Erik Esche, Saskia Bublitz, Gregor Tolksdorf, Jens-Uwe Repke

Automatic Decomposition of Nonlinear Equation Systems for Improved Initialization and Solution of Chemical Engineering Process Models

Esche, E., Bublitz, S., Tolksdorf, G., & Repke, J.-U. (2018). Automatic Decomposition of Nonlinear Equation Systems for Improved Initialization and Solution of Chemical Engineering Process Models. In 13th International Symposium on Process Systems Engineering (PSE 2018) (Computer Aided Chemical Engineering ; 44). pp. 1387–1392. https://doi.org/10.1016/b978-0-444-64241-7.50226-3

Mario R. Eden, Marianthi Ierapetritou and Gavin P. Towler (Editors) Proceedings of the 13th International Symposium on Process Systems Engineering – PSE 2018
July 1-5, 2018, San Diego, California, USA

Automatic Decomposition of Nonlinear Equation Systems for Improved Initialization and Solution of Chemical Engineering Process Models

Erik Esche[a*], Saskia Bublitz[a], Gregor Tolksdorf[a], Jens-Uwe Repke[a]

[a]*Process Dynamics and Operations Group, Technische Universität Berlin, Sekr. KWT 9, Str. des 17. Juni 135, D-10623 Berlin, Germany*
erik.esche@tu-berlin.de

Abstract

The initialization of process models for chemical engineering systems is often a challenge. While dedicated strategies exist for state of the art column models, novel models need to be initialized manually. In this contribution, we present a first application of these decomposition methods for supporting the initialization and solution of equilibrium- and non-equilibrium-type column models for distillation and compare them to well-known sequential solution methods for each. We show that these methods can supply viable solution procedures for nonlinear systems and are especially helpful in deriving new initialization techniques for novel process equipment.

Keywords: nonlinear system, initialization scheme, distillation, absorption.

1. Status Quo - Initialization and Solution of Column Models

State of the art process simulation tools feature dedicated initialization and solution schemes for distillation and absorption columns, and many other unit operations. Famous examples are Amundson and Pontinen's (1958) equation tearing method for equilibrium stage models, the sum-rates method tailored for absorption problems (Friday and Smith, 1964), etc. Of course, these methods have seen many evolutions over time and by now the inside-out methods (Boston and Sullivan, 1974) and their descendants are dominating the field of equilibrium-based column models.

1.1. Inside-Out Method for Column Models

The main advantage of the inside-out method lies in the introduction of an intermediate layer between thermodynamic calculations and the column model itself. This additional layer contains simplifying surrogate models for the phase equilibrium coefficients K and enthalpies h, which are successively retrained with complex thermodynamic models based on updated state variables from the column (J.D. Seader et al., 2011). The inside-out method requires the user to guess a temperature profile for the entire column and to estimate a vapor/gas flow for each stage. Based thereon the column model is split into an outer and an inner iteration loop. For the inner loop the stripping factors are taken as iteration variables ($S_{c,tr} = K_{c,tr} \cdot V_{tr} / L_{tr}$). In the outer loop calculations, the K and h correlations are then retrained and returned to the inner loop. Further details can be found in (R.A. Russel, 1983) and (J. Jelinek, 1988). The inside-out methods have proven so successful that they are nowadays implemented in most modern flowsheet simulators and applied for equilibrium and rate-based calculations of distillation, absorption, and extraction. Nevertheless, there are still many cases, in which solution schemes inside process simulators fail to reliably solve them. One example are complex

absorptions with electrolytes, which are difficult to initialize given the wide range of concentrations of all ionic species (Esche et al., 2014).

1.2. Initialization of Novel Models

For new unit operation models, tailored initialization and solution schemes usually become necessary and need to be implemented by the designer of each unit operation. Naturally, these steady-state models are systems of nonlinear equations and the main reason for not solving all equations simultaneously upfront lies in the ill-conditioning or in how bad a user-supplied initial guess is. In this contribution, we will investigate techniques from linear algebra to derive initialization techniques for nonlinear models. To benchmark these, their performance is compared against the inside-out method for the initialization and solution of column models.

2. Review of Methods in Linear Algebra

In linear algebra, several techniques are known, which by preconditioning, sorting of equations and variables, definition of loops, etc. improve the convergence of linear solution methods. Two famous examples are the Dulmage-Mendelsohn Decomposition (Dulmage and Mendelsohn, 1958) and the Bordered Block Transformation (Erisman et al., 1985).

2.1. Dulmage Mendelsohn / Block Diagonal Decomposition

The Dulmage Mendelsohn decomposition (DM) is a block diagonal decomposition, i.e. the nonzero elements of the Jacobian matrix are aligned along the main diagonal, so far that individually solvable blocks are created, which can be computed in sequence. This is achieved by permuting the original order of both rows and columns of the Jacobian. To this extent the DM decomposition can also be used to identify over- and underdetermined subsystems of a nonlinear equation systems. Further details hereon can be found in (Bublitz et al., 2017). The DM decomposition has two modes. In its “fine decomposition” the number of blocks along the main diagonal is maximized, leading to as small as possible subblocks. The block at the bottom right of the reorganized matrix needs to be evaluated first and then step by step each further block along the diagonal. The upper triangular section of the matrix contains additional occurrences of variables in equations, which are not sorted into any of these blocks. The respective variables are computed in prior blocks and have fixed values for their later occurrence. Several implementations of the DM decomposition are available. For this contribution, the implementation contained in (HSL, 2017) is used.

2.2. Bordered Block Transformation

Whilst the DM decomposition has the goal of a mostly empty lower triangular section of the Jacobian matrix, the bordered block transformation (BBTF) leads to a sparse upper triangular section. The system is split into a dense and a sparse part. The BBTF is a maximum matching algorithm, which primarily identifies variables, which appear in numerous equations. These variables appear as spikes above the main diagonal in the resorted Jacobian matrix. These spikes are subsequently moved to the right side of the Jacobian. Variables with few appearances on the other hand are sorted below the main diagonal. Similarly, equations with few variables are moved to the top left of the matrix. As a result, the BBTF lists variables with a great influence on the whole equation system, albeit from a linear perspective. The variables corresponding to the spikes are possible candidates for tearing the nonlinear system for an outer iterative loop. Therefore, initialization of these variables is essential and needs to be carefully done to allow for successful solution of the whole nonlinear system. All other variables could

theoretically be left uninitialized seeing that they are calculated in sequence by iterative solution of the resorted and teared system starting from the tearing variables. In this contribution, the BBTF is also taken from the Harwell Subroutine Library (HSL, 2017).
As outlined above, the DM decomposition and the BBTF can be used to ease the computational effort for the solution of a nonlinear system, solely based on the structure of the Jacobian matrix. Given that the conditioning of the individual subsystems identified by DM decomposition or the BBTF-teared system with its corresponding subsystems cannot be worse than the conditioning of the entire nonlinear system, it can be expected that the solution strategies outlined above will show a better numeric performance than the equation-oriented solution of the original system. However, the question remains, whether these decompositions lead to better solution strategies compared to dedicated column algorithms, e.g. the inside-out method outlined above.

2.3. Implementation in MOSAICmodeling and Export to MATLAB

To support the application of both decompositions (DM & BBTF), they have been implemented in our department's equation-based modeling and code generation environment MOSAICmodeling (Merchan et al., 2016). Hence, initialization and solution sequences can be automatically generated for all equation systems implemented in MOSAICmodeling, which are tailored to the specific numerical needs of an individual system. To benchmark the solution strategies provided by DM decomposition and BBTF, all four different options have been implemented in MATLAB. As the baseline, the entire NLE model implemented in MOSAICmodeling is exported to MATLAB and solved with a standard Newton-Raphson algorithm. This solution path will subsequently be referred to as NR. The DM and BBTF options are similarly exported to MATLAB and the same Newton method is employed for the solution of all nonlinear subsystems. Finally, for the comparison against the state of the art column algorithms, an inside-out method is manually implemented in MATLAB, which also uses the same Newton method for all implicit, nonlinear solution steps on the inside. This will subsequently be referred to as IO. The resulting calculation sequences for DM, BBTF, and IO are sketched in Fig. 1.

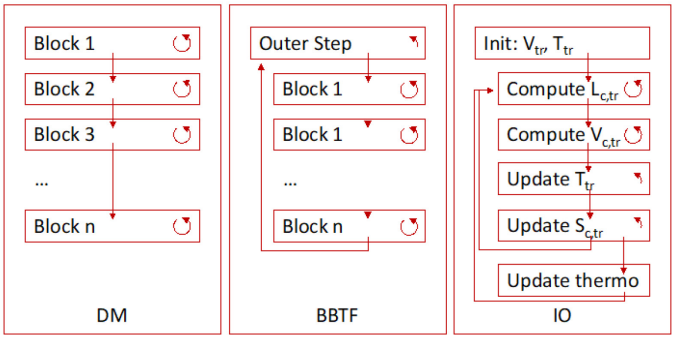

Figure 1: Graphical representation of calculation sequences based on DM and BBTF and for the implemented inside-out method. A full circular arrow implies a complete Newton-Raphson execution, a partial circular arrow depicts a single Newton step with dampening.

3. Case Studies

In this contribution two different applications of column models are investigated to benchmark the aforementioned implementations of DM and BBTF against an IO method and a basic NR algorithm.

3.1. Case Study 1 - Deisobutanizer

The first application is a distillation column for the separation of n- and i-butane. The column model consists of 20 trays using Raoult's law for the phase equilibrium and tray efficiencies based on DIPPR 102. Energy and molar component balances are formulated for each tray with enthalpies and densities based on further correlations from the DIPPR database (127 for enthalpy of vapor phase, 101 for the vapor pressure, 106 for the heat of evaporation, 105 for liquid phase densities). The model is further equipped with a total condenser and a common reboiler. The feed is positioned at stage 13 and consists of 40 % i-butane and 60 % n-butane. The pressure at the top of the column is set to 700 kPa and there is a constant pressure drop of 42 Pa per stage. Fig. 2 shows the incidence matrices of the distillation column model. As expected the DM decomposition (Fig. 2, left) shows a large block, which needs to be solved simultaneously. This block refers to the column itself and the interdependence of all trays given the countercurrent vapor and liquid flows. In any case, there is a sizeable reduction of the size of the largest sub-block compared to the overall system, by roughly 60 %. The BBTF (Fig. 2, right) generates a number of tear variables (spikes at the right above the main diagonal). Close inspection shows that these are temperatures on each tray and the vapor flows of i-butane leaving each tray. This is very similar to the guess values required by the IO methods, which demand temperature profiles and total vapor flow profiles entered by the user for the initialization of the IO procedure.

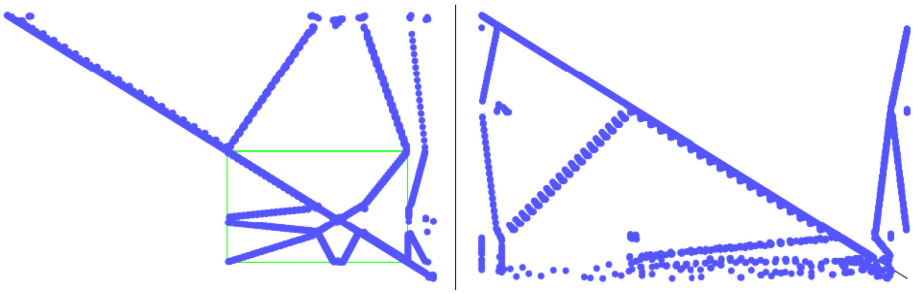

Figure 2: DM decomposition of the distillation column model (left) and BBTF (right).

For the subsequent case study the entire model is first of all manually initialized, i.e. all variables are given adequate guess values and then solved with the NR. The obtained solution is then used as a starting point for all further discussions. All state variables are then perturbed by ± 10, 20, and 30 % of their absolute values and the solution is then attempted with NR, DM, BBTF, and IO methods in turn. The results of this variation are summed up in Tab. 1. Apparently, the pure NR method has great issues with the variation of the initial points, which leads to large numbers of iterations and even failures to converge. The DM is coping better, despite the large block of equations it still has to solve simultaneously. Quite surprisingly the BBTF and the IO method show a highly similar behavior despite the fact that the IO performs a lot finer iterative computation compared to the BBTF with only two iterative layers. However, for both cases initializing temperature and vapor flows at reasonable values proved essential for obtaining any solution at all, while the values of all other state variables can be left essentially uninitialized.

Table 1: Comparison of variation of initial point and its effect on the convergence of algorithmic solution methods of the deisobutanizer column in case study 1. For BBTF and IO the maximum number of iterations in inner (inn.) and outer (out.) iterations is given. For DM the maximum number of iterations in one subblock is given.

Variation	NR	DM	BBTF	IO
-30 %	failure	31 iterations	21 out. / 52 inn.	10 out. / 57 inn.
-20 %	failure	29 iterations	20 out. / 51 inn.	5 out. / 89 inn.
-10 %	4198 iterations	28 iterations	17 out. / 30 inn.	4 out. / 80 inn.
+10 %	4943 iterations	26 iterations	17 out. / 34 inn.	5 out. / 23 inn.
+20 %	failure	34 iterations	17 out. / 44 inn.	7 out. / 101 inn.
+30 %	failure	failure	19 out. / 68 inn.	7 out. / 75 inn.

3.2. Case Study 2 – Absorption Column

The second application is an absorption column for the removal of CO_2 from a flue gas stream using an aqueous solution of monoethanolamine (MEA) as scrubbing liquid. The column model has 35 theoretical equilibrium stages, which employ a tray efficiency correlation fitted to experimental data (Esche et al., 2014). Similarly, to the first example, molar component and energy balances are computed on each tray with correlations for enthalpies, heats of absorption, etc. The flue gas fed to the bottom of the column has a molar fraction of 18 % CO_2 and the liquid feed at the top a residual CO_2 load of 0.1 mol CO_2 per mol MEA. The column is operated at 2 MPa. The gas feed has a temperature of 20 and the liquid feed of 40 °C. Both models describe the same basic tray structure, the results regarding the decompositions are highly similar. For the individual absorption column, a large block will always remain based on DM. A slight change can be seen regarding the BBTF. Given that a ternary system is now in place, the tear variables change slightly: all stage temperatures and stage vapor flows for CO_2 an H_2O are recognized as important initials. Yet again, the model is initialized at a point close to the solution and then variations are carried out for the four different solution methods detailed above. The results are reported in Tab. 2.

Table 2: Comparison of variation of initial point and its effect on the convergence of algorithmic solution methods of the absorption column in case study 2 (same notation as in Tab. 1).

Variation	NR	DM	BBTF	IO
-30 %	107 iterations	21 iterations	15 out. / 60 inn.	3 out. / 77 inn.
-20 %	108 iterations	21 iterations	15 out. / 60 inn.	5 out. / 248 inn.
-10 %	86 iterations	20 iterations	14 out. / 60 inn.	5 out. / 82 inn.
+10 %	235 iterations	19 iterations	14 out. / 60 inn.	5 out. / 101 inn.
+20 %	727 iterations	21 iterations	15 out. / 60 inn.	5 out. / 52 inn.
+30 %	failure	529 iterations	15 out. / 60 inn.	4 out. / 56 inn.

The DM-based solution strategy outperforms the pure NR. Regarding the BBTF and the IO, variation of the initial temperatures for both by even 10% fails right away. Hence, the temperatures are initialized at reasonable values near the solution, while all other tear variables can be varied freely. For both latter techniques, the tear variables become absolutely essential for a successful solution. Of course, this is also a major advantage,

seeing that the set of variables to be initialized can be heavily reduced. It can be concluded that the maximum matching as implemented by the BBTF algorithm shows a reliable performance comparable to the IO method for complex column models. Seeing that the implementation of the solution strategy is quite straight forward based on the equation-oriented form of the nonlinear system, there is a clear advantage regarding novel, custom-made unit operations. The DM-based strategy appears robust and highly versatile for aiding in the solution, although more variables need to be initialized.

4. Conclusions and Outlook

The solution of nonlinear models remains an issue in chemical engineering. While dedicated solution strategies exist for common unit operations, these need to be designed for new models. Chemical engineering models can be reliably decomposed to improve their convergence compared to the monolithic solution with a single Newton-Raphson solver. In this contribution, we have presented an application of the DM decomposition and of the BBTF to find initialization strategies for custom unit operations. To evaluate their performance a comparison has been carried out against the inside-out method. The case studies of a deisobutanizer and of a CO_2 absorption column show the merit of the solution strategy in reducing the required effort on the initialization of all state variables. In future work, these decompositions are extended by new algorithms for maximum matching in nonlinear systems. In addition, the results of the decompositions will be made available inside MOSAICmodeling's user-defined language specification environment. Hence, the users will be able to design their own solution methods using the solution environment of their choice.

References

N.R. Amundson, A.J. Pontinen, 1958, Multicomponent Distillation Calculations on a Large Digital Computer, Ind. Eng. Chem., 50, 730-736

J.F. Boston, S.L. Sullivan, 1974, A new class of solution methods for multicomponent, multistage, separation process, Can. J. Chem. Engr., 52, 52-63

S. Bublitz, E. Esche, G. Tolksdorf, V. Mehrmann, J.-U. Repke, 2017, Analysis and Decomposition for Improved Convergence of Nonlinear Process Models in Chemical Engineering, Chem. Ing. Tech., 89, 11, 1503-1514

A.L. Dulmage, N.S. Mendelsohn, 1958, Coverings of bipartite graphs, Canad. J. Math., 10, 517-534

A.M. Erisman, R.G. Grimes, J.G. Lewis, W.G. Pool, 1985, A structurally stable modification of Hellerman-Rarick's P4 algorithm for reordering unsymmetric sparse matrices, SIAM J. Anal. 22, 369-385

E. Esche, D. Müller, R. Kraus, G. Wozny, 2014, Systematic approaches for model derivation for optimization purposes, Chem. Eng. Sci., 115, 214-224

J.R. Friday, B.d. Smith, 1964, An analysis of the equilibrium stage separations problem – formulation and convergence, AIChE J., 10, 698-707

HSL. A collection of Fortran codes for large scale scientific computation. http://www.hsl.rl.ac.uk, last accessed: 01.12.2017.

J. Jelinek, 1988, The calculation of multistge equilibrium separation problems with various specifications, Comp. Chem. Eng., 12, 2-3, 195-198

V.A. Merchan, E. Esche, S. Fillinger, G. Tolksdorf, G. Wozny, 2016, Computer-Aided Process and Plant Development. A Review of Common Software Tools and Methods and Comparison against an Integrated Collaborative Approach, Chem. Ing. Tech., 88, 50-69

R.A. Russel, 1983, A flexible and reliable method solves single-tower and crude-distillation column problems, Chem. Eng., 17, 52-59

J.D. Seader, E.J. Henley, D.K. Roper, 2011, Separation Process Principles, 3rd Edition, Wiley