

Agent-based modeling of the energy network for hybrid cars

José María Gonzalez de Durana*, Oscar Barambones

*University College of Engineering, University of the Basque Country, Nieves Cano 12,
01006 Vitoria-Gasteiz, Spain*

Enrique Kremers

*European Institute for Energy Research (Electricité de France & Karlsruhe Institute of
Technology), Emmy-Noether-Strasse 11, 76131 Karlsruhe, Germany*

Liz Varga

Complex Systems Group, Cranfield University, Cranfield, United Kingdom

Abstract

Studies in complex energy networks devoted to the modeling of electrical power grids, were extended in previous work, where a computational multi-layered ontology, implemented using agent-based methods, was adopted. This structure is compatible with recently introduced Multiplex Networks which using Multi-linear Algebra generalize some of classical results for single-layer networks, to multilayer networks in steady state. Static results do not assist overly in understanding dynamic networks in which the values of the variables in the nodes and edges can change suddenly, driven by events, and even where new nodes or edges may appear or disappear, also because of other events. To address this gap, a computational agent-based model is developed to extend the multi-layer and multiplex approaches. In order to demonstrate the benefits of a dynamical extension, a model of the energy network in a hybrid car is presented as a case study.

Keywords: complex systems, agent based modeling, electrical grid, smart

*Corresponding author

Email address: josemaria.gonzalezdedurana@ehu.es
(José María Gonzalez de Durana)

Nomenclature

m	mechanical energy	I_d	input intensity for torque reference
e	electrical energy	I_t	torque controlled intensity output
g	chemical energy	k_c	torque control tuning coefficient
h	thermal energy	PS	power-split system
p	power flow	s, c, p, r	sun, carrier, planetary, ring
l	load flow	F_{sc}, F_{cr}	interaction forces
i, j	type indices $\in \{m, e, g, h\}$	$\omega_s, \omega_p, \omega_r$	planetary angular velocities
S_i	i -type input socket	τ_s, τ_p, τ_r	planetary torques
p_i, l_i	i -type p or l	r_s, r_p, r_r	planetary radii
p_{ij}, l_{ij}	p or l converted from i to j	d_{sc}, d_{cr}	planetary angular dampings
c_{ij}	fractional coupling factors	a_{ij}, b_{ij}	A, B state space matrix entries
r_{ij}	output store coupling factors	c_{ij}, d_{ij}	C, D state space matrix entries
η_{ij}	conversion efficiencies	g	gravity acceleration
e_i	stored energy	M	mass of vehicle
DC	direct current	$v_{abs_{air}}$	velocity of vehicle
u, i	motor electrical voltage or current	α	road slope
R	motor induced resistance	R_{air}	air resistance
L	autoinduction coefficient	R_{slope}	weight resistance at slop
J	moment of inertia	R_{roll}	asphalt friction resistance
B	viscous friction coefficient	ρ	air density
ω	angular velocity	C_d	vehicle drag coefficient
τ	mechanical torque	Ad	vehicle frontal area
V_t	supply voltage to motor	τ_{ext}	external resistance torque
k_t	motor torque constant	D	$\frac{d}{dt}$ operator
T_e	torque input to motor	SD	System Dynamics

grid, renewable energy systems, microgrids modeling, multi-carrier energy systems.

1. Introduction

For some time now, the importance of Smart Energy Networks has been published repeatedly by energy organizations in Europe and worldwide [1], [2], [3]. These networks use advanced information and communication technologies to monitor and manage the energy flows from multiple energy sources to meet the demand of end users.

A large number of studies and articles that deal with combined mechanical, electrical, chemical, heat [4] and other technologies have been published. However, as far as we know there is no standard representation in which one can schematically view all the system, even at a top level. So current

published studies are difficult to understand mainly because they do not use a clear way of representation but instead they adopt technology-close special drawings to try to explain things.

On the other hand there are mathematical like studies which use standard notations and graphic elements so they are very easy to understand for all people who know these standard notations. So it seems clear that any step towards standard representations of these complex systems could help to understand their associated applications.

One of those mathematical like studies are electrical systems. These systems are modeled as networks, using a weighted graph where node weights represent voltages and edge weights represent intensities in the network.

In the case of combined mechanical, electrical, chemical, heath and other technologies, using a multilayer network may be a convenient method for establishing models. And although its mathematical background is in a developing phase today (using multilinear algebra), its conceptual ideas can be used now as well as the representation scheme arising from them. The main idea here is to try to extend the graph-like ideas and representations used in electrical networks, to the more complex multilayer networks [5], using layers to represent the different energy kinds [6].

This idea is not original, but it has been used before by a number of authors [7], relevant to our studies.

In our opinion energy networks are the best candidate to represent complex combined (mechanical, electrical, chemical, heath, etc.) systems [8], [9], [10], due to the following reasons:

- They admit a graph-type representation.
- Multilayer and multiplex network theory, at developing state, can be used for modeling in some instances.
- Practically any physical system can be represented as an energy network, from the smaller ones at biological levels to the bigger ones at the entire world level [11].
- Some experiments may be done in which further “energy” carriers will be added, such as money flow, product flow, etc. and try to model the influence from one to the others.

Our approach consists of using these ideas to develop an agent based modeling method which serves at the same time to represent the system and to calculate the flows in the network.

2. Previous work

The idea of a network, its rationale being founded on classical graph theory, has been widely used to represent complex systems like the electricity grid, microgrids [12], transport networks, fluid and mechanical structures and lately more complex systems such as computer networks, communication systems, socio-technical systems [13], [14, 15, 16], the Internet, etc. Major development of science and technology during the last two centuries, has however been implemented in a piecemeal manner. Different engineering disciplines (mechanical, electrical, chemical, energy, communications, etc.) have developed mathematical apparatus and tools and, although based in graph theory, evolved iteratively to simplify disciplinary needs with the result that the basis of the method is recognizable only by true specialists in each discipline.

This situation has worsened further with the advent of progressive software tools, also increasingly specialized in each discipline, so that some specialist engineers only know how to manage some specific software tools for solving special (even complicated) application problems. But today, in order to solve problems relating to networked complex systems, in which different energy flows are involved, interdisciplinary teams are required, and the mismatches in disciplinary specializations (language, methods, etc.) creates significant barriers to integration.

As a way to deal with this situation, an abstract and rather general agent-based representation, in which the idea of *hub* [7], has been proposed [17].

In order to arrange that, previous work of the authors on electrical power grids [18] and microgrids [19], was extended to include complex energy networks, by modeling a computational multi-layered approach using agent-based methods.

The main contribution of this approach, which the authors try to show, in part, in this paper, is that it allows representation of multi-carrier energy networks, using a single sheet view in which the different energy flows (mechanical, electrical, chemical, heat etc.) are shown at once, and also calculates them dynamically in a local way at each agent.

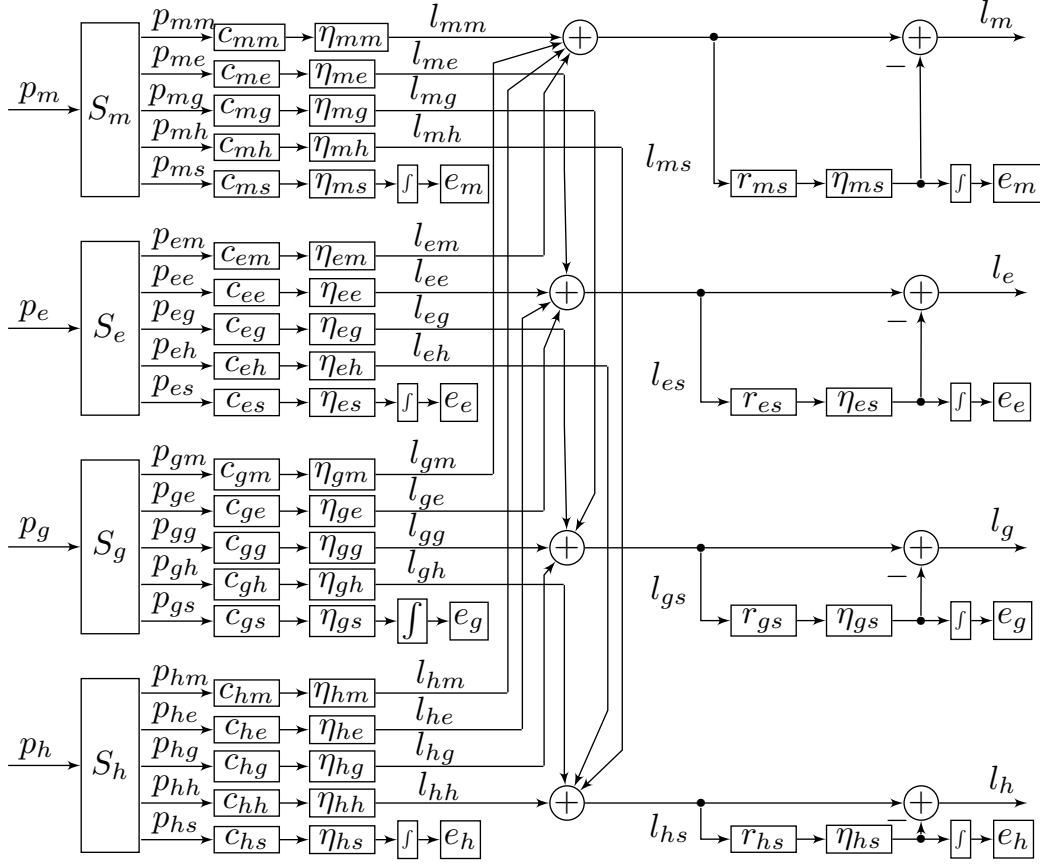


Figure 1: Hub internal diagram

As far as the authors know, no other studies with similar features have been published yet.

Using this scheme, a model with five layers, one layer for each energy type (mechanical, electrical, chemical and thermal), and another one for communications, was developed.

A description of these methods was published in [17], but only one example was provided due to space limitations, in which a microgrid specified in [20] was slightly modified in order to apply them. It is complicated to define a scenario and model it, because of the large amount of data required for any instance, even small.

Therefore, looking for some simple but complete instance, we have chosen

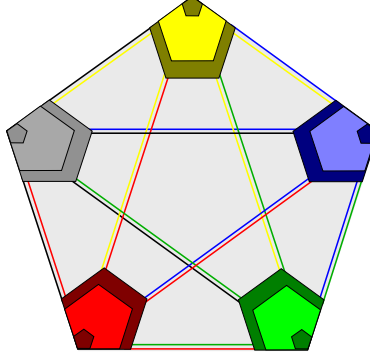


Figure 2: Hub internal graph

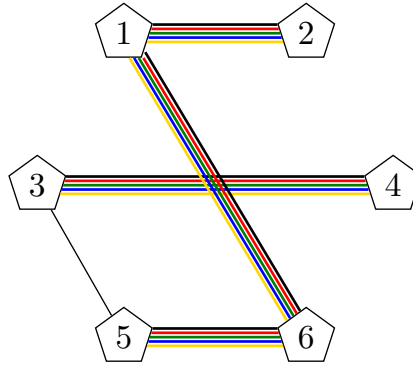


Figure 3: Multilayer network

the modeling of a hybrid car. There are well documented descriptions of some of them, the model is reduced in size and includes the four types of energy (m, e, g, h) referred in our approach.

Furthermore, the choice may be interesting to see the flexibility, versatility and generality of our method because just as in the former study calculations were made for three-phase alternating current, using the known *Power-Flow* algorithm, in this case the method of flows calculation is totally different, solving directly the differential equations arising at each node, using the System Dynamics computational method, by which the values of the dynamic variables at the nodes are calculated for each time t , and then, from these values, the power flows are calculated in a concurrent way.

3. Modeling a hybrid car

Hybrid cars have been studied and manufactured during the last few years so their dynamic equations for modeling are known because publications exist about the topic [21], [22]. As the Toyota Hybrid System has been one of most studied and documented [23], [24], it has been chosen to apply our multi-layer energy methods as described in this paper.

3.1. The energy network in a hybrid car

The Toyota Hybrid System built for Prius (THS-II) cars is a nice example to show our layered energy network, because it includes the four energy kinds: mechanical, electrical, chemical and thermal.

In the sequel the model of a DC motor is explained in detail, as a way to introduce the SD paradigm. After that, this model is extended to model a torque controlled DC motor and after that the model of a Planetary System (PS) is illustrated. Finally the two motors MG1 and MG2 and the PS are assembled to perform a model of the THS.

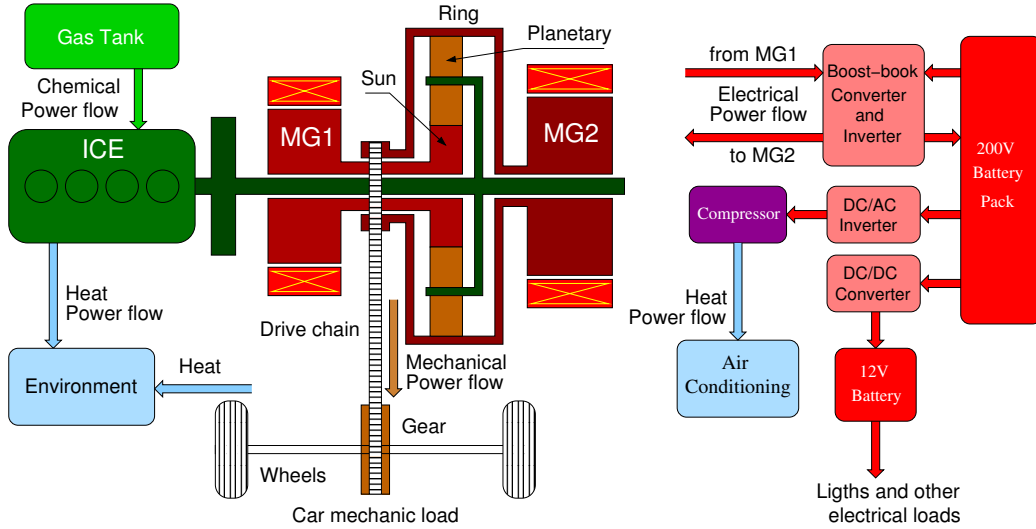


Figure 4: Hybrid car multi-carrier energy system

The main data and dynamic equations have been taken from [25] and [26]. The model was built using all three methods available in AnyLogic: System Dynamics (SD), Discrete Event (DE) and Agent Based (AB). The

power-split device and each motor were modeled using SD to implement the energy-based graphical modeling technique named *Power-Oriented Graphs* (POG) [27].

3.2. DC motor

In order to simplify the model, DC motors are assumed to be used, instead synchronous poly-phase motors used in actual cars. The system of ordinary differential equations of a DC motor is

$$\begin{aligned}\frac{di}{dt} &= \frac{1}{L}(-Ri + u) \\ \frac{d\omega}{dt} &= \frac{1}{J}(-B\omega + \tau) \\ u &= V_t - k_t \omega \\ \tau &= k_t i - T_e\end{aligned}\tag{1}$$

3.2.1. Torque controlled DC motor

Only one more equation

$$V_t = (R - k_c)I_t + k_c I_d + k_t \omega\tag{2}$$

is necessary to add to the DC motor system of equations (1) to obtain the torque controlled DC motor system.

3.3. THS Planetary

The dynamic equations of the elastic model of the planetary gear in state space form, given in [25, Fig. 3, p. 2971], can be written as

$$\begin{aligned}\begin{bmatrix} \dot{F}_{sc} \\ \dot{\omega}_p \\ \dot{F}_{cr} \end{bmatrix} &= \begin{bmatrix} 0 & a_{12} & 0 \\ a_{21} & a_{22} & a_{23} \\ 0 & a_{32} & 0 \end{bmatrix} \begin{bmatrix} F_{sc} \\ \omega_p \\ F_{cr} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & 0 \\ b_{21} & b_{22} & b_{23} \\ 0 & b_{32} & b_{33} \end{bmatrix} \begin{bmatrix} \omega_s \\ \omega_p \\ \omega_r \end{bmatrix} \\ \begin{bmatrix} \tau_s \\ \tau_c \\ \tau_r \end{bmatrix} &= \begin{bmatrix} 0 & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ 0 & c_{32} & 0 \end{bmatrix} \begin{bmatrix} F_{sc} \\ \omega_p \\ F_{cr} \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} & 0 \\ d_{21} & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix} \begin{bmatrix} \omega_s \\ \omega_p \\ \omega_r \end{bmatrix}\end{aligned}\tag{3}$$

where $a_{12} = r_p$, $a_{21} = -r_p$, $a_{22} = -r_p^2 d_{sc} - b_p - r_p^2 d_{cr}$, $a_{23} = -r_p$, $a_{32} = r_p$, $b_{11} = r_s$, $b_{12} = -r_s$, $b_{21} = -r_s d_{sc} r_p$, $b_{22} = r_s d_{sc} r_p - r_r d_{cr} r_p$, $b_{23} = r_r d_{cr} r_p$, $b_{32} = r_r$, $b_{33} = -r_r$, $c_{12} = r_p$, $c_{21} = -r_p$, $c_{22} = -r_p^2 d_{sc} - b_p - r_p^2 d_{cr}$, $c_{23} = -r_p$, $c_{32} = r_p$, $d_{11} = r_s$, $d_{12} = -r_s$, $d_{21} = -r_s d_{sc} r_p$, $d_{22} = r_s d_{sc} r_p - r_r d_{cr} r_p$, $d_{23} = r_r d_{cr} r_p$, $d_{32} = r_r$, $d_{33} = -r_r$

3.4. Vehicle mechanical model

The block diagram of the POG [27] of the vehicle mechanic transmission is shown in [25], Fig. 8 (see also its equivalent SD diagram in Figure 10). In this diagram, the resistance force F_r represents the forces applied to the car by the environment. The model takes into account the external loads to the vehicle, such as the air resistance R_a , the rolling resistance R_{roll} , the friction force with the asphalt f_r and the vehicle weight component on slopes R_{slope} .

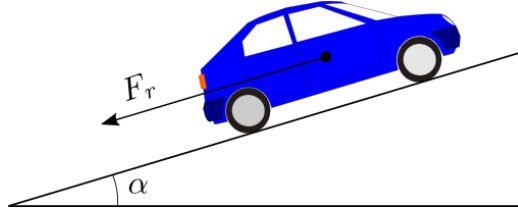


Figure 5: Car resistance forces

To implement the wind load, an arbitrary wind speed is considered within a given range, according to the current geographical area which, together with the vehicle speed, the absolute speed $v_{abs_{air}}$ is obtained. With this speed and the density of air ρ , the drag coefficient of the vehicle C_d and its front area A_d , the air resistance R_{air} pushing back the vehicle is obtained.

$$R_{air} = 0.5 \rho C_d A_d (v_{abs_{air}}^2)$$

For the weight action of the vehicle on slopes, a given angle α is considered within a possible range, and together with the values of vehicle mass M , gravity acceleration g , the resistance caused by the weight of the vehicle R_{slope} is obtained, depending on the angle α of the road.

$$R_{slope} = M g \sin(\alpha)$$

For the implementation of the friction with the asphalt R_{roll} , the values of the the vehicle mass M , the acceleration due to gravity g and the friction coefficient of the asphalt f_r are used.

$$R_{roll} = M g f_r$$

The random values that dynamically modify external loads of the vehicle are modified by an event that is activated in a user-specified time.

Finally, the three resistance values are added and multiplied by the radius of the wheels R_w to obtain the value of the external resistance torque τ_{ext} applied to the vehicle.

$$\tau_{ext} = (R_{air} + R_{slope} + R_{roll})R_w \quad (4)$$

3.5. Power Control Unit

The THS-II PCU is a complex electronic and computerized system whose description is outside the scope of this work. However, since it is a discrete event system, a brief description of it is given below, which could be used as a basis for designing or improving the system.

3.6. Other elements

The internal combustion engine (ICE), batteries, electronic converters and heat transmission system are not objects of modeling in this paper and they were simulated using appropriate parameters and tables in the AnyLogic model.

4. Implementation in AnyLogic

AnyLogic has friendly graphical user interfaces for System Dynamics (SD), Discrete Event (DE) and Agent Based (AB) methods, which simplify the implementation of models. Initially used mostly by businesses management collectives, its qualities are also being recognized in other fields [28], [29].

To model ODE systems using SD, a **Stock** variable must be used for each unknown function and also a **Flow** variable for its corresponding derivative.

To make a SD model, the user creates a new **Agent Type (Active Object Class in AnyLogic 6)**, from which a new (gridded) blank page appears. Then he drags the appropriate graphical elements from the SD palette to the blank page and then draws the appropriate dependence arrows. Finally he enters the corresponding numeric value or expression for each variable, into the appropriate properties box which opens by clicking at each of them.

4.1. DC motor

We are going to implement the model of the DC motor given in (1) using the SD method. To begin, we click **File** → **New** → **Agent Type**, whereupon a new blank page opens, in which we place the right graphical elements from the System Dynamics palette.

For the left hand side of each differential equation we should include two SD variables: a **Stock** variable and a **Flow** variable. We use the capital letter D to denote ordinary derivatives:

$$D := \frac{d}{dt}$$

Once the variables have been placed at some points in the page, we draw the dependences given at the right hand side of each equation, using arrows.

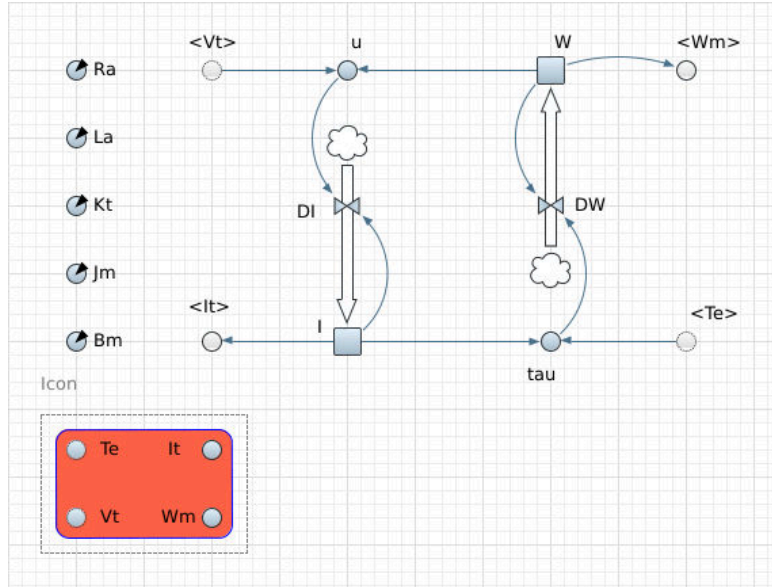


Figure 6: SD diagram of a DC motor

Figure 6 shows the SD diagram obtained. It contains *stock* variables W , I , *flow* variables DI , DW , *auxiliary* variables u , τ , T_e , V_t , I_t , W_m , and *parameters* R_a , L_a , K_t , J_m , B_m (on the left). There are also arrows indicating the SD dependences between variables.

The *shadow* variables $\langle Vt \rangle$, $\langle Wm \rangle$, $\langle It \rangle$, $\langle Te \rangle$, act as dummy copies of their respective auxiliary variables and are used for encapsulating the dc-motor object for using as a component in models. To do this we drag a **Rounded Rectangle**, from the **Presentation** palette, and select some of its **Properties**: **Fill Color** = **tomato**, ☒ **Visible on upper level**, ☒ **Icon**, and place input and output variables within it.

Once the diagram is built, the values for parameters and the expressions for variables must be entered. This is made simple by clicking at each respective graphical symbol. However, for flexible use we used a tricking mechanism: parameters do not appear directly in ODE expressions, but other variables are used instead. To do so these variables must be created and some values must be assigned to them. So


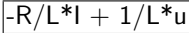
- Clicking at the gridded blank page, the **Properties** window of **Motor Agent Type** opens and then the text

double R=2, L = 0.06, kt=2.5, J = 0.53, B = 0.08;

is entered in the **Additional class code** box. In this way, these variables are created and they take the given values. These values can be any because they will be changed at startup time: in the same window, in **On startup** box, we enter the text

$R = Ra; L = La; kt = Kt; J = Jm; B = Bm;$

so that the initial values of ODE variables are assigned to be equal to their respective parameter values.

- Clicking for example at **Ra** parameter, its **Properties** window opens and then the value **Ra=1.33** can be entered in the **Default value** box. Repeat for the other parameters. These parameter values can be changed at instantiation time (ie when this object is used as a component for some model), but even also they could be changed (p driven by events) while the model is running. As you can see, the initial values given to variables are really not used. But it is necessary to give some value to them to avoid execution errors (perhaps this could be improved).
- Clicking at **DI** flow variable, its **Properties** window opens with the box  in which we enter 

- Clicking at DW flow variable, its Properties window opens with the box $\boxed{DW=}$ in which we enter $\boxed{-B/J*W + 1/J*\tau}$
- Clicking at u auxiliary variable, its Properties window opens with the box $\boxed{u=}$ in which we enter $\boxed{Vt - kt*W}$
- Clicking at tau auxiliary variable, its Properties window opens with the box $\boxed{\tau=}$ in which we enter $\boxed{kt*I - Te}$
- Clicking at It auxiliary variable, its Properties window opens with the box $\boxed{It=}$ in which we enter \boxed{I}
- Clicking at Wm auxiliary variable, its Properties window opens with the box $\boxed{Wm=}$ in which we enter \boxed{W}

The last two variables It and Wm will act as outputs of the model block, so they must be declared as ☒ Visible on upper level (or External in Anylogic 6) in their Properties window.

There are another two variables, Te and Vt that will act as inputs of the model block, so they must be declared as ☒ Visible on upper level and also as ☒ Dependant in their Properties window.

Once the object (called Agent in AnyLogic 7) has been finished, it can be used as an element, like a block, to build models in which is used as a component. To do so we simply drag the agent into the blank page of the model (usually Main if it corresponds to the main model).

4.2. Torque controlled DC motor

The SD diagram can be obtained from the DC motor one, simply by adding a new auxiliary variable Id and, clicking at Vt, its Properties window opens with the box $\boxed{Vt=}$ in which $\boxed{(R-kc)*It + kc*Id + kt*W}$ should be entered.

Note that Vt variable which was declared as ☒ Dependant in the DC motor model, now must be unset at this property and Id variable must be set as ☒ Dependant, because now it is an input.

Also a new parameter Kc has been added, whose value can be set by clicking at it, its Properties window opens with the box $\boxed{\text{Default value}}$ in which we enter $\boxed{0.2}$.

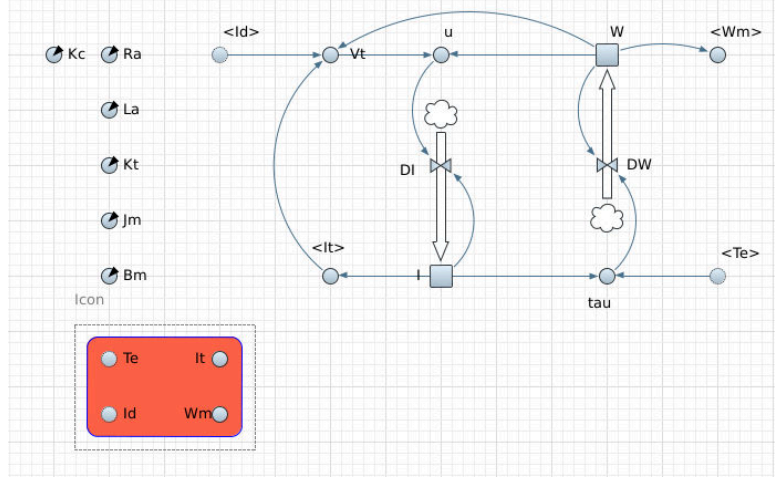


Figure 7: SD diagram of a torque controlled DC motor

4.3. THS Planetary

The dynamic model of the planetary system (3) is implemented in the same way as the previous objects. After clicking **File** → **New** → **Agent Type**, a blank page opens that we name for example **Planetary**, and then we place the desired graphical elements from the System Dynamics palette on it.

Since we have three derivatives \dot{F}_{sc} , \dot{w}_p and \dot{F}_{cr} , three flow variables are necessary which we call **DFsc**, **DWp** and **DFcr** and its corresponding stock variables **Fsc**, **Wp** and **Fcr**. Also we used **Ws**, **Wc**, **Wr** for inputs and **Ts**, **Tc**, **Tr** for outputs. Each of these six variables have a corresponding shadow variable, denoted $\langle \cdot \rangle$ with the same name, for encapsulating the object and making the icon. Note that the three inputs must be declared as **Dependent** auxiliary variables, and all inputs and outputs must be declared as **Visible on upper level**.

In this way we obtained the SD diagram shown in Figure 8. Once the diagram is built, values for parameters and expressions for variables must be entered. This is made by clicking at each respective graphical symbol. To simplify the process, we defined some more auxiliary variables for variables a_{ij} , b_{ij} , c_{ij} , d_{ij} , $1 \leq i, j \leq 3$ in Equation (3). Then,

- In the **Properties** window of **Planetary Agent Type** we enter the following text [25, p.2971]:

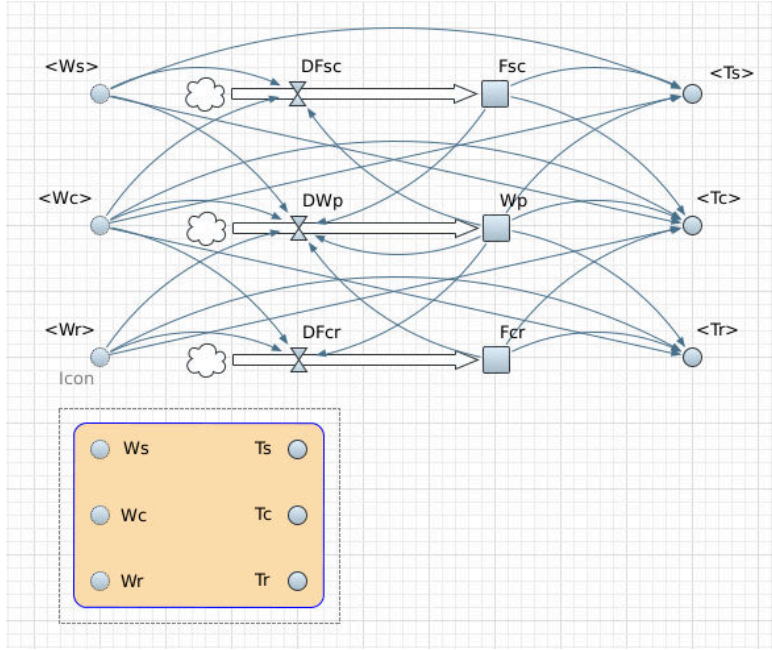


Figure 8: THS Planetary

```
// Parameters
double Nr=78, Ns=30, Np=23;
double rs=0.102, rr=0.248, rp=Np*rs/Ns;
double Jr=0.0081, Js=1.48e-4, Jc=0.0071;
double Ksc=1e7, Kcr=1e7;
double Jice=0.2596, Jp=Jice+Jc;
double dsc=300, dcr=300;
double bp=0, bs=0, bc=0, br=0;
// matrix elements
// A = -Le^(-1)*Ae
double a12=Ksc*rp;
double a21=-rp/Jp,
a22=(-rp*rp*dsc-bp-rp*rp*dcr)/Jp,
a23=-rp/Jp;
double a32=Kcr*rp;
// B = -Le^(-1)*Be
double b11=Ksc*rs, b12=-Ksc*rs;
```

```

double b21=-rs*dsc*rp/Jp,
      b22=(rs*dsc*rp-rr*dcr*rp)/Jp,
      b23=rr*dcr*rp/Jp;
double b32=Kcr*rr, b33=-Kcr*rr;
// C = Ce
double c11=rs, c12=rs*dsc*rp;
double c21=-rs, c22=-rs*dsc*rp+rr*dcr*rp,
      c23=rr;
double c32=-rr*dcr*rp, c33=-rr;
// D = De
double d11=bs+rs*rs*dsc, d12=-rs*rs*dsc;
double d21=-rs*rs*dsc,
      d22=bc+rs*rs*dsc+rr*rr*dcr,
      d23=-rr*rr*dcr;
double d32=-rr*rr*dcr, d33=br+rr*rr*dcr;

```

in the Additional class code box. No code is entered in On startup box, because there are no parameters to initialize in this case.

- Clicking at DFsc flow variable, its Properties window opens with the box $\boxed{DFsc=}$ in which we enter $\boxed{a12*Wp + b11*Ws + b12*Wc}$
- Clicking at DWp flow variable, its Properties window opens with the box $\boxed{DWp=}$ in which we enter $\boxed{a21*Fsc + a22*Wp + a23*Fcr + b21*Ws + b22*Wc + b23*Wr}$
- Clicking at DFcr flow variable, its Properties window opens with the box $\boxed{DFcr=}$ in which we enter $\boxed{a32*Wp + b32*Wc + b33*Wr}$
- Clicking at Ts auxiliary variable, its Properties window opens with the box $\boxed{Ts=}$ in which we enter $\boxed{c11*Fsc + c12*Wp + d11*Ws + d12*Wc}$
- Clicking at Tc auxiliary variable, its Properties window opens with the box $\boxed{Tc=}$ in which we enter $\boxed{c21*Fsc + c22*Wp + c23*Fcr + d21*Ws + d22*Wc + d23*Wr}$
- Clicking at Tr auxiliary variable, its Properties window opens with the box $\boxed{Tr=}$ in which we enter $\boxed{c32*Wp + c33*Fcr + d32*Wc + d33*Wr}$

Ws, Wc, Wr are inputs declared as visible, public and dependent.

4.4. THS Model

Once the models of the torque controlled DC motor and Planetary mechanical system have been built, we can build the THS placing (by dragging them from the **Projects** tab) two instances of the Tcmotor object, that we call Tc_MG1 and Tc_MG2, and an instance of the Planetary object into the Main gridded blank sheet, and connecting them, as figure 11 shows.

Now, clicking at each Tcmotor instance, we can enter its actual parameter values.

Parameter	Value
Name	Tc_MG2
Visible	yes
Single agent	Selected
Population of agents	Unselected
Ra	1.33
Jm	0.00128
Kt	2
La	0.06
Bm	0.07
Kc	1

Figure 9: MG2 parameters

4.5. Vehicle transmission

Obtaining the SD model from the block diagram shown in [25], Fig. 8, is pretty straightforward, by replacing each integrator ($1/s$) block for a type flow variable and placing the remaining variables as auxiliary variables, as Figure 10 shows, where k_t is the transmission stiffness, d_t is the transmission friction coefficient, R_t is the transmission ratio, J_w is the wheel inertia, R_w is the wheel radius, K_w is the tire longitudinal stiffness, b_w is the friction coefficient and M_v is the mass of the vehicle.

The torque τ_t is the transmission torque and τ_b is the resistant torque at the wheels,

$$\tau_b = \tau_{brake} + \tau_{ext}$$

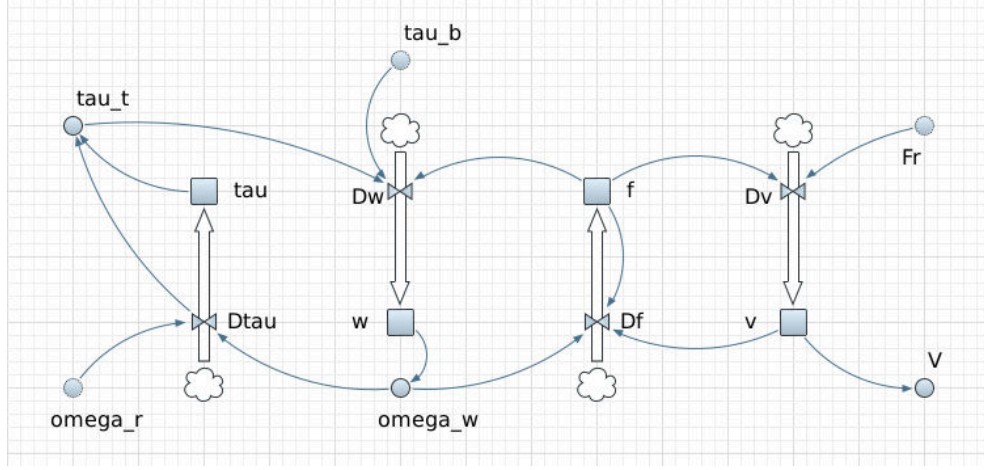


Figure 10: Car transmission SD diagram

where τ_{brake} is the braking torque and τ_{ext} is the external resistant torque before calculated in Equation (4). Using these formulas a new active object, called *Vehicle* was implemented and connected to the other components, as it is shown in Figure 11.

4.6. Power Control Unit

The control of energy management of the hybrid vehicle is a complex system that dynamically varies depending on internal and external vehicle parameters and environment parameters.

We used the UML statechart approach offered by AnyLogic for designing the system. A statechart with 8 states and substates was created to determine the operation of the vehicle.

The main state in which the vehicle is demanding energy, is the **Motion_mode** state. Within this, the vehicle will start in the **Start mode** state where the initial values of parameters are loaded. Parallel to this state is the **Normal_mode** state, in which it is assumed that the vehicle is demanding energy to accelerate or maintain constant the vehicle speed.

Within **Normal_mode** state, there are the **Cruise_mode** and **Accelerate_mode** states, from which the vehicle is maintaining a constant velocity and charging the battery, or is accelerating, respectively. These states also differ in that the battery is charging in the first one, but not in the second one, although this also depends on various parameters such as for instance the battery level.

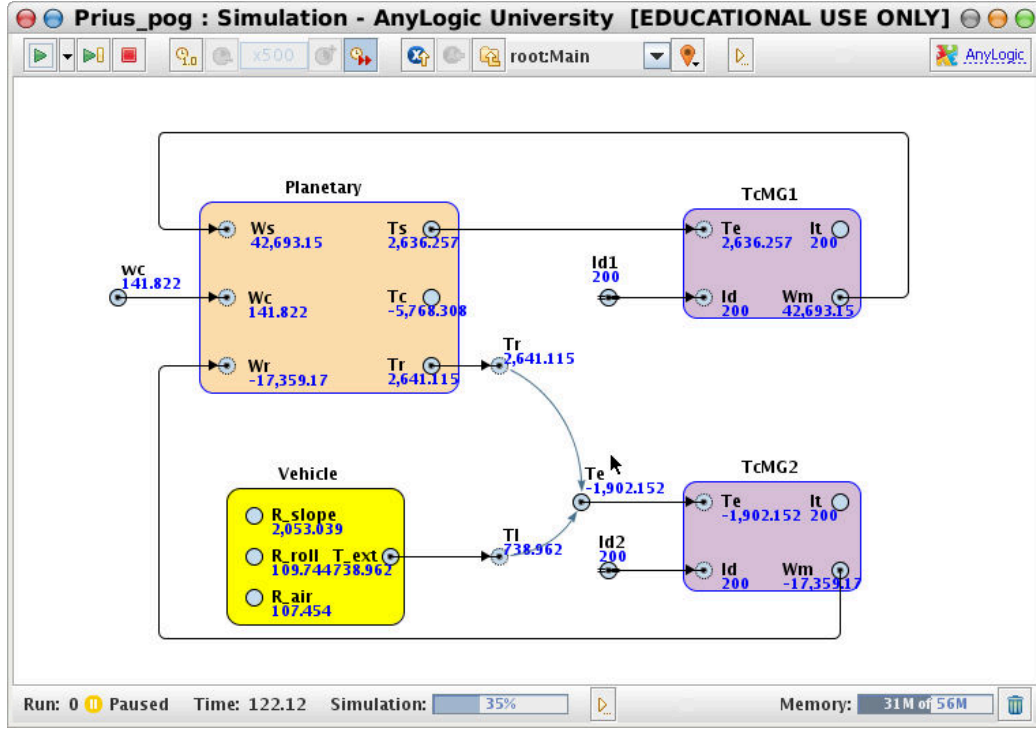


Figure 11: THS simulation

Finally, parallel to the **Motion** state is the **Brake_mode** state in which it is considered that the vehicle is braking.

All these states include parameters within them for use of the different engines and its management is really involved, beyond the scope of these notes.

The Power Control Unit (PCU) was modeled using DE (Fig. 12).

Also the other elements in THS-II were modeled. Once all of them were ready, definition of the network nodes (hubs) for this special case and integration of them into the appropriate multilayer network structure was done.

Each hub was modeled as an agent with a layer for each energy type (mechanical, electrical, chemical and thermal) and one more for communications, as Fig.13 shows.

Then the entire multilayer energy network was built using these agents and some other elements of the AB paradigm:

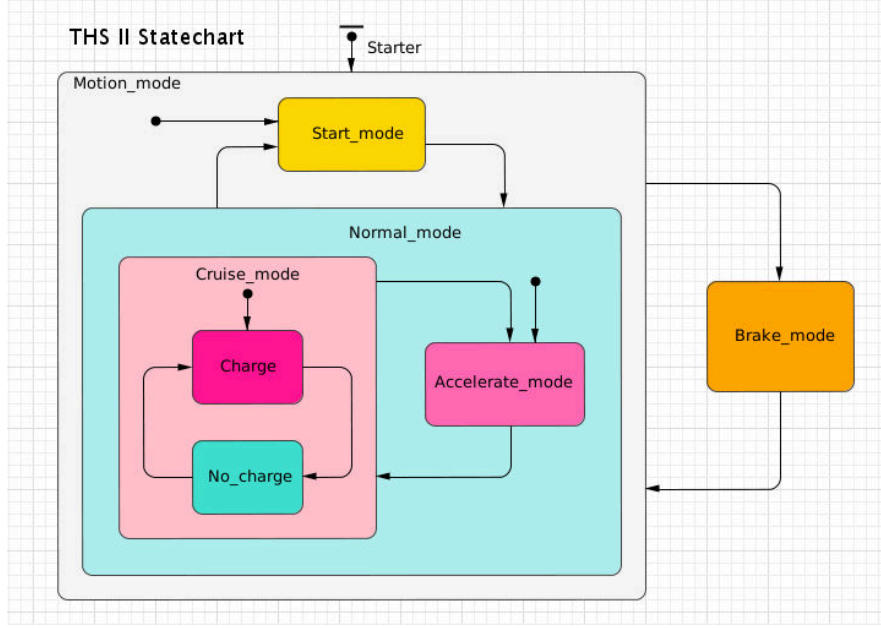


Figure 12: Statechart for THS

Velocities and torques in the Power Split device: ω_s , T_s (red, sun), ω_c , T_c (green, planetary) and ω_r , T_r (brown, ring), are similar to those obtained by other authors [25].

5. Model simulation

Figure 14 show a simulation window in which the different energy flows can be observed. The diagram consists of a special graph, where the nodes represent the energy transformations points, with names corresponding to their associated physical elements, and the colored edges represent the different flows: mechanical flow (brown), electrical flow (red), thermal flow (blue) and chemical flow (green). Also colored small numbers are shown at the nodes, representing the power flows entering each node (with positive sign if flow is entering and negative if it is leaving the node). These numbers change dynamically during the simulation.

Further graphics plots can be obtained for each element at which a more detailed output data representation is required. For example, Figure 15 shows the main variables, torques and angular velocities, in the power split device.

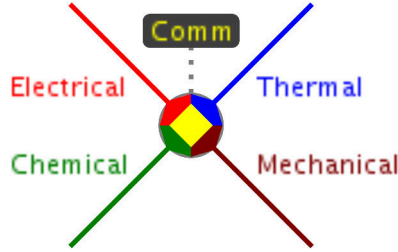


Figure 13: Hub represented as agent

6. Conclusion

An agent based modeling approach to represent and calculate multicarrier energy networks has been developed, from which a specific example showing a good part of its capabilities is explained in this article.

After comparing the results with those of other published versions of this well known system, the Toyota Hybrid System, noting that they are similar, we can say that our method works well for this example.

But our method has a number of advantages and improvements, when compared with other methods used as standard modeling tools, which can be summarized as follows.

- First, and in our opinion the most important, it provides a modeling method based on agents, for multicarrier energy networks.
- It allows the system representation on a single sheet, in which energy flows circulating in the system can be observed dynamically during simulation.
- All main power flow calculations are made in a local way at the agents which represent energy hubs.
- It is technology independent.
- It can be applied to a large class of dynamic systems provided they admit a multi-layer energy representation.

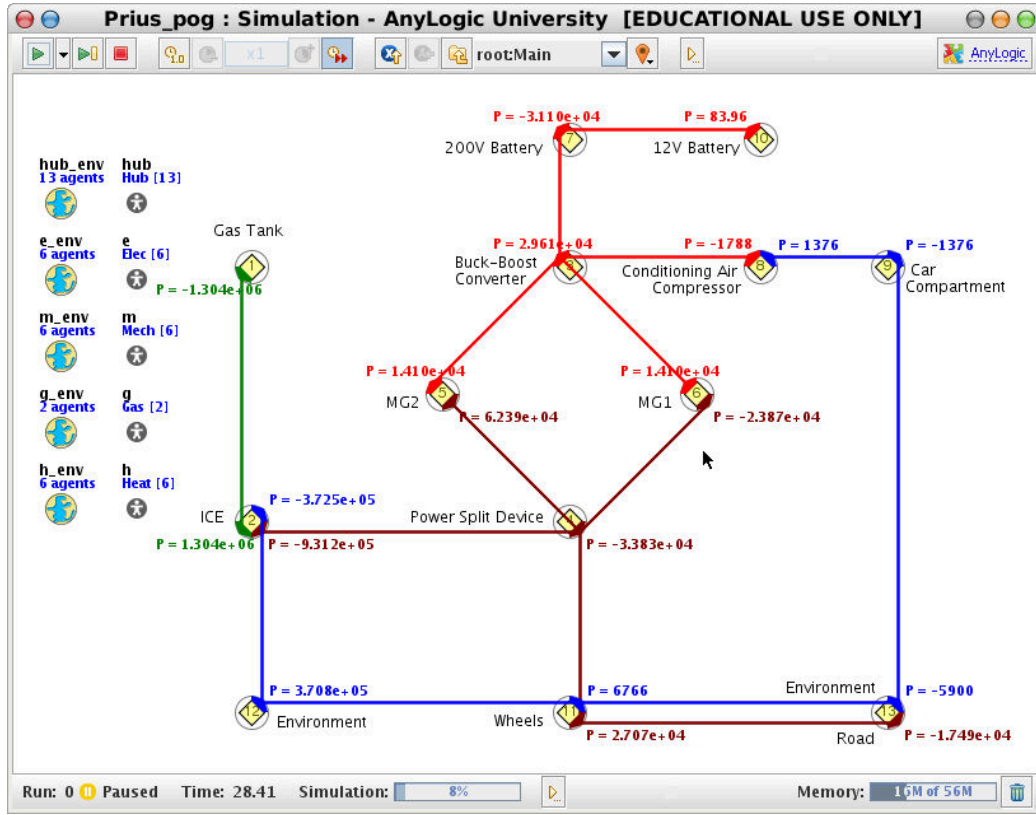


Figure 14: Simulation window of the energy network in the car

In the authors' opinion, our methods may be complementary to others developed on "multilayer-networks" or multiplex-networks. As they, of course, let you make calculations, they require the use of multilinear algebra which may involve certain degree of mathematical difficulty. But our methods are easier to apply, as they do not require a complicated mathematical formulation, and further allow the incorporation of events in the calculations, they may drop nodes, and communication between nodes, which is essential for the simulation and possible subsequent implementation of intelligent, multilayer energy networks.



Figure 15: Angular velocities and torques in the power split device

7. Future extensions

While the present structure of five layers can be used for modeling a large class of systems, allowing modeling with exogenous variables, cooling system technologies, fuel-cells, alternative fuels, etc., future extensions may include more energy carriers, representing flows of goods and money and then, taking into consideration that management commands and responses could be modeled using the communication agents, perhaps it might be possible to model some simple socio-technical case studies.

8. Acknowledgements

Dr Gonzalez de Durana and Dr Barambones are very grateful to the Basque Government by the support of this work through the projects S-PE12UN015 and S-PE13UN039, and to the UPV/EHU by its support through the projects GIU13/41 and UFI11/07. Dr Varga kindly acknowledges funding from both EPSRC grant EP/K012347/1 “International Centre for Infrastructure Futures” and EC Framework 7 grant FP7- 613194 “EU-InnovatE” Sustainable Lifestyles 2.0.

References

- [1] Prasenjit Basak, S. Chowdhury, S. Halder nee Dey, and S.P. Chowdhury. A literature review on integration of distributed energy resources in the perspective of control, protection and stability of microgrid. *Renewable and Sustainable Energy Reviews*, 16(8):5545–5556, 2012.
- [2] B.V. Mathiesen, H. Lund, D. Connolly, H. Wenzel, P.A. Østergaard, B. Möller, S. Nielsen, I. Ridjan, P. Karnøe, K. Sperling, and F.K. Hvelplund. Smart energy systems for coherent 100solutions. *Applied Energy*, 145(0):139 – 154, 2015.
- [3] Paul E. Dodds, Iain Staffell, Adam D. Hawkes, Francis Li, Philipp Grünewald, Will McDowall, and Paul Ekins. Hydrogen and fuel cell technologies for heating: A review. *International Journal of Hydrogen Energy*, 40(5):2065 – 2083, 2015.
- [4] L. Yang, E. Entchev, M. Ghorab, E.J. Lee, and E.C. Kang. Energy and cost analyses of a hybrid renewable microgeneration system serving multiple residential and small office buildings. *Applied Thermal Engineering*, 65(1-2):477 – 486, 2014.
- [5] S. Boccaletti, G. Bianconi, R. Criado, C.I. del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendiña-Nadal, Z. Wang, and M. Zanin. The structure and dynamics of multilayer networks. *Physics Reports*, 544(1):1 – 122, 2014. The structure and dynamics of multilayer networks.
- [6] R. Niemi, J. Mikkola, and P.D. Lund. Urban energy systems with smart multi-carrier energy networks and renewable energy generation. *Renewable Energy*, 48(0):524 – 536, 2012.

- [7] M. Geidl. *Integrated Modeling and Optimization of Multi-carrier Energy Systems*. PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, 2007.
- [8] Mattias Vesterlund and Jan Dahl. A method for the simulation and optimization of district heating systems with meshed networks. *Energy Conversion and Management*, 89(0):555 – 567, 2015.
- [9] E. Saloux, A. Teyssedou, and M. Sorin. Development of an exergy-electrical analogy for visualizing and modeling building integrated energy systems. *Energy Conversion and Management*, 89(0):907 – 918, 2015.
- [10] Robert D. Raine, Vida N. Sharifi, and Jim Swithenbank. Optimisation of combined heat and power production for buildings using heat storage. *Energy Conversion and Management*, 87(0):164 – 174, 2014.
- [11] Sgouris Sgouridis and Denes Csala. A framework for defining sustainable energy transitions: Principles, dynamics, and implications. *Sustainability*, 6(5):2601–2622, 2014.
- [12] Eklas Hossain, Ersan Kabalci, Ramazan Bayindir, and Ronald Perez. Microgrid testbeds around the world: State of art. *Energy Conversion and Management*, 86(0):132 – 153, 2014.
- [13] L. Varga. Transforming critical infrastructure. *International Innovation*, pages 35–37, 2013.
- [14] L Varga, T Grubic, P Greening, S Varga, F Camci, and T Dolan. Characterizing conversion points and complex infrastructure systems: Creating a system representation for agent-based modeling. *Complexity*, pages 35–37, 2014.
- [15] Catherine S.E. Bale, Liz Varga, and Timothy J. Foxon. Energy and complexity: New ways forward. *Applied Energy*, 138(0):150 – 159, 2015.
- [16] Marguerite Robinson, Liz Varga, and Peter Allen. An agent-based model for energy service companies. *Energy Conversion and Management*, 94(0):233 – 244, 2015.

- [17] J Gonzalez de Durana, O Barambones, E Kremers, and L Varga. Agent based modeling of energy networks. *Energy Conversion and Management*, 82(0):308 – 319, 2014.
- [18] E Kremers, J Gonzalez de Durana, and O Barambones. Emergent synchronisation properties of a refrigerator demand side management system. *Applied Energy*, 101(0):709–717, 2013. Sustainable Development of Energy, Water and Environment Systems.
- [19] E Kremers, J Gonzalez de Durana, and O Barambones. Multi-agent modeling for the simulation of a simple smart microgrid. *Energy Conversion and Management*, 75(0):643 – 650, 2013.
- [20] E. Kremers, P. Viejo, J. M. González de Durana, and O. Barambones. A complex systems modelling approach for decentralized simulation of electrical microgrids. In *15th IEEE International Conference on Engineering of Complex Computer Systems*, Oxford, 2010.
- [21] Mihael Cipek, Danijel Pavković, and Joško Petrič. A control-oriented simulation model of a power-split hybrid electric vehicle. *Applied Energy*, 101(0):121 – 133, 2013. Sustainable Development of Energy, Water and Environment Systems.
- [22] Guang Wu, Xing Zhang, and Zuomin Dong. Powertrain architectures of electrified vehicles: Review, classification and comparison. *Journal of the Franklin Institute*, 352(2):425 – 448, 2015. Special Issue on Control and Estimation of Electrified vehicles.
- [23] Guizhou Ren, Guoqing Ma, and Ning Cong. Review of electrical energy storage system for vehicular applications. *Renewable and Sustainable Energy Reviews*, 41(0):225 – 236, 2015.
- [24] Kehe Wu and Huan Zhou. A multi-agent-based energy-coordination control system for grid-connected large-scale wind-photovoltaic energy storage power-generation units. *Solar Energy*, 107(0):245 – 259, 2014.
- [25] F. Grossi, M. Fei, and R. Zanasì. Dynamic modeling and control of power-split hev with multi-phase electric machines under fault condition. In *2013 European Control Conference (ECC)*, pages 2969–2975, July 2013.

- [26] Roberto Zanasi and Federica Grossi. Modelling hybrid automotive systems with the pog technique. *Journal of Asian Electric Vehicles*, 8(1):1379–1384, 2010.
- [27] R. Zanasi, F. Grossi, and N. Giuliani. Extended and reduced {POG} dynamic model of an automatic corking machine for threaded plastic caps. *Mechatronics*, 24(1):1 – 11, 2014.
- [28] Jiayun Zhao, Sadik Kucuksari, Esfandiyar Mazhari, and Young-Jun Son. Integrated analysis of high-penetration {PV} and {PHEV} with energy storage and demand response. *Applied Energy*, 112(0):35 – 51, 2013.
- [29] Sgouris Sgouridis and Denes Csala. A framework for defining sustainable energy transitions: Principles, dynamics, and implications. *Sustainability*, 6(5):2601–2622, 2014.

Agent-based modeling of the energy network for hybrid cars

Gonzalez de Durana, Jose Maria

2015-04-17

Gonzalez de Durana JM, Barambones O, Kremers E, Varga L (2015) Agent-based modeling of the energy network for hybrid cars, Energy Conversion and Management, Volume 98, July 2015, pp. 376-386

<http://dx.doi.org/10.1016/j.enconman.2015.04.003>

Downloaded from CERES Research Repository, Cranfield University