). When the sun comes up in the morning, the temperature rises rapidly, and with it rises the saturation vapor pressure. The air is no longer completely saturated, and the fog dissolves quickly for a few hours. Fig. 10b shows the relative humidity of the air calculated as:

$$hump = \min\left(100\% \cdot \frac{W_{\text{air}}}{W_{\text{sat}}}, 100\%\right) \quad (25)$$

during the same first three days of December. The sun breaks through the fog for at least a few hours each day.

Clearly, this is not what really happens inside Biosphere 2. The air conditioning system extracts hot air from the savannah region during daytime hours, and cools the air down. Thereby, the water vapor pressure is reduced, and large quantities of

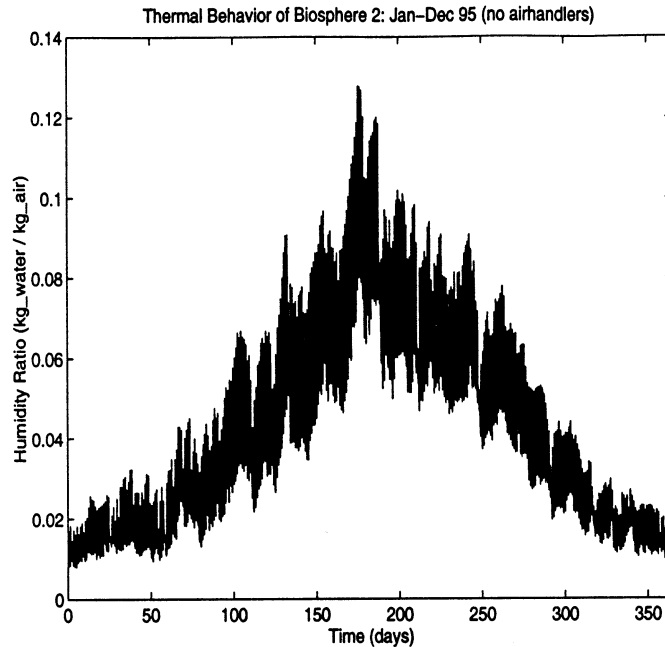


Fig. 8. Simulated annual variation of air humidity ratio inside Biosphere 2 without air conditioning.

liquid water are extracted from the air. This water is reintroduced into the system as artificial rain over the tropical rain forest, and also to replenish the pond to compensate for the evaporation.

Remember that the system is materially closed, i.e. the total amount of water available is constant. However, it is not all kept as moisture in the air, which would be the case without proper air conditioning. This means that the air conditioning system not only controls the temperature of the biomes within Biosphere 2, but also the moisture content of the air. In reality, Biosphere 2 is usually kept fog-free except over the rain forest. In the winter, the glass panels fog up as the evening sets in, because the inside temperature is kept warmer than the outside temperature, and the cover cools down sufficiently to allow condensation to take place. However, since the air conditioning system is not contained in the current model yet, its effects do not show up in the simulation results.

## 6. Conclusions

In this work, a model capturing the thermic behavior of Biosphere 2 in a non-controlled environment, i.e. without the active temperature control units, has been presented. In a future research effort, this model will be enhanced by adding the effects of the control elements such as: air handlers, fans, and heat exchanger

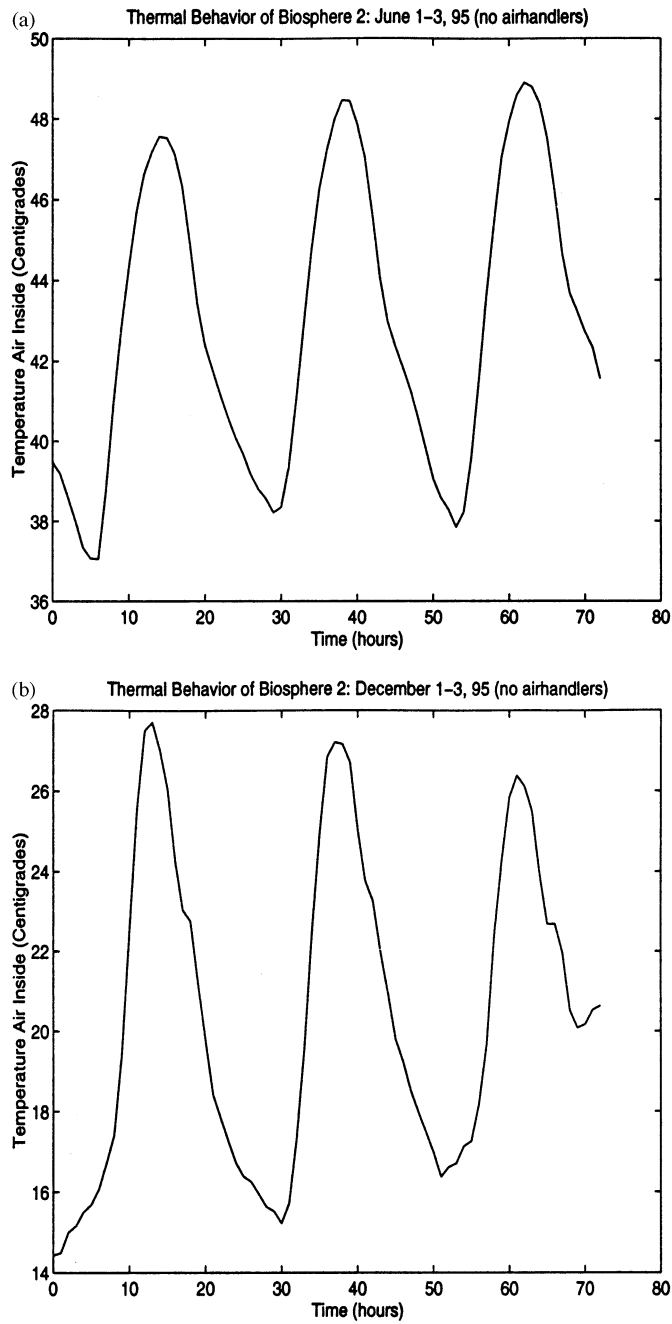


Fig. 9. Air temperature inside Biosphere 2, (a) June 1–3, 1995; (b) December 1–3, 1995.

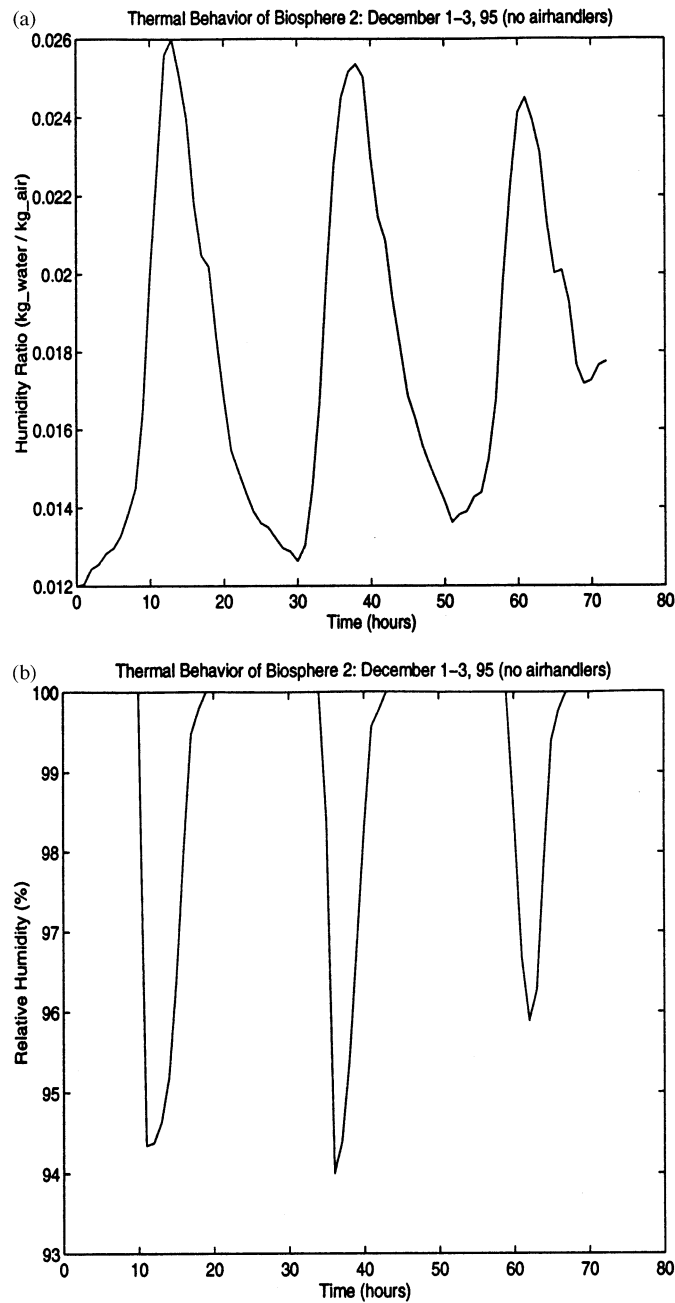


Fig. 10. Simulated air humidity ratio and (b) relative air humidity inside Biosphere 2, December 1–3, 1995.

coils. The controlled environment model will be an excellent tool for studying the system under different operational conditions. Such a tool is not currently available to the scientists designing experiments to be performed in Biosphere 2. Consequently, each new experiment presents a risk to the integrity of the system. The new model, once fully developed, shall reduce these risks and allow the conceptualization and optimization of scientific experiments prior to their actual implementation in Biosphere 2.

The bond graph modeling paradigm turned out to be an excellent choice for describing the different thermic properties of Biosphere 2 in a modular, object-oriented framework.

The Dymola software is excellently suited for supporting this modeling methodology, and the resulting Dymola models proved very useful for communicating knowledge about the physics underlying the thermal comportment of the instrument.

## Acknowledgements

Angela Nebot acknowledges gratefully the financial support received for the research described in this article by the *Direcció General de Recerca* of the *Generalitat de Catalunya* in the form of a *ajuda per estancies postdoctorals per a joves investigadors (BE)*, grant number 1995BEAI300068.

## References

- Ashrae, 1981. ASHRAE Handbook of Fundamentals. American Society of Heating, Refrigerating, and Air Conditioning Engineers, Atlanta, Georgia.
- Cellier, F.E., 1991. Continuous System Modeling. Springer-Verlag, New York, p. 755.
- Cellier, F.E., Elmqvist, H., Otter, M., 1995. Modeling from physical principles. In: Levine, W.S. (Ed.), The Control Handbook. CRC Press, Boca Raton FL, pp. 99–108.
- Duffie, J.A., Beckman, W.A., 1980. Solar Engineering of Thermal Processes. John Wiley and Sons, New York.
- Dynasim, 1995. Dymodraw Users' Manual Dynasim AB, Lund, Sweden.
- Dynasim, 1997. Their world wide web page can be visited at the URL <http://www.dynasim.se/>.
- Elmqvist, H., 1975. Simnon: An Interactive Simulation Program for Nonlinear Systems. MS Thesis, Report CODEN:LUTFD2/(TFRT-7502). Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.
- Elmqvist, H., 1978. A Structured Model Language for Large Continuous Systems. Ph.D. Dissertation. Report CODEN:LUTFD2/(TFRT-1015). Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.
- Elmqvist, H., 1995. Dymola: Dynamic Modeling Language, User's Manual. Dynasim AB, Lund, Sweden.
- Elmqvist, H., Åström, K.J., Schönthal, T., Wittenmark, B., 1990. Simnon: User's Guide for MS-DOS Computers. SSPA Systems, Gothenburg, Sweden.
- Korn, G.A., 1989. Interactive Dynamic-System Simulation. McGrawHill, New York.
- Luttmann, F., 1990. A Dynamic Thermal Model of a Selfsustaining Closed Environment Life Support System. Ph.D. Dissertation, University of Arizona, Tucson.

- Mathworks, 1992. Simulink: User's Manual, Mathworks, South Natick, MA.
- MGA, 1991. ACSL: Advanced Continuous Simulation Language: Reference Manual, 10th ed. Mitchell and Gauthier, Concord, MA.
- Nebot, A., Cellier, F.E., Mugica, F., 1996. Modeling the thermal behaviour of Biosphere II in a non-controlled environment using bond graphs. Sixth Annual Conference on AI, Simulation, and Planning in High Autonomy Systems, San Diego, California, March 25–27, pp. 189–196.
- Otter, M., 1992. DSblock: A Neutral Description of Dynamic Systems. Version 3.2, Technical Report TR R81-92, DLR, Institute for Robotics and System Dynamics, Wessling, Germany.
- Otter, M., 1996. Dymosim: User's Manual. Dynasim AB, Lund, Sweden.
- Rohsenow, W.M., Hartnett, J.P., Ganić, E.N., 1985. Handbook of Heat Transfer Applications, 2nd ed. McGraw-Hill, New York.