

# “Notes on the simulation of the heat equation in Firedrake”

October 10, 2017; rev. November 9, 2017

Simon Pirkelmann

## 1 Setting

Consider the following equation

$$y_t - \alpha \Delta y = 0 \text{ on } Q := \Omega \times [0, T] \quad (1)$$

where  $y : Q \rightarrow \mathbb{R}$  is the temperature,  $\alpha \in \mathbb{R}$  is the diffusion coefficient. We use the shorthand  $y_t = \frac{\partial y}{\partial t}$  to denote the time derivative.

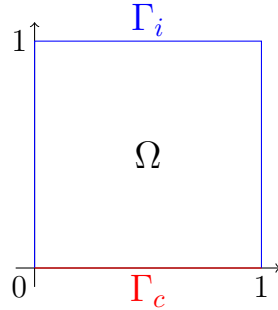
As a domain we consider the unit square. The boundary is partitioned in an isolating boundary  $\Gamma_i$  and a control boundary  $\Gamma_c$ . On the control part of the boundary heating/cooling is applied which is described by a Dirichlet condition

$$y = u \text{ on } \Gamma_c. \quad (2)$$

On the isolating part we have

$$-\frac{\partial y}{\partial n} = 0 \text{ on } \Gamma_i \quad (3)$$

where  $\frac{\partial y}{\partial n}$  is the derivative of  $y$  in normal direction.



For the moment we assume that  $u : [0, T] \rightarrow \mathbb{R}$  and  $y_{out} : [0, T] \rightarrow \mathbb{R}$  are constant on  $\Gamma_c$  and  $\Gamma_{out}$  respectively, i.e. that they are independent of space.

TODO: how to get to this boundary condition!

For simplicity we use a single Robin type boundary condition instead

$$-\beta \frac{\partial y}{\partial n} = \gamma(y - z) \text{ on } \Gamma. \quad (4)$$

By choosing  $\gamma := \begin{cases} \gamma_{out} & \text{on } \Gamma_i \\ \gamma_c & \text{on } \Gamma_c \end{cases}$  and  $z := \begin{cases} y_{out} & \text{on } \Gamma_{out} \\ u & \text{on } \Gamma_c \end{cases}$  we can approximate both types of boundary conditions in a uniform way. This will also allow us to extend the setting more easily in the future.

## 2 Numerical simulation of the heat equation

We simulate the equation using the finite element method.

### 2.1 Weak form

For the weak form we first discretize in time by picking a sampling rate  $h > 0$ . We define  $y_k := y(\cdot, t_0 + hk)$ ,  $y_{out,k} := y_{out}(\cdot, t_0 + hk)$ ,  $u_k := u(t_0 + hk)$ ,  $z_k := z(t_0 + hk)$  for  $k \in \{0, 1, \dots, N\}$ .

The initial value  $y_0$  is given. To compute the next state  $y_{k+1}$  for each  $k \in \{0, 1, \dots, N-1\}$  we replace  $y_t$  in equation (1) by  $\frac{y_{k+1} - y_k}{h}$  and  $y$  by  $y_{k+1}$  using backward Euler. Multiplying with a test function  $v$  and integrating over the domain  $\Omega$  yields

$$\int_{\Omega} \frac{y_{k+1} - y_k}{h} v \, dx - \alpha \int_{\Omega} \Delta y_{k+1} v \, dx = 0 \text{ for } k \in \{0, 1, \dots, N-1\}. \quad (5)$$

Using partial integration on the second integral we get

$$\int_{\Omega} \frac{y_{k+1} - y_k}{h} v \, dx - \alpha \int_{\Gamma} \frac{\partial y_{k+1}}{\partial n} v \, ds + \alpha \int_{\Omega} \nabla y_{k+1} \cdot \nabla v \, dx = 0 \quad (6)$$

Substituting the boundary condition (4) we obtain

$$\int_{\Omega} \frac{y_{k+1} - y_k}{h} v \, dx + \alpha \int_{\Gamma} \frac{\gamma}{\beta} (y_{k+1} - z_k) v \, ds + \alpha \int_{\Omega} \nabla y_{k+1} \cdot \nabla v \, dx = 0 \quad (7)$$

TODO: Strictly speaking, for backward Euler we may actually have to use  $z_{k+1}$  here instead. Think about this!

Substituting the definition of  $z_k$  and  $\gamma$  on the different parts  $\Gamma_{out}$  and  $\Gamma_c$  of the boundary we get

$$\int_{\Omega} \frac{y_{k+1} - y_k}{h} v \, dx + \alpha \int_{\Gamma_c} \frac{\gamma_c}{\beta} (y_{k+1} - u_k) v \, ds + \alpha \int_{\Gamma_{out}} \frac{\gamma_{out}}{\beta} (y_{k+1} - y_{out,k}) v \, ds + \alpha \int_{\Omega} \nabla y_{k+1} \cdot \nabla v \, dx = 0 \quad (8)$$

### 2.2 Implementation in Firedrake

The above variational equation is solved in Firedrake for each  $k \in \{0, 1, \dots, N-1\}$ :

```
# Definitions
S = FunctionSpace(self.mesh, "CG", 1)
v = TestFunction(S)
y1 = Function(S)
y0 = Function(S)

h = Constant(0.001)
gamma = [0.0, 0.0, 1.0e3, 0.0] # gamma for different parts of the
                                boundary
ac = 0.2
u = [0.0, 0.0, 25.0, 0.0]
```

```
a = (y1 * v + h * ac * inner(grad(y1), grad(v))) * dx
for i in range(1,5):
    a += h * ac * Constant(gamma[i-1]) * y1 * v * ds(i)
```

```
F = inner(y0, v) * dx
for i in range(1, 5):
    F += h * ac * Constant(gamma[i-1]) * Constant(u[i-1]) * v * ds(i)
```

### 3 Optimal Control Problem

Now that the simulation is running we want to implement an optimal control problem on top of it. Our goal is to solve the following problem:

$$\begin{aligned} \min_{y,u} J(y,u) &= \frac{1}{2} \int_{\Omega} (y(x,T) - y_{\Omega}(T,x))^2 dx + \frac{1}{2} \int_0^T \int_{\Omega} (y(x,t) - y_{\Omega}(t,x))^2 dx dt \\ &\quad + \frac{\lambda}{2} \int_0^T \int_{\Gamma_c} (u(x,t))^2 ds dt \\ \text{s.t. (1), (4)} \\ \underline{u}(x,t) &\leq u(x,t) \leq \bar{u}(x,t) \\ \underline{y}(x,t) &\leq y(x,t) \leq \bar{y}(x,t) \end{aligned}$$

While the constraints on the control can be dealt with rather straightforwardly, the state constraints present some challenges. We will use Lavrentiev regularisation to handle the state constraints. The Lavrentiev regularisation replaces the state constraint by a mixed state-control constraint. For this we introduce an additional control variable  $v : \Omega \times [0, \infty) \rightarrow \mathbb{R}$  defined on the whole domain. This control variable will also be penalized in the cost functional, and so the cost functional is modified to

$$\begin{aligned} \min_{u,v} J(y,u,v) &= \frac{1}{2} \int_{\Omega} (y(x,T) - y_{\Omega}(T,x))^2 dx + \frac{1}{2} \int_0^T \int_{\Omega} (y(x,t) - y_{\Omega}(t,x))^2 dx dt \\ &\quad + \frac{\sigma}{2} \int_0^T \int_{\Omega} (v(x,t))^2 dx dt + \frac{\lambda}{2} \int_0^T \int_{\Gamma_c} (u(x,t))^2 ds dt \end{aligned}$$

The state constraint is replaced by the auxiliary control constraint

$$\begin{aligned} \underline{y}(x,t) &\leq y(x,t) + \varepsilon v(x,t) \leq \bar{y}(x,t) \\ \Leftrightarrow \underbrace{\frac{1}{\varepsilon}(\underline{y}(x,t) - y(x,t))}_{\underline{v}(x,t)} &\leq v(x,t) \leq \underbrace{\frac{1}{\varepsilon}(\bar{y}(x,t) - y(x,t))}_{\bar{v}(x,t)} \end{aligned}$$

This kind of optimal control problem (with both a control on the boundary and a control in the domain) is similar to the one considered in [Tröltzsch, p.221ff]. We look at the general cost functional

$$\begin{aligned} \min_{y,u,v} J(y,u,v) &= \int_{\Omega} \phi(x, y(T)) dx + \int_Q \varphi(x, t, y, v) dx dt \\ &\quad + \int_{\Sigma_c} \psi(x, t, y, u) ds dt \end{aligned}$$

subject to

$$\begin{aligned}
 y_t - \alpha \Delta y + d(x, t, y, v) &= 0 && \text{in } Q \\
 \partial_n y + b(x, t, y, z) &= 0 && \text{in } \Sigma \\
 y(\cdot, 0) &= y_0 && \text{on } \Omega \\
 v_a(x, t) &\leq v(x, t) \leq v_b(x, t) && \text{in } Q \\
 u_a(x, t) &\leq u(x, t) \leq u_b(x, t) && \text{in } \Sigma
 \end{aligned}$$

In our case we have the following identities:

$$\begin{aligned}
 \phi(x, y(x, T)) &= \frac{1}{2}(y(x, T) - y_\Omega(x, T))^2 \\
 \phi_y(x, y(x, T)) &= y(x, T) - y_\Omega(x, T)
 \end{aligned}$$

$$\begin{aligned}
 \varphi(x, t, y(x, t), v(x, t)) &= \frac{1}{2}(y(x, t) - y_\Omega(x, t))^2 + \frac{\sigma}{2}(v(x, t))^2 \\
 \varphi_y(x, t, y(x, t), v(x, t)) &= y(x, t) - y_\Omega(x, t) \\
 \varphi_v(x, t, y(x, t), v(x, t)) &= \sigma v(x, t)
 \end{aligned}$$

$$\begin{aligned}
 \psi(x, t, y(x, t), u(x, t)) &= \frac{\lambda}{2}(u(x, t))^2 \\
 \psi_y(x, t, y(x, t), u(x, t)) &= 0 \\
 \psi_u(x, t, y(x, t), u(x, t)) &= \lambda u(x, t)
 \end{aligned}$$

$$\begin{aligned}
 d(x, t, y(x, t), v(x, t)) &= 0 \\
 d_y(x, t, y(x, t), v(x, t)) &= 0 \\
 d_v(x, t, y(x, t), v(x, t)) &= 0
 \end{aligned}$$

$$\begin{aligned}
 b(x, t, y(x, t), z(x, t)) &= \frac{\gamma(x, t)}{\beta(x, t)}(y(x, t) - z(x, t)) \\
 b_y(x, t, y(x, t), z(x, t)) &= \frac{\gamma(x, t)}{\beta(x, t)} \\
 b_u(x, t, y(x, t), z(x, t)) &= -\frac{\gamma(x, t)}{\beta(x, t)} \text{ ( on } \Sigma_c \text{ )}
 \end{aligned}$$

## 4 Derivation of the adjoint system

We introduce the adjoint states  $p_1$  and  $p_2$  to remove the PDE equality constraints. The Lagrangian for the problem is given by

$$\begin{aligned}\mathcal{L}(y, u, v, p_1, p_2) &= J(y, u, v) - \int_0^T \int_{\Omega} \left( \frac{\partial y}{\partial t} - \alpha \Delta y + d(x, t, y, v) \right) p_1 \, dx \, dt \\ &\quad - \int_0^T \int_{\Gamma} (\partial_n y + b(x, t, y, z)) p_2 \, dx \, dt \\ &= J(y, u, v) - \int_0^T \int_{\Omega} \left( \frac{\partial y}{\partial t} - \alpha \Delta y + d(x, t, y, v) \right) p_1 \, dx \, dt \\ &\quad - \int_0^T \int_{\Gamma_c} (\partial_n y + b(x, t, y, u)) p_2 \, dx \, dt \\ &\quad - \int_0^T \int_{\Gamma_{out}} (\partial_n y + b(x, t, y, y_{out})) p_2 \, dx \, dt\end{aligned}$$

For optimality we need

$$\begin{aligned}D_y \mathcal{L}(\bar{y}, \bar{u}, \bar{v}, p_1, p_2) y &= 0, \text{ for all } y \text{ with } y(\cdot, 0) = 0 \\ D_u \mathcal{L}(\bar{y}, \bar{u}, \bar{v}, p_1, p_2) (u - \bar{u}) &\geq 0, \quad \text{for all } u \in U_{ad} \\ D_v \mathcal{L}(\bar{y}, \bar{u}, \bar{v}, p_1, p_2) (v - \bar{v}) &\geq 0, \quad \text{for all } v \in V_{ad}\end{aligned}$$

In the following we compute  $D_y \mathcal{L}$ ,  $D_u \mathcal{L}$  and  $D_v \mathcal{L}$ . We start with  $D_y \mathcal{L}$ :

$$\begin{aligned}D_y \mathcal{L}(y, u, v, p_1, p_2) h &= D_y J(y, u, v) h - \int_0^T \int_{\Omega} \left( \frac{\partial h}{\partial t} - \alpha \Delta h + d_y(x, t, y, v) h \right) p_1 \, dx \, dt \\ &\quad - \int_0^T \int_{\Gamma_c} (\partial_n h + b_y(x, t, y, u) h) p_2 \, dx \, dt \\ &\quad - \int_0^T \int_{\Gamma_{out}} (\partial_n h + b_y(x, t, y, y_{out}) h) p_2 \, dx \, dt,\end{aligned}$$

where for  $D_y J(y, u, v)$  it holds that

$$\begin{aligned}D_y J(y, u, v) h &= \int_{\Omega} \phi_y(x, y) h(T) \, dx + \int_0^T \int_{\Omega} \varphi_y(x, t, y, v) h \, dx \, dt \\ &\quad + \int_0^T \int_{\Gamma_c} \psi_y(x, t, y, u) h \, ds \, dt.\end{aligned}$$

First we consider the term

$$\begin{aligned}& - \int_0^T \int_{\Omega} \left( \frac{\partial h}{\partial t} - \alpha \Delta h + d_y(x, t, y, v) h \right) p_1 \, dx \, dt \\ &= - \int_{\Omega} \int_0^T \frac{\partial h}{\partial t} p_1 \, dx \, dt + \int_0^T \int_{\Omega} \alpha \Delta h p_1 \, dx \, dt - \int_0^T \int_{\Omega} d_y(x, t, y, v) h p_1 \, dx \, dt.\end{aligned}$$

Using partial integration (in time) for the first term and Green’s second formula (in space) for the second term we obtain

$$\begin{aligned}
 & - \int_{\Omega} \int_0^T \frac{\partial h}{\partial t} p_1 \, dx \, dt + \alpha \int_0^T \int_{\Omega} \Delta h p_1 \, dx \, dt - \int_0^T \int_{\Omega} d_y(x, t, y, v) h p_1 \, dx \, dt \\
 &= - \int_{\Omega} [h p_1]_0^T \, dx + \int_{\Omega} \int_0^T \frac{\partial p_1}{\partial t} h \, dx \, dt + \alpha \int_0^T \int_{\Omega} \Delta p_1 h \, dx \, dt \\
 & \quad + \alpha \int_0^T \int_{\Gamma} p_1 \frac{\partial h}{\partial n} \, ds \, dt - \alpha \int_0^T \int_{\Gamma} \frac{\partial p_1}{\partial n} h \, ds \, dt - \int_0^T \int_{\Omega} d_y(x, t, y, v) p_1 h \, dx \, dt \\
 &= \int_{\Omega} p_1(0) h(0) \, dx - \int_{\Omega} p_1(T) h(T) \, dx + \int_0^T \int_{\Omega} \frac{\partial p_1}{\partial t} h \, dx \, dt + \alpha \int_0^T \int_{\Omega} \Delta p_1 h \, dx \, dt \\
 & \quad + \alpha \int_0^T \int_{\Gamma} p_1 \frac{\partial h}{\partial n} \, ds \, dt - \alpha \int_0^T \int_{\Gamma} \frac{\partial p_1}{\partial n} h \, ds \, dt - \int_0^T \int_{\Omega} d_y(x, t, y, v) p_1 h \, dx \, dt \\
 & \stackrel{h(0)=0}{=} - \int_{\Omega} p_1(T) h(T) \, dx + \int_0^T \int_{\Omega} \frac{\partial p_1}{\partial t} h \, dx \, dt + \alpha \int_0^T \int_{\Omega} \Delta p_1 h \, dx \, dt \\
 & \quad + \alpha \int_0^T \int_{\Gamma} p_1 \frac{\partial h}{\partial n} \, ds \, dt - \alpha \int_0^T \int_{\Gamma} \frac{\partial p_1}{\partial n} h \, ds \, dt - \int_0^T \int_{\Omega} d_y(x, t, y, v) p_1 h \, dx \, dt
 \end{aligned}$$

In the last step we used that  $h(0) = 0$ . This is explained in [Tröltzsch, p.96] and follows from a substitution  $y := y - \bar{y}$  for the state.

By substituting the above equation in the original equation and reordering by integration domains we obtain

$$\begin{aligned}
 D_y J(y, u, v) h &= \int_{\Omega} (\phi_y(x, y) - p_1(T)) h(T) \, dx \\
 & \quad + \int_0^T \int_{\Omega} \left( \frac{\partial p_1}{\partial t} + \alpha \Delta p_1 - d_y(x, t, y, v) p_1 + \varphi_y(x, t, y, v) \right) h \, dx \, dt \\
 & \quad + \int_0^T \int_{\Gamma} (\alpha p_1 - p_2) \frac{\partial h}{\partial n} \, ds \, dt \\
 & \quad + \int_0^T \int_{\Gamma} \left( -\alpha \frac{\partial p_1}{\partial n} - b_y(x, t, y, z) p_2 + \psi_y(x, t, y, u) \right) h \, ds \, dt
 \end{aligned}$$

Now we make special choices of  $h$ , first  $h \in C_0^\infty(Q)$ , then with arbitrary  $h(T)$ , arbitrary  $h|_{\Sigma}$  and finally with arbitrary  $\frac{\partial h}{\partial n}$ . From this we obtain the following adjoint equation:

$$\begin{aligned}
 -p_{1,t} - \alpha \Delta p_1 + d_y(x, t, \bar{y}, \bar{v}) p_1 &= \varphi_y(x, t, \bar{y}, \bar{v}) & \text{in } Q \\
 \alpha \partial_n p_1 + b_y(x, t, \bar{y}, \bar{z}) p_2 &= \psi_y(x, t, \bar{y}, \bar{u}) & \text{in } \Sigma \\
 \alpha p_1 &= p_2 & \text{in } \Sigma \\
 p_1(x, T) &= \phi_y(x, \bar{y}(x, T)) & \text{in } \Omega
 \end{aligned} \tag{9}$$

By setting  $p := p_1$ ,  $p_2 = \alpha p$  we obtain:

$$\begin{aligned}
 -p_t - \alpha \Delta p + d_y(x, t, \bar{y}, \bar{v}) p &= \varphi_y(x, t, \bar{y}, \bar{v}) & \text{in } Q \\
 \partial_n p + b_y(x, t, \bar{y}, \bar{z}) p &= \frac{1}{\alpha} \psi_y(x, t, \bar{y}, \bar{u}) & \text{in } \Sigma \\
 p(x, T) &= \phi_y(x, \bar{y}(x, T)) & \text{in } \Omega
 \end{aligned} \tag{10}$$

A similar result (but without the diffusion coefficient  $\alpha$ ) can be found in [Tröltzsch, p.225f].

Finally we also compute  $D_u \mathcal{L}$  and  $D_v \mathcal{L}$ :

$$\begin{aligned} D_u \mathcal{L}(y, u, v, p)h &= D_u J(y, u, v)h - \int_0^T \int_{\Gamma_c} b_u(x, t, y, u)ph \, ds \, dt \\ &= \int_0^T \int_{\Gamma_c} (\psi_u(x, t, y, u) - b_u(x, t, y, u)p)h \, ds \, dt \end{aligned}$$

$$\begin{aligned} D_v \mathcal{L}(y, u, v, p)h &= D_v J(y, u, v)h - \int_0^T \int_{\Omega} d_v(x, t, y, v)ph \, dx \, dt \\ &= \int_0^T \int_{\Omega} (\varphi_v(x, t, y, v) - d_v(x, t, y, v)p)h \, dx \, dt \end{aligned}$$

## 5 Weak form of the adjoint system

To derive the weak form of the adjoint system (10) we first note that the equation evolves backward in time. To get a forward equation we use the substitution  $\tau := T - t$ . Define

$$\begin{aligned} \tilde{p}(x, \tau) &:= p(x, T - \tau) = p(x, t) \\ \tilde{y}(x, \tau) &:= y(x, T - \tau) \\ \tilde{u}(x, \tau) &:= u(x, T - \tau) \\ \tilde{v}(x, \tau) &:= v(x, T - \tau) \end{aligned}$$

Noting that  $D_\tau \tilde{p}(x, \tau) = -D_t p(x, t)$ , the adjoint system changes to

$$\begin{aligned} \tilde{p}_\tau - \alpha \Delta \tilde{p} + d_y(x, T - \tau, \tilde{y}, \tilde{v})\tilde{p} &= \varphi_y(x, T - \tau, \tilde{y}, \tilde{v}) \quad \text{in } Q \\ \partial_n \tilde{p} + b_y(x, T - \tau, \tilde{y}, \tilde{u})\tilde{p} &= \frac{1}{\alpha} \psi_y(x, T - \tau, \tilde{y}, \tilde{u}) \quad \text{in } \Sigma \\ \tilde{p}(x, 0) &= \phi_y(x, \tilde{y}(x, 0)) \quad \text{in } \Omega \end{aligned} \tag{11}$$

Now the system can be solved again using backward Euler. Replace  $\tilde{p}_\tau$  by  $\frac{\tilde{p}_{k+1} - \tilde{p}_k}{h}$  and  $\tilde{p}$ ,  $\tilde{y}$ ,  $\tilde{u}$  and  $\tilde{v}$  by their discrete counterparts. We multiply with the test function  $v$  and integrate:

$$\int_{\Omega} \frac{\tilde{p}_{k+1} - \tilde{p}_k}{h} v \, dx - \int_{\Omega} \alpha \Delta \tilde{p}_{k+1} v \, dx + \int_{\Omega} d_y(x, T - \tau_k, \tilde{y}_k, \tilde{v}_k) \tilde{p}_{k+1} v \, dx = \int_{\Omega} \varphi_y(x, T - \tau_k, \tilde{y}_k, \tilde{v}_k) v \, dx$$

Using partial integration on the second integral and inserting the boundary condition yields

$$\begin{aligned} \int_{\Omega} \alpha \Delta \tilde{p}_{k+1} v \, dx &= - \int_{\Gamma} \alpha \frac{\partial \tilde{p}_{k+1}}{\partial n} v \, ds + \alpha \int_{\Omega} \nabla \tilde{p}_{k+1} \nabla v \, dx \\ &= \int_{\Gamma} \alpha b_y(x, T - \tau_k, \tilde{y}_k, \tilde{u}_k) \tilde{p}_{k+1} v \, ds - \int_{\Gamma} \psi_y(x, T - \tau_k, \tilde{y}_k, \tilde{u}_k) v \, ds \\ &\quad + \alpha \int_{\Omega} \nabla \tilde{p}_{k+1} \nabla v \, dx \end{aligned}$$

We insert this in the first equation again and order by integration domain

$$\begin{aligned} \int_{\Omega} \left( \frac{\tilde{p}_{k+1} - \tilde{p}_k}{h} + d_y(x, T - \tau_k, \tilde{y}_k, \tilde{v}_k) \tilde{p}_{k+1} - \varphi_y(x, T - \tau_k, \tilde{y}_k, \tilde{v}_k) v + \alpha \nabla \tilde{p}_{k+1} \nabla v \right) dx \\ + \int_{\Gamma} (\alpha b_y(x, T - \tau_k, \tilde{y}_k, \tilde{u}) \tilde{p}_{k+1} - \psi_y(x, T - \tau_k, \tilde{y}_k, \tilde{u}_k)) v \, ds = 0 \end{aligned}$$

After the modified adjoint system was solved we get the original adjoint by substituting back  $p(x, t_k) = \tilde{p}(x, \tau_k)$ .

## 5.1 Adjoint equation for our case

In our case the equation simplifies to:

$$\int_{\Omega} \left( \frac{\tilde{p}_{k+1} - \tilde{p}_k}{h} - (\tilde{y}_k - \tilde{y}_{\Omega,k}) \right) v + \alpha \nabla \tilde{p}_{k+1} \nabla v \, dx + \int_{\Gamma} \frac{\alpha \gamma}{\beta} \tilde{p}_{k+1} v \, ds = 0$$

Additionally, we have the initial condition:

$$\tilde{p}_0 = \phi_y(x, \tilde{y}(x, 0)) = y_N - y_{\Omega,N}$$

Rewriting (without the time substitution for  $y$ ) and inserting the different boundaries  $\Gamma_{out}$  and  $\Gamma_c$ :

$$\begin{aligned} \int_{\Omega} \left( \frac{\tilde{p}_{k+1} - \tilde{p}_k}{h} - (y_{N-k} - y_{\Omega,N-k}) \right) v + \alpha \nabla \tilde{p}_{k+1} \nabla v \, dx \\ + \alpha \int_{\Gamma_{out}} \frac{\gamma_{out}}{\beta} \tilde{p}_{k+1} v \, ds + \alpha \int_{\Gamma_c} \frac{\gamma_c}{\beta} \tilde{p}_{k+1} v \, ds = 0, \end{aligned}$$

for  $k \in \{0, \dots, N-1\}$ .

## 6 Solution by Projected Gradient Method

Let  $N \in \mathbb{N}$  be the MPC horizon and let  $u^n := (u_0^n, u_1^n, \dots, u_{N-1}^n)$ ,  $v^n := (v_0^n, v_1^n, \dots, v_{N-1}^n)$ ,  $y^n := (y_0^n, y_1^n, \dots, y_N^n)$ ,  $p^n := (p_0^n, p_1^n, \dots, p_N^n)$  be the iterates of the optimization algorithm. The gradient of the reduced cost functional  $f(v, u) = J(y(v, u), v, u)$  is given by

$$\begin{aligned} f'(v^n, u^n)(v, u) = \int \int_Q (\varphi_v(x, t, y^n, v^n) - d_v(x, t, y^n, v^n) p^n) v \, dx \, dt \\ + \int \int_{\Sigma} (\psi_u(x, t, y^n, u^n) - b_u(x, t, y^n, u^n) p^n) u \, ds \, dt \end{aligned}$$

This can also be found in [Tröltzsch, p. 243f], for the special case of  $d(x, t, y, v) = v$ ,  $b(x, t, y, u) = u$ .

Solution algorithm:

1. Solve forward system for given  $(u^n, v^n) \rightsquigarrow y_n$



2. Solve adjoint system  $\rightsquigarrow p^n$

3. Descent directions

$$\begin{aligned} h^n &:= -(\varphi_v(\cdot, y^n, v^n) - d_v(\cdot, y^n, v^n)p^n) \\ r^n &:= -(\psi_u(\cdot, y^n|_\Sigma, u^n) - b_u(\cdot, y^n, u^n)p^n|_\Sigma) \end{aligned}$$

4. Compute step size  $\rightsquigarrow s^n$  (e.g. use  $\min_{s>0} f(\mathbb{P}_V(v^n + sh^n), \mathbb{P}_U(u^n + sr^n))$ ).

5. New iterates:

$$(v^{n+1}, u^{n+1}) := (\mathbb{P}_V(v^n + s^n h^n), \mathbb{P}_U(u^n + s^n r^n)) \quad (12)$$

## 6.1 Gradient in our case

$$\begin{aligned} f'(v^n, u^n)(v, u) &= \int \int_Q (\sigma v^n) v \, dx \, dt \\ &\quad + \int \int_\Sigma ((\lambda u^n) + \frac{\gamma}{\beta} p^n) u \, ds \, dt \end{aligned}$$

## 7 Solver Options/Preconditioning

## Reference