AIChE WILEY InterScience DISCOVER SOMETHING GREAT

Generalization and Numerical Investigation of QMOM

R. Grosch, H. Briesen, and W. Marquardt
Lehrstuhl für Prozesstechnik, RWTH Aachen University, Turmstraße 46, D-52056 Aachen, Germany

M. Wulkow
Computing in Technology GmbH, Oldenburger Str. 200, D-26180 Rastede, Germany

DOI 10.1002/aic.11041
Published online November 20, 2006 in Wiley InterScience (www.interscience.wiley.com).

A generalized framework is developed for the quadrature method of moments (QMOM), which is a solution method for population balance models. It further evaluates the applicability of this method to industrial suspension crystallization processes. The framework is based on the concepts of generalized moments and coordinate transformations, which have been used already in earlier solution approaches. It is shown how existing approaches to QMOM are derived from the suggested unified framework. Thus, similarities and differences between the various QMOM methods are uncovered. Further, potential error sources involved in the different approaches to QMOM are discussed and assessed by means of a series of test cases. The test cases are selected to be challenging. The error in the QMOM solution is evaluated by comparison to an adaptive, error controlled solution of the population balance. The behavior of a range of different QMOM formulations is analyzed by means of numerical quadrature, dynamic simulation, as well as numerical continuation and bifurcation analysis. As a result of this detailed analysis, some general limitations of the method are detected and guidelines for its application are developed. This article is limited to lumped population balance models with one internal coordinate. © 2006 American Institute of Chemical Engineers *AIChE J,* 53: 207–227, 2007

Keywords: crystallization, population balance, integral approximation, method of moments, Gaussian quadrature, numerical investigation

Introduction

The quadrature method of moments (QMOM)[1–3] is a recent approach to the solution of the population-balance equation (PBE).[4–6] The idea of QMOM is to construct a Gauss-Christoffel quadrature closure by means of the knowledge of the distribution moments. This quadrature scheme is then used to approximate the integrals that appear in the process of transforming the population balance to moment equations. The Gauss-Christoffel quadrature scheme is known to yield the highest-order approximation to integrals with smooth integrands for a given number of quadrature abscissas and weights.[7] It is probably for these promising theoretical properties that a lot of attention has been dedicated to this method in literature. Since the conception of the method by McGraw[1] many different variants have been published independently from each other.

McGraw[1] develops QMOM and validates the method against an analytical solution for a problem describing diffusion controlled growth in an aerosol dynamics context. Wright et al.[8] develop a bivariate extension of the method

Correspondence concerning this article should be addressed to H. Briesen at briesen@lpt.rwth-aachen.de.

for coagulation and sintering problems. Later McGraw and Wright[9] apply the one-dimensional (1-D) method to multicomponent populations, and introduce a formulation of the method called "jacobian matrix transformation". This formulation provides an alternative solution to the classical quadrature problem that arises in QMOM. Yoon and McGraw[10,11] derive a multidimensional variant of QMOM, based on quadrature along main directions of the distribution determined by principle component analysis (PCA).

Simultaneously, Piskunov and Golubev[12] derive a method which they call the "generalized approximation method" for pure aggregation problems. Their discussion starts with the idea of using generalized moments with respect to general functions rather than the standard moments, which are derived from an inner product with respect to the monomials. However, they only elaborate the method for fractional moments, that is, moments with respect to $x^{\frac{k}{2}}$, rather than just x^k. In a later article,[13] these authors show that their method is essentially equivalent to QMOM.

Marchisio et al.[2,14] explore the applicability of QMOM to population-balance problems with nucleation, growth and aggregation and later extend the discussion to aggregation and breakage problems. In the work of this group the method is implemented in a CFD code to facilitate the simultaneous solution of fluid and population dynamics.[15] Marchisio and Fox[16] also develop the "direct" QMOM which extends in a straightforward manner to multidimensional PBEs, and is especially suited for the implementation in CFD codes to facilitate a simultaneous solution of population and fluid-dynamic problems.

Motz et al.[3,17] apply the method to particulate processes in general with a focus on suspension crystallization. In these publications a broad range of kinetics like growth, nucleation, breakage, attrition and agglomeration is covered.

Apart from the earlier cited contributions that developed QMOM, the method has also been extensively evaluated in the literature and error analysis of varying depth have been performed. Almost all authors have validated QMOM against analytical solutions for simple test problems. QMOM has also been compared to other moment methods, high-resolution methods or Monte Carlo methods for harder problems: Piskunov et al.[13] compare their results to computations with a finite element method (FEM) and with a moving grid method. Marchisio et al.[2,14,16,18] have analyzed QMOM by comparison to different sectional methods, Monte Carlo simulations, and with numerical quadrature of analytical solutions for macroscopic population balances, and perform a comparison of the direct QMOM to QMOM for the macroscopic PBE for a range of phenomena. Motz et al.[3,17] compare their results to high-resolution computations with the conservation element solution element method,[19] and with a sectional method. Furthermore, the method has been compared to other moment methods, to computations with a standard FEM, and to an FEM with transformed coordinate system for coagulation and sintering problems.[20] Upadhyay and Ezekoye[21] evaluate the 1-point QMOM for the solution of aerosol processes involving nucleation, growth, and coagulation by comparison to a moment closure which assumes a log-normal shape of the particle distribution. Kostogolou and Karabelas[22] investigate the behavior of QMOM, and of a set of other moment methods for homogeneous breakage problems by comparison to the self-similar analytical solution. Note, that the foregoing review is limited to contributions either developing or evaluating QMOM. Contributions mainly focussing on the application of the method are not included.

In contrast to previous contributions developing QMOM,[1–3,8–17] this work focuses on the development of a generalized framework, which embraces the various approaches. It is shown, how the existing methods are derived from the suggested framework, and their similarities and differences are uncovered. The framework is based on the use of generalized moments, and a coordinate transformation of the PBE. It is presented such that the derivation of a problem-tailored QMOM method can easily be performed. As an example, a new QMOM variant is derived which is based on generalized moments formulated with respect to parameter dependent Laguerre polynomials.

After the presentation of the framework, the various approaches to QMOM are evaluated in a numerical study. In light of the many numerical studies cited earlier, one might ask what further benefit is expected from yet another study. First, most of the earlier analyses have the disadvantage that they compare QMOM results to a solution for which the solution error itself is unknown. For example, Motz assumes that the deviations observed between QMOM, and high-resolution computations originate from the errors in the high-resolution computations, and that QMOM is more accurate.[17] In our case studies, we compare QMOM results to solutions obtained with Parsival.[23] Parsival uses an hp-adaptive discontinuous FEM, where the user can prescribe a relative error bound on the solution that is guaranteed by the algorithm. Therefore, Parsival provides a highly accurate reference solution. Secondly, since the publication of QMOM by McGraw[1] many variants of the method have been published. This article develops a generalized framework for QMOM, thus, uncovering the similarities and differences between the approaches. Third, for the cases studied in the literature, the QMOM solution mostly seems to approach the true solution rapidly with increasing resolution. Only Marchisio et al.[14,18] report other observations. In our work, with typical suspension crystallization models, we rarely found the favorable behavior reported in most previous publications. Therefore, the behavior of QMOM and the error sources involved are analyzed systematically in this publication, and guidelines for the use of QMOM are developed. Only with help of such a detailed analysis the practitioner can reckon whether observed problems could be the result of inherent limitations of the method.

In our numerical study, the behavior of QMOM is evaluated using a set of test problems of varying complexity. The test problems comprise a simple fines dissolution process, a simple crystallizer with fines dissolution, and a complex, industrially relevant crystallization process with size-dependent growth, attrition and fines dissolution. Consequently, the influence of the interaction of the phenomena can be studied in a systematic way. The study uncovers significant general limitations of the method for the approximation of phenomena that act strictly locally on a given size range of the particle distribution. Based on our findings, guidelines for the application of the method to crystallization models are developed which may also carry over to the solution of other types of population balance models.

In the numerical study, the quadrature error as well as the qualitative and quantitative solution behavior is investigated by means of dynamic simulation, numerical continuation, and bifurcation analysis. To the authors' knowledge, the ability of the method to properly represent the stability behavior of the process, which is essential for control applications, is investigated for the first time.

The focus of the present study is on models with one internal coordinate that are relevant in the field of suspension crystallization. Nevertheless, many of the findings directly carry over to the use of QMOM with more general population balance models involving multiple internal coordinates.

Generalized Framework for QMOM

Population balance models for suspension crystallization processes

In this contribution, the application of QMOM to models of well-mixed suspension crystallization processes is addressed. According to the population balance framework discussed in the textbook by Randolph and Larson,[5] the solid phase is assumed to be well described by a *particular dispersive state*, rather than having to consider individual crystals. Hence, it can be represented by a number density function $n(x, t)$, which is a function of a characteristic coordinate x, and time t. In this contribution we assume that the x-coordinate represents the particle size. The rate of change of the particle density n is described by the population balance equation[4,5]

$$\frac{\partial n(x,t)}{\partial t} = -\frac{\partial (G(x,t)n(x,t))}{\partial x} + \sum_{p\in P}[B_p(x,t) - D_p(x,t)] - n(x,t)\frac{1}{V(t)}\frac{dV(t)}{dt}. \quad (1)$$

Here, G is a possibly size-dependent growth function, B_p and D_p are general birth and death terms due to phenomena p. V represents the suspension volume.

In addition to the solid phase, the liquid phase has to be modeled in terms of mass balances. The solute balance has to account for the potential exchange of matter between the solid, and the liquid phases

$$\frac{dm_e(t)}{dt} = F_{\text{in}}(t)\, c_{\text{in e}}(t) - F_{\text{out}}(t) \in_{\text{out}}(t)\, c_e(t) - \sum_{p\in P} S_p(t), \quad (2)$$

where m_e is the mass of solute, F_{in} and F_{out} are the inlet and outlet suspension flow rates, $c_{\text{in,e}}$ and c_{e} are the mass concentrations of solute in the feed and the vessel, ε_{out} is the void fraction in the outlet stream, and the S_p describe the exchange between the solid and the liquid phases. Usually, an analogous mass balance without exchange terms also has to be added for the solvent. Commonly, exchange terms for growth, nucleation and ideal dissolution have to be taken into account in the mass balance of the crystallizing species by

$$S_{\text{gr}}(t) = 3k_v\, \rho_s\, V(t) \int_0^\infty G(x,t)\, n(x,t)\, x^2\, dx, \quad (3)$$

$$S_{\text{nuc}}(t) = k_v\, \rho_s\, V(t) \int_0^\infty B_{\text{nuc}}(x,t)\, x^3\, dx, \quad (4)$$

$$S_{\text{diss}}(t) = k_v\, \rho_s\, F_{\text{diss}}(t) \int_0^\infty h_{\text{diss}}(x,t)\, n(x,t)\, x^3\, dx. \quad (5)$$

respectively. In Eqs. 3–5 k_v is the shape factor, ρ_s is the mass density of the solid phase, B_{nuc} is a nucleation rate, P_{diss} is the suspension flow with which particles are removed, and h_{diss} is a classification function. The term S_{diss} might for instance be used to model the effect of an ideal fines trap. Another coupling between the mass balances, and the population balances is introduced via the void fractions ε_{out}.

Transformation of the size coordinate

Under certain circumstances, it is favorable to transform the characteristic coordinate of the population balance. A frequently encountered example is the transformation of a model from a particle volume to a particle size coordinate. Coordinate transformations in conjunction with QMOM have been successfully applied to solve pure coagulation, as well as combined coagulation and condensation problems in aerosol science.[9,10,12] Note, however, that the idea of using a transformed population balance is not new. The concept has already been used in earlier attempts to solve the PBE.[24] In the Appendix, the relations for a general transformation of the phenomenological terms typically occurring in suspension crystallization models are presented. Therefore, this paragraph only introduces the basic notation.

We use a bijective transformation function T to express the population balance, originally formulated in the x-coordinate, in the new z-coordinate:

$$x = T(z) \quad \text{and} \quad z = T^{-1}(x). \quad (6)$$

Since the number of particles $n(x, t)dx$ in the size range $[x, x + \text{dx}]$ has to be preserved, the expression for the density function $n^{(z)}(z, t)$ in the new coordinate system is given by

$$n^{(z)}(z,t) = n^{(x)}(T(z),t)\frac{dT(z)}{dz}. \quad (7)$$

The superscripts (z) and (x) denote the coordinate system of the density function n. The transformed population balance and the expressions for the transformed kinetics are presented in Appendix A.

Weighting function

The idea to use generalized moments with QMOM is adopted from Piskunov and Golubev,[12] who did, however, not fully exploit this idea.

The standard moments of the number density function are defined as

$$\mu_k(t) = \mu_k^{(x)}(t) = \int_0^\infty n(x,t)\, x^k\, dx = \int_{T^{-1}(0)}^{T^{-1}(\infty)} n^{(z)}(z,t) T(z)^k\, dz. \quad (8)$$

The lower-order standard moments are of direct physical relevance if x has the meaning of a particle size or of a volume coordinate. It can, however, be favorable to compute generalized moments $\mu_k^{(z,\phi)}$ in a transformed space (superscript z) with respect to a general parameter dependent, linearly independent weighting function $\phi_k(z, \mathbf{p})$

$$\mu_k^{(z,\phi)}(t,\mathbf{p}(t)) = \int_{T^{-1}(0)}^{T^{-1}(\infty)} n^{(z)}(z,t)\ \phi_k(z,\mathbf{p}(t))\ dz. \qquad (9)$$

We adopt here the name *weighting functions* for the set of functions $\phi_k(z, \mathbf{p}(t))$ due to the analogy with the weighting (or test) functions used in the method of weighted residuals (MWR).[25] These functions are not to be confused with the weight function used in the terminology of quadrature and orthogonal polynomials.[7] Specific weighting functions have been used already in the earliest attempts to solve the PBE. See for example the work by Singh and Ramkrishna[26] or Chiu and Christofides[27,28] and references, therein, for problem tailored weighting functions. Parameter dependent weighting functions are treated in depth by Deuflhard and Wulkow[29] in the context of MWR.

In the following, let $\mathbf{p} \in R^{N_p}$. In general the parameter vector $\mathbf{p}$ is a function of time. If so, these parameters can be chosen to have a direct physical meaning in terms of the properties of a distribution, such as a mean-particle size. In this setting, the parameters are computed from functions $\tilde{f}_s^{(p)}$ of the standard moments by

$$p_s(t) = \tilde{f}_s^{(p)}(\boldsymbol{\mu}(t)), \quad s = 1 \ldots N_p. \qquad (10)$$

The functions $\tilde{f}_s^{(p)}$ are usually chosen such that the generalized moments of first up to N_p-th order vanish, that is, $\mu_1^{(z,\phi)}(t, \mathbf{p}(t)) = \ldots = \mu_{N_p}^{(z,\phi)}(t, \mathbf{p}(t)) = 0$.[4,29] Consequently, N_p degrees of freedom are shifted from the moment space to the space of time-varying parameters. In fact, the procedure acts like a nonlinear transformation of the solution space.

Quadrature closure

The essence of the quadrature closure is to view the number density $n(z, t)$ as a general-weight function in the sense of classical Gauss-Christoffel quadrature,[7] and to approximate the integrals that appear during the transformation of the population balance to moment equations in terms of the N_q quadrature abscissas and weights

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} n^{(z)}(z,t)\, f(z)\ dz \approx \sum_{i=1}^{N_q} w_i(t)\, f(\zeta_i(t)). \qquad (11)$$

Here, ζ_i and w_i are the quadrature abscissas and weights respectively, and f denotes an arbitrary function. This quadrature scheme is exact if f is a polynomial up to the order $2N_q - 1$ in z. Therefore, the standard moments up to this order are represented exactly by the quadrature scheme. This property is exploited in QMOM because conversely the abscissas and weights can be computed from the knowledge of the moments.[1,30]

In the following we sort the generalized moments and time-varying parameters according to $\xi = [\mu_0^{(z,\phi)}, \mu_{N_{p+1}}^{(z,\phi)}, \ldots, \mu_{2N_q-1}^{(z,\phi)}, p_1, \ldots, p_{N_p}]^T$, where the vanishing moments are replaced by the parameters (see Eq. 10). Further, we comprise the abscissas and weights in one vector $\boldsymbol{\eta} = [\mathrm{w}^T, \zeta^T]^T$.

In the following the time dependence is omitted, when it is clear from the context. From the defining equation for the generalized moments (Eq. 9) one obtains with Eq. 11

$$0 = \mu_k^{(z,\phi)}(\mathbf{p}) - \tilde{g}_k^{(z,\phi)}(\boldsymbol{\eta},\mathbf{p}) = \mu_k^{(z,\phi)}(\mathbf{p}) - \sum_{i=1}^{N_q} w_i\ \phi_k(\zeta_i,\mathbf{p}),$$
$$k = 1, \ldots, 2N_q - 1 \qquad (12)$$

where the map $\tilde{g}_k^{(z,\phi)}(\boldsymbol{\eta}, \mathbf{p})$ from the abscissas, weights, and time-varying parameters to the generalized moments is introduced. The equality in Eq. 12 only holds if $\phi_k(z, \mathbf{p})$ are polynomials up to degree $2N_q - 1$ in z.[7,30] Therefore, the use of polynomial ϕ_k is the preferred choice. This publication is also limited to this choice even though the derived equations hold for the general case unless otherwise stated.

The earlier relations establish a correspondence between the generalized moments in the transformed space, and the abscissas and weights. In most cases, however, QMOM is employed to compute the standard moments. It follows from Eqs. 8 and 11 that one can approximate these by

$$\mu_k \approx \tilde{g}_k^{(z,T)}(\eta) = \sum_{i=1}^{N_q} w_i\ T(\zeta_i)^k\ , \quad k = 0 \ldots 2N_q - 1 \qquad (13)$$

It is clear that this expression is not exact in the sense of the Gauss-Christoffel-quadrature for general choices of T. Nevertheless, it is exact (at least for a number of the standard moments) for certain special choices of T, such as $T(z) = z$, $T(z) = z^{\frac{1}{\alpha}}$ with $\alpha \in N$, and for ϕ_k being polynomials of at most degree $2N_q - 1$.

If time-varying parameters $\mathbf{p}$ are to be considered, it is convenient to formulate the system Eq. 12 in a slightly different manner: the N_p equations of the vanishing generalized moments are replaced by the defining equations for the parameters (Eq. 10). If we work with a transformation for which Eq. 13 is exact, it is further convenient to substitute Eq. 13 into Eq. 10. Then, the following system is obtained.

$$\boldsymbol{\xi} = \mathbf{g}^{(z,\phi)}(\boldsymbol{\eta}). \qquad (14)$$

Recall that $\boldsymbol{\xi}$ comprises all nonvanishing moments and time-varying parameters, and $\boldsymbol{\eta}$ includes the abscissas and weights. Note, that Eq. 14 reduces to Eq. 12 if no time-varying parameters, and no transformation are considered as in the standard QMOM.

There are a number of ways to compute the abscissas and nodes $\boldsymbol{\eta}$ of the Gauss-Christoffel quadrature scheme for given moments and parameters $\boldsymbol{\xi}$. The different approaches can be divided into two groups. The first group solves Eq. 12 or Eq. 14 directly in a brute force approach by means of a nonlinear equation solver. In the literature on Gaussian quadrature it is stated that this approach has a justification in its own right, especially for nonstandard quadrature problems.[31] This approach obviously is very flexible. However, it is often feasible for a small number of abscissas and weights N_q

only. This is usually no limitation for QMOM, because the number of abscissas and weights almost never exceeds $N_q = 10$. For example, most publications on QMOM only work with $1 \leq N_q \leq 4$. Such an approach also has to be used, if non-polynomial ϕ_k were selected.[1,7,32] The second group of solution approaches views the quadrature problem in the light of orthogonal polynomials. The moments contain information on the distribution, which can be exploited in order to set up a three-term recurrence relation for a set of polynomials orthogonal with respect to the particle distribution n. From the coefficients of the three-term recurrence relation a tri-diagonal matrix, the so called *Jacobi* or *terminal* matrix, is setup. The eigenvalues of this matrix are the abscissas of the quadrature scheme, whereas the weights can be determined from its eigenvectors. The approach is known for its high-degree of accuracy and efficiency.

Derivation of moment equations

In order to compute the abscissas and weights of the Gauss-Christoffel quadrature scheme, the moments are needed as functions of time. The general moment equations are derived in the following. These equations hold for an arbitrary choice of linearly independent weighting functions and arbitrary bijective transformations unless otherwise stated.

In analogy to the standard QMOM the derivation starts from the transformed PBE in Eq. A11 by multiplication with the weighting function ϕ_k followed by an integration over the domain.[1] Using the definition of the moments (Eq. 9), and the quadrature approximation of the integrals (Eq. 11) the contributions of the single phenomena are derived term by term.

Holdup

In order to transform the holdup term in Eq. 1, the time derivative of Eq. 9 is considered. After an interchange of differentiation and integration, the product rule is applied to compute the time derivative of the integrand in the righthand side of Eq. 9. Then, a comparison of the resulting terms with the transformed holdup term from Eq. 1 yields

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \frac{\partial n^{(z)}(z,t)}{\partial t} \phi_k(z,\mathbf{p}(t))\, dz = \frac{d}{dt}\mu_k^{(z,\phi)}(t,\mathbf{p}(t)) - \int_{T^{-1}(0)}^{T^{-1}(\infty)} n^{(z)}(z,t) \sum_{s=1}^{N_p} \frac{\partial \phi_k(z,\mathbf{p}(t))}{\partial p_s} \frac{dp_s(t)}{dt}\, dz. \quad (15)$$

The integral on the righthand side can qualitatively be understood as a correction term for the time derivative of the generalized moments due to the presence of time-varying parameters $\mathbf{p}$. The term can be approximated in terms of the quadrature (Eq. 11)

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} n^{(z)}(z) \frac{\partial \phi_k(z,\mathbf{p})}{\partial p_s}\, dz \approx \sum_{i=1}^{N_q} w_i \frac{\partial \phi_k(\zeta_i,\mathbf{p})}{\partial p_s}. \quad (16)$$

Particle growth

In the general case of size-dependent particle growth one obtains by partial integration

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \frac{\partial G^{(z)}(z) n^{(z)}(z)}{\partial z} \phi_k(z,\mathbf{p})\, dz = \left[G^{(z)}(z)\, n^{(z)}(z)\, \phi_k(z,\mathbf{p})\right]_{T^{-1}(0)}^{T^{-1}(\infty)} - \int_{T^{-1}(0)}^{T^{-1}(\infty)} G^{(z)}(z)\, n^{(z)}(z) \frac{\partial \phi_k(z,\mathbf{p})}{\partial z}\, dz. \quad (17)$$

Here, we assume the boundary conditions: $n^{(z)}\ (z = T^{-1}(0)) = 0$ and $n^{(z)}(z \to T^{-1}\ (\infty)) = 0$, which imply that nucleation takes place inside the z-domain and not on the boundary. In this case, the first term on the righthand side vanishes. Again, the integral on the righthand side can be approximated with Eq. 11 to yield

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} G^{(z)}(z)\, n^{(z)}(z) \frac{\partial \phi_k(z,\mathbf{p})}{\partial z}\, dz \approx \sum_{i=1}^{N_q} w_i\, G^{(z)}(\zeta_i) \left.\frac{\partial \phi_k(z,\mathbf{p})}{\partial z}\right|_{z=\zeta_i}. \quad (18)$$

The boundary condition $n^{(z)}(z = T^{-1}(0)) = 0$ does not hold, if small particles dissolve and vanish in an undersaturated solution due to negative growth and crossing of the left boundary of the particle-size domain. In fact, none of the existing moment methods describes such a behavior. However, Motz et al.[3] suggest an ad hoc solution to this problem. In all our computations the growth rate remains positive in the small particle range. Hence, this problem does not occur here.

Nucleation

In the nucleation phenomenon $B_{\text{nuc}}^{(x,t)}(x) = b_{\text{nuc}}(t)\ n_{\text{nuc}}^{(x)}(x)$ particles are formed at a time-dependent rate $b_{\text{nuc}}(t)$ with a given normalized-size distribution $n_{\text{nuc}}^{(x)}(x)$. This phenomenon is transformed to generalized moment equations as follows

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[B_{\text{nuc}}^{(z)}(z)\right] \phi_k(z,\mathbf{p})\, dz = b_{\text{nuc}}(t) \int_{T^{-1}(0)}^{T^{-1}(\infty)} n_{\text{nuc}}^{(z)}(z)\, \phi_k(z,\mathbf{p})\, dz = b_{\text{nuc}}(t)\, \mu_{k,\text{nuc}}^{(z,\phi)}. \quad (19)$$

Here, $\mu_{k,\text{nuc}}^{(z,\phi)}$ are the generalized moments of the distribution of nuclei in the transformed space. A normalization in terms of the standard moments of the distribution of nuclei also holds in the transformed space, according to the meaning of $b_{\text{nuc}}(t)$ as a nucleation rate. For example $\mu_{0,\text{nuc}}^{(z)} = \mu_{0,\text{nuc}}^{(x)} = 1$ has to hold if $b_{\text{nuc}}(t)$ specifies the number of nuclei formed per unit suspension volume and unit time.

Particle removal

The sink term due to removal of particles from the control volume with the suspension flow rate F_{out} and classification

function $h(x)$ is treated as follows

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[-D_{\text{out}}^{(z)}(z)\right] \phi_k(z,\mathbf{p})\, dz = -\int_{T^{-1}(0)}^{T^{-1}(\infty)} F_{\text{out}} h^{(z)}(z)\, n^{(z)}(z)\, \phi_k(z,\mathbf{p})\, dz \approx -F_{\text{out}} \sum_{i=1}^{N_q} w_i\, h^{(z)}(\zeta_i)\, \phi_k(\zeta_i,\mathbf{p}). \quad (20)$$

For an unclassified removal, the sink term can be expressed directly in terms of the generalized moments $\mu_k^{(z,\phi)}$. This is also exploited if particles are added to the control volume, for example, by a feed stream. Complete dissolution of particles (for example, in a fines trap) can also be treated analogously.

Agglomeration

If one starts with the source and sink terms for agglomeration (Eqs. A4–A6) and performs similar calculations as Litster et al.[33] or Motz[17] involving a coordinate transformation of the integral, one obtains

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[B_{\text{agg}}^{(z)}(z') - D_{\text{agg}}^{(z)}(z')\right] \phi_k(z',\mathbf{p})\, dz' = \frac{1}{2}\int_{T^{-1}(0)}^{T^{-1}(\infty)} \int_{T^{-1}(0)}^{T^{-1}(\infty)} [\phi_k(T^{-1}\{(T(z')^3 + (T(z'')^3)^{1/3}\},\mathbf{p}) - \phi_k(z',\mathbf{p}) - \phi_k(z'',\mathbf{p})]\beta^{(z)}(z',z'')n(z')n(z'')\, dz'\, dz'' \quad (21)$$

$$\approx \frac{1}{2}\sum_{i=1}^{N_q}\sum_{j=1}^{N_q} [\phi_k(T^{-1}\{(T(\zeta_i)^3 + T(\zeta_j)^3)^{1/3},\mathbf{p}) - \phi_k(\zeta_i,\mathbf{p}) - \phi_k(\zeta_j,\mathbf{p})]\beta^{(z)}(\zeta_i,\zeta_j)\, w_i\, w_j. \quad (22)$$

Here, $\beta^{(z)}(z',z'')$ is the aggregation frequency in the transformed space. If we would choose $\phi_k = z^k$ and $x = T(z) = z$, then Eq. 21 is just the expression given by Litster et al.[33] The result is obtained for a symmetric agglomeration kernel β under the assumption that $\lim_{z\to\infty} T(z) = \infty$. This means that the formula does not apply if the transformation maps the infinite particle-size domain into a finite domain, as done for example by a tan-shaped transformation. Nevertheless, the formula can be used for the usual transformations as power, exponential, and the like, or for a transition to a particle-volume frame.

Breakage

For breakage or dispersion, the source and sink terms are treated separately. The sink term is transformed to generalized moment equations by

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[-D_{\text{break}}^{(z)}(z)\right] \phi_k(z,\mathbf{p})\, dz = -\int_{T^{-1}(0)}^{T^{-1}(\infty)} d_{\text{break}}^{(z)}(z)\, n^{(z)}(z)\, \phi_k(z,\mathbf{p})\, dz \quad (23)$$

$$\approx -\sum_{i=1}^{N_q} w_i\, d_{\text{break}}^{(z)}(\zeta_i)\, \phi_k(\zeta_i,\mathbf{p}). \quad (24)$$

Here, $d_{\text{break}}^{(z)}(z)$ is the breakage rate in the transformed space, that is, the rate at which particles of size $T(z)$ disappear from the control volume.

In a similar fashion the formation of the fragments, that is, the source term, is treated.

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[B_{\text{break}}^{(z)}(z)\right] \phi_k(z,\mathbf{p})\, dz = \int_{T^{-1}(0)}^{T^{-1}(\infty)} \phi_k(z,\mathbf{p}) \times \int_{T^{-1}(0)}^{T^{-1}(\infty)} N_{\text{break}}\gamma^{(z)}(z,z')\, d_{\text{break}}^{(z)}(z')\, n^{(z)}(z')\, dz'\, dz. \quad (25)$$

Here, $\gamma^{(z)}(z, z')$ is the fragment distribution in the transformed space, and N_{break} denotes the number of fragments formed in each breakage event. From this point, two directions to proceed to the moment equations are found in the literature. Marchisio et al.[14] change the order of integration

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[B_{\text{break}}^{(z)}(z)\right] \phi_k(z,\mathbf{p})\, dz = \int_{T^{-1}(0)}^{T^{-1}(\infty)} N_{\text{break}}\, d_{\text{break}}^{(z)}(z')\, n^{(z)}(z') \times \left(\int_{T^{-1}(0)}^{T^{-1}(\infty)} \gamma^{(z)}(z,z')\phi_k(z,\mathbf{p})\, dz\right) dz'. \quad (26)$$

and perform an analytical preprocessing of the fragment distribution equivalent to

$$\tilde{\gamma}_k^{(z,\phi)}(z') = \int_{T^{-1}(0)}^{T^{-1}(\infty)} \phi_k(z,\mathbf{p})\, \gamma^{(z)}(z,z')\, dz. \quad (27)$$

Then, the following quadrature approximation is obtained

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[B_{\text{break}}^{(z)}(z)\right] \phi_k(z,\mathbf{p})\, dz \approx N_{\text{break}} \sum_{i=1}^{N_q} w_i\, \bar{\gamma}_k^{(z,\phi)}(\zeta_i)\, d_{\text{break}}(\zeta_i) \quad (28)$$

for the source term. Motz et al.[3] suggest a different route. They split up the integral over the fragment distribution into a sum over the N_{break} contributions from the breakage event. If their derivation is followed for the generalized moments in the transformed space the source term can be approximated as

$$\int_{T^{-1}(0)}^{T^{-1}(\infty)} \left[B_{\text{break}}^{(z)}(z)\right] \phi_k(z,\mathbf{p})dz \approx \sum_{j=1}^{N_{\text{break}}}\sum_{i=1}^{N_q} w_i \phi_k(T^{-1}\{T(\zeta_i) - \Delta x_j(T(\zeta_i))\})d_{\text{break}}(\zeta_i). \quad (29)$$

Here, $\Delta x_j(x)$ denotes the difference in size of the original particle, and the particle produced by the breakage event.

Interaction with the liquid phase

The interaction with the liquid phase defined by Eqs. 3–5 is likewise approximated using the identity in Eq. 7 for the transformed density function, and the quadrature approximation in Eq. 11

$$S_{\mathrm{gr}} = 3\, k_v\, \rho_s\, V \int_{T^{-1}(0)}^{T^{-1}(\infty)} G^{(z)}(z)\, n^{(z)}(z)\, T(z)^2\, \frac{dT(z)}{dz} dz \approx 3\, k_v\, \rho_s\, V \sum_{i=1}^{N_q} w_i\, G^{(z)}(\zeta_i)\, T(\zeta_i)^2\, \left.\frac{dT(z)}{dz}\right|_{z=\zeta_i}, \quad (30)$$

$$S_{\mathrm{nuc}} = k_v\, \rho_s V \int_0^{\infty} B_{\mathrm{nuc}}(x)\, x^3 dx = k_v\, \rho_s\, V\, \mu_{3,\mathrm{nuc}}, \quad (31)$$

$$S_{\mathrm{diss}} = k_v\, \rho_s\, F_{\mathrm{diss}} \int_{T^{-1}(0)}^{T^{-1}(\infty)} h_{\mathrm{diss}}^{(z)}(z)\, n^{(z)}(z)\, T(z)^3\, dz \approx k_v\, \rho_s\, F_{\mathrm{diss}} \sum_{i=1}^{N_q} w_i\, h_{\mathrm{diss}}^{(z)}(\zeta_i)\, T(\zeta_i)^3. \quad (32)$$

We have used Eq. A1 in Eq. 30. $\mu_{3,\mathrm{nuc}}$ in Eq. 31 denotes the third-order standard moment of the distribution of nuclei.

Overall structure of the system of equations

The earlier derived Eqs. 15–29 can be cast in the form of an implicit differential equation system

$$\mathbf{M}(\boldsymbol{\xi}, \boldsymbol{\eta}) \frac{d\boldsymbol{\xi}}{dt} = \mathbf{f}^{(z,\phi)}(\boldsymbol{\xi}, \boldsymbol{\eta}). \quad (33)$$

The matrix $\mathbf{M}$ comprises the factors of the time differentials of the generalized moments and the time-varying parameters in Eq. 15. The functions $\mathbf{f}^{(z,\phi)}$ comprise the contributions of the individual phenomena in Eqs. 17–29.

A closed system of equations can be obtained in the following ways. First, a linearly implicit differential algebraic system comprising Eqs. 33 and 14 can be set up to result in

$$\begin{pmatrix} \mathbf{M}(\boldsymbol{\xi}, \boldsymbol{\eta}) & 0 \\ 0 & 0 \end{pmatrix} \frac{d}{dt} \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{pmatrix} = \begin{bmatrix} \mathbf{f}^{(z,\phi)}(\boldsymbol{\xi}, \boldsymbol{\eta}) \\ \boldsymbol{\xi} - \mathbf{g}^{(z,\phi)}(\boldsymbol{\eta}) \end{bmatrix} := \mathbf{F}^{(z,\phi,DAE)}(\boldsymbol{\xi}, \boldsymbol{\eta}). \quad (34)$$

To our knowledge, this DAE system has not yet been studied in the literature. We denote it as the QMOM-DAE formulation.

Second, index reduction by differentiation[34,35] results in the (linearly implicit) system of differential equations

$$\mathbf{M}(\mathbf{g}^{(z,\phi)}(\boldsymbol{\eta}) : \boldsymbol{\eta}) \mathbf{B}(\boldsymbol{\eta}) \frac{d}{dt} \boldsymbol{\eta} = \mathbf{f}^{(z,\phi)}(\mathbf{g}^{(z,\phi)}(\boldsymbol{\eta}), \boldsymbol{\eta}) := \mathbf{F}^{(z,\phi,ODE)}(\boldsymbol{\eta}) \quad (35)$$

denoted as the QMOM-ODE. In the derivation of Eq. 35 we take advantage of the fact that the vector $\boldsymbol{\xi}$ is given explicitly by Eq. 14. The matrix $\mathbf{B}(\boldsymbol{\eta}) = -\frac{\partial \mathbf{g}^{(z,\phi)}(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}}$ refers to the Jacobian of the system in Eq. 14, which should not be confused with the *Jacobi* matrix of the quadrature algorithm. The detailed derivation of the above system is presented by Motz[17] for the special case of standard moments, even though the connection to the index reduction is not established there. From this system, direct QMOM[16] for lumped models is derived after a short calculation, if one replaces the abscissas ζ_i by new variables $\lambda_i := w_i\, \zeta_i$ in the differential terms only. Direct QMOM is further designed as an explicit method, where the integrator only sees an explicit ODE in the new variables, which has the form

$$\frac{d}{dt} \tilde{\boldsymbol{\eta}} = \boldsymbol{\varsigma} \quad (36)$$

Here, the vector $\tilde{\boldsymbol{\eta}} = [\mathbf{w}^T, \boldsymbol{\lambda}^T]^T$ includes the new variables. According to Marchisio and Fox,[16] ζ can be interpreted as the vector of source terms for the new variables. It is computed explicitly in each integration step from the linear system

$$\mathbf{M}(\mathbf{g}^{(z,\phi)}(\boldsymbol{\eta}), \boldsymbol{\eta}) \tilde{\mathbf{B}}(\boldsymbol{\eta}) \boldsymbol{\varsigma} = \mathbf{f}^{(z,\phi)}(\mathbf{g}^{(z,\phi)}(\boldsymbol{\eta}), \boldsymbol{\eta}) \quad (37)$$

The matrix $\tilde{\mathbf{B}}(\boldsymbol{\eta})$ is given in the Appendix. In the absence of parameter dependent weighting functions this matrix only depends on $\boldsymbol{\zeta}$ and not on $\mathbf{w}$. It is assumed that the matrix, therefore, has favorable properties if some weights vanish.[16] This is probably the reason, why the abscissas are not substituted entirely. The explicit form of the method also makes it particularly amenable to the implementation in CFD solvers. Note, however, that in each solution step $\zeta_i = \frac{\lambda_i}{w_i}$ has to be computed before the linear system in Eq. 37 can be solved for $\boldsymbol{\varsigma}$. Therefore, an implicit dependence on the $\mathbf{w}$ is nevertheless present in $\tilde{\mathbf{B}}(\boldsymbol{\eta})$. In the following, we refer to the direct QMOM as QMOM-DODE method, because without the change of variables it is essentially an explicit reformulation of QMOM-ODE, where Eq. 35 is multiplied by the inverse of the matrix ($\mathbf{MB}$) from the left.

Third, the abscissas and weights can be computed implicitly by an algorithm as described above[1–3]

$$\mathbf{M}(\boldsymbol{\xi}, \mathbf{g}^{(z,\phi),-1}(\boldsymbol{\xi})) \frac{d}{dt} \boldsymbol{\xi} = \mathbf{f}^{(z,\phi)}(\boldsymbol{\xi}, \mathbf{g}^{(z,\phi),-1}(\boldsymbol{\xi})) := \mathbf{F}^{(z,\phi,EIG)}(\boldsymbol{\xi}). \quad (38)$$

With $\mathbf{g}^{(z,\phi),-1}(\boldsymbol{\xi})$, we refer to the inversion of the system $\boldsymbol{\xi} = \mathbf{g}^{(z,\phi)}(\boldsymbol{\eta})$ by means of an appropriate algorithm. In our implementation we used the same algorithm as Motz et al.,[3] that is, the *long quotient difference algorithm*.[32] We refer the reader to the book of Gautschi[7] and references, therein, for an overview on algorithms for the inversion of $\mathbf{g}^{(z,\phi)}$. Since it is standard nowadays to compute the abscissas and weights by solution of an eigenvalue problem of the *Jacobi* (or *terminal*) matrix, we call this approach QMOM-EIG.

Note, that all systems have to be augmented by the mass balances and further constitutive equations. For the numerical solution of the above systems, derivative information is required. In our computations we use analytical derivatives which are readily obtained by application of the chain rule.

Existing QMOM approaches

The current QMOM approaches can be derived from the QMOM-EIG, QMOM-ODE, or QMOM-DODE formulations. All existing approaches work with a set of weighting functions $\phi_k(z)$ without time-varying parameters. Therefore, $\mathbf{M}$ is the identity matrix for all these approaches.

The standard QMOM method[1,3,14] is obviously obtained if we let $\phi_k(z) = z^k$ and $x = T(z) = z$, and use the QMOM-EIG formulation in Eq. 38

$$\frac{d\boldsymbol{\xi}}{dt} = \mathbf{f}^{(x)}(\boldsymbol{\xi}, \mathbf{g}^{(x),-1}(\boldsymbol{\xi})). \tag{39}$$

The approach taken by Piskunov and Golubev[12] is obtained if one chooses $\phi_k(z) = z^{\frac{k}{\alpha}}$ and $x = T(z) = z$. The earlier derivations show that this choice is equivalent to using $\phi_k(z) = z^k$ and $x = T(z) = z^\alpha$, that is, the standard QMOM formulation with exponential transformation, which was used later by Yoon and McGraw.[10] Piskunov and Golubev[12] implement a variant of Eq. 35. Their calculations are equivalent to a multiplication of the system in Eq. 35 by $\mathbf{B}^{-1}$ from the left in each time step. Most probably this is done in order to solve the system with an ODE integrator. Essentially, the same approach is taken by McGraw and Wright.[9] They call this the *Jacobian transformation* approach. They suggest, that one could use an analytical inverse $\mathbf{B}^{-1}$.[9] This approach is, however, limited to small systems only. In the general case, a numerical inversion of $\mathbf{B}$ would be necessary as it is also done in direct QMOM.[16] In contrast to the other contributions, Motz[17] works directly with the linearly implicit differential system stated in Eq. 35, specialized for standard moments to

$$\mathbf{B}(\boldsymbol{\eta})\frac{d}{dt}\boldsymbol{\eta} = \mathbf{f}^{(x)}(\mathbf{g}^{(x)}(\boldsymbol{\eta}), \boldsymbol{\eta}). \tag{40}$$

Motz[17] also suggests to use the quadrature algorithm for an initialization of the abscissas and nodes for given initial moments. This strategy is also employed in this work. Considering that the QMOM-ODE system is derived by an index reduction of the differential-algebraic system, a consistent initialization of the initial values is of paramount importance in order to obtain accurate results.

The standard direct QMOM of Marchisio and Fox[16] follows directly from the QMOM-DODE formulation in Eqs. 36 and 37. The source terms $\boldsymbol{\zeta}$ are then computed from

$$\tilde{\mathbf{B}}(\boldsymbol{\eta})\boldsymbol{\varsigma} = \mathbf{f}^{(x)}(\mathbf{g}^{(x)}(\boldsymbol{\eta}), \boldsymbol{\eta}). \tag{41}$$

To our knowledge only Yoon and McGraw[10] rigorously address the subject of coordinate transformation. The derivations in their article are performed for coagulation and sintering for two specific transformations, namely the power transformation $T(z) = z^\alpha$, and an exponential transformation $T(z) = \exp(z)$. This leads us to the question how a beneficial transformation can be found in general. Since the choice of the weighting functions ϕ_k by Piskunov and Golubev[12] is equivalent to the use of the power transformation with the standard weighting functions x^k, we also ask the question what other choices of ϕ_k could be used. Both questions are addressed in detail in the numerical study below.

QMOM using Laguerre polynomials

With help of the earlier relations, the derivation of a QMOM method for a specific choice of the weighting functions ϕ_k is straightforward. Continuous crystallization processes with nucleation and growth kinetics usually produce a particle-size distribution of near exponential shape. Therefore, we choose a parameter dependent weighting function of exponential shape $\psi(z, p) = pe^{-pz}$ as done by Deuflhard and Wulkow,[29] in the context of polymerization processes with a discrete internal coordinate. The polynomials, which are orthogonal with respect to this function are then the parameter-dependent Laguerre polynomials $\phi_k(z, \mathbf{p}) = l_k(z, p)$. A detailed description of the discrete version of these polynomials is found in Deuflhard and Wulkow.[29] If one chooses $p = (\mu_0^{(z)}/\mu_1^{(z)})$ as an instance of Eq. 10,[4,29] the first-order generalized moment vanishes, $\mu_1^{(l,z)} = 0$, and a differential equation for the parameter p is obtained. In the following, we denote the resulting method by LAG-QMOM. The prefix LAG denotes that equations for generalized moments with respect to Laguerre polynomials are considered. Note, that these equations can be solved by means of any of the three formulations {DAE, ODE, DODE, EIG} introduced above.

The proposed method is similar to the global Galerkin method described by Deuflhard and Wulkow.[29] Due to the orthogonality of the Laguerre polynomials, the generalized moments have the meaning of the coefficients of a weighted polynomial expansion of the distribution n. This shows how the generalized framework presented above incorporates the current QMOM, and ideas of the older global MWR approach.

When using LAG-QMOM-EIG, we compute the abscissas and nodes with respect to the generalized moments. This is done by means of the algorithm stated by Sack and Donovan.[32] There, the computation of the recurrence relation of the polynomials orthogonal to $n^{(z)}$ is then better conditioned. Accordingly, the computation of the abscissas and weights is facilitated if $n^{(z)}$ is close in shape to the exponential distribution. For a continuous crystallizer with nucleation, growth and product removal, $n^{(z)}$ directly has an exponential shape. In this case, one would then use a transformation that does not distort this shape too much. Of course a distortion is completely avoided if the identity transformation $x = T(z) = z$ is used.

It is also important to notice that no matter whether the algorithm is used with standard moments or with Laguerre moments, it should be indifferent to the results when ideal computations could be performed. Nevertheless, it is known that equal results are not necessarily obtained for finite precision computations.[7,32,36] Similarly, it should be expected that also the behavior of the other formulations improves over the standard ones.

Numerical studies

In the numerical study, reference simulations with the error controlled, adaptive population balance solver Parsival[23] are performed in order to asses the overall error of QMOM. In Parsival, the relative weighted error of the particle-size distribution can be controlled up to a user specified tolerance. In our computations we work with a value of tol = 1e-3. The

simulations of the QMOM models are performed in Matlab[37] using a mex-interface to the integrator DASSL.[38] The continuation study is performed using a Matlab based continuation code.[39] The computations are carried out on a Laptop with an Intel Pentium M processor with 1.4 GHz and 1 GB RAM.

Fines-dissolution process

In the first case study we investigate the behavior of the standard QMOM in DAE, ODE, DODE, and EIG formulation for a discontinuous fines dissolution process. The process takes place in a vessel with volume $V = 1\ \mathrm{m}^3$. The initial particle size distribution assumes a Gaussian shape ($\tilde{x} = 0.4$ mm, $\sigma = 0.05$ mm, $N_0 = 1(10^9 \#/\mathrm{m}^3)$). From the vessel, a stream of suspension with flow rate $F_f = 10\ \mathrm{m}^3/\mathrm{h}$ is removed with a classification function

$$h(x) = \frac{1}{2}\left[1 - \tanh\left(p_{f2}\frac{x - p_{f1}}{p_{f1}}\right)\right], \tag{42}$$

where we choose $p_{f1} = 0.4$ mm and $p_{f2} = 20$. The dissolved material is fed back to the vessel. Since we assume the same density for the crystalline material and the dissolved liquid component, the volume of the vessel remains constant. The time horizon for the integration starts at $t_0 = 0$ h and ends at $t_{end} = 10$ h.

Transformations

In a first step, this process is simulated with the standard QMOM-ODE without a coordinate transformation. For all simulations, the relative and absolute integrator tolerances in DASSL are set to 1e-4. Note, that this time integration tolerance is not comparable to the tolerance specified in Parsival, because Parsival controls the error of the overall distribution. In Figure 1a, the results for μ_0 are compared to a simulation in Parsival. For two abscissas and weights, $N_q = 2$, the simulation reaches the final time $t_{end} = 10$ h. This simulation takes less than 0.27 CPUs. However, for $N_q \geq 3$ the integrator gets stuck at a certain time taking ever smaller steps. We aborted the simulations at about $t = 5.7$ h for $N_q = 3$, $t = 5.0$ h for $N_q = 4$, $t = 2.1$ h for $N_q = 5$, $t = 1.1$ h for $N_q = 6$, $t = 0.7$ h for $N_q = 7$, because the simulations did not show any progress at these points even after very long integration times. For higher-values of N_q the integrator exited with an error already in the first step. In the following, we abort all simulations, when they take longer than about 100 times the time of the last successful run with fewer abscissas and weights.

Since the number N_q is usually limited to a small range, it is desirable in practice that the error in the QMOM solution for a selected set of moments, that is, $\mu_0, \ldots, \mu_4$, decreases monotonically with increasing N_q for each time t. Figure 1 shows that standard QMOM-ODE indeed approaches the reference solution with increasing N_q for very short integration times. However, for longer integration times the QMOM solution oscillates around the reference solution with increasing N_q. Therefore, at a given time the QMOM solutions with a higher-value of N_q do not necessarily lie closer to the reference solution than those with a lower N_q. This makes the behavior of QMOM difficult to predict. Also remember that the solution cannot be computed over the full time horizon for all $N_q > 3$.

Therefore, in a second step the same process is simulated using the power transformation $T(z) = z^2$. With this transformation the final time is reached for $N_q = 2, 3, 4$, but not for $N_q \geq 5$. The obtained trajectories of all moments lie comparatively close to the trajectories of the model without transformation. However, at $N_q = 4$ two abscissas cross during the simulation, and for $N_q = 5$ the first weight becomes significantly negative. These results are inconsistent with the underlying mathematical theory of the quadrature approximation and the physics of the process. With the exponential transformation $T(z) = \exp(z) - 1$, none of the simulations could be completed successfully. However, when choosing a problem tailored transformation $T(z) = (p_{f1}/p_{f2})\ \mathrm{arctanh}\ln(z) + p_{f1}$ the results displayed in Figure 1B are obtained. Here, almost all simulations for $N_q = 2, \ldots, 10$ reach the final time. Only the simulation for $N_q = 8$ had to be aborted at about $t = 8.1$ h. Even though the difference between the QMOM solution, and the reference solution is quite large for low-numbers of abscissas and weights and usually gets worse with increas-

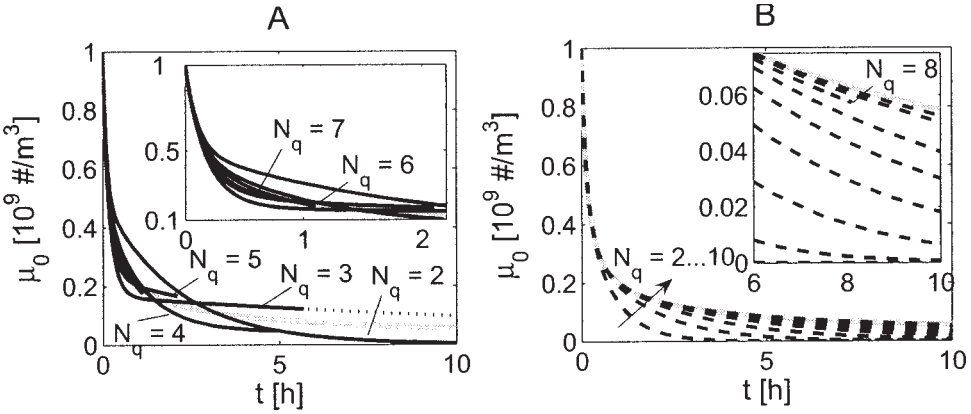

Figure 1. Comparison of simulation results for QMOM-ODE (solid black line) and QMOM-DODE (dotted black line) without transformation to Parsival (bold gray line) (A), and comparison of results for QMOM-ODE with transformation (dashed black line) to Parsival (bold gray line) (B).

Table 1. Approximation Error of the Righthand Side without Transformation $\varepsilon_{f_k}^{N_{q=3}}$ and with Transformation $\varepsilon_{f_k^{(z)}}^{N_{q=3}}$, and Error of the Moments in the Original Space Computed with the Abscissas and Weights Obtained from the Transformed moments $\varepsilon_{\mu_k}^{N_{q=3}}$.

k	0	1	2	3	4	5
$\varepsilon_{f_k}^{N_{q=3}}$	−5.07e−10	+9.34e−03	+1.96e−02	+3.08e−02	+4.26e−02	+5.50e−02
$\varepsilon_{f_k^{(z)}}^{N_{q=3}}$	−1.35e−14	+2.40e−14	−2.51e−14	+3.21e−14	−3.18e−14	−8.65e−02
$\varepsilon_{\mu_k}^{N_{q=3}}$	+4.55e−12	+4.55e−12	−2.38e−03	−7.07e−03	−1.40e−02	−2.31e−02

k is the order of the moment. The errors in $\varepsilon_{f_k^{(z)}}^{N_{q=3}}$ which are in the order of 1e-14 stem from the fact that the numerical quadrature which is used to compute the reference solution is not exact itself.

ing order of the compared moment, the solution converges to the reference solution with increasing Nq. The relative errors that is, the relative deviation from the reference solution in Parsival of μ_0 and μ_3 at the final time are $\varepsilon_{\mu_0}^{N_{q=2}} \approx 100\%$, and $\varepsilon_{\mu_3}^{N_{q=0}} \approx 100\%$ for $Nq = 2$. They improve with increasing N_q to $\varepsilon_{\mu_0}^{N_{q=10}} = 4.5\%$, $\varepsilon_{\mu_3}^{N_{q=10}} = 9.9\%$. The simulation takes 0.1 CPUs for $N_q = 2$ and up to 0.25 CPUs for $N_q = 10$. This compares favorably to the case without transformation, where the smallest error at the final time is achieved by QMOM-DODE (discussed later) with $\varepsilon_{\mu_0}^{N_{q=3}} = 75.7\%$, $\varepsilon_{\mu_3}^{N_{q=3}} = 69.1\%$. The solution at the final time cannot be obtained for a higher-number of abscissas and weights.

The success of this specific transformation is based on the fact that it linearizes the fines classification function in Eq. 42 which acts locally on the untransformed particle size axis. Consequently, a large quadrature error is observed in the righthand side of the untransformed QMOM model. Contrary to this, the righthand side in the transformed QMOM model is approximated exactly for all moments, but the last one, due to the properties of the Gaussian quadrature. Nevertheless, the QMOM model with transformation has the additional disadvantage that the standard moments are only approximately obtained via Eq. 8. The errors involved in the approximation of the righthand side and the recovery of the standard moments are displayed for the initial conditions at time $t = 0$ h in Table 1. Another disadvantage of working with a transformation is that the approximation error of the standard moments might directly couple back into the solution via the exchange terms (see for example Eqs. 30 and 32) if the kinetics depend on the liquid phase concentration. This is not the case for the present example.

By means of an analogous analysis of the quadrature error, the behavior of the power and exponential transformations can be explained. The approximation errors in the righthand side with these transformations are comparable to those without transformation given in the first row of Table 1. Compared to the exponential transformation, the power transformation has the advantage that it does not produce an approximation error for the first N_q standard moments. Note, that analytical formulas exist for the computation of the standard moments of a Gaussian distribution,[6] whereas the initial moments in the transformed space have to be computed by numerical quadrature with sufficient accuracy.

Another important aspect of a quadrature scheme is the condition number of the map $[\mathbf{g}^{(z)}(\boldsymbol{\eta})^{-1}]$ from the moments to the abscissas and weights. One can study the condition number of $\mathbf{g}^{(z)}(\boldsymbol{\eta})^{-1}$ by regarding its inverse $\mathbf{g}^{(z)}(\boldsymbol{\eta})$ (see Eq. 14). However, we do not compute the condition number of $\mathbf{g}^{(z)}(\boldsymbol{\eta})$ directly, but use the condition number of the Jacobian $\mathbf{B} = (\partial g^{(z)}(\eta)/\partial\eta)$ as a qualitative measure for a high or a low relative condition number of the map itself. We resort to this procedure, because the condition number of the Jacobian is indicative of the error amplification, that is, for example, experienced by a Newton solver in the linear system for each Newton step. For conciseness we will refer to the condition number of a map and to the condition number of its Jacobian in the subsequent text synonymously, knowing, however, that they are not the same. For a rigorous treatment of the subject, we refer to Gautschi.[7,40] It turns out that the condition number of $\mathbf{B}$ at the initial point is cond($\mathbf{B}$) = 3.44e+04 for $N_q = 2$ and cond($\mathbf{B}$) = 4.22e+12 for $N_q = 5$ without transformation. In contrast to this, we obtain cond($\mathbf{B}$) = 3.78e+00 for $N_q = 2$ and cond($\mathbf{B}$) = 9.24e+02 for $N_q = 5$ with transformation. Note, that these condition numbers occur, even though the problem itself can be regarded as well scaled. For example, at the initial point the weights and abscissas in the case without transformation for $N_q = 5$ are $\mathbf{w} = [0.01, 0.22, 0.53, 0.22, 0.01]^T$ $(\#/\mathrm{m}^3)$ and $\zeta = [0.31, 0.36, 0.40, 0.44, 0.49]^T$ mm. Clearly, from these numbers one would not expect such an extreme condition number. The finding that the map $\mathbf{g}^{(z)}(\boldsymbol{\eta})^{-1}$ can be extremely ill-conditioned, especially if the weight function has infinite support, is in agreement with the literature on orthogonal polynomials,[7] but the influence of a coordinate transformation on the condition number is not explicitly studied in these publications.

The condition numbers of the Jacobians of the overall systems behave in an analogous fashion. Obviously, the condition numbers of the map from the moments to the abscissas and weights is strongly improved by the use of the problem tailored transformation.

Formulations

When comparing the ODE versus the DODE, DAE, and EIG formulation, the following results are obtained for the standard case without transformation. For QMOM-DODE the solution can be computed for $N_q = 2,3$ over the whole integration interval. The solution for $N_q = 3$ is given as dotted line in Figure 1. For higher-values of N_q the linear solver used for the solution of the source terms in Eq. 37 issued warnings saying that the matrix was too ill-conditioned to be inverted. Eventually DASSL aborted the simulation. Note, that for $N_q > 3$ the QMOM-DODE simulations had to be aborted at an earlier time point than for QMOM-ODE. For QMOM-DAE without transformation the integrator failed to cover the whole time horizon for every simulation and reported a convergence error of the corrector. QMOM-EIG

frequently issued warnings (starting from $t = 1.69$ h for $N_q = 2$), indicating that negative entries in ρ were computed in the long quotient difference algorithm. The square root of the entries of ρ are used in the computation of the abscissas and weights.[32] From a theoretical point of view, it cannot be guaranteed any longer that the quadrature scheme is real and positive under these circumstances.[31] There are variants of the quadrature algorithm which could still compute a complex quadrature scheme. However, our implementation of the algorithm is not suitable for this case. By lack of a better means and in the hope that the algorithm might recover, we simply set the negative entries of ρ to zero, but actually these results should be discarded. In some cases the smallest abscissa becomes negative when this strategy is applied. Nevertheless, for the part of the solution that could be computed the obtained moments are identical to within the specified tolerances in all methods.

In the case with transformation, QMOM-DODE solves the system over the full integration horizon for $N_q = 2,\ldots,7$ and aborts early for all higher values of N_q. QMOM-DAE successfully solves the systems for $N_q = 2,3,5,7,9$, but aborts early for all even $N_q > 2$ with corrector convergence problems. For QMOM-EIG similar problems are observed as in the case without transformation, but the errors occur later (for example, at $t = 7.18$ h for $N_q = 2$). Again, the required accuracy for the computed parts of the solutions in terms of the moments is met.

The key to the interpretation of the grossly different behavior of the QMOM formulations is given in Figure 2. The figure compares the solutions for the abscissas and weights computed with QMOM-ODE in the cases with (dashed lines), and without (solid lines) transformation to abscissas and weights computed from the reference solution for the moments obtained in Parsival (gray lines) by means of the long quotient difference algorithm for $N_q = 2$. Notice that the abscissas computed in the case with transformation are transformed back into the original space for comparison. The figure shows a similar behavior of the first-weight for both QMOM-ODE solutions. The weight rapidly approaches zero at the beginning, and stays close to this value for the rest of the simulation time. This explains the problems of the QMOM-DAE and QMOM-EIG formulations. The algebraic system in QMOM-DAE, becomes singular if one or more weights vanish or if abscissas coincide. The long-quotient difference algorithm in QMOM-EIG experiences the reported problems because the quadrature scheme indeed becomes non-positive for the computed moments. Note, however, that the weights in the ODE and DAE cases never reach zero exactly, but vary around this value with a small bandwidth. A comparison to the abscissas and weights computed from the error controlled solution in Parsival shows that QMOM fails to reproduce the trend of the abscissas towards higher values, which is necessary to avoid the vanishing weights. Evidently, the moments computed by QMOM, are so much affected by error that the abscissas are essentially insensitive. Also observe that the abscissas and weights computed from the error controlled Parsival solution are so noisy, while the error in the solution is controlled to be small. The small numerical disturbances in the moments are obviously amplified in the abscissas and weights. When Eq. 14 is used to map these abscissas and weights back onto the Parsival solution, the moments are met within the error tolerance prescribed for Parsival. Note, that these observations are most probably the consequences of the ill-conditioning of the map $\mathbf{g}(\boldsymbol{\eta})^{-1}$ that is investigated above. The condition numbers indicate that the map is ill-conditioned in the untransformed case, even though none of the weights is close to zero, nor are the abscissas close to each other at the initial point where these numbers are calculated. During the simulation the condition numbers rise while the first weight approaches zero in the transformed and in the untransformed case. The transformation improves this situation in so far, that the condition numbers are always orders of magnitudes lower. These results explain again the behavior of QMOM-DAE, but it is not clear yet why QMOM-ODE could solve the problem. When a weight approaches zero the mass matrix, that is, $\mathbf{B}$ in the implicit system in Eq. 35 (for standard QMOM $\mathbf{M} = \mathbf{I}$), becomes numerically singular. The standard integrator DASSL is an implicit DAE solver, which can readily handle the case of a singular mass matrix. For the integrator this just means that the ODE system becomes a DAE system. That is, the differential index of the system changes from zero to one during the simulation. Therefore, the behavior of QMOM-ODE is also dependent on the integrator used. Note, however, that in addition to $\mathbf{B}$ also the Jacobian of the right hand side becomes singular in the same column as $\mathbf{B}$ if a weight approaches zeros or if nodes coincide. For a perfect singularity of both matrices, the system would not be solvable, because there would essentially be more variables than equations. In the case of a near singularity of the matrices during the numerical solution, the situation, must appear to the integrator like a further increment of the differential index. Whatever the interpretation of the situation, it is clear that with each vanishing weight the system is harder to solve numerically. For QMOM-DODE a similar argument holds. At the initial point the condition numbers of $\mathbf{B}$ and $\tilde{\mathbf{B}}$ are of compa-

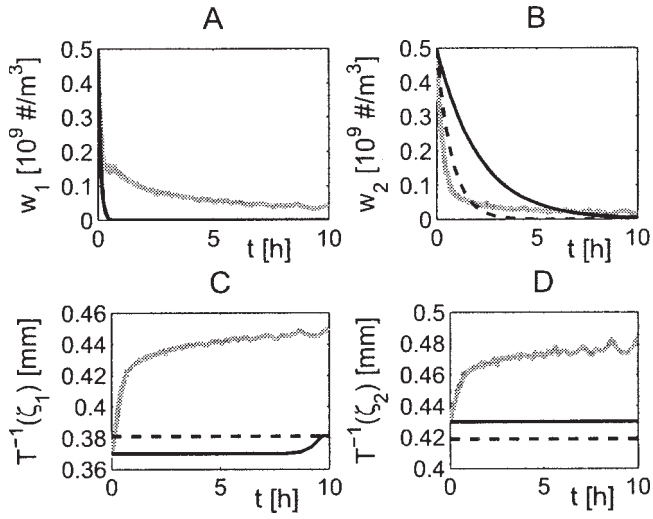

Figure 2. Comparison of abscissas and weights obtained from the QMOM-ODE simulation without transformation (solid black line), the QMOM-ODE simulation with transformation (dashed black line) and those computed from the Parsival solution for the moments (bold gray line).

rable orders of magnitude in all runs. During the simulation, the condition number of $\mathbf{B}$ increases strongly. Compared to this, the condition number of $\tilde{\mathbf{B}}$ increases mildly only for low numbers of abscissas and weights (N_q). In the case without transformation this is observed for $N_q = 2,3$. For $N_q = 2$ the condition number even remains almost constant. However, for all higher values of N_q the conditioning of $\tilde{\mathbf{B}}$ is comparable to or worse than that of **B.** This explains, why for $N_q = 3$ the solution could be computed over the whole integration horizon with QMOM-DODE, but not with QMOM-ODE. Also note that $\tilde{\mathbf{B}}$ has to be inverted numerically in Eq. 37. If the matrix becomes very ill-conditioned (cond ($\tilde{\mathbf{B}}$) $\approx 1e20$) this inversion is not possible any more, whereas the implicit formulation of QMOM-ODE still works fine if such condition numbers are observed in **B.** For this reason QMOM-DODE aborted the simulations with large N_q earlier than QMOM-ODE. In the case with transformation the condition number of $\tilde{\mathbf{B}}$ remains lower than that of $\mathbf{B}$ for $N_q < 6$. Obviously QMOM-DODE behaves favorably for low-numbers of abscissas and weights, the exact number of N_q being problem dependent.

The discussion also explains the differences observed between the simulations with and without transformation. With transformation the systems are generally much better conditioned, so that the influence of a weight which is close to zero is not as severe as in the case without transformation. However, the behavior of QMOM is not improved in a fundamental way by using the transformation, because the insensitivity is the same with and without transformation. Also, no particular reason for the failure of QMOM-ODE with transformation, and $N_q = 8$ was found. As in all other cases the first few weights vary around zero. Inspection of the values of the weights just shows that the integrator happens to take a step which produces a number of very small weights in one step and refines into the detected singularity. This could equally well have happened in the simulations with higher values of N_q, but due to the comparatively crude integration tolerance the integrator did not notice the approaching singularity.

It is stressed that for all investigated transformations and all numbers of abscissas and weights the same problem of vanishing weights occurred in this example, leading to a singularity. Notice, however, that the original distributed problem is far from being singular in the prescribed time horizon. Figure 2 shows that none of the weights computed from the Parsival solution vanish. Therefore, it has to be concluded that QMOM fails altogether to approximate the low-order behavior of the distribution which is inferred from the detailed Parsival solution in Figure 2. Nevertheless, the solution for the low-order moments could be approximated quite reasonably with the problem tailored transformation and high-numbers of N_q. We found that the problems mainly have to be attributed to the highly localized nature of the used classification function. With a smoother classification function, for example for $p_{f2} = 10$ a good agreement of the QMOM and Parsival results is observed.

Simple crystallizer with fines dissolution

This section deals with the model of a continuously operated pilot plant crystallizer ($V = 1$ m^3) taken from Pathath and Kienle.[41] In the model simple power law kinetics for growth and nucleation are considered, as well as a fines dissolution loop. Two aspect of the fines dissolution loop are taken into account. The first one is the removal of a fines fraction from the crystallizer volume with a suspension flow rate F_f. The second aspect is the dissolution of large crystals at the bottom of the vessel, where the hot stream with dissolved fines is reinserted into the apparatus. The overall effect is described by the following classification function

$$h(x) = \frac{1}{2}\left[\left(1 - \tanh\left(p_{f2}\frac{x - p_{f1}}{p_{f1}}\right)\right) + k_f\left(1 + \tanh\left(p_{fc2}\frac{x - p_{fc1}}{p_{fc1}}\right)\right)\right] \quad (43)$$

Here, the parameter k_f is used to rate the effect of fines dissolution against the dissolution of the large crystals. Note, that we use a variant of the classification function used by Pathath and Kienle[41] for comparability with the fines dissolution process discussed in the previous section. However, the chosen parameter values render our results comparable to theirs: $p_{f1} =$ 1e-4 m, $p_{f2} = 5$, $k_{fd} = 0.5$, $p_{fc1} =$ 9e-4 m, $p_{fc2} = 7.5$. We choose as initial state for our simulation the steady-state of the process without fines dissolution, that is, $F_f = 0$ m^3/h.

In the model, the growth and nucleation kinetics can be transformed into moment equations analytically. Therefore, the major part of the quadrature error stems from the dissolution kinetics.

Scaling

When analyzing the QMOM-DAE model of this process it turns out that the model is ill-conditioned. At the initial point the condition number of the Jacobian of the DAE system Eq. 34 is cond($\partial F^{DAE}(\xi,\eta)/\partial(\xi,\eta)$) = 1.38e+26, and the condition number of $\mathbf{B}$ is cond **(B)** = 3.97e+20 for $N_q = 2$ and cond ($\partial F^{DAE}(\xi,\eta)/\partial(\xi,\eta)$) = 1.73e+29 and cond($\mathbf{B}$) = 8.57e +39 for $N_q = 5$. What options are there to influence the condition number? Clearly, to consider the scaling of the model is important. However, also the choice of problem-tailored transformation, weighting function, or choice of QMOM formulation may improve the situation. This set of options is investigated briefly in the following, starting with scaling.

At the initial point the standard moments of the distribution have highly varying values ranging for example from μ_0(t = 0) = 1.34e+10 #/m^3 to μ_9(t = 0) = 2.37e-19 #m^2. During the solution of the system the moments stay in this order of magnitude. Obviously, a simple diagonal scaling of the model states could drastically improve the situation. Another scaling approach is taken by Wang and Fox,[42] who use a logarithmic transformation of the moments. Singular perturbation concepts and techniques as used in the work of Chiu and Christofides[27,28] in the context of nonlinear controller synthesis might also result in an improved conditioning of the system by dealing with the multiple time-scales present in the system.

Here, we use, however, yet another way of scaling, because it is intuitive, easily implemented, and maintains the physical meaning of the variables. We change the unit system of the model. If, for instance, the particle number is

measured in groups of 10^9 # particles instead of just 1 #/ and if the x-coordinate is measured in 10^{-3} m = 1 mm instead of 1 m, the values of the standard moments vary much less in the new unit system. This is due to the individual scaling factors that are implied by the transition from the old to the new unit system, i.e. scaling factors $C_a = 10^9$ and $C_x = 10^{-3}$ for the number and the particle size units. The values are chosen such that the connected variables are in the order of one to ten. The value of the moment μ_k in the new unit system then is $\{\mu_k\}^{\text{new}} = \{\mu_k\}^{\text{old}}/C_a C_x^k = 10^{3k}/10^9\{\mu_k^{\text{old}}\}$. Thus, one obtains for the chosen units: $\mu_0 = 13.4\ 10^9\#/\text{m}^3$ and $\mu_9 = 2.37\text{e-}1\ 10^9\#\text{mm}^9/\text{m}^3 = 2.37\text{e-}1\ \#\text{mm}^6$. A deeper analysis, not given here, shows that this choice of unit system is close to an optimal scaling in terms of C_a and C_x. We also scale the mass balance by measuring the mass in 10^3 kg = t instead of kg. The improvement of the condition numbers is significant: cond$(\partial F^{DAE}(\boldsymbol{\xi},\boldsymbol{\eta})/\partial(\boldsymbol{\xi},\boldsymbol{\eta})) = 3.24\text{e+}6$ and cond$(\mathbf{B}) = 5.58\text{e+}2$ for $N_q = 2$; cond $(\partial F^{DAE}(\boldsymbol{\xi},\boldsymbol{\eta})/\ \partial(\boldsymbol{\xi},\boldsymbol{\eta})) = 8.82\text{e+}8$, and cond$(\mathbf{B}) = 1.96\text{e+}7$ for $N_q = 5$.

Transformations

Inspired by the previous case study we are also tempted to improve the condition number by means of an appropriate transformation. It turns out, however, that a suitable transformation could not be found for this process. Neither of the standard transformations, that is, exponential or power, improve the condition number or the quadrature accuracy. Also a problem tailored transformation is not found. For the present model, the inverse of the classification function h is not bijective, and, therefore, not suitable as transformation. One could consider to take only the inverse of the fines dissolution part as transformation. Nevertheless, this transformation is also not suited for the wide stretched distributions encountered in this model. The part of the x-coordinate with $x > 0.48\text{e-}3$ m is mapped into $T^{-1}(x) = 1.0$. Thus, even though theoretically favorable in the case of $k_f = 0$, the transformation is intractable from a numerical point of view. In general one also finds that transformations tailored to improve the behavior of a specific phenomenon most likely introduce an increased quadrature error in the other phenomena. This is readily observed from equations for the phenomena in the transformed space derived in Appendix A.

Choice of weighting function

What are other possibilities available to cope with the problem of a high condition number? The LAG-QMOM method derived above could be well suited. At the initial condition the distribution assumes a perfect exponential shape. Indeed the condition number of the overall system Jacobian, and of $\mathbf{B}$ at the initial condition are significantly reduced when using the LAG-QMOM for the unscaled model: cond$(\partial F^{(z,1,DAE)}(\boldsymbol{\xi},\boldsymbol{\eta})/\partial(\boldsymbol{\xi},\boldsymbol{\eta})) = 9.28\text{e+}22$ and cond$(\mathbf{B}) = 1.84\text{e+}14$ for $N_q = 2$, cond$(\partial F^{(z,1,DAE)}(\boldsymbol{\xi},\boldsymbol{\eta})/\partial(\boldsymbol{\xi},\boldsymbol{\eta})) = 8.59\text{e+}22$ and cond$(\mathbf{B}) = 2.83\text{e+}16$ for $N_q = 5$. However, in the well-scaled case the improvement is moderate only: cond $(\partial F^{(z,1,DAE)}(\boldsymbol{\xi},\boldsymbol{\eta})/\partial(\boldsymbol{\xi},\boldsymbol{\eta})) = 2.15\text{e} + 6$, and cond$(\mathbf{B}) = 1.90\text{e+}2$ for $N_q = 2$, cond$(\partial F^{(z,1,DAE)}(\boldsymbol{\xi},\boldsymbol{\eta})/\partial(\boldsymbol{\xi},\boldsymbol{\eta})) = 8.10\text{e+}6$ and cond(B) = 4.65e+4 for $N_q = 5$.

A further analysis, not given here, shows that for the present example the condition number of $\mathbf{B}$ for LAG-QMOM always lies below the condition number of $\mathbf{B}$ for standard QMOM regardless of the scaling used. The higher the condition numbers in general, the larger usually the positive effect of using LAG-QMOM. The condition number of both matrices is, however, most significantly affected by the scaling of the system. For example, for $N_q = 3$, the condition numbers in the case of optimum scaling only differ by a factor of less than 10. The findings are qualitatively in agreement with the literature on orthogonal polynomials.[7,40] A choice of special weighting functions does not always significantly improve the condition number of the map $\mathbf{g}^{(z)}(\boldsymbol{\eta})^{-1}$ from the moments to the abscissas and weights significantly, especially for weight functions with infinite support.

Formulation

Another possibility to reduce the condition number might be the use of the QMOM-DODE formulation. However, in the well-scaled case the condition numbers are only improved for $N_q < 4$, for example, cond$(\tilde{\mathbf{B}}) = 9.29\text{e+}1$ for $N_q = 2$. For $N_q > 4$ the condition number of $\tilde{\mathbf{B}}$ is worse than that of $\mathbf{B}$ for example, cond$(\tilde{\mathbf{B}}) = 1.74\text{e+}8$ for $N_q = 5$. The overall best condition number for small values of N_q is obtained by using LAG-QMOM-DODE, for example, cond $(\tilde{\mathbf{B}}^{(J)}) = 2.40\text{e+}1$ for $N_q = 2$. Even though, the values of the condition numbers are compared at the initial condition, the behavior remains comparable over the whole integration range in this example.

Simulation results

For the simulations a relative and absolute error tolerances of $tol = 1\text{e-}8$ are chosen in DASSL. First, we observe the results for standard QMOM. With QMOM-ODE it is possible to simulate the problem for all considered values of $N_q = 2,\ldots,10$. QMOM-DODE completes the simulations for $N_q = 2,\ldots,7$ without errors. For $N_q = 8,\ldots,10$ the solution of the linear system leads to frequent warnings that the matrix $\tilde{\mathbf{B}}$ is ill-conditioned, and that the solution might be inaccurate. Nevertheless, a comparison of the solutions shows that the relative deviation of the moments in QMOM-DODE and QMOM-ODE is less than 3e-6 at the final time. For QMOM-DAE the simulation is successful for $N_q = 2,\ldots,9$. For $N_q = 10$ the integration is aborted, because the minimum step-size is reached. QMOM-EIG successfully solves the process model for $N_q = 2,\ldots,7$. For higher values of N_q the simulation issued frequent warnings, because the inverse of the ill-conditioned matrix $\mathbf{B}$ is used in the computation of the analytical derivatives. With LAG-QMOM-ODE the problem is solved without difficulties for $N_q = 2,\ldots,7$. The problem is here that the derivative of the mass matrix is more complex for LAG-QMOM-ODE than for QMOM-ODE. Therefore, we automate its generation using Maple. For $N_q = 8$ the created file becomes so large that Matlab crashes with a segmentation fault, when reading it. However, when the simulation is switched to finite differences it completes successfully for $N_q = 8,\ldots,10$. LAG-QMOM-DODE solves the problem for $N_q = 2,\ldots,9$ without warnings. LAG-QMOM-DAE solves the system for $N_q = 2,\ldots,8$ over the full integration horizon. For higher-values of N_q the integrator aborts the simulation early, because the minimum step-size is reached. LAG-QMOM-EIG solves the problem for $N_q = 2,\ldots,10$ without

difficulties. The results for the moments with the different formulations in QMOM and LAG-QMOM are identical to within the tight specified-relative error tolerance, when the simulations complete without warnings or errors.

A comparison of the integration times shows that QMOM-EIG usually is the computationally fastest formulation. For $N_q = 5$ the simulation takes only 0.61 CPUs. QMOM-ODE and QMOM-DODE usually take approximately 40–70% and QMOM-DAE, about 60–100% longer than QMOM-EIG. The situation is similar for LAG-QMOM, only that LAG-QMOM takes approximately twice the computational time of QMOM. This is mainly due to the fact that the LAG-QMOM equations (see Eq. 33) are more complex, and their evaluation is more expensive particularly for large-values of N_q.

The foregoing comparison allows to draw the following conclusions. For the present example QMOM-ODE is the most robust formulation for standard QMOM and QMOM-EIG is the fastest one. LAG-QMOM has the drawback of being more complex than QMOM, which results in longer integration times. Also, as a result of this complexity, problems are encountered with the automatically generated derivatives of the mass matrix. On the other side, the strength of LAG-QMOM is evident when the EIG formulation is used.

Since the results for the moments are identical within the specified integration tolerances in QMOM and LAG-QMOM, only the results of QMOM are compared to simulation results in Parsival in Figures 3A–D. Figures 3A and 3C show the results for μ_0 and μ_3, respectively, for $N_q = 2, 3, 4$, and Figures 3B and 3D display the results for $N_q = 6, 8, 10$. For $N_q = 2$ the QMOM results deviate largely from the Parsival results in terms of the dynamic behavior. The steady-state is attained after less then 15 h, whereas the Parsival solution does not attain a steady-state within the displayed integration time of 60 h. In contrast to this, the QMOM solution is unstable and performs continuous oscillations for $N_q = 3$. Furthermore, for $N_q = 4$ the QMOM solution attains the steady-state slightly earlier than the Parsival solution and the steady-state error (compared to the reference solution in Parsival) for μ_0 is larger than this error for $N_q = 2$. Nevertheless, for $N_q = 4$ the solution for μ_3 is reproduced quite well. From the solution for the first few values of $N_q = 2, 3, 4$ one could conclude that the solution is reproduced more accurately for larger values of N_q. However, the results for $N_q = 6, 8, 10$ evidently show that this is not the case. A direct approach with increasing N_q towards the reference solution cannot be inferred from these results.

Continuation results

A similar picture is inferred from Figures 4A–4D. Figures 4A and 4C display the results of a parameter continuation for μ_0 and μ_3 over the parameter F_f of QMOM-DAE models for $N_q = 2, 3, 4$, and Figures 4B and 4D show the results for

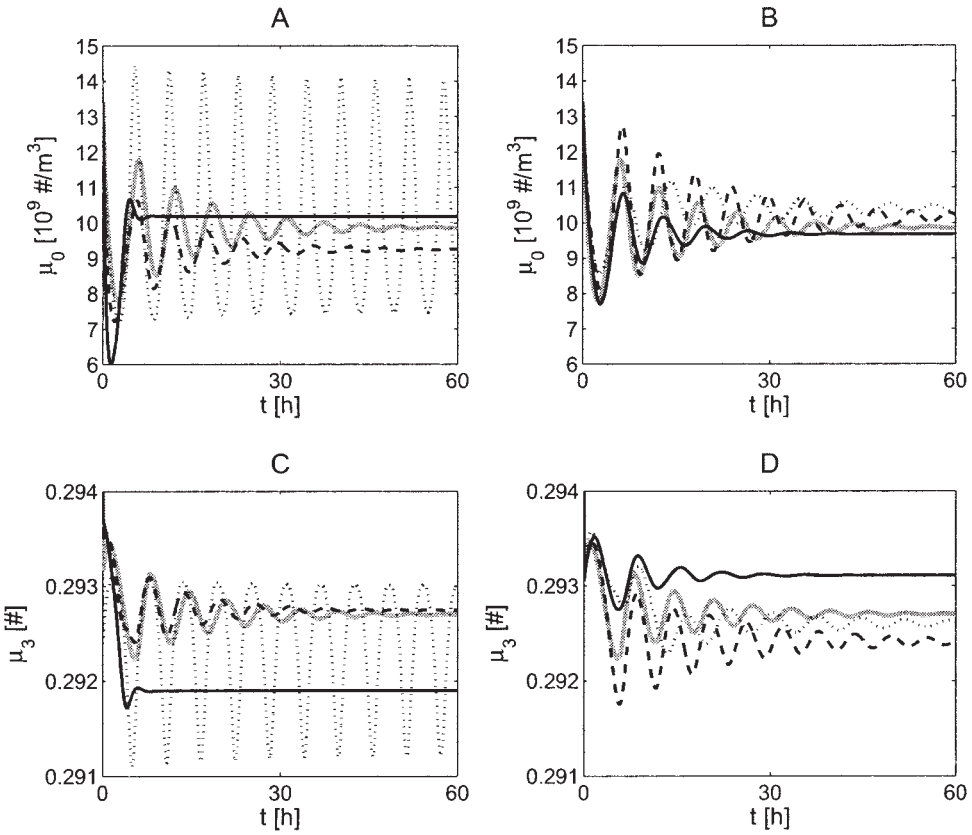

Figure 3. Comparison of simulation results for μ_0 (A) and μ_3 (C) in Parsival (solid gray line), and standard QMOM with $N_q = 2$ (solid line), $N_q = 3$ (dotted line), $N_q = 4$ (dashed line), and for μ_0 (B), and μ_3 (D) with $N_q = 6$ (solid line), $N_q = 8$ (dotted line), $N_q = 10$ (dashed line).

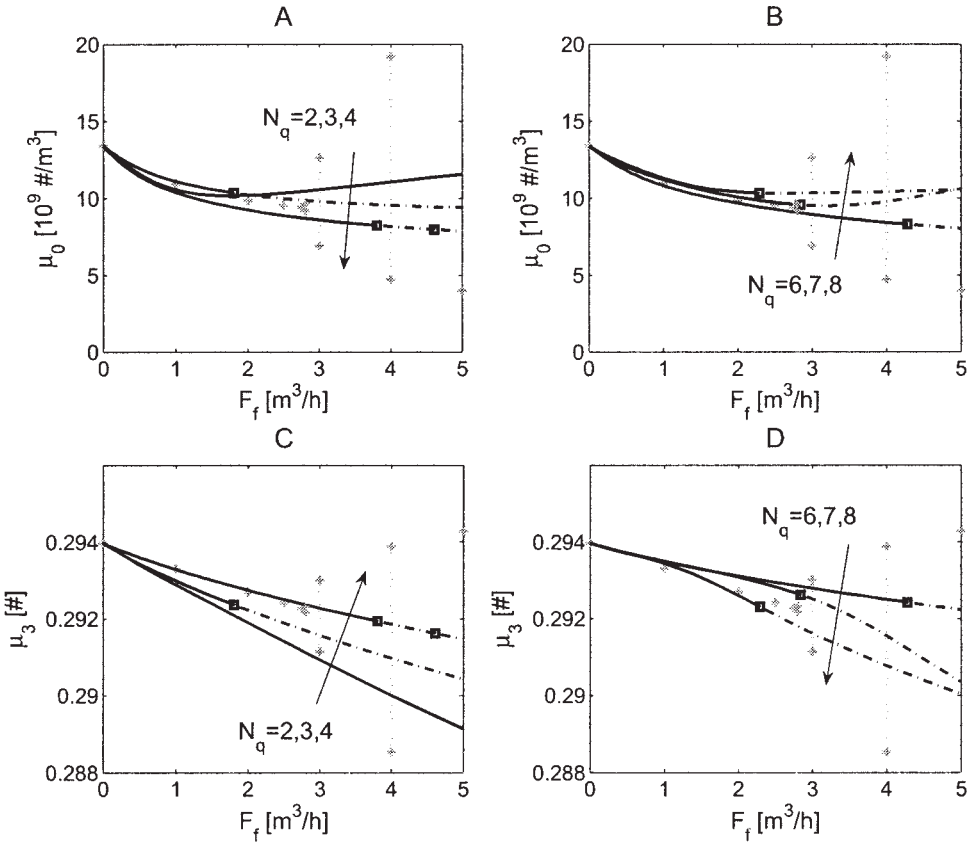

Figure 4. Comparison of continuation results for μ_0 (A) and μ_3 (C) with QMOM-DAE with N_q = 2, 3, 4, and for μ_0 (B) and μ_3 (D) with N_q = 6, 7, 8 (stable solutions (solid line), unstable solutions (dash dotted line), bifurcation points (squares)). For comparison steady-states (crosses) and amplitudes (crosses connected by dotted line) from Parsival simulations are also displayed.

the same variables for $N_q = 6, 7, 8$. In the graph stable, and unstable solutions are distinguished (solid and dash-dotted lines, respectively), and bifurcation points (squares) are displayed. In all cases the bifurcation points are Hopf points which result from a crossing of a complex conjugate pair of eigenvalues over the imaginary axis. In comparison the steady-states computed with Parsival by simulation into the steady-state at fixed values of F_f are displayed as crosses. When the Parsival solution is unstable and continuous oscillations are performed, the amplitudes of the oscillation are plotted as crosses. The bifurcation of the Parsival solution occurs around $2.75\ \mathrm{m^3/h} < F_f^{Hopf} < 2.8\ \mathrm{m^3/h}$.

Figures 4A and 4C show the same trend with respect to the dynamic behavior as the dynamic simulations in Figures 3A and 3C. For $N_q = 2$ no bifurcation occurs over the whole parameter range $F_f \in [2, 5]\ \mathrm{m^3/h}$. For $N_q = 3$ the bifurcation occurs with $F_f^{Hopf} \approx 1.81\ \mathrm{m^3/h}$ at smaller values of F_f than the bifurcation of the Parsival solution and for $N_q = 4$ the bifurcation occurs with $F_f^{Hopf} \approx 3.8\ \frac{\mathrm{m^3}}{\mathrm{h}}$ at larger values. Notice that for $N_q = 4$ a second pair of eigenvalues crosses the imaginary axis at $F_f^{Hopf} \approx 4.61\ \mathrm{m^3/h}$, which does not happen for any other QMOM model within the specified parameter range. Over the range from $F_f \in [0, 2.75]\ \mathrm{m^3/h}$ (that is, the stable branch of the Parsival solution) the QMOM solution with $N_q = 4$ reproduces the Parsival solution for μ_3 quite well ($\varepsilon_{\mu_3}^{N_q=4} < 0.1\%$), but the deviation for μ_0 is large ($\varepsilon_{\mu_0}^{N_q=4} < 8\%$). Compared to this, the steady-state errors for $N_q = 2$ are $\varepsilon_{\mu_0}^{N_q=2} < 11\%$ and $\varepsilon_{\mu_3}^{N_q=2} < 0.5\%$. Since the unstable steady-states cannot be computed in Parsival, a quantitative analysis is not realized for the unstable branch. It is nevertheless striking that some QMOM models predict an upward curvature in μ_0, and others predict only a saturation of the decay of μ_0 in the regarded parameter range. Again, the increment of N_q does not improve the quality of the approximation in all respects. For higher values of N_q the location of the bifurcation point still varies largely, and the true location of the point is approximated best for $N_q = 7$ where $F_f^{Hopf} = 2.84\ \mathrm{m^3/h}$. The solution for $N_q = 7$ also captures the basic trends of the Parsival solution on the stable branch. An upward curvature is predicted for the unstable branch. The approximation errors in this case are $\varepsilon_{\mu_0}^{N_q=7} < 2.3\%$ and $\varepsilon_{\mu_0}^{N_q=3} < 0.15\%$ on the stable branch.

Complex crystallizer model

This section deals with a state of the art model[43] of the same pilot-plant crystallizer treated as in the previous section. Size-dependent particle growth and secondary nucleation due to attrition are considered in the model. The attrition of large particles is modeled by means of a negative size-

dependent growth rate which is superimposed to the kinetic particle growth. Depending on the survival efficiency which is a function of supersaturation, a fraction of the abraded small particles is dissolved, whereas the rest is reinserted into the solid phase as secondary nuclei. The fines dissolution loop of the apparatus is modeled in the same way as in the previous section by means of a classification function (Eq. 43). Only the parameters of the classification function are slightly modified in order to ensure a similar dynamic behavior of both models. We choose $k_{fd} = 0.195$ and $p_{fc1} = 0.92$ mm. All other parameters remain unchanged. Like the simpler model, the Parsival model is stable for $F_f \leq 2.75$ m^3/h, and unstable for $F_f \geq 2.8$ m^3/h with these settings.

This model allows to study the influence of a number of non-trivial kinetics on the QMOM solution. The growth and attrition kinetics are relatively smooth, while the fines dissolution kinetics act very locally on the particle-size domain with steep gradients.

Numerical results

Based on the experience gathered in the previous section the investigation is started with a scaled model using the same unit system as in the previous section. In order to analyze the behavior of QMOM, simulation and continuation case studies similar to the ones presented in the previous section are performed. In the simulations the QMOM-ODE formulation is employed, whilst QMOM-DAE is used in the continuation case studies.

In a first step the behavior of QMOM is investigated by performing separate simulation studies for the growth and attrition kinetics. Here a reduction of the approximation error of QMOM with increasing N_q, and a very good agreement of the QMOM and Parsival results is observed. For $N_q \geq 6$ the results are within the solution tolerance of the Parsival simulation. When the model is simulated for simultaneous growth and attrition kinetics (without fines dissolution) the QMOM solution still approaches the Parsival solution with increasing N_q, but the errors turn-out to be much larger. For this model the overall error is obviously larger than the sum of the errors when individual phenomena are studied one at a time. Nevertheless, the QMOM steady-state deviates only slightly from the Parsival solution for $N_q \geq 6$. When fines dissolution is used, the deviation from the Parsival solution is substantially larger with respect to both the quantitative, as well as the qualitative behavior.

The behavior described earlier for the simulation case study can also be inferred from the results of the continuation case study displayed in Figure 5. The steady-states of the QMOM models are plotted over the continuation parameter F_f in the range $F_f = [0, 5]$ m^3/h. For $F_f = 0$ m^3/h the steady-states generally approach the Parsival steady-state (indicated by a cross) for increasing N_q. However, for higher-values of F_f QMOM steady-state solutions approach the Parsival solution for $N_q = 2, 3, 4$, but deviate again from the Parsival solution for $N_q = 5, 6, 7$. Therefore, the deviation of the QMOM steady-state for $N_q = 5$ from the Parsival steady-state is smaller than that for $N_q = 6, 7$. This trend, however, is stopped with $N_q = 8$, and the QMOM solution approaches the Parsival solutions again for $N_q = 9$. The solution for $N_q = 10$ could not be computed over the whole range due to corrector convergence problems. The problem is here the high-condition number of the system Jacobian.

In Figure 5, it is striking that the Parsival model becomes unstable for $F_f \geq 2.8$ m^3/h, but none of the displayed QMOM solutions shows a bifurcation towards an oscillatory behavior in the displayed range of F_f. If the continuation range is extended one finds, for example, that a Hopf bifurcation occurs at $F_f^{Hopf} = 5.66$ m^3/h for $N_q = 3$ and at $F_f^{Hopf} = 7.00$ m^3/h for $N_q = 4$. Only for $N_q \geq 9$ one finds an oscillatory solution in the regarded range. However, due to the high-condition numbers of the system Jacobian, the test function that detects the stability of the solution in the continuation software did not work reliably for $N_q \geq 9$. For this reason, the bifurcation point could only approximately be found by simulation. Here one finds $F_f^{Hopf} \in [2.95, 3.0]$ m^3/h for $N_q = 9$, which gives the best agreement with the Parsival solution for all $N_q \in [2,10]$. A further analysis of the situation shows that the larger qualitative and quantitative disagreement again mainly has to be attributed to the locally acting classification function. If the same parameter setting is used as for the simple crystallizer with fines dissolution ($k_{fd} = 0.5$) the behavior of the QMOM model is improved. Here, the Parsival model is unstable for $F_f > 1.8$ m^3/h, and the QMOM solution for $N_q = 3$ is unstable for $F_f^{Hopf} = 1.74$ m^3/h. However, the location of the stability boundary again deviates largely with the number of abscissas and weights N_q ($F_f^{Hopf} = 3.02$ m^3/h for $N_q = 4$). Opposed to this, if only the dissolution of small crystals is considered ($k_{fd} = 0.0$), the disagreement between QMOM and the reference solution in Parsival is worst.

The results can be interpreted as follows. Obviously, part of the error is due to the more complex interplay the different phenomena. The main error, however, stems from the poor approximation of the very localized-size dependent classification function. The smaller the value of k_{fd}, the more localized is the overall classification function and the worse are the results.

Summary and Conclusions

The aims of this article are twofold. First, a generalized framework for QMOM is formulated, on the basis of which the existing approaches are derived. Thus, similarities and differences of the approaches are established. Also, a new formulation based on generalized moments which are formed with respect to parameter dependent Laguerre polynomials is derived. By means of the presented equations any problem tailored method with a combination of transformation and weighting function can easily be derived.

Second, the work analyzes merits and pitfalls of QMOM by means of test problems that include highly-localized fines dissolution and classification functions. Notice, that the examples are, however, by no means constructed. Most crystallization processes of industrial relevance involve classification of some sort. Additionally, the findings directly carry over to other localized phenomena, such as some agglomeration or breakage kernels.

Based on the numerical evidence, guidelines for the use of the QMOM for general population balance models are devel-

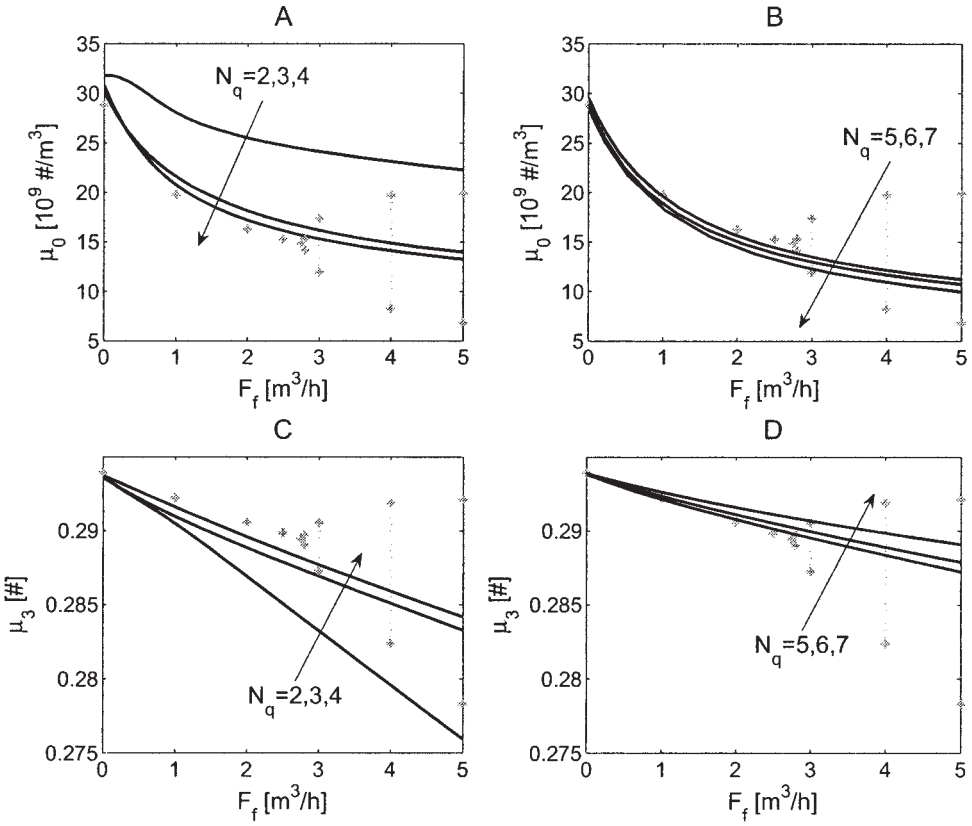

Figure 5. Comparison of continuation results for μ_0 (A) and μ_3 (C) with QMOM-DAE with N_q = 2, 3, 4, and for μ_0 (B) and μ_3 (D) with N_q = 5, 6, 7 (stable solutions (solid line)). For comparison steady-states (crosses), and amplitudes (crosses connected by dotted line) from Parsival simulations are also displayed.

oped. They are summarized together with the main results in the following paragraphs.

Performance. In terms of the computational performance of the investigated QMOM formulations the following results are obtained. In all test problems QMOM-ODE proves to be the most robust formulation, if the implicit formulation together with a DAE integrator is used. QMOM-DODE usually shows the same computational speed, but proves to be less reliable for large numbers of abscissas and weights. QMOM-EIG is always the computationally fastest formulation, but also the least robust one. The LAG-QMOM variant is found to be computationally more expensive than standard QMOM. Nevertheless, with the LAG-QMOM-EIG formulation solutions for high-numbers of abscissas and weights can be obtained, which are not obtainable with QMOM-EIG.

Error Sources. By means of the numerical study the impact of the various potential error sources is investigated, and procedures are sought to deal with these errors in a satisfactory manner.

One error source is the error in the initial data. The accurate computation of the initial moments turns out to be difficult if moments of high-order of nonstandard distributions or if generalized moments are required.[40] For QMOM-ODE and QMOM-DODE the accurate initialization of the initial abscissas and weights, which is fundamental for these QMOM formulations, is achieved by means of the quadrature algorithm.

Further error sources are the error in the recovery of the standard moments from generalized moments, and the quadrature error in the righthand side. The former error directly couples back to the QMOM model, if the crystallization phenomena depend on the concentration of the solute. Simulation results for the power transformation ($T(z) = z^{\alpha}$ with non integer α) not presented here, indicate that this error can become the dominant error source. The quadrature error is observed to increase drastically for phenomena which act locally on the particle-size domain. Therefore, such phenomena have to be treated with care. Preferentially, the phenomena should be chosen as smooth as possible. As a guideline, an analysis of the quadrature error as done in Table 1 for a set of characteristic particle distributions provides a good insight into both errors.

The last error source stems from the ill-conditioning of the map from the moments to the abscissas and weights. It is found that this map is usually ill-conditioned. As a consequence the abscissas and weights are not always computed accurately for given moments during the simulation. Nevertheless, the overall solution of the moments is in many cases observed to be rather insensitive to errors in the abscissas and weights. Therefore, the overall impact on the solution cannot be easily quantified. The results indicate that the ill-conditioning fundamentally influences the overall approximation quality of QMOM. The factors influencing the condition number of the map from the moments to the abscissas and

weights are discussed in the subsequent paragraph that deals with the condition numbers of the QMOM models in general.

Condition Numbers. The condition numbers of the system Jacobians are studied in order to assess the difficulty of finding an accurate numerical solution to the models. It is discovered that the Jacobian of the map from the moments to the abscissas, and weights is ill-conditioned for large numbers of abscissas and weights in all investigated examples. In the Section "A fines dissolution process" it is shown that the condition number is improved considerably by means of a problem tailored transformation. If the condition number of this map is high, then usually it is also high for the overall QMOM model. For all test problems presented here, none of the standard transformations used in the literature, that is, the power or the exponential transformation, improves the behavior of the QMOM models. Besides, it is found in a case study not presented here that the power transformation can significantly reduce the condition numbers of QMOM models for pure agglomeration processes. Obviously, the merits of a given transformation depend highly on the considered problem.

For the two continuous crystallization test problems a problem tailored transformation cannot be found. Furthermore, the possibility of choosing a different set of weighting functions ϕ_k is explored, by using the newly developed LAG-QMOM variant. It turns out, however, that this formulation only moderately improves the condition numbers of the QMOM model when the model is well scaled.

Using the QMOM-DODE formulation with the changed coordinates (w_i, $\lambda_i = w_i\ \zeta_i$) also does not improve the situation fundamentally. In the presented examples, the map from the moments to the new coordinates (w_i, λ_i) is usually better conditioned for small numbers of abscissas and weights (N_q), compared to the standard map. This becomes particularly evident in the case of the batch fines dissolution process without transformation, and $N_q = 2$. Here, the condition number of $\tilde{\mathbf{B}}$ stays almost constant over the whole integration range. However, for large N_q the condition numbers are usually worse than the ones of the standard map. A disadvantage of this method is further, that it relies on the explicit reformulation. Also, for the computation of the new map, the abscissas have to be computed by inversion of the weights, that is $\zeta_i = \lambda_i/w_i$, in order to solve the explicit system in Eq. 37. As a consequence small or vanishing weights inevitably lead to problems during the simulation even though the matrix $\tilde{\mathbf{B}}$ does not depend explicitly on the weights. Therefore, the method is more vulnerable than the implicit QMOM-ODE formulation for high numbers of abscissas and weights (N_q). Consequently, taking into account the robustness of the different formulations, our recommendation is to attempt a first solution with the standard QMOM-ODE formulation.

Approximation accuracy. Furthermore, the overall accuracy of the approximation is studied. For the fines dissolution process presented earlier QMOM fails altogether to produce a low-order approximation of the distributed solution. Here, a problem tailored transformation helps to obtain results for the moments, but it does not cure the problem that the abscissas are basically insensitive. The convergence behavior of QMOM is assessed by comparison of the quantitative dynamic and steady-state behavior, and by comparison of the qualitative dynamic behavior. Only for models with smooth phenomena standard QMOM approaches the true solution with an increasing number of abscissas and weights (N_q). For models with phenomena that act strictly locally on a given particle-size range, QMOM tends to oscillate around the true solution for increasing N_q. Even though the width of the oscillation seems to become narrower with increasing N_q, this behavior is unfavorable, because the increased computational effort does not guarantee an improved solution.

What should QMOM be used for? For the investigated models with nonlocal, smooth phenomena, QMOM is able to reproduce the model behavior well. In such cases, the solution accuracy usually improves with an increasing number of abscissas and weights. However, for models containing highly-localized kinetics, QMOM, behaves in a rather unpredictable fashion. The problem is, that it is not clear how smooth a model has to be, in order to be smooth enough to result in a predictable behavior. In any case the solution error is unknown.

If only few phenomena are present and the involved functions have the right properties, then one may be able to improve the behavior of QMOM with a problem-tailored transformation. However, for a general model with multiple phenomena of which some act locally on a given size range one will at most be able to find a QMOM model that describes the given model appropriately with respect to a few selected quality measures in a selected parametric range. If necessary this can also be achieved by a fit of the QMOM model. However, one will generally not be able to obtain a QMOM model that is an accurate low order approximation of the overall distributed model.

Therefore, the conclusion has to be drawn that QMOM may be used as a model reduction technique for process control and optimization for simple models. If models with a range of different interacting and local phenomena are used, the results have to be inspected with care. Consequently, it is not recommended to use QMOM blindly as a model solution technique in order to reliably predict the behavior of an unknown process. Rather, the results should be validated against a reference solution of known error as part of QMOM model development.

Acknowledgments

We acknowledge the financial support of the Deutsche Forschungsgemeinschaft (DFG) (MA 1188/17-1).

Notation

For the QMOM formulations the following naming convention applies in the text: ⟨weighting function⟩-QMOM-⟨formulation⟩. For the standard case, where $\phi_k^{(z)} = z^k$ is selected, the first part of the name is omitted.

The units stated here, apply for the untransformed case if x in the above text is chosen as a particle-size coordinate, and both the particle size, as well as the space dimension are measured in m. Furthermore, note that # = 1.

b_{nuc} = overall nucleation rate, 1/s
B_p = general source term, #/(m^4 s)
$\mathbf{B}$ = Jacobian of g
$\tilde{\mathbf{B}}$ = transformed $\mathbf{B}$ matrix in QMOM-DODE
c_e = mass concentration of solute, kg/m^3

$\mathbf{C}$ = map from standard moments to generalized moments
d_{break} = breakage rate, 1/s
D_p = general sink term, #/(m^4 s)
f = an arbitrary function
$f_s^{(p)}$ = defining function for time-varying parameter p_s
$\mathbf{f}^{(z,\phi)}$ = system of moment equations
F = suspension flow rate, m^3/s
$\mathbf{F}$ = system of closed QMOM equations
$\tilde{g}_k^{(z,\phi)}$ = defining equation for k-th order generalized moment
$g_k^{(z,T)}$ = defining equation for approximation of k-th order standard moment, (# m^k)/m^3
$\mathbf{g}$ = combined system of defining equations for moments and time-varying parameters
h = classification function, dimensionless
$\mathbf{I}$ = identity matrix, dimensionless
k_f = parameter in classification function, dimensionless
k_v = volume shape factor, dimensionless
l_k = k-th order Laguerre polynomial, dimensionless
m_e = mass of solute, kg
$\mathbf{M}$ = mass matrix
n = number density per suspension volume, #/m^4
N_{break} = number of fragments in breakage event, dimensionless
N_p = number of phenomena, dimensionless
N_q = number of quadrature abscissas and weights, dimensionless
p_f = parameter in classification function, dimensionless
$\mathbf{p}$ = vector of time-varying parameters in weighting function
P = set of all phenomena, dimensionless
S = exchange term, kg/s
t = time, s
T = transformation map, m
V = suspension volume, m^3
w_i = quadrature weight, #/m^3
x = original coordinate, e.g. particle size, m
$\tilde{x}$ = mean-particle size, m
z = transformed coordinate, transformation dependent dimension

Greek letters

β = agglomeration kernel, m^3/(# s)
γ = fragment distribution function, 1/m
ε = void fraction, dimensionless
ζ_i = quadrature abscissa, transformation dependent dimension
$\boldsymbol{\eta}$ = vector of weights and abscissas
$\tilde{\boldsymbol{\eta}}$ = vector of weights and transformed abscissas in QMOM-DODE
μ_k = k-th order standard distribution moment, (# m^k)/m^3
$\mu_k^{(z,\phi)}$ = k-th order standard generalized distribution moment in transformed space
$\boldsymbol{\xi}$ = vector of generalized moments and parameters
λ = vector of transformed abscissas in QMOM-DODE, transformation dependent dimension
ρ_s = solid-mass density, kg/m^3
ρ_i = coefficient of three-term recurrence relation, dimensionless
σ = standard deviation
σ_i = coefficient of three-term recurrence relation, dimensionless
ς = vector of sources for weights and transformed abscissas in QMOM-DODE
ϕ_k = general weighting function
ψ = weight function, 1/m

Subscripts

agg = agglomeration
break = breakage
diss = dissolution
f = fines
gr = growth
in = inflow
nuc = nucleation
out = outflow
p = phenomenon

Superscripts

DAE = QMOM-DAE formulation
DODE = QMOM-DODE formulation
EIG = QMOM-EIG formulation
ODE = QMOM-ODE formulation

Literature Cited

1. McGraw R. Description of aerosol dynamics by the quadrature method of moments. *Aerosol Sci Technol*. 1997;27:255–265.
2. Marchisio DL, Pikturna JT, Fox RO, Virgil RD, Barresi AA. Quadrature method of moments for population-balance equations. *AIChE J*. 2003;49:1266–1276.
3. Motz S, Mannal S, Gilles E-D. Integral approximation–an approach to reduced models for particulate processes. *Chem Eng Sci*. 2004;59: 987–1000.
4. Hulburt HM, Katz S. Some problems in particle technology. *Chem Eng Sci*. 1964;19:555–574.
5. Randolph AD, Larson M.-A. *Theory of Particulate Processes*. Academic Press Inc. 2 ed. 1988.
6. Ramkrishna D. *Population Balances*. London:Academic Press; 2000.
7. Gautschi W. *Orthogonal Polynomials: Computation and Approximation*. Oxford University Press; 2004.
8. Wright DL, McGraw R, Rosner DE. Bivariate extension of the quadrature method of moments for modeling simultaneous coagulation and sintering of particle populations. *J Colloid Interface Sci*. 2001;236: 242–251.
9. McGraw R, Wright DL. Chemically resolved aerosol dynamics for internal mixtures by the quadrature method of moments. *J Aerosol Sci*. 2003;34:189–209.
10. Yoon C, McGraw R. Representation of generally mixed multivariate aerosols by the quadrature method of moments: I. Statistical foundation. *J Aerosol Sci*. 2004;35:561–576.
11. Yoon C, McGraw R. Representation of generally mixed multivariate aerosols by the quadrature method of moments: II. Aerosol dynamics. *J. Aerosol Sci*. 2004;35:577–598.
12. Piskunov VN, Golubev AI. The generalized approximation method for modeling coagulation kinetics-part 1: justification and implementation. *J Aerosol Sci*. 2002;33:51–63.
13. Piskunov VN, Golubev AI, Barret JC, Ismailova NA. The generalized approximation method for modeling coagulation kinetics-part 2: comparison with other methods. *J Aerosol Sci*. 2002;33: 65–75.
14. Marchisio DL, Vigil DR, Fox RO. Quadrature method of moments for aggregation–breakage processes. *J Colloid Interface Sci*. 2003;258: 322–324.
15. Marchisio DL, Vigil RD, Fox RO. Implementation of the quadrature method of moments in CFD codes for aggregation–breakage problems. *Chem Eng Sci*. 2003;58:3337–33351.
16. Marchisio DL, Fox RO. Solution of population balance equations using the direct quadrature method of moments. *J Aerosol Sci*. 2005; 36:43–73.
17. Motz S. *Reduktion populations dynamischer Modelle*. Fortschr.-Ber. VDI Reihe 3 Nr. 809 Düsseldorf: VDI Verlag; 2004.
18. Marchisio DL, Soos M, Sefcik J, Morbidelli M. Role of turbulent shear rate distribution in aggregation and breakage processes. *AIChE J*. 2006; 52:1–16.
19. Chang S-C. The method of space-time conservation element and solution element - A new approach for solving the Navier-Stokes and Euler equations. *J Comp Physics*. 1995;119:295–324.
20. Barrett JC, Webb NA. A comparison of some approximate methods for solving the aerosol general dynamic equation. *J Aerosol Sci*. 1998; 29:31–39.
21. Upadhyay RR, Ezekoye OA. Evaluation of the 1-point quadrature approximation in QMOM for combined aerosol growth laws. *J Aerosol Sci*. 2003;34:1665–1683.
22. Kostoglou M, Karabelas AJ. An assessment of low-order methods for solving the breakage equation. *Powder Technol*. 2002;127:116–127.
23. Wulkow M, Gerstlauer A, Nieken U. Modeling and simulation of crystallization processes using Parsival. *Chem Eng Sci*. 2001;56: 2575–2588.
24. Gelbard F, Seinfeld JH. Numerical solution of the dynamic equation for particulate systems *J Comp Physics*. 1978;28:357–375.
25. Finlayson BA, Scriven LE. The method of weighted residuals–a review. *Appl Mech Reviews*. 1966;19:735–748.

26. Singh PN, Ramkrishna D. Solution of population balance equations by MWR. *Chem Eng Sci*. 1977;1:23–31.
27. Chiu TY, Christofides PD. Nonlinear control of particulate processes. *AIChE J*. 1999;45:1279–1297.
28. Chiu TY, Christofides PD. Robust control of particulate processes using uncertain population balances. *AIChE J*. 2000;46:266–280.
29. Deuflhard P, Wulkow M. Computational treatment of polyreaction kinetics by orthogonal polynomials of a discrete variable. *Impact Comp Sci Eng*. 1989;1:269–301.
30. Stroud AH, Secrest D. *Gaussian Quadrature Formulas*. Prentice-Hall Inc.; 1966.
31. Laurie DP. Computation of Gauss-type quadrature formulas. *J Comp Appl Math*. 2001;127:201–217.
32. Sack RA, Donovan AF. An algorithm for Gaussian quadrature given modified moments. *Numer Math*. 1972;18:465–477.
33. Litster JD, Smit DJ, Hounslow MJ. Adjustable discretized population balance for growth and aggregation. *AIChE J*. 1995;41:591–603.
34. Bachmann R, Brüll L, Mrziglod Th, Pallaske U. On methods for reducing the index of differential algebraic equations. *Comp Chem Eng*. 1990;14:1271–1273.
35. Unger J, Kröner A, Marquardt W. Structural analysis of differential-algebraic equation systems–theory and applications. *Comp Chem Eng*. 1995;19:867–882.
36. Gautschi W. How and how not to check gaussian quadrature-formulas. *BIT*. 1983;23:209–216.
37. MathWorks, The Inc. *MatLab Version 7: User manual*. 3 Apple Hill Drive, Natick, MA 01760–2098, USA; 2004.
38. Petzold LR. A description of DASSL: A differential/algebraic system solver sand82–8637 Sandia National Laboratories; 1982.
39. Grosch R, Marquardt W. *DASCont: continuation of DAEs in Matlab*. Process Systems Engineering, RWTH Aachen University Aachen; 2003.
40. Gautschi W. On generating orthogonal polynomials. *SIAM J Sci Stat Comput*. 1982;3:289–317.
41. Pathath PK, Kienle A. Nonlinear oscillations in ammonium sulfate crystallization: a comparison of different model predictions. *Ind Eng Chem Res*. 2003;42:6949–6955.
42. Wang I, Fox RO. Application of in situ adaptive tabulation to CFD simulation of nano-particle formation by reactive precipitation. *Chem Eng Sci*. 2003;58:4387–4401.
43. O'Meadhra R, Rosmalen GM. Scale-up of ammonium sulphate crystallization in a DTB crystallize. *Chem Eng Sci*. 1996;51:3943–3950.

Appendix

Transformation of size coordinate

According to Eq. 7 the population balance (Eq. 1) is transformed term by term by substitution of x with $x = T(z)$, and multiplication of both sides of the equation by the derivative of the transformation $(dx/dz) = (dT(z)/dz)$. For the integral terms the bounds of integration have to be transformed as well.

The transformation of the holdup term is trivial for a transformation which is independent of time. Likewise, the source due to particle loaded inflow with a given distribution or due to the nucleation with a given distribution are trivially transformed by application of Eq. 7.

Noting that $G(x) = dx/dt$ by definition,[10] the growth rate is transformed as

$$G^{(z)}(z,t) = \left(\frac{dT(z)}{dz}\right)^{-1} G(T(z),t). \tag{A1}$$

The transformation of the classified removal yields

$$D^{(z)}_{\text{out}}(z,t) = D^{(x)}_{\text{out}}(T(z),t)\frac{dT(z)}{dz} = F_{j,\text{out}}(t)h^{(z)}(z,t)n^{(z)}(z,t), \tag{A2}$$

where we introduce

$$h^{(z)}(z,t) = h(T(z),t), \tag{A3}$$

for the classification function in the transformed space.

Starting from the formulation of the agglomeration source in the particle size coordinate system one obtains under the restrictive condition that in the limit $\lim_{z\to\infty} T(z) = \infty$

$$B^{(z)}_{\text{agg}}(z,t) = \frac{1}{2}T(z)^2\frac{dT(z)}{dz} \int_{T^{-1}(0)}^{z=T^{-1}(x)} \frac{\beta^{(z)}(u(z,z'),z',t)}{T(u(z,z'))^2, \frac{dT(z)}{dz}\Big|_{z=u(z,z')}} n^{(z)}(u(z,z'),t)n^{(z)}(z',t)dz'. \tag{A4}$$

Here, we introduce the function $u(z',z'') = T^{-1}\{(T(z')^3 - T(z'')^3)^{\frac{1}{3}}\}$ which has the meaning of a particle size in the transformed space. We also introduced the definition of the agglomeration frequency

$$\beta^{(z)}(z',z'',t) = \beta(T(z'),T(z''),t) \tag{A5}$$

in the transformed space.

Note, if the transformation $T(z) = (z/k_v)^{(1/3)}$ is applied with k_v being the shape factor, the familiar agglomeration source in the volume coordinate system is obtained. The transformation of the agglomeration sink is straightforward

$$D^{(z)}_{\text{agg}}(z,t) = n^{(z)}(z,t)\int_{T^{-1}(0)}^{T^{-1}(\infty)} \beta(z,z',t)n^{(z)}(z',t)dz'. \tag{A6}$$

The transformation of the breakage source and sink terms are performed similarly

$$B^{(z)}_{\text{break}}(z,t) = \int_z^{T^{-1}(\infty)} N_{\text{break}}\ \gamma^{(z)}(z,z',t)d^{(z)}_{\text{break}}(z',t)n^{(z)}(z',t)dz', \tag{A7}$$

$$D^{(z)}_{\text{break}}(z,t) = d^{(z)}_{\text{break}}(z,t)n^{(z)}(z,t), \tag{A8}$$

with the breakage rate

$$d^{(z)}(z,t) = d(T(z),t), \tag{A9}$$

and the fragment distribution

$$\gamma^{(z)}(z',z'',t) = \gamma(T(z'),T(z''),t)\frac{dT(z')}{dz'}. \tag{A10}$$

The result for $\gamma^{(z)}$ is explained by the fact that γ has the meaning of a distribution for the first argument, as it describes the probability to form a fragment of size z' by a breakage event of a particle of size z''.

The population balance in the transformed space is then

$$\frac{\partial n^{(z)}(z,t)}{\partial t} = -\frac{\partial (G^{(z)}(z,t) n^{(z)}(z,t))}{\partial z} + \sum_{p \in P} [B_p^{(z)}(z,t) - D_p^{(z)}(z,t)] - n^{(z)} \frac{1}{V(t)} \frac{dV(t)}{dt}. \quad \text{(A11)}$$

This population balance is equivalent to the balance in Eq. 1 which is represented in the particle size coordinate system.

Appendix B. Details of QMOM-DODE Formulation

After introduction of the transformed abscissas $\lambda_i = w_i \zeta_i$ into **B** one obtains for the elements of $\tilde{\boldsymbol{B}}$ the following expressions

$$\tilde{B}_{k+1,i} = \phi_k(\zeta_i, \mathbf{p}) - \zeta_i \frac{\partial \phi_k(\zeta_i, \mathbf{p})}{\partial \zeta_i} + \sum_{s=1}^{N_p} \sum_{l=1}^{N_q} \left(w_l \frac{\partial \phi_k(\zeta_l, \mathbf{p})}{\partial p_s} \frac{\partial f_s^{(p)}(\boldsymbol{\eta})}{\partial w_i} - \zeta_i \frac{w_l}{w_i} \frac{\partial \phi_k(\zeta_l, \mathbf{p})}{\partial p_s} \frac{\partial f_s^{(p)}(\boldsymbol{\eta})}{\partial \zeta_i} \right), \quad k = 0, \ldots, 2N_q - 1 - N_p, i = 1, \ldots, N_q, \quad \text{(B1)}$$

$$\tilde{B}_{k+1,i+N_q} = \frac{\partial \phi_k(\zeta_i, \mathbf{p})}{\partial \zeta_i} + \sum_{s=1}^{N_p} \sum_{l=1}^{N_q} \left(\frac{w_l}{w_i} \frac{\partial \phi_k(\zeta_l, \mathbf{p})}{\partial p_s} \frac{\partial f_s^{(p)}(\boldsymbol{\eta})}{\partial \zeta_i} \right), \quad k = 0, \ldots, 2N_q - 1 - N_p, i = 1, \ldots, N_q, \quad \text{(B2)}$$

$$\tilde{B}_{k+2N_q-N_p,i} = \frac{\partial f_k^{(p)}(\boldsymbol{\eta})}{\partial \zeta_i} - \frac{\zeta_i}{w_i} \frac{\partial f_k^{(p)}(\boldsymbol{\eta})}{\partial \zeta_i}, \quad k = 1, \ldots, N_p, \quad i = 1, \ldots, N_q \quad \text{(B3)}$$

$$\tilde{B}_{k+2N_q-N_p,i+N_q} = \frac{1}{w_i} \frac{\partial f_k^{(p)}(\boldsymbol{\eta})}{\partial \zeta_i}, \quad k = 1, \ldots, N_p, \; i = 1, \ldots, N_q. \quad \text{(B4)}$$

The summation terms stem from the application of the chain rule to the weighting functions $\phi_k(z)$, which also depend on the time-varying parameters **p**. The dependence of **p** on the abscissas and weights is then expressed in terms of the defining function $f_s^{(p)}(\boldsymbol{\eta}) = \tilde{f}_s^{(p)}(\tilde{\mathbf{g}}^{(z,T)}(\boldsymbol{\eta}))$ in Eq. 10 in which the defining equation for the standard moments, Eq. 13, has been substituted.

Manuscript received Sept. 23, 2005, and revision received Aug. 18, 2006.