Spatially distributed systems

aka Distributed-parameter systems

Zdeněk Hurák

January 5, 2020

All the models in mechanical and electrical domains that we derived so far were completely void of any *spatial* information. We assumed that either the systems stretch across negligible area (or fill a negligible volume) or that considering the spatial dimension is irrelevant since the variables of interest are identical throughout the whole system.

The first assumption was reasonable for electronic circuits provided they were not operated at too high frequencies. What happens at high frequencies (say, GHz and higher) is that the wavelengths become comparable with the size of the components and systems, hence they must be modeled as spatially distributed. We do not typically encounter this in electronics for automation systems.

The second assumption was perfectly valid for rigid-body mechanical systems. The translation velocities of all the elementary particles comprising the cart sliding over a flat surface are identical. It is sufficient to replace the whole bulky object with a single point (of the identical mass) and analyze the motion of the point. Of course, when rotation motion is superposed on the translation motion, each elementary volume of the object travels with a different translation velocity, but this does not add much more complication—the whole resulting motion can be easily characterized as a composition of the two velocities upon introducing the moment of inertia of the object.

On the other hand, we have already encountered an instance of a spatially distributed system in a hydraulic domain. While learning how to model inertia and friction of a fluid flowing in a pipe using bond graphs, we proposed to model the two phenomena as an R and I elements attached to a type-1 junction. But we found it confusing to see that the analogous electronic circuit contained the resistor connected in series with the inductor. Translating this back into the hydraulic setting, this would suggest that some section of the pipe will be purely frictional whereas the inertia effect will only be seen in the adjacent section of the pipe, which is not corresponding to the reality. Intuitively we feel that both the inertia and friction must be considered *continuously* along the pipe. But we were happy with the lumped model since it can provide reasonable accuracy and waved hands at the problematic interpretation.

Along the same line of reasoning, while discussing heat transfer, we considered two key phenomena: heat conductance and heat accumulation. When modeling heat transfer through a slab of some material (say, a wall of your house), we modeled each of the two phenomena in their respective layer. However, in reality there need not be any physical layers in the material, they are just our mental constructs. The whole wall could be 3D-printed from a concrete. Intuitively, both the conduction and the accumulation take place everywhere throughout the material. But we followed the multilayer approach to obtain a convenient lumped model in the form of a bond graph.

In this lecture we will take a bit more systematic approach to such systems. Note, however, that we are only able to scratch the surface here because the topic of spatially distributed systems (aka distributed-parameter systems) is overwhelmingly vast and whole courses and textbooks are dedicated to it.

There are some fundamental differences between the spatially distributed and lumped systems. The former can be described by partial differential equations (PDE) whereas the latter can be modeled using ordinary differential equations (ODE). While there is now a solid unified theory for ODEs, there is nothing like that available for PDEs. Instead, PDEs are classified into a few families and for each family its own bunch of theoretical results have been derived. This makes studying spatially distributed systems hard.

On the other hand, while the essence of modeling of lumped systems was in clever assembling of possibly a large number of coupled equations (that is what we used bond graphs or Euler-Lagrange equations for), there will be no such challenge for us in spatially distributed systems. Typically we will be given a physical phenomenon such as heat transfer, chemical diffusion or vibration, for which the equations are already well known and cataloged in textbooks. The only remaining modeling activity is in finding the values of physical parameters and setting the boundary conditions properly. Occasionally, a combination of two or three phenomena might appear but this still does not constitute a true challenge in assembling the equation into a single model. The true challenge with models described by PDEs will be to *analyze* the model, typically by means of *numerical simulations*.

The topic of numerical solution of PDEs is still being intensively studied but a lot of useful methods exist now. We have no ambitions in this course (and this single lecture) to give even an overview. Instead, we will see one particular way of systematic approximation of a PDE by a set of coupled ODEs—the *method of lines* (MOL). This will allow us to model spatially distributed systems using bond graphs.

1 Heat transfer—heat equation

1.1 Deriving the heat equation

First we consider heat transfer through a layer that separates two bulks of material, each at their respective temperatures θ_1 and θ_2, see Fig. 1.

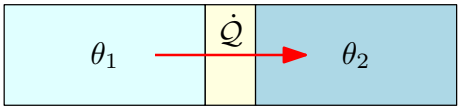

Figure 1: Two objects separated by a partial thermal isolation through which they can exchange energy by heat transfer. Each object is characterized by its temperature θ_1 and θ_2. We want to determine the heat flow $\dot{Q}$

Fourier's law tells us that the heat flow depends on the difference between the two temperatures. In the simplest case this dependence is just linear

$$\dot{Q}(t) = \underbrace{H}_{\frac{A\lambda}{h}}(\theta_1(t) - \theta_2(t)), \tag{1}$$

where H is the thermal conductance, A is the area of the contact, h is the thickness of the layer and λ is the (coefficient of) thermal conductivity and it depends on the material of the layer.

Let's now align our considered system with a global coordinate axis x extending horizontally to the right. For the thickness of the layer going to zero, we get

$$\dot{Q}(t) = -A\lambda \lim_{h\to 0} \frac{\theta(x+h,t) - \theta(x,t)}{h}, \tag{2}$$

which yields

$$\dot{Q}(t) = -A\lambda \frac{\partial \theta(x,t)}{\partial x}. \tag{3}$$

Let's now extend the notation such that both the temperature and the heat flow are the functions of time t and spatial coordinate x. The above equation formalizes to

$$\frac{\partial Q(x,t)}{\partial t} = -\kappa \frac{\partial \theta(x,t)}{\partial x}, \tag{4}$$

which is the differential version of Fourier's law.

We can also extend it to full 3D setup, in which the heat can flow in any direction. Then the differential version of Fourier's law generalizes to

$$\frac{\partial Q(x,t)}{\partial t} = -\kappa \nabla \theta(x,t), \tag{5}$$

that is, the heat flow now has a direction (it is a vector) and is given by the gradient of the temperature field, scaled by k and with the direction reverted.

Besides the heat conduction there is also a heat accumulation. Consider the scenario in Fig. 2. Heat flow $\dot{Q}_1$ is entering the slab of the material through its left wall and heat flow $\dot{Q}_2$ is leaving through the right wall. Consider in this introductory treatment that all the other walls are perfectly insulated.

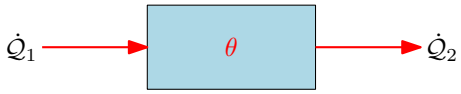

Figure 2: Thermal capacitor. Heat is entering through the left wall and leaving through the right wall. All other walls are perfectly insulated

Obviously, the difference between the two heat flows must be due to heat accumulated in the slab, which is observable through a change in the temperature. The heat accumulated in a slab of material of mass m and specific heat (heat capacity per mass) c is

$$Q_{\text{stored}}(t) = mc\theta(t). \tag{6}$$

If the temperature is not constant throughout the bulk of material, we need to rewrite the above expression in an integral form

$$Q_{\text{stored}}(t) = \int_V \rho c\theta(x,t)\mathrm{d}V. \tag{7}$$

If the density ρ and the specific heat c are constant throughout the space, they could be moved outside the integral, but we are not going to rely on it here.

Differentiating this with respect to time we get the rate at which the thermal energy accumulated in the slab is rising

$$\begin{aligned}\dot{Q}_{\text{stored}}(t) &= \frac{\mathrm{d}}{\mathrm{d}t}\int_V \rho c\theta(x,t)\mathrm{d}V \\ &= \int_V \rho c\dot{\theta}(x,t)\mathrm{d}V.\end{aligned} \tag{8}$$

Now if we assume that the only significant dimension of the slab is the horizontal (left-to-right), which ensures that the temperature of an arbitrary point only depends on its x (horizontal) coordinate, we can specialize the above integral expression into

$$\dot{Q}_{\text{stored}}(t) = \int_a^b \rho Ac\dot{\theta}(x,t)\mathrm{d}x. \tag{9}$$

Besides relating the stored energy with the temperature, it can also be related with the incoming and outgoing heat flows

$$\dot{Q}_{\text{stored}}(t) = \dot{Q}_1(t) - \dot{Q}_2(t). \tag{10}$$

Let's now shuffle this a bit. First, let's again consider a horizontal coordinate axis x. The left and right walls are at coordinates a and b, respectively. We can then rewrite the above equation using a single heat flow $Q(x,t)$ parameterized by x

$$\dot{Q}_{\text{stored}}(t) = -(\dot{Q}(b,t) - \dot{Q}(a,t)), \tag{11}$$

which can be further rewritten into

$$\dot{Q}_{\text{stored}}(t) = -\int_a^b \frac{\partial \dot{Q}(x,t)}{\partial x}\mathrm{d}x. \tag{12}$$

Now we have two expressions for $\dot{Q}_{\text{stored}}(t)$

$$\int_a^b \rho A c \dot{\theta}(x,t)\mathrm{d}x = -\int_a^b \frac{\partial \dot{Q}(x,t)}{\partial x}\mathrm{d}x. \tag{13}$$

In fact, this can be augmented with a term describing an internal generator of thermal energy (a heater), should there be one

$$\int_a^b \rho A c \dot{\theta}(x,t)\mathrm{d}x = -\int_a^b \frac{\partial \dot{Q}(x,t)}{\partial x}\mathrm{d}x + \int_a^b g(x,t)\mathrm{d}x, \tag{14}$$

where $g(x,t)$ is the generated power density.

Upon substituting for $\dot{Q}$ from (4) and removing the integral signs we get the differential equation

$$\rho A c \frac{\partial \theta(x,t)}{\partial t} = -\frac{\partial}{\partial x}\left(-\kappa \frac{\theta(x,t)}{\partial x}\right) + g(x,t). \tag{15}$$

Provided that κ is constant throughout the object, the equation simplifies to

$$\frac{\partial \theta(x,t)}{\partial t} = \alpha \frac{\partial^2 \theta(x,t)}{\partial x^2} + g(x,t). \tag{16}$$

where

$$\alpha = \frac{\kappa}{\rho A c} > 0. \tag{17}$$

This is the celebrated heat equation (for a homogeneous material). Very often in textbooks it is only shown without the term corresponding to the internal generators

$$\frac{\partial \theta(x,t)}{\partial t} = \alpha \frac{\partial^2 \theta(x,t)}{\partial x^2}. \tag{18}$$

In order to extend this result into full 3D, that is, to the scenarios in which the heat can be exchanged through all the walls, we need to go back to (11) and rewrite it using a surface integral

$$\dot{Q}_{\text{stored}} = -\oint_A (\dot{\mathbf{Q}} \cdot \mathbf{n})\mathrm{d}A, \tag{19}$$

where $\dot{\mathbf{Q}}$ is now in bold face because it is a vector and $\mathbf{n}$ is a unit vector normal to the surface. Note the minus sign which reflects the fact that by a common convention the normal vector is oriented out from the volume enclosed by the surface.

Invoking the *divergence theorem* we reshape the previous result into

$$\dot{Q}_{\text{stored}} = -\int_V \left(\nabla \cdot \dot{\mathbf{Q}}\right) \mathrm{d}V. \tag{20}$$

But after substituting from the vector version of differential Fourier's law (5) for $\dot{\mathbf{Q}}$ we get

$$\dot{Q}_{\text{stored}} = -\int_V \left(\nabla \cdot (-\kappa \nabla \theta)\right) \mathrm{d}V, \tag{21}$$

which, in case of κ constant throughout the volume eventually yields

$$\frac{\partial \theta(x,t)}{\partial t} = \alpha \Delta \theta(x,t), \tag{22}$$

where Δ stands for Laplace operator, which is defined for a scalar function θ as

$$\Delta \theta \coloneqq \nabla \cdot \nabla \theta = \frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2} + \frac{\partial^2 \theta}{\partial z^2}. \tag{23}$$

1.2 Discussion of Laplacian

Now, before we move on to exploring the heat equation, let's spend some time with this important function—the *Laplacian*. In order to get some insight into it, let's consider the 1D version first.

The second derivative $\frac{\mathrm{d}^2 f(x)}{\mathrm{d}x^2}$ characterizes the *curvature* of some univariate function f at the point x. It vanishes when the function is linear. That means that the solution of the differential equation

$$\frac{\partial^2 \theta(x,t)}{\partial x^2} = 0 \tag{24}$$

is a function with a linear graph. Well, actually affine, that is, a linear function plus some offset

$$f(x) = ax + b. \tag{25}$$

The values of the two parameters a and b have to be determined from the *boundary conditions*. For a second-order ODE, two conditions are needed. One possibility is to fix the values of the function f at the two ends of the interval, that is

$$f(a) = f_a \tag{26}$$
$$f(b) = f_b. \tag{27}$$

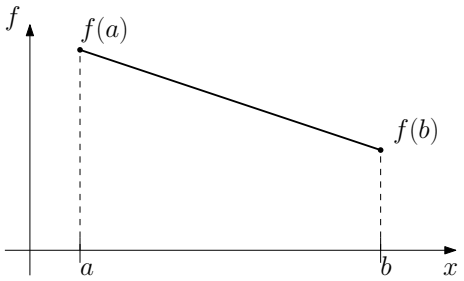

Figure 3: Solution of the differential equation $0 = f_{xx}$ for given boundary values

In other words, if we know the value of the function at the ends of the interval, we can immediately tell the value of the function everywhere inside the interval between the function just by linear interpolation as Fig. 3 illustrates

What implication does it have for the heat equation? Consider the asymptotic solution of (18), that is, consider what happens as $t \to \infty$. Obviously, the left-hand side vanishes (this is by the definition of steady state) but then

$$0 = \frac{\partial^2 \theta(x,t)}{\partial x^2}. \tag{28}$$

The implication is now clear, I suppose. After a sufficient time, the distribution of temperature along the one-dimensional domain is described by an affine graph. The actual slope and offset will be determined by the boundary conditions (the temperatures at the two ends).

The above mentioned fact introduces one important property of the steady-state solution of the heat equation—in every point in the solution domain (interval in our 1D case), the solution is given as an *average* of the values in the neighborhood (again, interval in our case).

The implications in the full 3D case are little bit less intuitive but equally important. In order to interpret the condition (called Laplace equation)

$$0 = \Delta\theta = \frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2} + \frac{\partial^2 \theta}{\partial z^2} \tag{29}$$

consider just a two-dimensional domain (for the ease of visualization), that is,

$$0 = \Delta\theta = \frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2}. \tag{30}$$

Now, consider as the solution domain a unit disk in the plane and the boundary values are given by $\theta(x, y, t) = xy$. The steady-state solution does not depend on the initial conditions, hence we do not need to know them and we can solve the Laplace equation to obtain Fig. 4.

Note that unlike in the 1D case, here the graph of the function is not just a tilted and shifted plane. After all, here it is not the second derivative(s) that

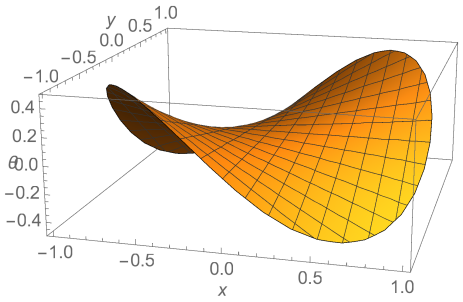

Figure 4: Solution to Laplace equation over a subset in R^2—a unit disk—for some given boundary conditions

must vanish but rather something more complicated—the Laplacian $\Delta\theta$. But the averaging property works here as well.

1.3 Heat equation vs diffusion equation

It is worth noting that this behavior can not only be found while modeling the heat conduction but also while modeling (chemical) diffusion. That is why the heat equation is sometimes also called a *diffusion equation*. It describes how a concentration of some chemical evolves in time (in absence of a chemical reaction).

1.4 Heat equation also describing the distributed consensus algorithm in networked systems

What may be perhaps even more surprising is to hear that the same principles are artificially embedded in some distributed computation schemes in networked systems. Although a computed network can hardly be regarded as spatially continuous, with a large enough nodes in the network (or agents in a multiagent system, or autonomous UAVs in a flight formation), the network can be viewed as a spatially discretized system. The individual nodes then periodically send their own value while receiving values from their neighbors and they update their value by averaging it with the received values. The process repeats, in which the network mimics the heat conduction or diffusion phenomenon. Some links to the literature are in the final section containing tips on literature.

1.5 Numerical solution by the method of lines (spatial discretization)

Now that we know that the fully accurate model of heat transfer in a slab of material is a partial differential equation, we may want to analyze it. One

way to do it is through numerical solution. Among the numerical techniques, there is one that is particularly suited to the spirit of our course because it will lead to a set of ordinary differential equations. It is called the *method of lines* (MOL) and essentially constitutes in spatial discretization of the solution domain..

Once again, have a careful look at the heat equation 18

$$\frac{\partial\theta(x,t)}{\partial t} = \alpha\frac{\partial^2\theta(x,t)}{\partial x^2}. \tag{31}$$

Does not it look familiar? On the right-hand side we have the first derivative with respect to time of the variable of interest. This looks pretty much like a state-space model, right? Well, the complication is that the "variable" of interest is not just a number of n-tuple of numbers, as we are used to in lumped systems, but it is a segment of a function. You can perhaps view the state vector as comprised of an infinite number of elements. That is why we call these systems "infinite-dimensional" in contrast with the finite-dimensional systems, with which are already familiar.

The theory of infinite-dimensional systems is very involved and certainly does not belong to an undergraduate curriculum. For example, just have a look at the "matrix A" in the state-space model, that is, have a look at what kind of operator we have on the right-hand side. The operator is $\alpha\frac{\partial^2}{\partial x^2}$, which is certainly much more involved than the mere real matrix A in the finite-dimensional state-space model $\dot{x}(t) = Ax(t)$.

Elaborating on the didactic advice to view $\theta(x,t)$ at a given time t as a "very long" vector, we can come up with an intuitive finite-dimensional approximation—just replace $\theta(x,t)$ by a long vector

$$\begin{bmatrix}\theta_1(t)\\ \theta_2(t)\\ \vdots\\ \theta_n(t)\end{bmatrix} \tag{32}$$

containing approximations to $\theta(x,t)$ a grid of points, that is,

$$\theta_1(t) \approx \theta(a+h), \qquad \theta_2(t) \approx \theta(a+2h), \qquad \ldots \qquad \theta_n(t) \approx \theta(a+nh). \tag{33}$$

The boundary values are

$$\theta_0(t) = \theta(a,t) \tag{34}$$

$$\theta_{n+1}(t) = \theta(b,t). \tag{35}$$

The solution domain is visualized in Fig. 5. We can see that instead of the full stripe in the $x-t$ plane, we now only consider a set of (half-)lines, hence the name of the technique.

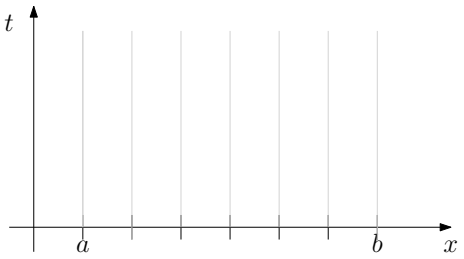

Figure 5: The solution domain for the method of lines (MOL)

What this spatial discretization brought in was that we no longer have to consider the infinite-dimensional "vectors" but rather can work with the standard (however long) vectors. However, the right hand side of our "state-space model" still needs some work. We need to express (actually approximate) the second derivative of the function using just those finite number of values that are now available. But that is a very common task. It is called *finite-difference* (FD) approximation. We have already used in while discussing numerical algorithms for ODES, do you remember? Forward Euler, Backward Euler, ... These were simple approximations of derivatives. For example, the Forward Euler scheme approximates the derivative as $\dot{x}(t) \approx \lim_{h\to 0} \frac{x(t+h)-x(t)}{h}$. For our purposes here they were not quite adequate because they were not symmetric. Intuitively we feel that the physics here calls for some symmetric approximation schemes. Such approximation is proposed by the *central difference(s)*

$$\frac{\partial \theta}{\partial x} \approx \frac{\theta(x+\frac{1}{2}h) - \theta(x-\frac{1}{2}h)}{h}. \tag{36}$$

The second derivative is then obtained by applied the same approximation scheme once again

$$\frac{\partial^2 \theta}{\partial x^2} \approx \frac{\frac{\theta(x+h)-\theta(x)}{h} - \frac{\theta(x)-\theta(x-h)}{h}}{h} \tag{37}$$

$$= \frac{1}{h^2}\left(\theta(x+h) - 2\theta(x) + \theta(x-h)\right). \tag{38}$$

Now, apply this approximation not just at any x but at those grid point $(x+h, x+2h, \ldots, x+nh)$ to spatially discretize the heat equation

$$\begin{bmatrix} \dot{\theta}_1(t) \\ \dot{\theta}_2(t) \\ \vdots \\ \dot{\theta}_n(t) \end{bmatrix} = \frac{\alpha}{h^2} \begin{bmatrix} \theta_2 - 2\theta_1 + \theta_0 \\ \theta_3 - 2\theta_2 + \theta_1 \\ \vdots \\ \theta_n - 2\theta_{n-1} + \theta_{n-2} \end{bmatrix}. \tag{39}$$

The final task is to reformat this into the popular matrix-vector state-space

format $\dot{x} = Ax + Bu$, which leads to

$$\begin{bmatrix} \dot{\theta}_1(t) \\ \dot{\theta}_2(t) \\ \dot{\theta}_3(t) \\ \vdots \\ \dot{\theta}_{n-1}(t) \\ \dot{\theta}_n(t) \end{bmatrix} = \frac{\alpha}{h^2} \underbrace{\begin{bmatrix} -2 & 1 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & \dots & 0 & 0 \\ 0 & -1 & -2 & \dots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \dots & -2 & 1 \\ 0 & 0 & 0 & \dots & 1 & -2 \end{bmatrix}}_{A} \begin{bmatrix} \theta_1(t) \\ \theta_2(t) \\ \theta_3(t) \\ \vdots \\ \theta_{n-1}(t) \\ \theta_n(t) \end{bmatrix} + \frac{\alpha}{h^2} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ \vdots & \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_0(t) \\ \theta_{n+1}(t) \end{bmatrix} \tag{40}$$

Voilà our first finite-difference approximation of a partial differential equation! The pattern of the matrix A is hard to overlook. It is a banded matrix, actually a very narrow banded matrix. It is a good idea to store the matrix in a computer as a *sparse matrix*. Other than that, we obtained a standard state-space model, which can be analyzed using standard tools from (finite-dimensional) linear systems theory.

It is fair to admit that this approach is far from superior as soon as we extend the solution domain to a subset of $\mathbb{R}^2$ or even $\mathbb{R}^3$. Since the order of the error introduced by spatial discretization should be comparable with the order of the error brought in by temporal discretization, the number of "lines" can be very high. The order of the resulting finite-dimensional system is then prohibitively high. Just imagine griding a box-like domain with $10 \times 10 \times 10$ points, which does not seem too dense. The resulting system is of order 1000, which is already quite a computational load. Nonetheless, the MOL method certainly deserves a space in our set of tools.

Let's evaluate the method by means of a numerical example. We consider a 1D segment located in the interval [0,1] on the x horizontal axis. For 20 grid points the lumped LTI model given by the MOL method can be created in Matlab using the following code.

```
%% Parameters

d = 1;                  % the length of the spatial interval
n = 20;                 % the number of points on the spatial interval
h = d/n;                % the length of a spatial segment

a = 1;

%% Building the state-space matrices

c = zeros(n-1,1);
r = zeros(n-1,1);

c(1:2) = [-2 1];
r(1:2) = [-2 1];

A = toeplitz(c,r);
A = a/h^2 * A;

B = zeros(n-1,2);
```

```
B(1,1) = 1;
B(n-1,2) = 1;
B = a/h^2 * B;

C = eye(n-1,n-1);
D = zeros(n-1,2);

G = ss(A,B,C,D);
```

If Fig. 6 we can see a visualization of the "sparsity pattern", that is, the nonzero entries of the matrix, as produced by the `spy` command in Matlab. The tridiagonal character of the matrix as derived in (39) is confirmed.

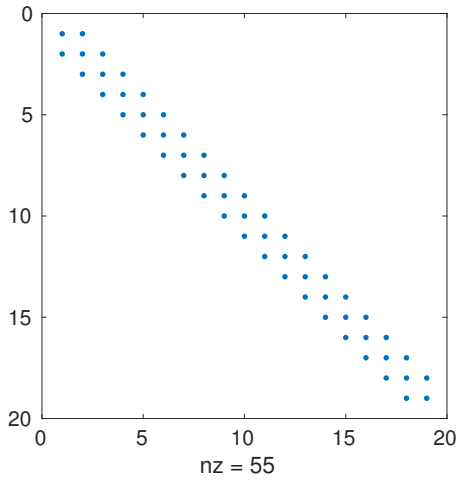

Figure 6: A "map" of nonzero entries of the state matrix A

In Fig. 7 we see three snapshots of temperature profiles in response to zero initial conditions and nonzero boundary conditions. Obviously, as time elapses, the temperature distribution approaches the linear (affine) shape as predicted.

In Fig. 8 we observe another response, this time it is a response to nonzero initial conditions (temperature profile) while both boundary conditions are kept at zero[1]

Let's formalize the two types of response into one combined problem statement. Generally, we are given a partial differential equation together with the *initial* and *boundary conditions*; we will refer to them as IC and BC, respectively. In our particular case, the PDE is first-order in time, hence we need just a single initial condition. But since this is a PDE and not an ODE, our independent variable is a function, and therefore in order to specify the initial condition, we need to specify the temperature along the whole interval! Similarly, the equation is of second-order in the spatial variable, hence we need two boundary conditions. One choice of boundary conditions is to have the

[1]For these introductory purposes I do not even bother to offset the working temperature to some more practical levels—everyone can certainly do such transformation by themselves.

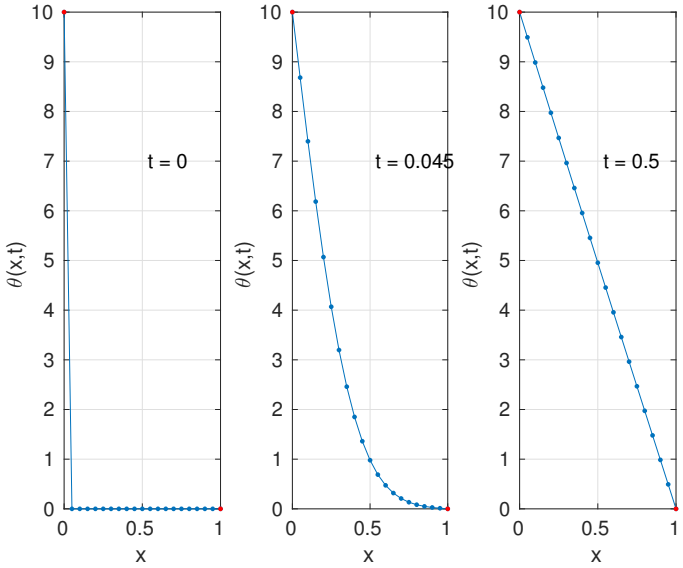

Figure 7: Snapshots of the temperature profiles at three time instances; zero initial conditions, the left boundary temperature set nonzero

values of the independent variable specified at both ends of the interval, as we suggest in the problem statement below

$$\frac{\partial\theta(x,t)}{\partial t} = \alpha\frac{\partial^2\theta(x,t)}{\partial x^2} \tag{41}$$

$$\text{IC}: \quad \theta(x,0) = f(x), \tag{42}$$

$$\text{BC}: \quad \theta(a,t) = \theta_a(t), \tag{43}$$

$$\theta(b,t) = \theta_b(t). \tag{44}$$

The boundary conditions in this case are called *Dirichlet's* conditions. Alternatively, we can specify the values of the derivatives of the independent variable at both ends, in which case the boundary conditions are called *Neumann's* conditions.

$$\frac{\partial\theta(x,t)}{\partial t} = \alpha\frac{\partial^2\theta(x,t)}{\partial x^2} \tag{45}$$

$$\text{IC}: \quad \theta(x,0) = f(x), \tag{46}$$

$$\text{BC}: \quad \left.\frac{\partial\theta(x,t)}{\partial x}\right|_{x=a} = h_a(t), \tag{47}$$

$$\left.\frac{\partial\theta(x,t)}{\partial x}\right|_{x=b} = h_b(t). \tag{48}$$

Note that by specifying $\frac{\partial\theta(x,t)}{\partial x}$ we actually specify the heat flow, that is, $\frac{\partial Q(x,t)}{\partial t}$ (see the differential Fourier's law). For example, specifying that the

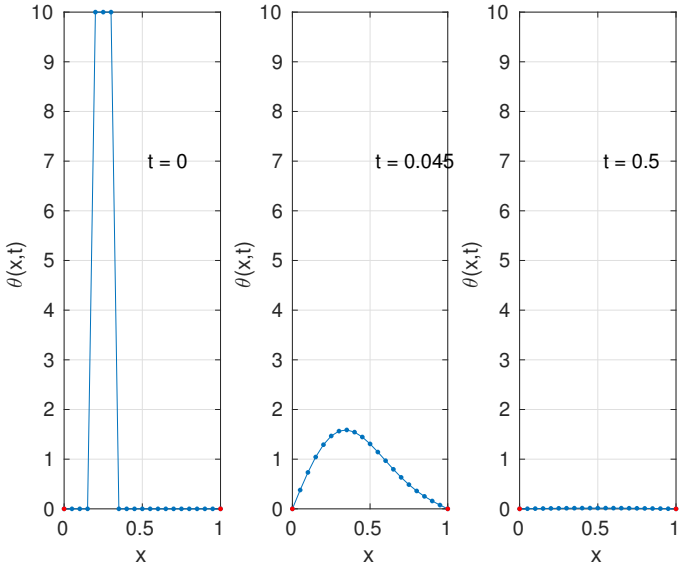

Figure 8: Snapshots of the temperature profiles at three time instances; zero boundary temperature, some nonzero initial temperature distribution

partial derivative of the temperature is vanishing at the boundaries, we are enforcing complete thermal insulation of the system.

$$\frac{\partial \theta(x,t)}{\partial t} = \alpha \frac{\partial^2 \theta(x,t)}{\partial x^2} \tag{49}$$

$$\text{IC}: \quad \theta(x,0) = f(x), \tag{50}$$

$$\text{BC}: \quad \dot{Q}_a(t) = 0, \tag{51}$$

$$\dot{Q}_b(t) = 0. \tag{52}$$

1.6 Spatial discretization with bond graphs

Now, finally, let's have a look how this is related to what we were developing during most of our course—the power bond graph modeling. In fact, we will use a pseudo-bond graph, which will model the two important phenomena—thermal resistance and thermal storage. Consider the multilayer thermal model of a heat conduction through a wall in Fig. 9.

The corresponding (pseudo-)bond graph is in Fig. 10.

We will leave the development of the state equations up to you. Make sure you can arrive at these equations

$$\begin{bmatrix} \dot{\theta}_1(t) \\ \dot{\theta}_2(t) \\ \dot{\theta}_3(t) \end{bmatrix} = \frac{1}{RC} \begin{bmatrix} -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} \theta_1(t) \\ \theta_2(t) \\ \theta_3(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{RC} & 0 \\ 0 & 0 \\ 0 & \frac{1}{RC} \end{bmatrix} \begin{bmatrix} \theta_a(t) \\ \theta_b(t) \end{bmatrix}, \tag{53}$$

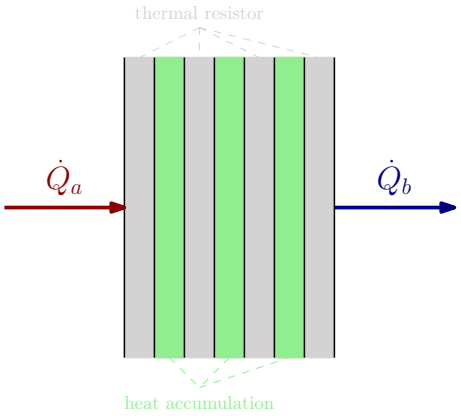

Figure 9: Sketch of a lumped approximation to a heat conduction through a slab of a material—layers representing thermal resistance are interleaved with layer representing heat storage

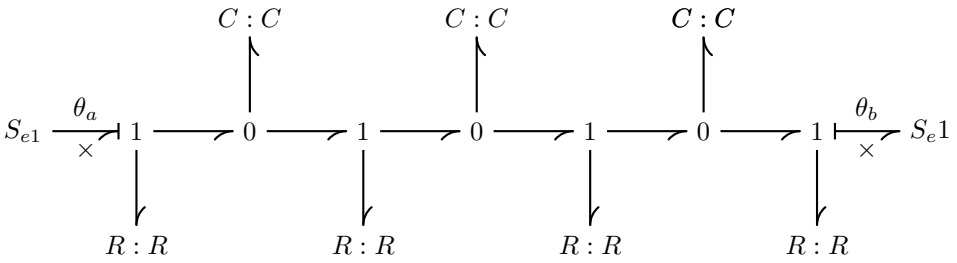

Figure 10: Bond graph for a multilayer thermal model

where

$$R = \frac{1}{H} = \frac{h}{A\lambda} \tag{54}$$

$$C = mc = Ah\rho c, \tag{55}$$

from which

$$\frac{1}{RC} = \frac{A\lambda}{hAh\rho c} = \underbrace{\frac{\lambda}{\rho c}}_{\alpha} \frac{1}{h^2}. \tag{56}$$

2 Flexible systems—wave equation

[...] In the meantime see the handwritten notes [...]

3 Literature

This lecture was prepared as a compilation of numerous resources.

Although our key textbook [1] discusses the distributed-parameter models in Chapter 10, we do not quite follow their treatment.

Instead, we used some bits from the chapter 10 in [8], which is a book that generally serves a good job as an alternative [1]. It is not required for you to get the book, though.

Our treatment is fairly standard and can be found in every other introductory textbook on partial differential equations, wave phenomena, flexible mechanical structures, heat transfer and alike. In particular, I favor [7] and [9] and [10] as practically oriented introductions to partial differential equations. The widely popular [3] and [6] are focusing more on the analysis and less on the modeling aspects. For the wave theory, I like [2], [5] and [11].

Distributed computing and in particular distributed control is a vast area but the seminal paper [4] is certainly one of the good places where to start.

References

[1] Forbes T. Brown. *Engineering System Dynamics: A Unified Graph-Centered Approach, Second Edition*. CRC Press, 2nd edition, August 2006.

[2] William C. Elmore and Mark A. Heald. *Physics of Waves*. Dover Publications, New York, revised edition edition, October 1985.

[3] Lawrence C Evans. *Partial differential equations*. American Mathematical Society, Providence, 2010.

[4] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004.

[5] A.P. French. *Vibrations and Waves*. CBS Publishers & Distributors, New Delhi, December 2003.

[6] John Fritz. *Partial differential equations*. Springer-Verlag, New York, 1982.

[7] Mark S. Gockenbach. *Partial Differential Equations: Analytical and Numerical Methods*. Society for Industrial and Applied Mathematics, Philadelphia, 2nd edition, November 2010.

[8] Dean C. Karnopp, Donald L. Margolis, and Ronald C. Rosenberg. *System Dynamics: Modeling and Simulation of Mechatronic Systems*. Wiley, 4th edition, January 2006.

[9] R.M.M. Mattheij, S.W Rienstra, and J.H.M. ten Thije Boonkkamp. *Partial Differential Equations: Modeling, Analysis, Computation*. Society for Industrial & Applied Mathematics, Philadelphia, November 2005.

[10] Michael Shearer and Rachel Levy. *Partial Differential Equations: An Introduction to Theory and Applications*. Princeton University Press, Princeton, March 2015.

[11] Dudley H. Towne. *Wave Phenomena*. Dover Publications, New York, revised edition edition, June 2012.