An Adaptive Model Switching and Discretization Algorithm for Gas Flow on Networks

Pia Domschke*, Oliver Kolb, Jens Lang
*domschke@mathematik.tu-darmstadt.de

TECHNISCHE UNIVERSITÄT DARMSTADT

Research Group
Numerical Analysis and
Scientific Computing

March 4, 2010

Abstract

We are interested in the simulation and optimization of gas transport in networks. Those networks consist of pipes and various other components like compressor stations and valves. The gas flow through the pipes can be modelled by different equations based on the Euler equations. For the other components, purely algebraic equations are used. Depending on the data, different models for the gas flow can be used in different regions of the network. We use adjoint techniques to specify model and discretization error estimators and present a strategy that adaptively applies the different models while maintaining the accuracy of the solution.

1 Introduction

Intense research has been done in the field of simulation and optimization of gas transport in networks in the last years [2, 6, 11, 12, 3, 5, 4]. The aim of operating a gas transmission network is to minimize the running costs of the compressor stations whereas the contractual demand of the customers has to be met. The large extent of gas networks and their high complexity make the optimization a difficult computational task. For the optimization, it is necessary to efficiently solve the underlying equations on networks within a certain tolerance. In this paper, we present an adaptive model switching and discretization algorithm that is suitable for these requirements.

The equations describing the flow of gas through a pipe are based on the Euler equations, a hyperbolic system of partial differential equations. The transient flow of gas may be described appropriately by equations in one space dimension. Other components of the network, like compressor stations and valves, follow algebraic equations. For the whole network, adequate initial, boundary (at sources and sinks) and coupling conditions have to be given.

Solving hyperbolic PDEs in one space dimension does not pose a challenge, but the complexity of the whole problem increases with the size of the network. Thus, we use a hierarchy of three models that describe the flow of gas with different accuracy, but also with different computational effort. We then want to use the simplified models in regions with low activity, while sophisticated models have to be used in regions, where the dynamical behaviour has to be resolved in detail.

Since the behaviour of the network changes both in space and time, an automatic steering of the model hierarchy as well as the discretization is necessary. We introduce error estimators for the discretization and the model errors using adjoint techniques and present a strategy to automatically balance those errors with respect to a given tolerance.

Existent software packages like SIMONE [1] may also use different models for the simulation task. However, one model has to be chosen in advance, which is often too restrictive.

The paper is organized as follows. The modelling of the network as well as the different models are introduced in Sect. 2. In Sect. 3, error estimators for both, the model and the discretization error, are derived using adjoint techniques. We present a strategy to adaptively balance model and discretization error in Sect. 4. Numerical Results are given in Sect. 5.

2 Model Hierarchy and Network

In this section, we describe the modelling of the network. We introduce a hierarchy consisting of three different models describing the flow of gas through a pipe. Each model results from the previous one by making further simplifying assumptions [2]. The most complex model is the nonlinear model followed by the linear model. The most simple model used is the algebraic model (see Fig. 1). Also, further network components are given.

Figure 1: Model hierarchy

2.1 Network

The gas network is modelled as a directed graph $\mathcal{G} = (\mathcal{J}, \mathcal{V})$ with edges $\mathcal{J}$ and vertices $\mathcal{V}$ (nodes, branching points). The set of edges $\mathcal{J}$ consists of pipes $j \in \mathcal{J}_p$, compressor stations $c \in \mathcal{J}_c$ and valves $\mathrm{v} \in \mathcal{J}_\mathrm{v}$. Each pipe $j \in \mathcal{J}_p$ is defined as an interval $[x_j^a, x_j^b]$ with a direction from x_j^a to x_j^b. In each pipe, one of the models holds and adequate initial and coupling as well as boundary conditions have to be specified. Valves and compressor stations are described by algebraic equations.

2.2 Nonlinear Model

The isothermal Euler equations, which describe the flow of gas, consist of the continuity and the momentum equation together with the equation of state for real gases. With some simplifying assumptions, as the pipe being horizontal and a constant speed of sound [4], the equations are

$$p_t + \frac{\rho_0 c^2}{A} q_x = 0, \qquad q_t + \frac{A}{\rho_0} p_x + \frac{\rho_0 c^2}{A} \left(\frac{q^2}{p} \right)_x = -\frac{\lambda \rho_0 c^2 |q| q}{2dAp} . \tag{1}$$

Here, q denotes flow rate under standard conditions (1 atm air pressure, temperature of 0 °C), p the pressure, c the speed of sound, λ the friction coefficient, d the diameter and A the cross-sectional area of the pipe, and ρ_0 the density under standard conditions.

2.3 Semilinear Model

If the velocity of the gas is much smaller than the speed of sound, i.e., $|\mathrm{v}| \ll c$ with $\mathrm{v} = \frac{\rho_0 q}{\rho A}$, we can neglect the nonlinear term in the spatial derivative of the momentum equation in (1). This yields a semilinear model

$$p_t + \frac{\rho_0 c^2}{A} q_x = 0, \qquad q_t + \frac{A}{\rho_0} p_x = -\frac{\lambda \rho_0 c^2 |q| q}{2dAp} . \tag{2}$$

2.4 Algebraic Model

A further simplification leads to the stationary model: Setting the time derivatives in (2) to zero results in an ordinary differential equation, which can be solved analytically:

$$q = \text{const.}, \qquad p(x) = \sqrt{p(x_0)^2 + \frac{\lambda \rho_0^2 c^2 |q| q}{dA^2} (x_0 - x)} . \tag{3}$$

Here, $p(x_0)$ denotes the pressure at an arbitrary point $x_0 \in [0, L]$. Setting $x_0 = 0$, that is, $p(x_0) = p(0) = p_{in}$ at the inbound of the pipe, and $x = L$, that is, $p(x) = p(L) = p_{out}$ at the end of the pipe, yields the algebraic model [13].

2.5 Further Network Components

Besides pipes, there are some other components a network can consist of. Those are, for example, compressor stations and valves. These components are, like the pipes, modelled as edges. This way, the coupling conditions at the intersections are still valid. Flow rate and pressure are determined by algebraic equations that can be nonlinear.

Compressor Station

A compressor station is a facility that increases the pressure of the gas. Running a compressor is relatively costly since the compressor station consumes some of the gas. The equation for the fuel consumption of a compressor is given by [9]

$$F(p_{in}, p_{out}, q_{in}) = c_F q_{in} \left(\left(\frac{p_{out}}{p_{in}} \right)^{\frac{\gamma - 1}{\gamma}} - 1 \right) , \tag{4}$$

the compressor power P is determined by

$$P(p_{in}, p_{out}, q_{in}) = c_P q_{in} \left(\left(\frac{p_{out}}{p_{in}} \right)^{\frac{\gamma - 1}{\gamma}} - 1 \right) . \tag{5}$$

Here, γ is the isentropic coefficient of the gas, and c_F and c_P are compressor specific constants. The control of the compressor is given by the pressure difference.

Valves

Valves are used to regulate the flow of the gas by opening or closing. In case of an open valve, the equations $q_{in} = q_{out}$, $p_{in} = p_{out}$ hold. If the valve is closed, it is $q_{in} = q_{out} = 0$.

3 Error Estimators

With the different models for the pipes and the other network components we can solve the whole network as a system using adequate initial, boundary and coupling conditions. A way to achieve a compromise between the accuracy of the solution and the computational costs is to use the more complex model only when necessary and to refine the discretization only where needed. Using the solution of adjoint equations as done in [7, 8, 4], we deduce a model and a discretization error estimator to measure the influence of the model and the discretization on a user-defined output functional M.

The functional M can be of any form, for example measuring the pressure value troughout the network over the whole time interval,

$$M(p) = \int_0^T \int_\Omega p(x,t)\,\mathrm{d}x.$$

Another possibility is to measure the costs of the compressor stations in form of the power consumption,

$$M(p,q) = \sum_{c \in \mathcal{J}_c} \int_0^T F_c(t)\,\mathrm{d}t. \tag{6}$$

Let $\xi = (\xi_1, \xi_2)^T$ be the adjoint pressure and flow rate of one of the models with respect to the functional M. We now use the adjoint equations to assess the simplified models with respect to the quantity of interest. Let $u = (p,q)^T$ be the solution of the nonlinear model (1) and $u^h = (p^h, q^h)^T$ the discretized solution of the semilinear model (2). For the discretization of the PDE, we apply an implicit box scheme [10]. Then the difference between the output functional $M(u)$ and $M(u^h)$ can be approximated using Taylor expansion. Inserting the solution ξ of the adjoint system, we get a first order error estimator for the model and the discretization error respectively as in [4] (see also [8]):

$$M(u) - M(u^h) \approx \eta_m + \eta_h \tag{7}$$

with the estimators η_m and η_h as follows:

$$\eta_m^{\text{LIN-NL}} = \int_0^T \int_\Omega -\xi^T \begin{pmatrix} 0 \\ \frac{\rho_0 c^2 (q^h)^2}{A p^h} \end{pmatrix}_x \mathrm{d}x\,\mathrm{d}t \tag{8}$$

$$\eta_h^{\text{LIN}} = \int_0^T \int_\Omega -\xi^T \begin{pmatrix} p_t^h + \frac{\rho_0 c^2}{A} q_x^h \\ q_t^h + \frac{A}{\rho_0} p_x^h + \frac{\lambda \rho_0 c^2 |q^h| q^h}{2 d A p^h} \end{pmatrix} \mathrm{d}x\,\mathrm{d}t. \tag{9}$$

If, the other way round, u denotes the solution of the semilinear model (2) and u^h the discretized solution of the nonlinear model, one gets the same estimator for the model error (except for the sign), and the discretization error reads as follows,

$$\eta_h^{\text{NL}} = \int_0^T \int_\Omega -\xi^T \begin{pmatrix} p_t^h + \frac{\rho_0 c^2}{A} q_x^h \\ q_t^h + \frac{A}{\rho_0} p_x^h + \frac{\rho_0 c^2}{A} \left(\frac{(q^h)^2}{p^h}\right)_x + \frac{\lambda \rho_0 c^2 |q^h| q^h}{2 d A p^h} \end{pmatrix} \mathrm{d}x\,\mathrm{d}t. \tag{10}$$

Since the algebraic model can be solved exactly, the discretization error disappears and one only gets an estimator for the model error

$$\eta_m^{\text{ALG-LIN}} = \int_0^T \int_\Omega -\xi^T \begin{pmatrix} p \\ q \end{pmatrix}_t \mathrm{d}x\,\mathrm{d}t \tag{11}$$

with ξ being the solution of the adjoint equations either of the semilinear model or of the algebraic model. Here, $u = (p,q)^T$ denotes the solution of the stationary model (3).

The discretization error estimators η_h^{NL} and η_h^{LIN} may be split up into a temporal and a spatial discretization error estimator as follows. Let u be the exact and u^h be the discretized solution of the nonlinear model (1). We use a short notation of (1), i.e., $u_t + f(u)_x = g(u)$, which yields

$$\begin{aligned}\eta_h^{\text{NL}} &= \int_0^T \int_\Omega -\xi^T \left(u_t^h + f(u^h)_x - g(u^h)\right) \mathrm{d}x\,\mathrm{d}t \\ &= \int_0^T \int_\Omega -\xi^T \left((u_t^h - u_t) + (f(u^h)_x - f(u)_x) - (g(u^h) - g(u))\right) \mathrm{d}x\,\mathrm{d}t\end{aligned}$$

since u is the exact solution of (1). We may split the integral into two parts

$$\eta_h^{\text{NL}} = \underbrace{\int_0^T \int_\Omega -\xi^T (u_t^h - u_t)\,\mathrm{d}x\,\mathrm{d}t}_{=:\eta_t^{\text{NL}}} + \underbrace{\int_0^T \int_\Omega -\xi^T \left((f(u^h)_x - f(u)_x) - (g(u^h) - g(u))\right) \mathrm{d}x\,\mathrm{d}t}_{=:\eta_x^{\text{NL}}} \tag{12}$$

The temporal and spatial discretization error estimator for the semilinear model are derived analogously. For the computation, the exact solution is approximated by a higher order reconstruction using neighboring points. For the time derivative, we use a polynomial reconstruction of order 2 and denote it by $u_t \approx R_t(u^h)$. The spatial derivative of f and the value of g are reconstructed with order 4, giving $f(u)_x \approx R_x(f(u^h))$ and $g(u) \approx R(g(u^h))$, respectively.

The estimators computed are then

$$\begin{aligned}\eta_t^{\text{NL}} &\approx \int_0^T \int_\Omega -\xi^T \left(u_t^h - R_t(u^h)\right) \mathrm{d}x\,\mathrm{d}t\,, \\ \eta_x^{\text{NL}} &\approx \int_0^T \int_\Omega -\xi^T \left(\left(f(u^h)_x - R_x(f(u^h))\right) - \left(g(u^h) - R(g(u^h))\right)\right) \mathrm{d}x\,\mathrm{d}t\,.\end{aligned}$$

4 Balancing Model and Discretization Error

With the above defined estimators, we can introduce a strategy to balance the model and discretization error inside the network. In practice, often smaller time horizons are optimized. Thus, we do not control the overall error any more, but the relative error piecewise. For this, we divide the time interval $[0, T]$ into subintervals of equal size $[T_{k-1}, T_k]$. Regarding one subinterval $[T_{k-1}, T_k]$, we can compute the forward as well as the backward/adjoint solution and evaluate the error estimators, which yields

$$M_k(u) - M_k(u^h) \approx \eta_{m,k} + \eta_{t,k} + \eta_{x,k}.$$

Given a tolerance TOL for the relative error, we can approximate the exact error by the estimators giving

$$\frac{\left|M_k(u) - M_k(u^h)\right|}{\left|M_k(u)\right|} \approx \frac{\left|\eta_{m,k} + \eta_{t,k} + \eta_{x,k}\right|}{\left|M_k(u^h)\right|} \leq \texttt{TOL}. \tag{13}$$

We first check the discretization error to ensure the discretization to be adequate. Then, the model error is taken into account. A scheme of the algorithm is given in Fig. 2.

Check discretization error

First, the discretization is checked. Given the tolerance TOL as above, we ensure the discretization error to be small compared to the model error by decreasing TOL by a user-defined factor $0 < \kappa < 1$ giving $\kappa \cdot \texttt{TOL} =: \texttt{TOL}_h$. We demand the discretization error estimator to satisfy

$$\left|\eta_{t,k} + \eta_{x,k}\right| < \texttt{TOL}_h \cdot \left|M_k(u^h)\right|.$$

If the error estimator exceeds the given upper bound, the temporal and spatial discretization errors are treated individually, that is,

$$\left|\eta_{t,k}\right| < \frac{1}{2}\texttt{TOL}_h \cdot \left|M_k(u^h)\right| \text{ and } \left|\eta_{x,k}\right| < \frac{1}{2}\texttt{TOL}_h \cdot \left|M_k(u^h)\right|.$$

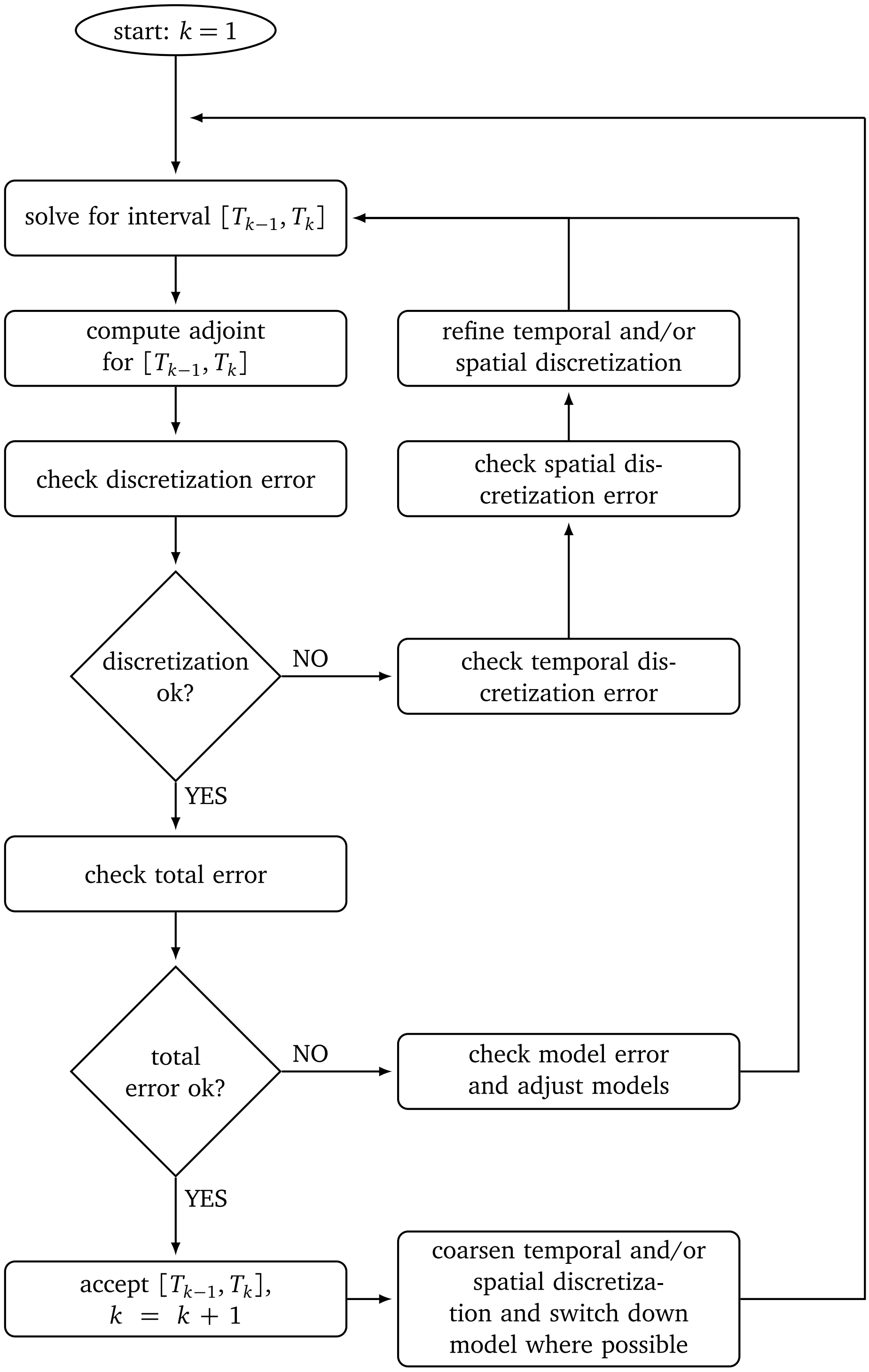

Figure 2: Scheme of the balancing algorithm

Check Temporal Discretization Error

If the temporal error estimator exceeds the given tolerance, the time step size is marked for refinement. After checking the spatial discretization error, the time interval $[T_{k-1}, T_k]$ has to be computed again. If, in contrast, the error estimator $\left|\eta_{t,k}\right|$ is much smaller than the upper bound, the time step size is marked for coarsening. If the current time interval has to be recomputed due to spatial or model errors, the temporal coarsening is nevertheless directly applied.

Check Spatial Discretization Error

Now, the spatial discretization error is estimated locally for each pipe. We can split up the discretization error estimator from (12) for each pipe, giving for the nonlinear model

$$\eta_{x,k}^{\mathrm{NL}} = \int_{T_{k-1}}^{T_k} \sum_{j\in\mathcal{J}_p} \int_{x_j^a}^{x_j^b} -\xi^T \Big((f(u^h)_x - f(u)_x) - (g(u^h) - g(u)) \Big) \,\mathrm{d}x\,\mathrm{d}t = \sum_{j\in\mathcal{J}_p} \eta_{x,k,j}^{\mathrm{NL}},$$

analogously for the other models. Thus, we can estimate the spatial discretization error for each pipe in the time interval $[T_{k-1}, T_k]$ with the corresponding model, which yields

$$\left| \sum_{j\in\mathcal{J}_p} \eta_{x,k,j} \right| < \frac{1}{2}\mathtt{TOL}_h \cdot \left|M_k(u^h)\right|.$$

For the inequality to hold, it suffices to claim

$$\sum_{j\in\mathcal{J}_p} \left|\eta_{x,k,j}\right| < \frac{1}{2}\mathtt{TOL}_h \cdot \left|M_k(u^h)\right|.$$

In order to get an upper bound for each pipe itself, we uniformly distribute the target functional, i.e., we divide it by the number of pipes, giving

$$\left|\eta_{x,k,j}\right| < \frac{1}{2}\mathtt{TOL}_h \cdot \frac{\left|M_k(u^h)\right|}{\left|\mathcal{J}_p\right|} \qquad \forall j \in \mathcal{J}_p.$$

If $\left|\eta_{x,k,j}\right|$ exceeds the given tolerance, the pipe is marked for refinement. If instead, the error estimator is much smaller than the right hand side, the pipe is marked for coarsening.

The time interval $[T_{k-1}, T_k]$ is computed again with a finer discretization where needed.

Check Total Error

If the discretization error is accepted, the total error estimator $\left|\eta_{m,k} + \eta_{t,k} + \eta_{x,k}\right|$ is evaluated. If

$$\left|\eta_{m,k} + \eta_{t,k} + \eta_{x,k}\right| \le \mathtt{TOL} \cdot \left|M_k(u^h)\right|,$$

the time interval $[T_{k-1}, T_k]$ is accepted, k is increased and the next interval is computed.

Check Model Error and Adjust Models

If the discretization error is small enough, but the total error is not, the model errors of all pipes are checked, i.e., the model error estimators (8) and/or (11) are evaluated for each pipe.

For the pipes using the semilinear or algebraic model, first the estimators with respect to the higher models are evaluated. If the error estimator exceeds the given tolerance, that is,

$$\eta_{m,k,j} \ge \mathtt{TOL}_m \cdot \frac{M_k(u^h)}{\left|\mathcal{J}_p\right|},$$

with $\mathtt{TOL}_m := (1-\kappa)\cdot\mathtt{TOL}$, the pipe is supposed to use the model above subject to the hierarchy.

Then, the estimators with respect to the lower models are computed for the pipes using the nonlinear or the semilinear model. If the error estimator is much less than the given tolerance, that is,

$$\eta_{m,k,j} < s \cdot \mathtt{TOL}_m \cdot \frac{M_k(u^h)}{\left|\mathcal{J}_p\right|},$$

with a "shift down factor" $s \ll 1$ (e.g. 10^{-1} or 10^{-2}), the pipe can use the lower model for the next calculations. The time interval $[T_{k-1}, T_k]$ is computed again with the adjusted models.

5 Numerical Results

In this section, we give numerical results for a medium sized real life network. The network consists of 12 pipes (P01 – P10, with lengths between 30km and 100km), 2 sources (S01 – S02), 4 consumers (C01 – C04), 3 compressor stations (Comp01 - Comp03) and one control valve (CV01). The graph of the network is shown in Fig. 3.

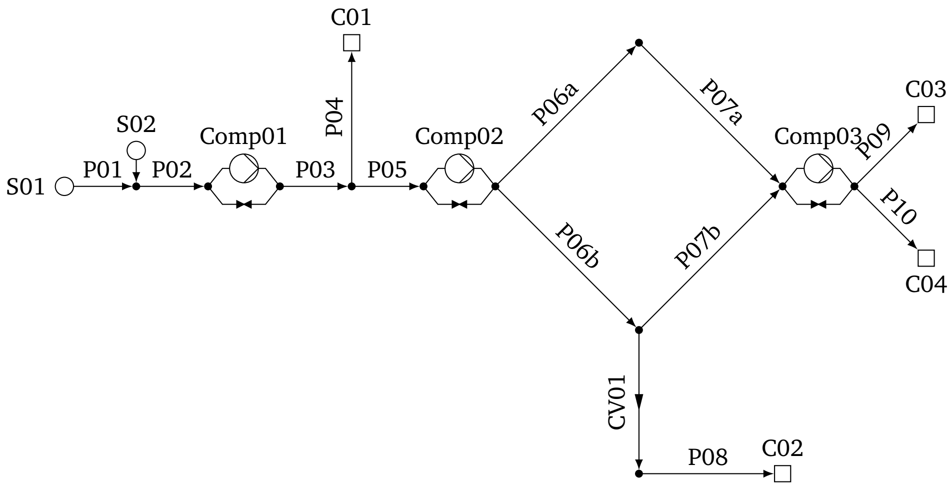

Figure 3: Network with compressor stations and control valve

The simulation starts with stationary initial data. The boundary conditions and the control for the compressor stations are time-dependent. The target functional is given by the total fuel gas consumption of the compressors, i.e.

$$M(u) = \sum_{c \in \mathcal{J}_c} \int_0^T F_c(t)\,\mathrm{d}t.$$

The simulation time is 14400 seconds with an initial time step size $\Delta t = 1800$s. The subintervals are 3600s each. The initial spatial step size is $\Delta x = 10000$m. The factor κ is set to 10^{-1}.

Table 1 shows the maximal relative error in the target functional

$$\text{rel.err.} = \max_k \frac{\left|M_k(u) - M_k(u^h)\right|}{\left|M_k(u)\right|},$$

the total target functional, the maximal and the minimal time and spatial step size used and the total time for the computation subject to the tolerance TOL. As an approximation of the exact solution we computed a solution with the nonlinear model and a finer discretization than used in the adaptive algorithm, which is shown in the last row. For a comparison of the computation time, we computed a solution without adaptivity using the nonlinear model. The discretization was chosen to be similar to the one used at tolerance 10^{-4}, but no error estimators were calculated. The time needed for computation was 4.20e+01 seconds, i.e., the overhead for the adaptive model switching and discretization is only 3.2%. Clearly, the adaptive algorithm also delivers reliable information about the accuracy of the discrete solution.

The computation was done on a 3100MHz AMD Athlon™ 64 X2 Dual Core 6000+.

Table 1: Result of the algorithm using different values of TOL

TOL	rel.err.	$M(u^h)$	max/min Δt	max/min Δx	time [s]
1e-01	5.5264e-02	13.240217	1800/1800	20,000/10,000	7.20e-02
1e-02	4.2304e-03	12.756843	900/900	20,000/10,000	1.84e-01
1e-03	2.5513e-04	12.730996	112.5/112.5	20,000/2,500	1.18e+00
1e-04	3.3264e-05	12.728721	7.03125/7.03125	10,000/625	4.34e+01
reference solution		12.728459	3.515625	312.5	1.46e+02

Table 2: Models used during calculation depending on the tolerance TOL

TOL	ALG	LIN	NL
1e-01	100%	0%	0%
1e-02	50.0%	50.0%	0%
1e-03	8.3%	91.7%	0%
1e-04	0.0%	75.0%	25.0%

Besides the discretization, it is also interesting how the model switching part works depending on TOL. Table 2 shows how often which model is used during calculation. It can be seen that the smaller the tolerance the better models are used.

Figure 4 shows a snapshot of the simulation at time $t = 3600$s with $\text{TOL} = 10^{-4}$. At the sources and sinks, the upper numbers denote the pressure in bar, the lower ones denote the flow rate. At the compressor stations, the upper numbers are the increase in pressure and also the flow rate is shown below. At all inner nodes, only the pressure is printed. At each pipe, also the used model is indicated. The small white dots in the thick black lines represent the discretization. Note that the picture is not to scale.

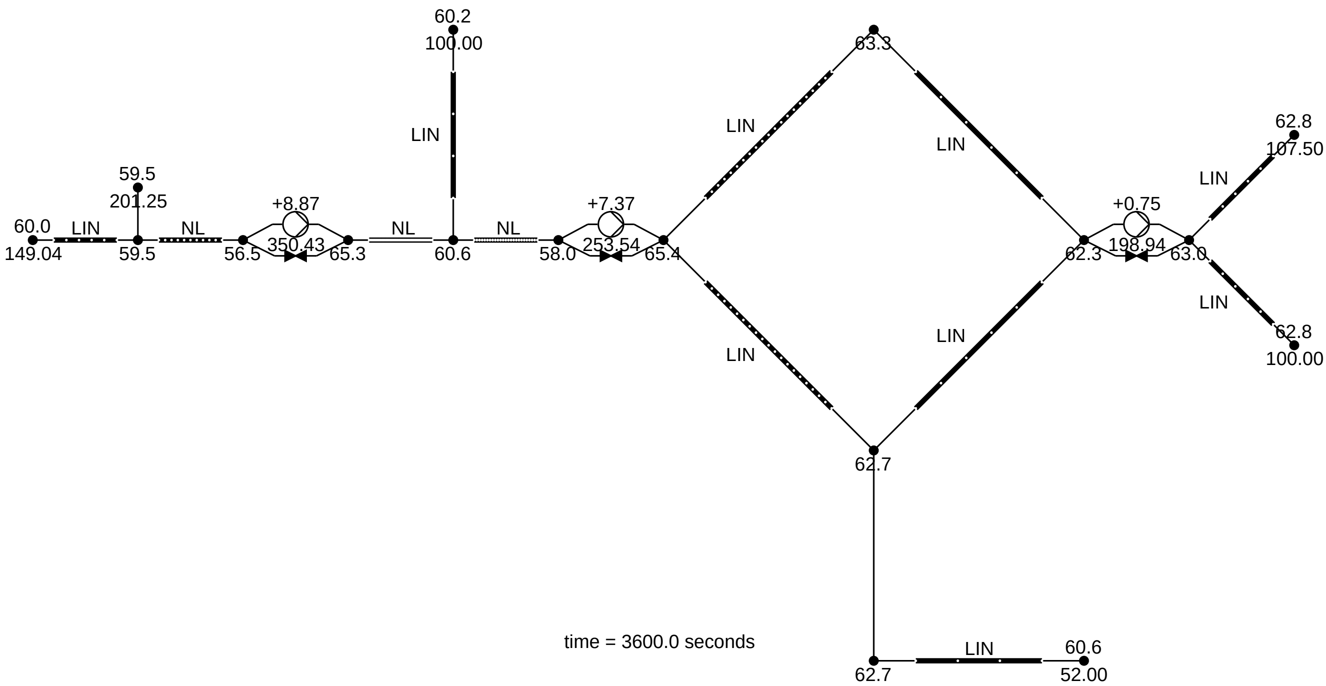

Figure 4: Snapshot of the network at time $t = 3600$ s

6 Summary

We presented an adaptive model switching and discretization algorithm. For that we used a hierarchy of models that describe the flow of gas through a pipe qualitatively different. Using adjoint techniques, we introduced error estimators for the model errors as well as for the discretization errors. With these estimators we developed an algorithm that balances the model and discretization errors subject to a given tolerance and automatically switches between the models in the hierarchy. It could be seen that for different tolerances, the discretization was adaptively adjusted and also the different models were used. Also, if the algorithm is used in an optimization framework, many nonlinearities can be avoided, since the nonlinear model is only used where needed. That means a dramatic reduction of complexity and degrees of freedom for the optimization part.

Acknowledgments

This paper was supported by the German Research Foundation (DFG) under the grant LA1372/5-1.

References

[1] *SIMONE Software*. http://www.simone.eu/simone-simonesoftware.asp

[2] BALES, Pia: *Hierarchische Modellierung der Eulerschen Flussgleichungen in der Gasdynamik*. Darmstadt, Technische Universität Darmstadt, Department of Mathematics, Diploma Thesis, 2005

[3] BALES, Pia ; GEISSLER, Björn ; KOLB, Oliver ; LANG, Jens ; MARTIN, Alexander ; MORSI, Antonio: Combination of Nonlinear and Linear Optimization of Transient Gas Networks / TU Darmstadt. 2008 (2552). – Forschungsbericht

[4] BALES, Pia ; KOLB, Oliver ; LANG, Jens: Hierarchical Modelling and Model Adaptivity for Gas Flow on Networks. In: *Computational Science - ICCS 2009* Bd. 5544, Springer Berlin / Heidelberg, 2009 (Lecture Notes in Computer Science), S. 337–346

[5] BANDA, Mapundi ; HERTY, Michael: Multiscale modeling for gas flow in pipe networks. In: *Math. Meth. Appl. Sci.* 31 (2008), August, S. 915–936. http://dx.doi.org/10.1002/mma.948. – DOI 10.1002/mma.948

[6] BANDA, Mapundi ; HERTY, Michael ; KLAR, Axel: Coupling conditions for gas networks governed by the isothermal Euler equations. In: *NHM* 1 (2006), Juni, Nr. 2, S. 295–314

[7] BECKER, Roland ; RANNACHER, Rolf: An optimal control approach to a posteriori error estimation in finite element methods. In: *Acta numerica* 10 (2001), S. 1–102

[8] BRAACK, M. ; ERN, A.: A posteriori control of modeling errors and discretization errors. In: *SIAM Journal of Multiscale Modeling and Simulation* 1 (2003), Nr. 2, S. 221–238

[9] HERTY, M.: Modeling, simulation and optimization of gas networks with compressors. In: *Networks and Heterogeneous Media* 2 (2007), Nr. 1, S. 81–97

[10] KOLB, Oliver ; LANG, Jens ; BALES, Pia: An implicit box scheme for subsonic compressible flow with dissipative source term. In: *Numerical Algorithms* 53 (2010), Nr. 2, S. 293–307. http://dx.doi.org/10.1007/s11075-009-9287-y. – DOI 10.1007/s11075–009–9287–y

[11] MARTIN, A. ; MÖLLER, M. ; MORITZ, S.: Mixed integer models for the stationary case of gas network optimization. In: *Mathematical Programming* 105 (2006), S. 563–582

[12] MORITZ, Susanne: *A Mixed Integer Approach for the Transient Case of Gas Network Optimization*, TU Darmstadt, Diss., 2006

[13] SEKIRNJAK, Erwin: Transiente Technische Optimierung / PSI AG. 2000. – Entwurf