Mixing Flows in Dynamic Fluid Transport Simulations

Mehrnaz Anvari
Fraunhofer Institute for Algorithms and Scientific Computing SCAI
Sankt Augustin, Germany
email: Mehrnaz.Anvari@scai.fraunhofer.de

Anton Baldin
PLEdoc GmbH and Fraunhofer Institute SCAI
Sankt Augustin, Germany
email: Anton.Baldin@scai.fraunhofer.de

Tanja Clees
University of Applied Sciences Bonn-Rhein-Sieg and Fraunhofer Institute SCAI
Sankt Augustin, Germany
email: Tanja.Clees@scai.fraunhofer.de

Bernhard Klaassen
Fraunhofer Research Institution for Energy Infrastructures IEG
Bochum, Germany
email: Bernhard.Klaassen@ieg.fraunhofer.de

Igor Nikitin
Fraunhofer Institute for Algorithms and Scientific Computing SCAI
Sankt Augustin, Germany
email: Igor.Nikitin@scai.fraunhofer.de

Lialia Nikitina
Fraunhofer Institute for Algorithms and Scientific Computing SCAI
Sankt Augustin, Germany
email: Lialia.Nikitina@scai.fraunhofer.de

***Abstract*—This paper presents a new numerically efficient implementation of flow mixing algorithms in dynamic simulation of pipeline fluid transport. Mixed characteristics include molar mass, heat value, chemical composition and the temperature of the transported fluids. In the absence of chemical reactions, the modeling is based on the universal conservation laws for molar flows and total energy. The modeling formulates a sequence of linear systems, solved by a sparse linear solver, typically in one iteration per integration step. The functionality and stability of the developed simulation methods have been tested on a number of realistic network scenarios. The main output of the paper is a functioning and stable implementation of flow mixing algorithms for dynamic simulation of fluid transport networks.**

Keywords-simulation and modeling; mathematical and numerical algorithms and methods; mixing flows; pipeline fluid transport.

I. Introduction

This paper continues a series of our works on modeling of fluid transport networks. Previous works presented stationary [1] and dynamic [2] modeling of fluid transport networks limited to a single chemical composition and constant temperature. In addition, some aspects of stationary modeling of mixing fluids of different compositions and/or temperatures were considered in [3]. In this paper, flow mixing modeling will be considered in more detail, with special emphasis on the thermodynamic layer of the model. In particular, dynamic mixing equations and algorithms for their solving will be presented. The developed approach is implemented in our Multi-phYsics Network Simulator (MYNTS) [4], which is used to solve actual transport scenarios for natural gas [5], hydrogen [6], carbon dioxide [7], water [8] and other fluids.

Fluid transport modeling is based on the conservation of mass flows in the form of dynamic Kirchhoff equations; Darcy-Weisbach pipeline pressure drop formula, with empirical friction term by Nikuradse [9] and Hofer [10]; equation of state computation by simplified analytical models by Papay [11], Peng-Robinson [12] and Soave-Redlich-Kwong [13] or more complex ISO-norm models AGA8-DC92 [14] and GERG2008 [15]–[17].

In state-of-the-art, a number of previous studies [18]–[24] considered modeling of pipeline fluid transport, both at the universal mathematical level [18], and in various application scenarios. Such scenarios include transport of natural gas [19] [21], steam transport in oil refineries [20], carbon dioxide transport [22]–[24]. All these works are characterized by the presentation of transport equations as laws of conservation of mass, momentum and energy. In the presence of various substances, conservation of molar flows is added, while the general relations of thermodynamics of open systems [25] regulate the relations of energy and temperature.

A common drawback of existing solutions is the closed nature of modeling within blackbox systems. If it is necessary to change the modeling, modify or introduce new equations and variables, the system must be reprogrammed. In addition, existing systems experience difficulties in solving large realistic network problems in the presence of numerical instabilities. The novelty of our approach consists in transparent modeling, where the user can freely change the equations and experiment with different forms of representing physical processes in fluid transport networks. We also pay special attention to the stability and performance of solution algorithms, which is especially important for realistic scenarios with a large number of elements. The purpose of this work is to extend transparent and numerically stable modeling to mixing flows present in realistic fluid transport scenarios.

In this work, Section II presents the modeling of mixing flows incorporating molar and temperature relationships. Section III describes the numerical experiments performed using the developed methods. Section IV summarizes the main results and conclusions of the work.

II. Modeling of mixing flows

This section describes the details of modeling of mixing flows, consisting of modeling fluid molar composition and

temperature distribution.

A. Molar fluid composition

A fluid transport network is described by a directed graph consisting of nodes and edges connecting them. The graph is described by an incidence matrix I_{ne}, in which each edge e has nonzero entries for the nodes n that this edge connects; -1 for the node that edge comes from, $+1$ for the node that edge enters. Mixing fluid flows are described by following equations

$$V_n \partial \rho_n / \partial t = \sum_e I_{ne} m_e, \quad (1)$$

$$V_n \partial (\rho_n \mu_n^{-1}) / \partial t = \sum_e I_{ne} m_e \mu_e^{-1}, \quad (2)$$

$$V_n \partial (\rho_n \mu_n^{-1} x_n) / \partial t = \sum_e I_{ne} m_e \mu_e^{-1} x_e, \quad (3)$$

where V_n is the volume assigned to the node; ρ_n represents the mass density at the node; t denotes time; the sum applies to all edges adjacent to the node; m_e is the mass flow in an edge, considered positive if the direction of flow coincides with the direction of the edge, and negative otherwise; $\mu_{n/e}$ is the molar mass assigned to both the node and the edge; $x_{n/e}$ are the mole fractions of the components that make up the fluid.

Physically, the above equations describe various conservation laws. In particular, (1) is the dynamic Kirchhoff equation and describes the conservation of mass. Here, $V_n \rho_n$ on the left side, with V_n representing a time-independent volume, describes the mass of fluid in the node. The sum on the right side accounts for the mass flow into the node, minus the flow out. Equation (2) describes the conservation of the total molar amount of a fluid, where $V_n \rho_n \mu_n^{-1}$ represents the number of moles in a node, and the sum on the right side is the total molar flow in the node. Finally, (3) describes the molar conservation for each component, $V_n \rho_n \mu_n^{-1} x_n$ represents the number of moles of a given component in a node, and the sum is the molar flow of that component. Equations (1) and (2) are valid in the absence of chemical reactions between the components of the fluid.

The x-vector may also include other quantities to which linear molar mixing applies, such as the molar heat value H_m, and linear approximations (T_c, P_c) used in certain equations of state for critical temperature and critical pressure, among others. Alternatively, such quantities can be calculated in post-processing as a linear combination over the molar composition. Explicit inclusion in the mixing equation allows these quantities to be calculated even when the determination of molar composition is disabled.

The conservation equations of type (1)–(3) are standard, can be found in a textbook, e.g., eq.(4.1) in [25]. Now we will rewrite them in a more convenient form, resolved with respect to derivatives:

$$V_n \rho_n \partial \mu_n^{-1} / \partial t = \sum\nolimits'_e I_{ne} m_e (\mu_e^{-1} - \mu_n^{-1}), \quad (4)$$

$$V_n \rho_n \mu_n^{-1} \partial x_n / \partial t = \sum\nolimits'_e I_{ne} m_e \mu_e^{-1} (x_e - x_n), \quad (5)$$

$$\sum\nolimits'_e = \sum\nolimits_{e, I_{ne} m_e > 0}, \quad (6)$$

where the sum is taken over the flows incoming to the node. To prove it, it is necessary to perform the differentiation in (2) and take into account (1), which will result in (4), in which the sums are taken over all flows, incoming and outgoing. Further, if one takes into account that μ_e^{-1} for an outgoing flow is equal to μ_n^{-1} at a node, the sum can be reduced to the incoming flows. The proof for (5) is similar. The condition of equality of mixed quantities in the node and in the outgoing flow can also be used to reduce the total number of variables. Namely, one can completely eliminate the variables in the edge e, replacing them with the values in the upstream node n', $\mu_e^{-1} \to \mu_{n'}^{-1}$, $x_e \to x_{n'}$. When time derivatives are set to zero, these equations are reduced to stationary formula (see eq.(13) in [3]).

Boundary conditions: $\mu = \mu_{set}$, $x = x_{set}$ are fixed to the specified values in the network entry nodes. The system of (4)–(6) and boundary conditions is closed. Its stationary part on the right side of the equations is non-degenerate if all nodes are connected to at least one entry node in the upstream direction. A complete dynamical system can be non-degenerate even if this rule is violated, for example if all flows are zero. In this case, the dynamic term ensures the preservation of the transported quantities, keeping them at the starting values.

Startup algorithm: at entry nodes, the transported values are initialized to set values to satisfy the boundary conditions. In all other nodes, values are initialized to default values, which are either specified by the user or averaged over all set values. As a part of the general procedure [2], the initial flows are set to zero and all fluid composition-dependent quantities, such as density ρ, are calculated from the appropriate equations of state. This procedure provides a smooth startup, with all equations initially satisfied. Then, fluid starts to propagate from entries to the neighbor nodes with growing massflow, replacing default values with current ones.

Scaling factors: according to the general procedure [2], all equations are scaled to cover the range of 100 units when variables are changed in their physical domain. Such factors can influence convergence of the solver and should be selected carefully.

V_n-definition: in accordance with the discretization scheme formulated in [2], each pipe contributes half of its volume to the end nodes, and all other elements contribute a nominally specified volume V_0.

Linearity of the system: with known m-flows, the μ^{-1}-subsystem (4) is linear; also, for known m and μ^{-1}, the x-subsystem (5) is linear. This property is convenient for controlling convergence, since each linear subsystem in the non-degenerate case is solvable in one iteration. The following algorithm is used to integrate the equations.

Algorithm (simulation workflow):

```
init;
repeat{ mumix; xmix; Tmix; PM; t+=dt; }
```

Here, `init` represents the initialization of all variables according to the startup algorithm described above. `mumix` is the solution of the μ^{-1}-subsystem, `xmix` is the solution

of the x-subsystem, `Tmix` is the solution of the temperature subsystem formulated below, and `PM` is the solution of the pressure-massflow subsystem as formulated in [2]. As in [2], our primary goal is to determine stationary solutions of the system by integrating with as large steps as possible until stationarity is achieved. The most stable method suitable for this purpose is time discretization of the implicit Euler type: $\partial v/\partial t \to (v - v_{prev})/dt$, for all dynamic variables v, where v_{prev} is the value from the previous step, dt is the integration step. For a detailed study of dynamic processes, more sophisticated finite-difference schemes [26] [27] can be used.

B. Temperature modeling

The starting point is the law of conservation of energy for open systems (see, for example, eq.(4.14) in [25]):

$$V_n \partial(\rho_n \mu_n^{-1} U_n)/\partial t = \textstyle\sum_e I_{ne} m_e \mu_e^{-1} H_e, \tag{7}$$

where U is the molar internal energy, $H = U + P\mu/\rho$ is the molar enthalpy, and P is the pressure. The equation is similar to the conditions of molar mixing in (3). The difference is that the derivative of the nodal internal energy is on the left side, and the total enthalpy flow in the node is on the right side. Physically, with each flow, internal energy is introduced into the node, as well as the work of the fluid against the pressure in the node. This work can be combined with internal energy, giving enthalpy on the right side of the equation. On the left side, under the derivative, there is still nodal internal energy. In general case, other terms can be present in the conservation law, vanishing for simple mixing in the node. In particular, no additional work is performed in the node, and due to the assumed absolute thermal insulation of the node, heat transfer becomes zero. Possible processes with additional work and heat transfer are assigned to special edge elements and are described below.

We rewrite the equation (7) as follows:

$$\begin{aligned} V_n \rho_n \mu_n^{-1} \partial H_n/\partial t - V_n \partial P_n/\partial t = \\ = \textstyle\sum_e' I_{ne} m_e \mu_e^{-1} (H_e - H_n), \end{aligned} \tag{8}$$

the derivation is similar to (5), also here the nodal internal energy is re-expressed in terms of enthalpy and pressure in the node.

Boundary conditions: $H = H_{set}$, enthalpy is fixed to the specified value in entry nodes. Alternatively, one can use the condition $T = T_{set}$, which fixes the temperature at the entry nodes.

In addition, according to eq.(4.14) in [25], gravitational and kinetic terms can be added to the internal energy and enthalpy: $H \to H + \mu g h + \mu v^2/2$, where g is the acceleration of free fall, h is the height, and v is the speed of translational motion of the fluid. To calculate the kinetic term, one needs to know the diameter, which is not available for all types of elements. For example, a compressor is a very complex structure to be described by a single diameter. Also, at nodes where many edges join, complex internal motion occurs, which does not coincide with the simple translational motion described by a kinetic term with a single diameter. On the other hand, for the transport of gases, the kinetic term is usually significantly less than the internal energy, for translational velocities significantly lower than the speed of sound. In our simulation, we made it possible to optionally turn off the kinetic term in the temperature equations.

In (8), H_n represents the nodal value, and H_e represents the edge downstream value. The difference from x-mixing is that here the edge downstream value in the general case cannot be replaced by the upstream nodal value, since there are elements that change the enthalpy value. The system cannot be reduced to a purely nodal one; in addition, the system also includes the temperature T of the fluid.

HT-constraint:

$$H = H_{mod}(P, T, x), \tag{9}$$

$$H = H_{mod}(P, T_{prev}, x) + c_p(T - T_{prev}), \tag{10}$$

where H_{mod} is the thermodynamic model for enthalpy, $c_p = \partial H_{mod}/\partial T$ is the molar heat capacity calculated at point (P, T_{prev}, x). The equations (9)–(10) and (H, T) variables are introduced per node and edge.

The first equation relates enthalpy and temperature according to the thermodynamic model used. We use GERG2008 [15]–[17] as a concrete implementation of such relation. For software-technical reasons, it cannot be used directly; its call once per internal iteration produces too many total calls of GERG2008 module, resulting in significant slowdown. In addition, the equation is nonlinear, violating the desired linearity property of the Tmix subsystem. The second equation is a linearization of the first, it can be used in internal iterations, with a less frequent update of the coefficients. When using the workflow formulated above, (m, P, ρ) in all mix phases are considered as fixed parameters, updated in PM-phase. For H_{mod} and c_p, updates occur immediately before the start of the Tmix phase. In addition, to increase stability, the temperature is clamped to a given range, by default set to $[223.15, 423.15]K$.

Default element equation:

$$H_e = m_e > 0 ? H_{n1} : H_{n2} \tag{11}$$

formulates isenthalpic process [25], where the edge enthalpy is taken from the upstream node, similar to x-mixing. In this and further equations, the edge e goes from node n_1 to node n_2, conditions are written in C-notation: *x?y:z = if(x) then y; else z*. This model is applied to the most of element types, in particular, to valves, regulators, resistors and shortcuts; while the exceptional types are listed below.

Pipe equation:

$$\begin{aligned} (m_e > 0 ? (H_{n1} - H_e)\mu_{n1}^{-1} : (H_{n2} - H_e)\mu_{n2}^{-1}) |m_e| = \\ = \pi D L c_{ht} (T_e - T_{soil}), \end{aligned} \tag{12}$$

the change of enthalpy over the pipe is equal to a heat exchange with the soil, eq.(33.3) in [28]. Here T_{soil} is soil temperature, D is pipe diameter, L is pipe length, and c_{ht} is

heat transfer coefficient. The pipe should have sufficiently fine subdivision to model the heat exchange appropriately.

Compressor equation:

$$m_e > 0?(T_e - T_{n1}((|P_{n2}/P_{n1}|^{(\kappa-1)/\kappa} - 1)/\eta + \\ +1)z_{n1}/z_e) : (H_e - H_{n2}) = 0, \quad (13)$$

for positive flow, the change of temperature is described by eq.(38.51) in [28], or a similar formula (eq.(13-31)) without z-correction from [29]; otherwise, isenthalpic process is used. Here κ is isentropic exponent, η is efficiency, z is compressibility factor. This basic model is designed for gas transport, while for liquids, e.g., CO_2 pumps, customer-specific models can be used.

Coolers and heaters:

$$m_e > 0?(A_{set} > 0?(T_e - T_{set}) : (H_e - H_{n1})) \\ : (H_e - H_{n2}) = 0, \quad (14)$$

at the simplest modeling level, we implement these elements by clamp formulas: $T_e = \min(T_{n1}, T_{set})$ for coolers and $T_e = \max(T_{n1}, T_{set})$ for heaters. These formulas are piecewise-linear. Their linearization leads to the common formula above and the active set flag described by the following algorithm.

Algorithm (active set):

```
cooler:
  if(Aset==1&&He>Hn1) then Aset=0
  if(Aset==0&&Te>Tset) then Aset=1
heater:
  if(Aset==1&&He<Hn1) then Aset=0
  if(Aset==0&&Te<Tset) then Aset=1
```

Here $A_{set} = 1$ corresponds to an active mode, $A_{set} = 0$ to a standby mode. The algorithm is applied after Tmix-phase, its convergence is tracked.

III. Numerical Experiments

We performed a series of simulations on networks of different complexity levels to study in detail the effects of flow mixing, integration stability, and iteration convergence.

N1 network: the network shown in Figure 1 contains 100 nodes, 111 edges and is used for numerical experiments with the transport of natural gas and hydrogen. Detailed settings of supplies in the considered scenario are presented in Table I. Selected time discretization is $dt = 3 \cdot 10^4 s$, $nsteps = 100$. The network has a simple Y-shaped topology, with two supply nodes n99_gm and n56_gm, as well as a mixing node n89, where the flows from the supplies come together, and the rest of the network, ending with the most distant exit node n76.

Figure 2a shows the evolution of inverse molar mass. Figure 2b presents molar heat value, and Figure 2c demonstrates molar fraction of CH_4, representative for chemical composition in the considered test scenario. In all these plots, the values in supply nodes n99_gm and n56_gm are kept constant at set values. In stationary solution, the simple topology of the network leads to a single mixed state, formed in node n89 and propagated downstream to the rest of the network.

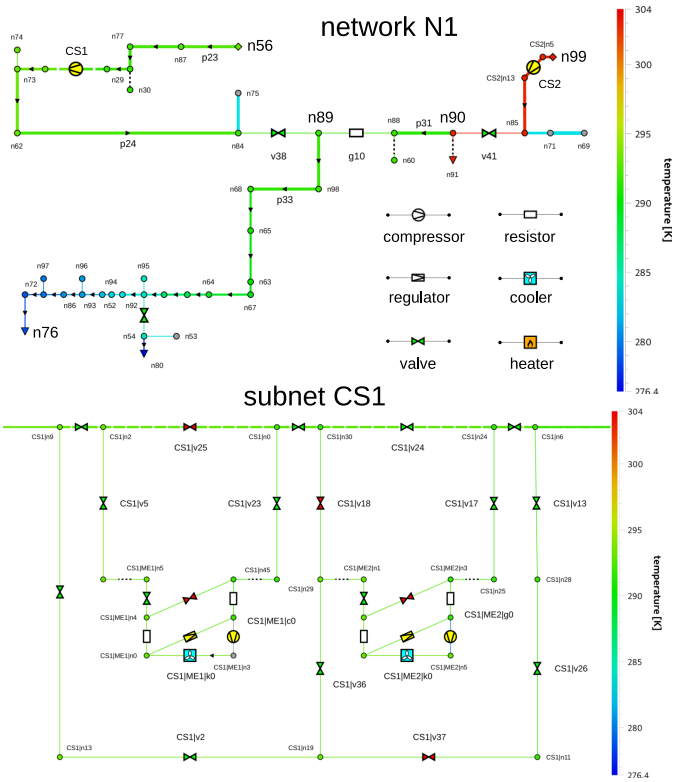

Figure 1. Test network N1.

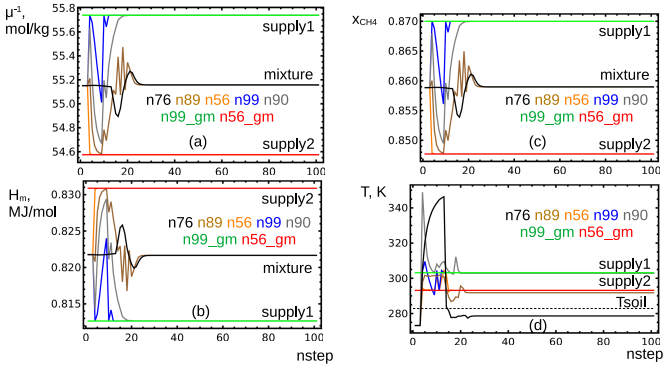

Figure 2. Simulation results (see text for details).

In the evolution, the values in all nodes tend either to supply values or to this mixed state. Interestingly, in the startup of the evolution, the curves perform several large oscillations between the boundary states, before they relax at the stationary state. This happens due to a complex distribution of flows at the startup phase.

Note that the graphs Figure 2a and Figure 2c have an identical shape, and Figure 2b has the same shape vertically reflected. This happens because there are only two supplies in the network, and the default composition is a linear combination of them. As a result, the trajectory of the system in x-space is limited to a 1-dimensional subspace. Graphs

TABLE I
SUPPLY SETTINGS IN VARIOUS SCENARIOS

scenario	entry	composition	temperature
N1 nat.gas	n99_gm	87% CH_4, 1% C_2H_6, 1% C_3H_8, 1% CO_2, 10% N_2	$303.15K$
N1 nat.gas	n56_gm	85% CH_4, 3% C_2H_6, 1% C_3H_8, 1% CO_2, 10% N_2	$293.15K$
dyn-pipe H_2	n0000	95% H_2, 5% N_2	$313.15K$
dyn-pipe CO_2	n0000	95% CO_2, 3% N_2, 2% O_2	$313.15K$

TABLE II
TESTING VARIOUS IMPLEMENTATIONS OF HEATERS ON N85 NETWORKS SET

implementation of heaters	num. of divergent cases
disabled	3
local	0
nonlocal	85
joined	2

Figure 2a-c are projections of this trajectory to different directions and therefore have the same shape.

Figure 2d shows temperature dependence in selected nodes. During startup evolution, strong heating occurs due to the inverse Joule-Thomson (JT) effect and the influence of the $\partial P/\partial t$-term in (8). With further evolution, the temperature in nodes close to supplies tends to the corresponding constant temperature values of the incoming fluid. In more detail, in the considered scenario, after each supply there is a compressor station, the outlet temperature of which is regulated by a cooler. The outlet temperature of the cooler is set to the same value as that of the corresponding supply. The temperature in network nodes remote from the supply tends to a constant value, slightly below $T_{soil} = 283.15K$, due to the influence of the JT-effect.

N85 networks set: contains 85 realistic natural gas networks, obtained for benchmarking from our industrial partner. The networks are highly resolved, containing up to 4 thousands of nodes each. We used these networks for numerical experiments testing the stability of simulation with a different implementation of heaters. Unlike coolers, which control their own output temperature, heaters must control the temperature in an adjacent element, the regulator. In dynamic formulation of the problem, especially at low flows, heaters do not have time to regulate their temperature in order to constantly ensure the set temperature values in the regulator. This leads to divergences. We have tested several options for implementation of heaters, shown in Table II. For disabled heaters, 3 scenarios out of 85 are divergent. For the most stable implementation option, when heaters control their own local temperature, all scenarios are convergent. If the heaters try to control the temperature nonlocally, in the attached regulators, all scenarios diverge, making such implementation impossible. For our selected option, the heaters are joined with regulators, the unified element controls its own output temperature, 2 scenarios out of 85 are divergent, slightly better than the complete disabling of the heaters. On average, simulating one converging scenario from the N85 set takes about one minute on a 2.6 GHz CPU 16 GB RAM computer.

Hydrogen and carbon dioxide pipelines: this is one of our standard test cases, $L = 150km$ $D = 0.5m$ horizontally laid pipeline, transporting gaseous H_2 or CO_2 in liquid or supercritical phase. The case supports variable spatial discretization, for the considered scenario selected to $nsubdiv = 50$. Time discretization is the same as for N1 network. Supply setting is presented in Table I. The considered scenario has a single fluid composition and is used mainly for testing of the temperature modeling. The dynamic simulation starts from $T_{soil} = 283.15K$ and a different $T_{set} = 313.15K$ at the pipeline entry. The simulation converges to stationary solution with nearly exponential fall of temperature from T_{set} to T_{soil}. For CO_2, an observed stronger deviation from the exponent is due to JT-effect and the nonlinear enthalpy model.

Convergence of iterations: in our implementation, we use the globally convergent Newton's solver with Armijo line search rule [30], applied at every time step. For linear problems, it just forwards the solution to the underlying sparse linear solver, that for non-degenerate problems converges in 1 iteration. Due to proper initialization, at the first time step all phases converge in 0 iteration, just keeping the starting values. This provides a good method to test that all variables are correctly initialized. At the second time step, all mix phases also converge in 0 iteration, while in the last PM phase the network filling begins, and PM phase starts to increase its iteration number. For N1 network and H_2/CO_2 pipe scenarios, all mix phases are solved in 1 iteration on intermediate timesteps, as it should be for non-degenerate linear systems; and in 0 iteration at the last timesteps, due to convergence to stationary solution. For large N85 networks, Tmix phase can have intermediately 2-3 iterations, indicating the remaining degeneracy or the disbalance of scaling factors in Tmix system. This effect is planned to be studied in more details, with the application of principal component analysis [31].

The numerical experiments performed show that the purpose of this work has been fully achieved, the modeling has been extended to include mixing flows and is working for scenarios of varying complexity. The modeling in our system is presented in open text form, as a list of variables and equations, which both we and the users can freely modify. This distinguishes us from the existing solutions, in which the modeling is usually hardcoded within the system. We also provide numerical stability of the modeling and the solution algorithms, which allows us to solve large realistic scenarios in fluid transport simulation.

IV. CONCLUSION AND FUTURE WORK

This paper considered the modeling of mixing flows in dynamic simulation of pipeline fluid transport. Mixed characteristics include molar mass, heat value, chemical composition and temperature of the transported fluids. In the absence of

chemical reactions, the modeling is based on the universal conservation laws for molar flows and total energy. The modeling leads to a system of differential algebraic equations, including linear molar mixing formulas, nonlinear temperature-energy relationships, and piecewise-linear element equations for coolers and heaters. In our approach, for nonlinear relations, linearization is carried out in the vicinity of the previous integration step, piecewise-linear relations are reduced to linear ones using the active set method. The resulting sequence of linear systems is solved by a sparse linear solver, typically in one iteration per integration step. The functionality and stability of the developed approach have been tested in a number of realistic network scenarios.

Numerical experiments on the moderate size N1 network allow us to follow the mixing processes in detail, including the evolution of molar mass, heat value, chemical composition, and temperature. Experiments on the N85 set of large-scale networks demonstrate the stability of the developed methods and its sensitivity to such details as nonlocality of equations used in the implementation of heaters. Hydrogen and carbon dioxide pipeline scenarios are used for testing the temperature modeling and the convergence of simulation. Due to the linearity of the mixing equations, their solution is typically carried out in one iteration, representing a minor overhead to solving the main system of nonlinear equations describing the distribution of pressures and flows over the fluid transport networks.

Our future plans include studying the balance of scaling factors in the temperature mixing system for large-scale networks and applying the developed methods to more complex hydrogen and carbon dioxide scenarios.

ACKNOWLEDGMENTS

The work has been partly supported by Fraunhofer research cluster CINES and partly by BMBF project TransHyDE, grant no. 03HY201M. We also acknowledge support from Open Grid Europe GmbH in the development and testing of the software.

REFERENCES

[1] T. Clees, I. Nikitin, and L. Nikitina, “Making network solvers globally convergent”, Advances in Intelligent Systems and Computing, vol. 676, 2018, pp. 140-153.

[2] M. Anvari et al., “Stability of dynamic fluid transport simulations”, J. Phys.: Conf. Ser., vol. 2701, 2024, 012009.

[3] A. Baldin et al., “On advanced modeling of compressors and weighted mix iteration for simulation of gas transport networks”, Lecture Notes in Networks and Systems, vol. 601, 2023, pp. 138-152.

[4] T. Clees et al., “MYNTS: Multi-physics network simulator”, in Proc. of SIMULTECH 2016, International Conference on Simulation and Modeling Methodologies, Technologies and Applications, pp. 179-186, SciTePress, 2016.

[5] A. Baldin, T. Clees, B. Klaassen, I. Nikitin, and L. Nikitina, “Topological reduction of stationary network problems: example of gas transport”, International Journal On Advances in Systems and Measurements, vol. 13, 2020, pp. 83-93.

[6] T. Clees et al., “Efficient method for simulation of long-distance gas transport networks with large amounts of hydrogen injection”, Energy Conversion and Management, vol. 234, 2021, 113984.

[7] M. Anvari et al., “Simulation of pipeline transport of carbon dioxide with impurities”, in Proc. of INFOCOMP 2023, the 13th International Conference on Advanced Communications and Computation, pp. 1-6, IARIA, 2023.

[8] A. Baldin et al., “Universal translation algorithm for formulation of transport network problems”, in Proc. SIMULTECH 2018, International Conference on Simulation and Modeling Methodologies, Technologies and Applications, vol. 1, pp. 315-322.

[9] J. Nikuradse, “Laws of flow in rough pipes”, NACA Technical Memorandum 1292, Washington, 1950.

[10] P. Hofer, “Error evaluation in calculation of pipelines”, GWF-Gas/Erdgas, vol. 114, no. 3, 1973, pp. 113-119 (in German).

[11] J. Saleh, ed., Fluid Flow Handbook, McGraw-Hill 2002.

[12] D.-Y. Peng and D.P. Robinson, “A new two-constant equation of state”, Ind. Eng. Chem. Fundam, vol. 15, 1976, pp. 59-64.

[13] G. Soave, “Equilibrium constants from a modified Redlich-Kwong equation of state”, Chemical Engineering Science, vol. 27, 1972, pp. 1197-1203.

[14] ISO 12213-2: Natural gas – calculation of compression factor, International Organization for Standardization, 2020.

[15] ISO 20765-2: Natural gas – Calculation of thermodynamic properties – Part 2: Single-phase properties (gas, liquid, and dense fluid) for extended ranges of application, International Organization for Standardization, 2020.

[16] O. Kunz and W. Wagner, “The GERG-2008 wide-range equation of state for natural gases and other mixtures: An expansion of GERG-2004”, J. Chem. Eng. Data, vol. 57, 2012, pp. 3032-3091.

[17] W. Wagner, Description of the Software Package for the Calculation of Thermodynamic Properties from the GERG-2008 Wide-Range Equation of State for Natural Gases and Similar Mixtures, Ruhr-Universität Bochum, 2022.

[18] E. Egger and J. Giesselmann, “Stability and asymptotic analysis for instationary gas transport via relative energy estimates”, Numerische Mathematik, Springer 2023.

[19] J. K. van Deen and S. R. Reintsema, “Modelling of high-pressure gas transmission lines”, Appl. Math. Modelling, vol. 7, 1983, pp. 268-273.

[20] C.-Y. Chang, S.-H. Wang, Y.-C. Huang, and C.-L. Chen, “Transient response analysis of high pressure steam distribution networks in a refinery”, in Proceedings of the 6th International Symposium on Advanced Control of Industrial Processes (AdCONIP), May 28-31, 2017, Taipei, Taiwan, pp. 418-423, IEEE 2017.

[21] T.-P. Azevedo-Perdicoúlis, F. Perestrelo, and R. Almeida, “A note on convergence of finite differences schemata for gas network simulation”, in Proceedings of the 22nd International Conference on Process Control, June 11-14, 2019, Štrbské Pleso, Slovakia, pp. 274-279, IEEE 2019.

[22] M. Chaczykowski and A. J. Osiadacz, “Dynamic simulation of pipelines containing dense phase/supercritical CO2-rich mixtures for carbon capture and storage”, International Journal of Greenhouse Gas Control, vol. 9, 2012, pp. 446-456.

[23] P. Aursand, M. Hammer, S. T. Munkejord, and Ø. Wilhelmsen, “Pipeline transport of CO2 mixtures: models for transient simulation”, International Journal of Greenhouse Gas Control, vol. 15, 2013, pp. 174-185.

[24] S. T. McCoy and E. S. Rubin, “An engineering-economic model of pipeline transport of CO2 with application to carbon capture and storage”, International Journal of Greenhouse Gas Control, vol. 2, 2008, pp. 219-229.

[25] M. J. Moran and H. N. Shapiro, Fundamentals of Engineering Thermodynamics, John Wiley and Sons, 2006.

[26] C. Himpe, S. Grundel, and P. Benner, “Next-gen gas network simulation”, in: Progress in Industrial Mathematics at ECMI 2021, pp. 107-113, Springer 2022.

[27] C. Himpe, S. Grundel, and P. Benner, “Model order reduction for gas and energy networks”, J. Math. Industry, vol. 11:13, 2021, pp. 1-46.

[28] J. Mischner, H.-G. Fasold, and K. Kadner, gas2energy.net, Systemplanung in der Gasversorgung, Gaswirtschaftliche Grundlagen, Oldenbourg Industrieverlag GmbH, 2011 (in German).

[29] GPSA Engineering Data Book, 14th Edition, Gas Processors Suppliers Association, 2016.

[30] C. T. Kelley, Iterative Methods for Linear and Nonlinear Equations, SIAM, 1995.

[31] A. Baldin et al., “Principal component analysis in gas transport simulation”, in Proc. of SIMULTECH 2022, International Conference on Simulation and Modeling Methodologies, Technologies and Applications, pp. 178-185, SciTePress, 2022.