

Supplementary Material – Application of generalized (hyper-) dual numbers in equation of state modeling

1 LINEAR ALGEBRA

1.1 Systems of linear equations

In some cases, the evaluation of the model itself requires the solution of a system of nonlinear equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{x}, \mathbf{b} \in \mathbb{R}^N, \quad \mathbf{A} \in \mathbb{R}^{N \times N}. \quad (\text{S1})$$

We are here only concerned with cases in which the real valued equation has a unique solution. To calculate the derivatives of \mathbf{x} , the system of linear equations has to be solved for generalized (hyper-) dual numbers instead. Then, the coefficients of \mathbf{A} , \mathbf{x} and \mathbf{b} are all generalized (hyper-) dual numbers. Since direct solution methods like LU or Cholesky decomposition only rely on basic algebraic operations, they can be implemented unaltered for generalized (hyper-) dual numbers. However, it might not be desired to redo very efficient implementations for linear algebra like LAPACK. Then the solution of a system of linear equations using generalized (hyper-) numbers can be split into multiple systems of real linear equations.

For dual numbers, eq. (S1) can be expanded using the product rule as

$$\mathbf{A}_0\mathbf{x}_0 + (\mathbf{A}_0\mathbf{x}_1 + \mathbf{A}_1\mathbf{x}_0)\varepsilon = \mathbf{b}_0 + \mathbf{b}_1\varepsilon. \quad (\text{S2})$$

By collecting the coefficients of ε , eq. (S2) can be split into two equations

$$\mathbf{A}_0\mathbf{x}_0 = \mathbf{b}_0 \quad \text{and} \quad \mathbf{A}_0\mathbf{x}_1 = \mathbf{b}_1 - \mathbf{A}_1\mathbf{x}_0 \quad (\text{S3})$$

that can be solved sequentially for \mathbf{x}_0 and \mathbf{x}_1 . The same approach can be used for any generalized (hyper-) dual number. For scalar hyper dual numbers for example, the real systems of equations are

$$\mathbf{A}_0\mathbf{x}_0 = \mathbf{b}_0 \quad (\text{S4})$$

$$\mathbf{A}_0\mathbf{x}_1 = \mathbf{b}_1 - \mathbf{A}_1\mathbf{x}_0 \quad (\text{S5})$$

$$\mathbf{A}_0\mathbf{x}_2 = \mathbf{b}_2 - \mathbf{A}_2\mathbf{x}_0 \quad (\text{S6})$$

$$\mathbf{A}_0\mathbf{x}_{12} = \mathbf{b}_{12} - \mathbf{A}_1\mathbf{x}_2 - \mathbf{A}_2\mathbf{x}_1 - \mathbf{A}_{12}\mathbf{x}_0. \quad (\text{S7})$$

The fact that only \mathbf{A}_0 appears as the system matrix of a linear equation can be exploited to speed up the calculation by obtaining an appropriate decomposition and using that to calculate all entries of \mathbf{x} .

1.2 Eigenvalues of symmetric matrices

Real symmetric matrices $\mathbf{A} \in \mathbb{R}^{N \times N}$ are orthogonally diagonalizable. This can be written as

$$\mathbf{A}\mathbf{u}_i = \lambda_i\mathbf{u}_i \quad i = 1 \dots N \quad (\text{S8})$$

with \mathbf{u}_i the eigenvector corresponding to the eigenvalue λ_i and $\mathbf{u}_i^\top \mathbf{u}_j = \delta_{ij}$. Again, a solution method that finds eigenvalues and eigenvectors can be written using generalized (hyper-) dual numbers, but it might be

desirable to use an off the shelf algorithm. Therefore, the eigenvalue problem can be split into individual components. For scalar dual numbers, this results in

$$\mathbf{A}_0 \mathbf{u}_{0i} = \lambda_{0i} \mathbf{u}_{0i} \quad (\text{S9})$$

$$\mathbf{A}_0 \mathbf{u}_{1i} + \mathbf{A}_1 \mathbf{u}_{0i} = \lambda_{1i} \mathbf{u}_{0i} + \lambda_{0i} \mathbf{u}_{1i}. \quad (\text{S10})$$

Equation (S9) is a real valued eigenvalue problem and is solved using an established algorithm from a linear algebra library. Multiplying (S10) with \mathbf{u}_{0i}^\top from the left and using $\mathbf{u}_{0i}^\top \mathbf{u}_{0i} = 1$ leads to

$$\mathbf{u}_{0i}^\top \mathbf{A}_0 \mathbf{u}_{1i} + \mathbf{u}_{0i}^\top \mathbf{A}_1 \mathbf{u}_{0i} = \lambda_{1i} + \lambda_{0i} \mathbf{u}_{0i}^\top \mathbf{u}_{1i}. \quad (\text{S11})$$

Since \mathbf{A}_0 is symmetric, $\mathbf{u}_{0i}^\top \mathbf{A}_0 = (\mathbf{A}_0^\top \mathbf{u}_{0i})^\top = (\mathbf{A}_0 \mathbf{u}_{0i})^\top = \lambda_{0i} \mathbf{u}_{0i}^\top$ and thus the first and last terms in eq. (S11) cancel and the calculation of the derivatives of the eigenvalues simply become

$$\lambda_{1i} = \mathbf{u}_{0i}^\top \mathbf{A}_1 \mathbf{u}_{0i}. \quad (\text{S12})$$

Since the eigenvalues \mathbf{u}_{0i} are orthogonal, they form a basis of \mathbb{R}^N . Thus, the derivatives of the eigenvalues can be written in this basis as

$$\mathbf{u}_{1i} = \sum_k \alpha_{ik} \mathbf{u}_{0k}. \quad (\text{S13})$$

Using this in eq. (S10) simplifies to

$$\sum_k \alpha_{ik} \mathbf{A}_0 \mathbf{u}_{0k} + \mathbf{A}_1 \mathbf{u}_{0i} = \lambda_{1i} \mathbf{u}_{0i} + \sum_k \alpha_{ik} \lambda_{0i} \mathbf{u}_{0k} \quad (\text{S14})$$

and with eq. (S9)

$$\sum_k \alpha_{ik} (\lambda_{0i} - \lambda_{0k}) \mathbf{u}_{0k} = \mathbf{A}_1 \mathbf{u}_{0i} - \lambda_{1i} \mathbf{u}_{0i} \quad (\text{S15})$$

Multiplying with \mathbf{u}_{0j}^\top from the left leads to

$$\sum_k \alpha_{ik} (\lambda_{0i} - \lambda_{0k}) \delta_{jk} = \mathbf{u}_{0j}^\top \mathbf{A}_1 \mathbf{u}_{0i} - \lambda_{1i} \delta_{ij} \quad (\text{S16})$$

and therefore

$$\mathbf{u}_{1i} = \sum_{j \neq i} \frac{\mathbf{u}_{0j}^\top \mathbf{A}_1 \mathbf{u}_{0i}}{\lambda_{0i} - \lambda_{0j}} \mathbf{u}_{0j}. \quad (\text{S17})$$

2 NONLINEAR EQUATIONS

A nonlinear system of equations can be written in the form

$$F(\mathbf{x}(t), t) = 0 \quad (\text{S18})$$

where the goal is to find the variables \mathbf{x} for a given value of t . The equation for the derivatives of \mathbf{x} are obtained by deriving eq. (S18) with respect to t , as

$$F_{\mathbf{x}}(\mathbf{x}(t), t)\mathbf{x}'(t) + F_t(\mathbf{x}(t), t) = 0 \quad (\text{S19})$$

showing that even if F is a nonlinear function, the equation to find the derivatives of its solution is always linear. If F is solved for a dual number $\mathbf{x} = \mathbf{x}_0 + \mathbf{x}_1\varepsilon$ and $t + \varepsilon$, from comparing

$$F(\mathbf{x}, t + \varepsilon) = F(\mathbf{x}_0, t) + (F_{\mathbf{x}}(\mathbf{x}_0, t)\mathbf{x}_1 + F_t(\mathbf{x}_0, t))\varepsilon = 0 \quad (\text{S20})$$

to eqs. (S18) and (S19), it follows that the dual number indeed contains the value and the derivative of the solution.

For a general solution procedure $\mathbf{x}^{k+1} = G(\mathbf{x}^k, t)$ that converges to the solution \mathbf{x}_0^* , it is not guaranteed, that the derivative parts of the solution also converge (Bartholomew-Biggs, 1998). However, since the equations for the derivatives are always linear, a single step of a Newton iteration suffices to find the solution, if the real part is already known from $F(\mathbf{x}_0^*, t) = 0$. This can be demonstrated by applying one Newton step

$$F_{\mathbf{x}}(\mathbf{x}^k, t + \varepsilon)\Delta\mathbf{x}^k = -F(\mathbf{x}^k, t + \varepsilon) \quad (\text{S21})$$

to the starting value $\mathbf{x}_0^0 = \mathbf{x}_0^*$ and $\mathbf{x}_1^0 = 0$. Then from $F(\mathbf{x}_0^0, t) = 0$ follows $\Delta\mathbf{x}_0^0 = 0$. The derivative part of eq. (S21)

$$F_{\mathbf{x}}(\mathbf{x}_0^k, t)\Delta\mathbf{x}_1^k + \left(F_{\mathbf{x}\mathbf{x}}(\mathbf{x}_0^k, t)\mathbf{x}_1^k + F_{\mathbf{x}t}(\mathbf{x}_0^k, t)\right)\Delta\mathbf{x}_0^k = -\left(F_{\mathbf{x}}(\mathbf{x}_0^*, t)\mathbf{x}_1^k + F_t(\mathbf{x}_0^*, t)\right) \quad (\text{S22})$$

simplifies to

$$F_{\mathbf{x}}(\mathbf{x}_0^*, t)\mathbf{x}_1^1 = -F_t(\mathbf{x}_0^*, t). \quad (\text{S23})$$

which is the equation for the derivative of the solution as shown in eq. (S19). The generalization to higher order derivatives is straightforward when taking into account that the calculation of the derivatives only progresses one order for every Newton iteration. Therefore, for second order (partial) derivatives, two Newton steps are required, for third order derivatives three and so on.

REFERENCES

Bartholomew-Biggs, M. (1998). Using forward accumulation for automatic differentiation of implicitly-defined functions. *Computational Optimization and Applications* 9, 65–84. doi:10.1023/A:1018382103801