EUROPEAN CONSORTIUM FOR MATHEMATICS IN INDUSTRY

ECMI Modelling Week 2018
University of Novi Sad, Faculty of Sciences

MODELLING OF ICE AROUND COOLING PIPES

Instructor: Jürgen Dölz (Technical University of Darmstadt)

Florian Bürger	(Technical University of Dresden)
Nataša Mišanović	(University of Novi Sad)
Tiago Sequeira	(Technical Superior Institute of Lisbon University)
Spasimir Nonev	(University of Sofia "St. Kliment Ohridski")
Agnes Printemps	(University of Grenoble Alpes and Grenoble INP)

15th Octobre 2018

Contents

1 Introduction

At the present time, people are interested in artificially freezing the soil in certain areas, whether for recreational purposes on ice rinks or as a necessity in the economy. In times of global warming, especially people have to struggle with additional problems living in permafrost regions. The soil in such areas is permanently frozen due to the low temperatures, thus providing additional stability for buildings and roads constructed thereon. However, as temperatures rise, the thawing of the earth threatens, resulting in the loss of this additional stability. Collapses of buildings or tunnels can be the result. Also, the volume of the substrate decreases with increasing temperatures, as the water in the earth continues to contract below 4 °C. As a result, paved roads are ruptured and so impassable for vehicles. In such situations one is therefore interested in keeping the soil artificially frozen. By using cooling pipes in the soil, through which cooling liquid flows, this effect can be effected. The mathematical modeling leads to the so called "*Bernoulli's* free boundary problem". This report is about the derivation of an iterative formula to calculate the null level set of the temperature u.

2 Physical Setting

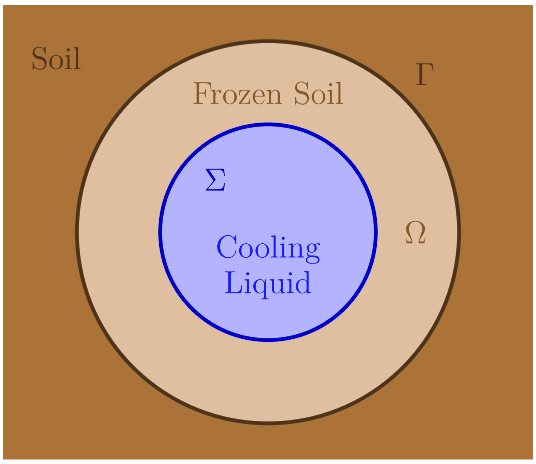


Figure 1: Schematic illustration of our problem

We consider our problem in two dimensions. Precisely, the cross section of our cooling pipe and the surrounding soil should be a part of $\mathbb{R}^2$. We denote the boundary of our cooling pipe by Σ. We assume, the temperature on Σ is known and given by a function $g\colon \Sigma \to \mathbb{R}$. After a long time of cooling process, the temperature u has reached a stationary state. In this state, there is a subset of the soil around the cooling pipe where the soil is frozen. Let us denote this part by Ω. And also there is a part of the soil, where the temperature u is $0\ °C$, which is denoted by Γ. Figure 1 gives an illustration of our problem. In this setting, u is satisfying the following properties:

$$\begin{cases} \Delta u &= 0 \quad \text{in } \Omega \\ u &= g \quad \text{on } \Sigma \\ u &= 0 \quad \text{on } \Gamma \\ \frac{\partial u}{\partial \boldsymbol{n}} &= h \quad \text{on } \Gamma, \end{cases} \tag{1}$$

where $\frac{\partial u}{\partial \boldsymbol{n}} = \nabla u \cdot \boldsymbol{n}$ is the normal derivative of u and $\boldsymbol{n}$ is the outer unit normal of Γ. Finally, the function $h > 0$ depends on various properties of soil, for example its specific thermal conductivity.

The last line of equation (1) is the so called "Stefan condition". This condition is motivated by the fact that we have no source of heat in the soil around the pipe, so the energy in form of temperature flowing into the ice is the energy which flows out from the surrounding soil.

3 Finding Γ - Gradient Descent

For the numerical calculation of our quantity of interest Γ we will construct an iterative procedure. Let us consider a procedure in the following form:

$$\Gamma_{k+1} = \Gamma_k + z\, d_k, \quad k \in \mathbb{N}_0 \tag{2}$$

where

- $\Gamma_k \subset \mathbb{R}^2$,
- $z\colon \Gamma_k \to \mathbb{R}^2$ is a vector field on the actual Γ_k, (we interpret it as pointwise search direction for our new Γ_{k+1}),
- $d_k\colon \Gamma_k \to \mathbb{R}$ is the pointwise stepsize for every search direction.

In this case we can not handle our formula in a good way, so we replace Γ_k by a parametrization $\gamma_k\colon [0,1] \to \Gamma_k$. We are getting this new formula:

$$\gamma_{k+1}(s) = \gamma_k(s) + z(s)\, d_k(s), \qquad k \in \mathbb{N}_0,\ s \in [0,1]. \tag{3}$$

We read (3) pointwise for every $s \in [0,1]$. We have z and d_k replaced by new functions on $[0,1]$ and still the same range but we do not have relabeled them. In Figure 2 we see an illustration, how we calculate our new Γ_{k+1}. For our iterative formula, z and d_k are still unknown. So, for z we will choose simply the outer unit normal vector field of the actual Γ_k and for d_k we need to do some calculations.

We are interested of the level set Γ where u is zero. So let us look at an arbitrary but fixed $s \in [0,1]$. At the point $\gamma_k(s) \in \Gamma_k$ we are searching for a root of u or in other

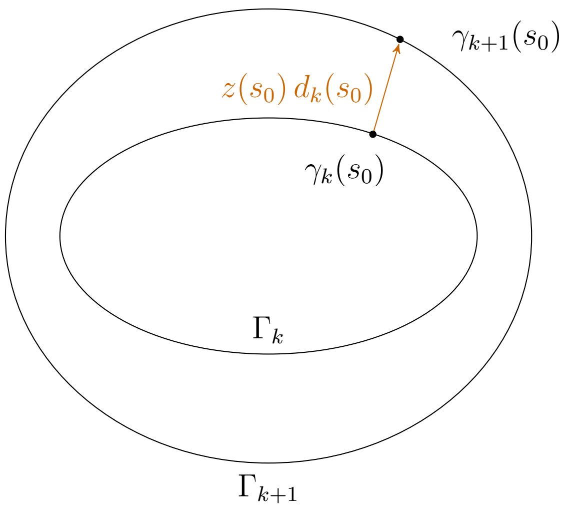


Figure 2: Scheme, how we calculate our new Γ_{k+1}

terms: we want to find a value $d_k(s)$ such that $u(\gamma_k(s) + z(s)d_k(s)) = 0$. Regrettably we do not know much about our function u except $\Delta u = 0$ in Ω. Furthermore Ω is also unknown, because of $\Omega = \mathrm{conv}\{\Sigma, \Gamma\}$, where

$$\mathrm{conv}\{\Sigma, \Gamma\} = \left\{x \in \mathbb{R}^2 \mid \exists x_\Sigma \in \Sigma,\, \exists x_\Gamma \in \Gamma \colon \exists \alpha \in [0,1] \colon x = \alpha\, x_\Sigma + (1-\alpha)\, x_\Gamma\right\}$$

is the convex hull of Σ and Γ. So for the actual iteration k we are computing u on a discrete Ω_k and we are trying to compute d_k by a Taylor expansion of u, because it allows us to relate values of u on Γ_{k+1} to values of u on Γ_k. We calculate the Taylor expansion until the first order and omit higher order terms.

$$0 \overset{!}{=} u(\gamma_{k+1}(s)) \overset{(3)}{=} u(\gamma_k(s) + z(s)\, d_k(s)) \underset{\text{at } d_k=0}{\overset{\text{Taylor}}{\approx}} u(\,\gamma_k(s)) + \nabla u(\gamma_k(s))\,) \cdot z(s)\, d_k(s)$$

$$\Rightarrow \qquad d_k(s) = \frac{u(\gamma_k(s))}{\nabla u(\gamma_k(s))) \cdot z(s)} \qquad s \in [0,1]. \tag{4}$$

4 Computing u

We know from Section 2 u is satisfying (1). Since we do not know Γ exactly, we can not solve this PDE. We have to replace (1) by the following, semi-discretized problem:

$$\begin{cases} \Delta u &= 0 \quad \text{in } \Omega_k \\ u &= g \quad \text{on } \Sigma \\ u &= 0 \quad \text{on } \Gamma_k \\ \frac{\partial u}{\partial \boldsymbol{n}} &= h \quad \text{on } \Gamma_k, \end{cases} \tag{5}$$

where $\Omega_k = \mathrm{conv}\{\Sigma, \Gamma_k\}$. By [Harbrecht 2012] we will see that u can be computed in the following way:

$$u(\boldsymbol{x}) = -\frac{1}{2\pi} \int\limits_{\Sigma\cup\Gamma} \log\|\boldsymbol{x}-\boldsymbol{y}\| \frac{\partial u}{\partial \boldsymbol{n}}(\boldsymbol{y})\, d\sigma_{\boldsymbol{y}} - \frac{1}{2\pi} \int\limits_{\Sigma\cup\Gamma} \frac{\langle \boldsymbol{x}-\boldsymbol{y}, \boldsymbol{n}_{\boldsymbol{y}}\rangle}{\|\boldsymbol{x}-\boldsymbol{y}\|^2} u(\boldsymbol{y})\, d\sigma_{\boldsymbol{y}}, \quad \boldsymbol{x} \in \Omega. \quad (6)$$

In the next Section we will discuss two different approaches to discretize this formula. With their help we will get a formula to calculate the discretized u on Γ.

5 Discretization

5.1 Linear Equality Systems for u

The following two Sections are about the discretization of equation (6). Every approach discretize one of the occuring integrals in (6) and will produce a SLE. To solve the discrete version of (6), we solve the sum of both SLE.

5.1.1 Single Layer Approach

For a domain D with diameter $\mathrm{diam}(D) < 1$ and compact Lipschitz boundary Γ let u satisfy

$$\begin{cases} \Delta u &= 0 \quad \text{in } D \\ u &= g \quad \text{on } \Gamma. \end{cases}$$

There exists a function w, such that

$$u(\boldsymbol{x}) = -\frac{1}{2\pi} \int\limits_{\Gamma} \log ||\boldsymbol{x}-\boldsymbol{y}||\, w(\boldsymbol{y})\, d\sigma_{\boldsymbol{y}}, \qquad \boldsymbol{x} \in D. \quad (7)$$

Taking traces on both sides, we are solving:

$$\mathcal{V}w = g \qquad \text{on } \partial D,$$

to obtain w. Setting $\boldsymbol{x} = \gamma(s) \in \Gamma$, gives us from (7):

$$\begin{aligned}(\mathcal{V}w)(\boldsymbol{x}) &= -\frac{1}{2\pi} \int\limits_{\Gamma} \log ||\boldsymbol{x}-\boldsymbol{y}|| w(\boldsymbol{y}) d\sigma_{\boldsymbol{y}} \\ &= -\frac{1}{4\pi} \int\limits_{0}^{1} k(s,t)\tilde{w}(t) dt \\ &=: (\tilde{\mathcal{V}}\tilde{w})(s), \qquad \boldsymbol{x} \in D,\end{aligned}$$

where k and $\tilde{w}$ are suitable short notations, i.e.

$$k(s,t) := \log ||\gamma(s) - \gamma(t)||^2, \qquad \tilde{w}(t) := ||\gamma'(t)|| w(\gamma(t)).$$

Now, we will use another trick. We split k in the following form $k(s,t) = k_1(s,t) + k_2(s,t)$, where

$$\begin{cases} k_1(s,t) := \log\left(\frac{||\gamma(s)-\gamma(t)||^2}{4\sin^2(\pi(s-t))}\right), \\ \\ k_2(s,t) := \log\left(4\sin^2(\pi(s-t))\right), \end{cases} \tag{8}$$

with

$$\lim_{s\to t} k_1(s,t) = \log\left(\left\|\frac{\gamma'(t)}{2\pi}\right\|^2\right). \tag{9}$$

This is necessary to calculate the diagonal elements in the discretization matrix. For trigonometric Lagrange polynomials $L_i(t)$, we assume $\tilde{w}(t) = \sum_{i=0}^{n-1} \tilde{w}_i L_i(t)$. For $n = 2m$ this yields

$$\int_0^1 k_2(s_i,t) L_j(t) dt = -\frac{1}{m}\left(\sum_{l=1}^{m-1} \frac{1}{l}\cos(2\pi l(s_i - t_j)) + \frac{1}{n}\cos(2\pi m(s_i - t_j))\right) := R_j(s_i).$$

For

$$\boldsymbol{V}_0 = \frac{1}{n}\Big(k_1(s_j,t_i)\Big)_{i,j=0}^{n-1}, \qquad \boldsymbol{V}_1 = \Big(R_j(s_i)\Big)_{i,j=0}^{n-1}, \qquad \boldsymbol{w} = \Big(\tilde{w}_i\Big)_{i=0}^{n-1},$$

$$\boldsymbol{g} = \Big(\tilde{g}(s_i)\Big)_{i=0}^{n-1}$$

we will get a SLE

$$-\frac{1}{4\pi}(\boldsymbol{V}_0 + \boldsymbol{V}_1)\boldsymbol{w} = \boldsymbol{g}.$$

Having solved for $\boldsymbol{w}$ allows to compute:

$$u(\boldsymbol{x}) = -\frac{1}{4\pi n}\sum_{i=0}^{n-1} \log ||\boldsymbol{x} - \gamma(t_i)||^2 \tilde{w}(t_i) \tag{10}$$

5.1.2 Double Layer Approach

For a domain D with compact Lipschitz boundary Γ, a function u which is satisfying

$$\begin{cases} \Delta u &= 0 \quad \text{in } D \\ u &= g \quad \text{on } \Gamma, \end{cases}$$

there exists a function w, such that

$$u(\boldsymbol{x}) = \frac{1}{2\pi} \int\limits_{\Gamma} \frac{\langle \boldsymbol{x} - \boldsymbol{y}, \boldsymbol{n_y} \rangle}{||\boldsymbol{x} - \boldsymbol{y}||^2} w(\boldsymbol{y}) d\sigma_{\boldsymbol{y}}, \qquad \boldsymbol{x} \in D. \tag{11}$$

We are solving:

$$\left(K - \frac{1}{2}\right) \mathbf{w} = \mathbf{g} \qquad \text{on } \Gamma,$$

to obtain w. Setting $\boldsymbol{x} = \gamma(s) \in \Gamma$, gives us from (11):

$$\begin{aligned}
(Kw)(\mathbf{x}) = &\ \frac{1}{2\pi} \int\limits_{\Gamma} \frac{\langle \mathbf{x} - \mathbf{y}, \mathbf{n}_y \rangle}{||\mathbf{x} - \mathbf{y}||^2} w(\mathbf{y}) d\sigma_{\mathbf{y}} \\
= &\ \frac{1}{2\pi} \int\limits_0^1 k(s,t) \tilde{w}(t) dt \\
=: &\ (\tilde{K}\tilde{w})(s), \qquad \mathbf{x} \in D,
\end{aligned}$$

where k and $\tilde{w}$ are practical short notations, i.e.

$$k(s,t) := \frac{\langle \gamma(s) - \gamma(t), \mathbf{n}(t) \rangle}{||\gamma(s) - \gamma(t)||^2} ||\gamma'(t)|| \qquad \tilde{w}(t) := w(\gamma(t)). \tag{12}$$

We are using a Nyström-Scheme and

$$\lim_{s \to t} k(s,t) = \frac{\langle \gamma''(t), \mathbf{n}(t) \rangle}{2||\gamma'(t)||}. \tag{13}$$

for the diagonal elements of the LES-matrix to obtain a linear system of equations

$$\left(\mathbf{K} - \frac{1}{2}\mathbf{I}\right) \mathbf{w} = \mathbf{g},$$

with

$$\mathbf{K} = \frac{1}{2\pi n} \Big(k(s_i, t_j) \Big)_{i,j=0}^{n-1}, \qquad \mathbf{w} = \Big(\tilde{w}(s_i) \Big)_{i=0}^{n-1}, \qquad \mathbf{g} = \Big(\tilde{g}(s_i) \Big)_{i=0}^{n-1}.$$

Having solved for $\mathbf{w}$ allows to compute:

$$u(\mathbf{x}) = \frac{1}{2\pi n} \sum_{i=0}^{n-1} \frac{\langle \mathbf{x} - \gamma(t_i), \mathbf{n}(t_i) \rangle}{||\mathbf{x} - \gamma(t_i)||^2} ||\gamma'(t_i)||\, \tilde{w}(t_i). \tag{14}$$

5.1.3 Mixed Boundary Value Problems

Let Ω, Σ, Γ be as above. Look at

$$\Delta u = 0 \quad \text{in } \Omega, \qquad u = g \quad \text{on } \Sigma, \qquad \frac{\partial u}{\partial \mathbf{n}} = h \quad \text{on } \Gamma.$$

We know a representation formula of u by (6). For using this formula, we require u and $\partial u/\partial\mathbf{n}$ on both, Σ and Γ. By the boundary conditions we have $u = g$ on Σ and $\partial u/\partial\mathbf{n} = h$ on Γ. So the values of u on Γ and the values of $\partial u/\partial\mathbf{n}$ on Σ are missing. To calculate it, we solve a SLE with block matrices:

$$\begin{pmatrix} \mathbf{V}_{\Sigma\Sigma} & -\mathbf{K}_{\Gamma\Sigma} \\ -\mathbf{V}_{\Sigma\Gamma} & \mathbf{K}_{\Gamma\Gamma} + \mathbf{I}/2 \end{pmatrix} \begin{pmatrix} \mathbf{u}_{N,\Sigma} \\ \mathbf{u}_{D,\Gamma} \end{pmatrix} = \begin{pmatrix} \mathbf{K}_{\Sigma\Sigma} + \mathbf{I}/2 & -\mathbf{V}_{\Gamma\Sigma} \\ -\mathbf{K}_{\Sigma\Gamma} & \mathbf{V}_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} \mathbf{g}_{\Sigma} \\ \mathbf{h}_{\Gamma} \end{pmatrix} \tag{15}$$

where $\mathbf{u}_{D,\Gamma}$ are the values of u on Γ and $\mathbf{u}_{N,\Sigma}$ are the values of $\partial u/\partial\mathbf{n}$ on Σ.

The matrices on the diagonals can be obtain from the previous approaches, the off-diagonal blocks are given by

$$\begin{aligned} \mathbf{V}_{\Sigma\Gamma} &= -\frac{1}{4\pi}\Big(\log \|\gamma_\Sigma(s_i) - \gamma_\Gamma(t_i)\|^2 \Big)_{i,j=0}^{n-1}, \\ \mathbf{K}_{\Sigma\Gamma} &= \frac{1}{2\pi}\Big(\frac{\langle \gamma_\Sigma(s_i) - \gamma_\Gamma(t_j), \mathbf{n}_\Gamma(t_j)\rangle}{\|\gamma_\Sigma(s_i) - \gamma_\Gamma(t_j)\|^2} \|\gamma'_\Gamma(t_j)\| \Big)_{i,j=0}^{n-1}, \end{aligned}$$

$\mathbf{V}_{\Gamma\Sigma}$ and $\mathbf{K}_{\Gamma\Sigma}$ are given similarly, and

$$\mathbf{u}_{D,\Gamma} = \Big(u\big(\gamma_\Gamma(s_i)\big)\Big)_{i=0}^{n-1}, \qquad \mathbf{u}_{N,\Sigma} = \left(\frac{\partial u}{\partial \mathbf{n}}\big(\gamma_\Sigma(s_i)\big) \|\gamma'_\Sigma(s_i)\| \right)_{i=0}^{n-1},$$

with γ_Γ the parametrization of Γ and γ_Σ the parametrization of Σ. $\mathbf{g}_\Sigma$ and $\mathbf{h}_\Gamma$ are the vectors of the given Dirichlet and Neumann data on the discretization points $s_0,\ s_1,\ \ldots,\ s_{n-1}$.

5.2 Discretization of Γ

We want to construct a discretization of Γ or in other terms of $\gamma\colon [0,1] \to \Gamma$. We assume there exists a function $r\colon [0,1] \to \mathbb{R}_+$ such that

$$\gamma(s) = r(s) \begin{pmatrix} \cos(2\pi s) \\ \sin(2\pi s) \end{pmatrix}, \quad s \in [0,1]. \tag{16}$$

For our formulas (9), (12) and (13) we will need γ' and γ'' respectively. With formula (16) we will need the first and second derivative of r. Further, the outer unit normal $\boldsymbol{n}$ of Γ is also necessary. For the derivatives of r it is useful to approximating r by a trigonometric polynomial in the following form

$$r(s) = \frac{a_0}{2} + \sum_{l=1}^{M-1} (\alpha_l \cos(2\pi l s) + \beta_l \sin(2\pi l s)) + \frac{\alpha_M}{2}\cos(2\pi M s), \quad M \in \mathbb{N}, \tag{17}$$

because then the differentiability is guaranteed. M is chosen skillfully (for example $M = \frac{n}{2} + 1$, where n is the number of discretization points of $[0,1]$ if n is even). The outer unit normal $\boldsymbol{n}$ is calculated in the following way:

$$\boldsymbol{n}(s) = \frac{1}{\|\gamma'(s)\|}\boldsymbol{R}\,\gamma'(s) \quad (s \in [0,1]), \text{ where} \tag{18}$$

$$\boldsymbol{R} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

This formula is motivated by the fact, γ' is the tangent vector on Γ, so if we rotate it and scale by its length, we will get our outer unit normal vector field. The derivatives of r can calculated by differentiating formula (17). Our final discretizing is to discretize $[0,1]$ by a finite set $\{0 = s_0, s_1, ..., s_{n-1} < 1\}$.

6 Implementation

We already have collected all the tools to compute Γ. Figure 3 illustrates the schematic procedure of the algorithm. In the following steps we see which files solve the specific subtasks:

1. First of all we solve equation (5) with an initial γ_0 using the numerical method presented in Section 5.1.3. This is implemented in the files `SLA.m`, `DLA.m` and `DLA2Boundaries.m`. The first file computes the Single Layer Problem and its corresponding matrix. The second file computes the Double Layer Problem and its corresponding matrix. The last file builds on the previous approaches to solve the mixed problem. After that the matrices in (15) are assembled and the SLE is solved to get the values of u on Γ_k.

2. We apply the gradient descent method, which is implemented in `newGamma.m`, in order to obtain γ_{k+1}, precisely we calculate the coefficients $(\alpha_j)_{j=0}^{n-1}$ and $(\beta_j)_{j=0}^{n-1}$ of the trigonometric polynomial for the radius function of γ_{k+1}.

3. Finally, we check the stopping condition $\|u(\gamma_{k+1})\| \leq \varepsilon$ (for a chosen $\varepsilon > 0$), where $u(\gamma_{k+1})$ means the Dirichlet values of u on Γ_{k+1}. If the condition is not satisfied, then we start at step 1 again, now with γ_{k+1} instead of γ_0. If the condition is satisfied, then we have found our approximation of Γ.

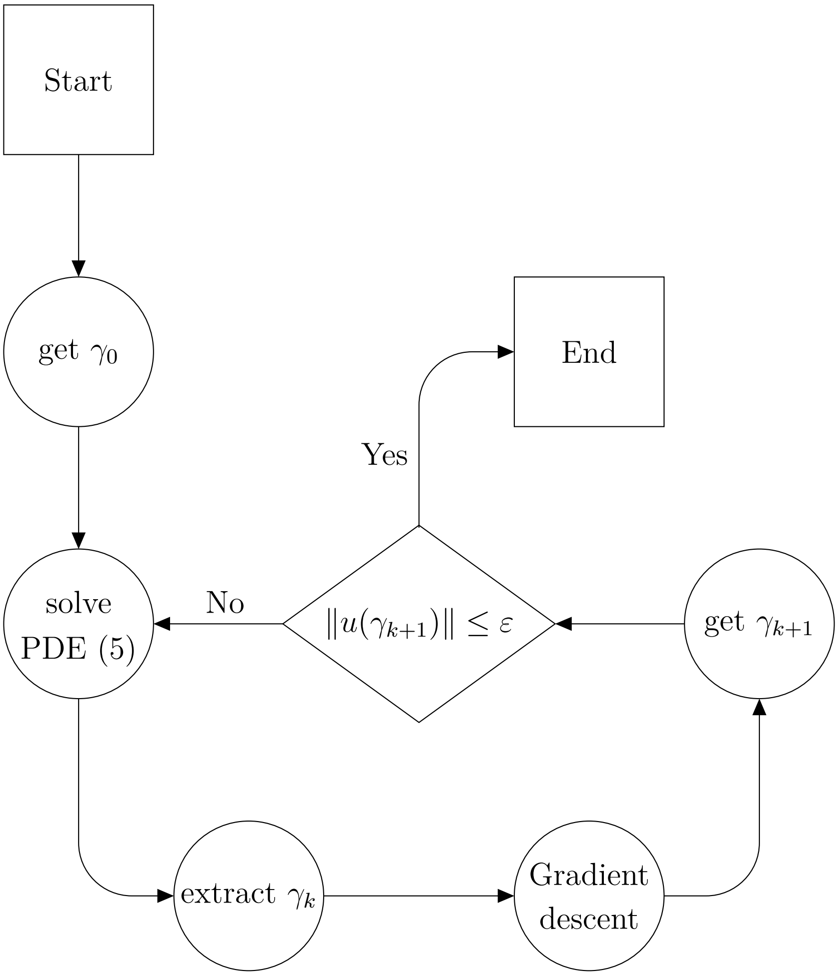


Figure 3: Workflow

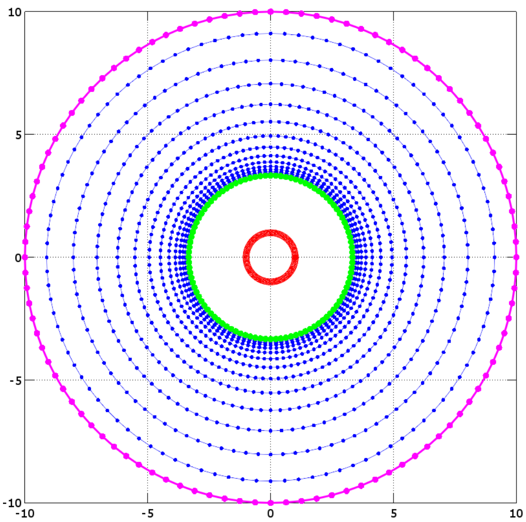
7 Numerical Examples

Here we can see some examples of calculation. Every Dirichlet data and every Neumann data was assumed to be constant on the respective sets. In the following Figures we see some illustrations of our iterative formula. The boundary Σ is colored in red and the initial guess Γ_0 is colored in magenta. Every fourth iteration of the iterative process was plotted and colored in blue. In Table 1 we can see, what parameters and boundary data were chosen, to get this examples. Each calculation falls below the maximum permissible error ε. The final Γ_k, which is our approximation for Γ, is also plotted and colored in green.

Table 1: Data for the different examples, illustrated in the Figures

Figure	#discretization points	max. Iterations	error tolerance ε	Dirichlet data on Σ	Neumann data on Γ_k, $\forall k$
4(a)	100	100	0.5	-20.0	5.0
4(b)	100	100	0.5	-20.0	5.0
5	100	30	0.5	-5.0	30.0

In Figure 4 we can see two different choices for Γ_0. The remaining parameters were chosen equally. We see the iteration (blue) converge against a set (approximated by green).



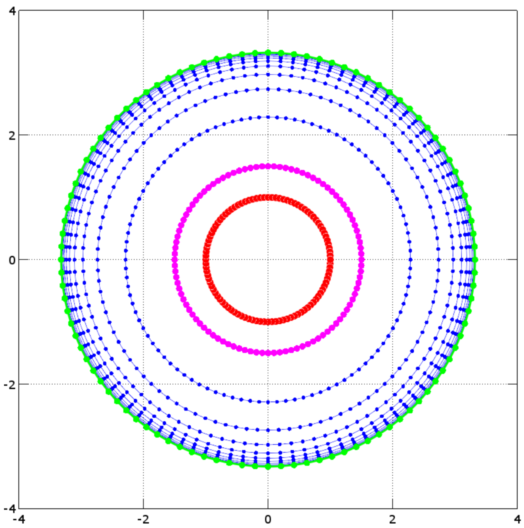


Figure 4: Circular pipe with different initial guess Γ_0

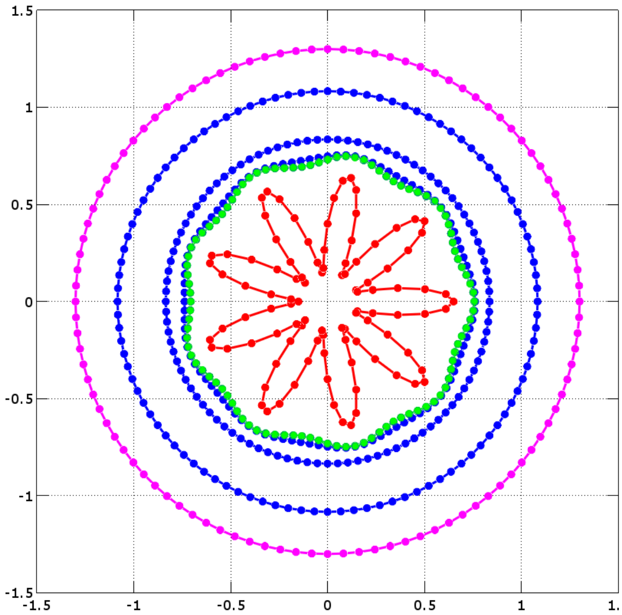


Figure 5: Star shaped pipe

A suggestion for further research is to obtain the optimal settings as well as creating a model which uses data for the soil and pipe specifications. Furthermore it is useful to construct higher order iterative processes, to get faster convergence.

8 Group Work Dynamics

The first day of ECMI Modelling Week we were introduced to the Modelling of Ice Around Cooling Pipes theme. We discussed about physical model (Figure 1) which gives an illustration of our problem and 'Stefan condition' that the temperature, u is satisfying. After that, we discussed about different methods to compute Γ.

Next day, we were splitted in two groups and discussed about the two different approaches of formula (6) discretization. One group worked on the Single Layer Approach while the other group worked on the Double Layer Approach. For the practical implementation of the theory above we used Octave. Later we mixed the groups such that anybody could tell everything to the other students about his/her approach. We also made sure that there was at least one student in every group with programming knowledge of Octave.

After this process the phase was starting to program all discretization functions in Octave and testing them. Each group programmed their special approach. After two days we could combine our programs to get the final version. Friday evening we visualized our results and prepared the presentation for Saturday, July 21st.

After the Modeling Week, we stayed in contact through social networks to work together on the report. Everyone wrote on the section he/she worked most during the week.

9 Instructor's Assessment

Although the phenomena of ice around cooling pipes is rather familiar nowadays, its modelling and formulation in mathematical terms is no easy task. I thought that the students did a very good job in splitting up the modelling problem into smaller tasks which they distributed among each other. Despite their very different backgrounds there was a constant collaboration and communication to put achieved results together to simulate a challenging problem. I find it in particular mentionable that all students were involved at all times during the modelling week, such that their results can be considered as the result of a true group work. The fact that the final simulations were done in two dimensions is especially mentionable.

References

Harbrecht, Helmut (2012). *Integralgleichungen und Randelemente.*

Kurz, Stefan (2015). *Fast Boundary Element Methods for Engineers.*

Rumpf, M. and M. Flucher (1997). “Bernoulli’s free-boundary problem, qualitative theory and numerical approximation”. In: *Journal für die reine und angewandte Mathematik (Crelles Journal)*, pp. 165–204.

Steinbach, Olaf (2008). *Numerical Approximation Methods for Elliptic Boundary Value Problems.* Springer-Verlag.