

Automatic Differentiation based QMOM for Population Balance Equations

Vinay Kariwala* Yi Cao** Zoltan K. Nagy***

* *School of Chemical & Biomedical Engineering, Nanyang
Technological University, Singapore 637459 (e-mail: vinay@ntu.edu.sg)*

** *School of Engineering, Cranfield University, Cranfield, Bedford
MK43 0AL, UK (e-mail: y.cao@cranfield.ac.uk)*

*** *Department of Chemical Engineering, Loughborough University,
Leicestershire LE11 3TU, UK (e-mail: z.k.nagy@lboro.ac.uk)*

Abstract: The quadrature method of moments (QMOM) has emerged as a promising tool for the solution of population balance equations in the past few years. The QMOM requires solving differential algebraic equations (DAEs) consisting of ordinary differential equations related to the evolution of moments as well as nonlinear algebraic equations resulting from the quadrature approximation of moments. In this paper, the use of automatic differentiation (AD) technique is proposed for solution of DAEs arising in QMOM. In the proposed method, the variables of interest are approximated using high-order Taylor series. Using AD, the high-order Taylor coefficients can be recursively calculated to obtain high-fidelity solution of the DAE system. Benchmark examples drawn from literature are used to demonstrate the superior accuracy and computational advantage of the AD-QMOM over existing state-of-the-art techniques, such as DAE-QMOM.

Keywords: Automatic differentiation, Dynamic simulation, Particulate processes, Population balance equations, Quadrature method of moments.

1. INTRODUCTION

Population balance models have been widely used for modelling particulate, droplet or bubble dynamics in single or multiphase processes (Ramkrishna, 2000). The solution of a population balance equation (PBE) usually requires computationally-expensive, complex numerical techniques. The variety of solution approaches proposed for the PBE include the standard method of moments (MOM) (Hulburt and Katz, 1964), the quadrature method of moments (QMOM) (McGraw, 1997), the method of characteristics (Aamir et al., 2009), direct numerical solution techniques, such as finite volume (Gunawan et al., 2004), lattice Boltzmann method (Majumder et al., 2010) and finite difference schemes (Nicmanis and Hounslow, 1998), and kinetic Monte Carlo simulation approaches (Rosner et al., 2003). The numerical robustness and computational efficiency of the solution methods are of significant importance especially in the cases of model based control and optimization, as well as in coupled computational fluid dynamics and PBE applications.

The QMOM proposed by McGraw (1997) has been recently accepted as one of the most efficient approaches for solving generic PBEs with growth, nucleation, aggregation/coalescence and breakage mechanisms. The QMOM utilizes the quadrature theory to avoid the closure problem encountered in the case of standard MOM simulations. The QMOM by McGraw (1997) is based on the product difference algorithm (PD) of Gordon (1968). Application of the QMOM has been extended into aggregation, coag-

ulation and breakage systems (Wright et al., 2001; Rosner and Pyykonen, 2002; Marchisio et al., 2003). However, the PD algorithm is not always the best approach for computing the quadrature points from the moments of the particle size distribution because for a larger number of moments, the method is sensitive to small errors. Therefore, the applicability of QMOM is limited to no more than six quadrature points (McGraw, 1997) and generally even fewer for more complex processes, such as diffusion-controlled growth with secondary nucleation. Several variants of the QMOM methods have been developed recently, such as Jacobian matrix transformation method (McGraw and Wright, 2003), direct QMOM (Fan et al., 2004) and fixed QMOM (Alopaeus et al., 2006). However, these QMOM solutions also suffer from numerical robustness deficiencies for more complex processes. Recently, an alternate solution technique for the QMOM was introduced based on the simultaneous solution of the moment equations and quadrature approximation as a semi-explicit differential-algebraic equation (DAE) system (Grosch et al., 2006; Gimbut et al., 2009). The DAE-QMOM method (Gimbut et al., 2009) showed increased robustness and significantly better computational efficiency than the PD-QMOM method. These advantages, however, can only be achieved with the analytical computation of the Jacobian matrix of the DAE system, which is not readily possible for complex mechanisms.

In this paper, a novel methodology is proposed for the solution of PBEs based on the automatic differentiation (AD) algorithm (Griewank, 2000). AD belongs to the class

of computational techniques used to evaluate derivatives of functions defined in computer programs. Such programs consist of a sequence of elementary operations whose derivatives are well known. By numerically applying the chain rule to these arithmetic sequences, not only can AD deliver truncation error free derivatives, hence superior to finite difference approximation, but also avoids code growth, which is a common issue associated with symbolic differentiation approaches. In AD, high-order Taylor coefficients of continuous functions can be recursively obtained. Recently, the superior computational efficiency and numerical accuracy of this method for solving various differential equation problems has been demonstrated (Cao, 2005; Al-Seyab and Cao, 2008a,b; Pryce, 1998; Barrio, 2005). In this work, the AD approach is extended to solve the DAEs arising in the QMOM. The AD-QMOM approach is evaluated for various mechanisms like growth and breakage. It is shown that the approach provides increased robustness and computational efficiency compared to the DAE-QMOM method (Gimbun et al., 2009).

2. QUADRATURE METHOD OF MOMENTS

The dynamic PBE for a closed homogeneous system can be written with diameter as the internal coordinate as:

$$\begin{aligned} \frac{\partial n(L)}{\partial t} = & \underbrace{\int_L^\infty b(L, \lambda) a(\lambda) n(\lambda) d\lambda}_{\text{birth due to breakage}} \\ & + \underbrace{\frac{L^2}{2} \int_0^L \frac{\beta((L^3 - \lambda^3)^{1/3}, \lambda) n((L^3 + \lambda^3)^{1/3}, \lambda) n(\lambda)}{(L^3 - \lambda^3)^{2/3}} d\lambda}_{\text{birth due to coalescence or aggregation}} \\ & - \underbrace{a(L) n(L)}_{\text{death due to breakage}} - \underbrace{n(L) \int_0^\infty \beta(L, \lambda) n(\lambda) d\lambda}_{\text{death due to coalescence or aggregation}} \\ & - \underbrace{\frac{\partial(G(L)n(L))}{\partial L}}_{\text{growth}} + \underbrace{\delta(0, L)B}_{\text{nucleation}} \quad (1) \end{aligned}$$

where β , a , G , B , b , and δ are the aggregation kernel, breakage kernel, growth rate, nucleation rate, the daughter particle size distribution and the Dirac delta function, respectively, whereas both L and λ are the particle characteristic length. The PBE in (1) can be simplified using a moment transformation, where the r^{th} moment of the distribution, μ_r , is given by:

$$\mu_r = \int_0^\infty n(L) L^r dL \quad (2)$$

After the moment transformation, the PBE in (1) is represented by a set of ordinary differential equations (ODEs) in terms of the moments:

$$\begin{aligned} \frac{d\mu_r}{dt} = & \int_0^\infty L^r \int_0^\infty a(\lambda) b(L, \lambda) n(\lambda) d\lambda dL \\ & + \frac{1}{2} \int_0^\infty n(\lambda) \int_0^\infty \beta(L, \lambda) (L^3 + \lambda^3)^{r/3} n(L) dL d\lambda \\ & - \int_0^\infty L^r a(L) n(L) dL - \int_0^\infty L^r n(L) \int_0^\infty \beta(L, \lambda) n(\lambda) d\lambda dL \\ & - \int_0^\infty r L^{r-1} G(L) n(L) dL + \delta(0, L) B \quad (3) \end{aligned}$$

The moment equations in (3) are solvable for growth and nucleation problems using the standard MOM technique, however, it is not possible to solve the breakage and coalescence terms due to the closure problem, since the integrations cannot be written in term of the moments. Therefore, (3) needs to be transformed again into a quadrature MOM formulation to eliminate the closure problem. The essence of the quadrature closure is to consider the number density $n(L)$ as a general weight function and to approximate the integrals that appear during the transformation of the PBE to moment equations in terms of a set of abscissas and weights. The QMOM employs a quadrature approximation (McGraw, 1997):

$$\mu_r \approx \sum_{\ell=1}^N w_\ell L_\ell^r \quad (4)$$

where w_ℓ are the weights, L_ℓ are the abscissas and N is the number of quadrature points. This quadrature approximation is exact if the function in (4) are polynomials up to the order $2N - 1$. After applying the quadrature rule, the moment transformed PBE can be written as:

$$\frac{d\mu_r}{dt} = f_r(w, L, \gamma) \quad (5)$$

where $\gamma = [\beta, a, G, B, b]$ and

$$\begin{aligned} f_r = & \sum_{\ell=1}^N w_\ell a(L_\ell) b(r, L_\ell) \\ & + \frac{1}{2} \sum_{\ell=1}^N w_\ell \sum_{m=1}^N w_m \beta(L_\ell, L_m) (L_\ell^3 + L_m^3)^{r/3} \\ & - \sum_{\ell=1}^N w_\ell a(L_\ell) L_\ell^r - \sum_{\ell=1}^N w_\ell L_\ell^r \sum_{m=1}^N w_m \beta(L_\ell, L_m) \\ & + r \sum_{\ell=1}^N w_\ell L_\ell^{r-1} G(L_\ell) + \delta(0, r) B \quad (6) \end{aligned}$$

Now the closure problem has been eliminated, and hence the PBE in (5) is solvable using QMOM by following the evolution of w_ℓ and L_ℓ , as well as μ_r . The moments are non-linearly related to the weights and abscissas by (4).

The QMOM calculations require integration of the ODEs in (5) generated from the moment equations for $r = 0, 1, \dots, 2N - 1$, alongside the solution of non-linear algebraic equations in (4) obtained from the quadrature rule. These equations may be numerically solved simultaneously as a set of coupled DAEs. The DAE method is an attractive method for solving the QMOM since it arises from the natural mathematical formulation of the QMOM approximation problem. Equations (4) and (5) together represent a semi-explicit DAE system which can be solved using standard DAE solution techniques and software. For increased robustness, the Jacobian of the DAE system should be computed analytically and used in the numerical integration of the system (Gimbun et al., 2009). Although previous studies have shown that the DAE-QMOM method provides a computationally more efficient and robust approach for solving PBEs than other QMOM-based approaches, the technique is still limited to a small number of quadrature points (typically not more than 5-6), and relies on the computation of the analytical Jacobian of the DAE system, which is not always possible.

Table 1. Useful identities for propagation of Taylor series coefficients (Griewank, 2000)

	z	$z^{[k]}$
1	cx	$cx^{[k]}$
2	$x + y$	$x^{[k]} + y^{[k]}$
3	xy	$\sum_{j=0}^k x^{[k-j]}y^{[j]}$
4	x^r	$\frac{1}{kx^{[0]}} \left(r \sum_{j=1}^k jz^{[k-j]}x^{[j]} - \sum_{j=1}^{k-1} jx^{[k-j]}z^{[j]} \right)$

3. AUTOMATIC DIFFERENTIATION BASED QMOM

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ be a d -time continuously differentiable function and $x(t) \in \mathbb{R}^n$ be given as a truncated Taylor series: $x(t) = \sum_{k=0}^d x^{[k]}t^k$. Then, the Taylor coefficients of $z(t) = f(x(t)) = \sum_{k=0}^d z^{[k]}t^k \in \mathbb{R}^m$ can be recursively calculated through AD based on the chain rule, $z^{[k]} \equiv f^{[k]}(x^{[0]}, \dots, x^{[k]})$; see Table 1 for examples. Furthermore, for a differential equation, $\dot{x} = f(x)$, since $\dot{x}^{[k]} = (k+1)x^{[k]}$, all Taylor coefficients, $x^{[k]}$ of the solution can be recursively calculated from $x^{[0]} = x(t_0)$, as $x^{[k+1]} = \frac{1}{k+1}f^{[k]}(x^{[0]}, \dots, x^{[k]})$. Therefore, the solution at $t_1 = t_0 + h$ can be explicitly obtained as $x(t_1) = \sum_{k=0}^d x^{[k]}h^k$. The integration step, h is determined according to the specified tolerance (Cao, 2005).

To solve the DAE system in (4)-(5) using the AD method, the variables of interest, namely μ , w and L , are approximated using d -order Taylor series,

$$\mu_r = \sum_{k=0}^d \mu_r^{[k]}t^k, \quad w_\ell = \sum_{k=0}^d w_\ell^{[k]}t^k, \quad L_\ell = \sum_{k=0}^d L_\ell^{[k]}t^k \quad (7)$$

where $\mu_r^{[k]}$, $w_\ell^{[k]}$ and $L_\ell^{[k]}$ are the Taylor coefficients of μ_r , w_ℓ and L_ℓ , respectively. These coefficients can be recursively obtained through AD. For tutorial purpose, these recursive expressions are explicitly derived in the following discussion. Firstly, the coefficients of $\mu_r^{[k]}$ are derived according to (5) through the Taylor expansion of $f_r = \sum_{k=0}^d f_r^{[k]}t^k$ as

$$\mu_r^{[k+1]} = \frac{\tau}{k+1} f_r^{[k]}; \quad k = 1, 2, \dots, d-1 \quad (8)$$

where τ is a time scaling factor so that $\mu_r(t_0 + h) = \sum_{k=0}^d \mu_r^{[k]}(h/\tau)^k$. Usually τ can be chosen to be unity, but the numerical stability can be improved by properly selecting τ . This is because the truncation error is mainly determined by the norm of the high-order term of the Taylor series, $\|\mu_r^{[d]}h^d\|$. If h is too small, for the same error tolerance, $\|\mu_r^{[d]}\|$ will be too large for numerical stability. On the other hand, if h is too large, then $\|\mu_r^{[d]}\|$ will be too small to maintain sufficient accuracy. Ideally, it is best to let $\tau = h$. However, h is determined after all $x^{[k]}$ are calculated. To avoid recalculation of $x^{[k]}$, we select τ to be the previous time step h .

Clearly, $f_r^{[k]}$ depends on the Taylor coefficients of w , L and μ , which are problem specific and are discussed for specific problems in the next section. The focus of rest of this section is on solving the algebraic equations in (4). Note that the zeroth order Taylor coefficients of w_ℓ and

L_ℓ are inherited from the previous step. The higher order Taylor coefficients of w_ℓ and L_ℓ can be found by solving a set of linear equations, as shown next.

When $r = 0$,

$$\mu_0^{[k]} = \sum_{\ell=1}^N w_\ell^{[k]}; \quad k = 1, \dots, d \quad (9)$$

For $r > 0$, using Identity 3 in Table 1, we have

$$\mu_r^{[k]} = \sum_{\ell=1}^N \sum_{i=0}^k w_\ell^{[k-i]} (L_\ell^r)^{[i]} \quad (10)$$

The right-hand side of (10) can be arranged as:

$$\sum_{\ell=1}^N \left(w_\ell^{[k]} (L_\ell^{[0]})^r + w_\ell^{[0]} (L_\ell^r)^{[k]} + \sum_{i=1}^{k-1} w_\ell^{[k-i]} (L_\ell^r)^{[i]} \right)$$

Using Identity 4 in Table 1 to express $(L_\ell^r)^{[k]}$ in terms of $L_\ell^{[k]}$ and noting that $(L_\ell^r)^{[0]} = (L_\ell^{[0]})^r$,

$$\tilde{\mu}_r^{[k]} = \sum_{\ell=1}^N w_\ell^{[k]} (L_\ell^{[0]})^r + \sum_{\ell=1}^N r w_\ell^{[0]} (L_\ell^{[0]})^{r-1} L_\ell^{[k]} \quad (11)$$

where

$$\tilde{\mu}_r^{[k]} = \mu_r^{[k]} - \sum_{\ell=1}^N \left(\frac{w_\ell^{[0]}}{k L_\ell^{[0]}} \sum_{j=1}^{k-1} ((r+1)j - k) (L_\ell^r)^{[k-j]} L_\ell^{[j]} + \sum_{i=1}^{k-1} w_\ell^{[k-i]} (L_\ell^r)^{[i]} \right) \quad (12)$$

Equations (9) and (11) can be put in matrix format as

$$Ax = b \quad (13)$$

where

$$A_{ij} = \begin{cases} (L_j^{[0]})^{i-1} & \text{for } 1 \leq j \leq N \\ (i-1)w_j^{[0]} (L_j^{[0]})^{i-2} & \text{for } N+1 \leq j \leq 2N \end{cases} \quad (14)$$

$$x = [w_1^{[k]} \dots w_N^{[k]} L_1^{[k]} \dots L_N^{[k]}]^T \quad (15)$$

$$b = [\mu_0^{[k]} \tilde{\mu}_1^{[k]} \tilde{\mu}_2^{[k]} \dots \tilde{\mu}_{2N-1}^{[k]}]^T \quad (16)$$

Thus, the unknowns ($w_\ell^{[k]}$ and $L_\ell^{[k]}$) can be found by solving the linear equations in (13). Note that the matrix A depends on the zeroth order Taylor coefficients only and thus needs to be inverted only once for all k .

Error control. The efficiency of the AD method greatly relies on error control to adopt the integration step as large as possible. Assume that $x(t) = [\mu^T(t) w^T(t) L^T(t)]^T$ at the next integration time $t = t_0 + h$ be $x(t_0 + h) = \sum_{k=0}^d x^{[k]}(t_0)h^k + \epsilon(h, d)$, where $\epsilon(h, d)$ is the truncation error. Then,

$$\epsilon(h, d) \approx C(h/r)^{d+1} \quad (17)$$

where r is the radius of convergence and C is a constant. For sufficiently large d ,

$$r \approx r_d := \frac{\|x^{[d-1]}\|_\infty}{\|x^{[d]}\|_\infty} \quad (18)$$

Since, $\epsilon(h, d-1) \approx \epsilon(h, d)(r_d/h) \approx \epsilon(h, d) + \|x^{[d]}\|_\infty h^d$, it leads to the following estimation of the truncation error:

$$\epsilon(h, d) \approx \frac{h^{d+1} \|x^{[d]}\|_\infty^2}{\|x^{[d-1]}\|_\infty - h \|x^{[d]}\|_\infty} \quad (19)$$

Therefore, for specified d and error tolerance δ , the integration step can be estimated to satisfy $\epsilon(h, d) \leq \delta$. For $h > 1$, it leads to

$$h \leq \left(\frac{\delta \|x^{[d-1]}\|_\infty}{\|x^{[d]}\|_\infty^2} \right)^{1/(d+1)} \quad (20)$$

The procedure for AD-QMOM is given in Algorithm 1.

Algorithm 1. Initially, compute $\mu_r(0)$, $r = 0, 1, \dots, 2N - 1$, based on the given initial distribution. Solve the nonlinear equations in (4) to obtain $w_\ell(0)$ and $L_\ell(0)$, $\ell = 1, 2, \dots, N$. Choose d and initial τ . While $t < t_f$:

- a. Set $\mu_r^{[0]} = \mu_r(t)$, $w_\ell^{[0]} = w_\ell(t)$ and $L_\ell^{[0]} = L_\ell(t)$.
- b. for $k = 0$ to d
 - i. if $k = 0$, compute A in (16) and its inverse. Set $(L_\ell^r)^{[0]} = (L_\ell^{[0]})^r$.
 - ii. if $k > 0$, compute $\tilde{\mu}_r^{[k]}$ and solve linear equations in (13) to get $w_\ell^{[k]}$ and $L_\ell^{[k]}$. Compute $(L_\ell^r)^{[k]} = \sum_{i=0}^k L_\ell^{k-i} (L_\ell^r)^{[i]}$.
 - iii. Compute $f_r^{[k]}$ and set $\mu_r^{[k+1]} = \frac{\tau}{k+1} f_r^{[k]}$.
- c. Select h to satisfy (20).
- d. Update μ_r , w_ℓ and L_ℓ using (7).
- e. Set $t = t + \tau h$ and update $\tau = \tau h$. Go to step a.

4. NUMERICAL EXAMPLES

In this section, we compare the efficiencies and accuracies of AD-QMOM and DAE-QMOM using benchmark examples, namely diffusion-controlled growth and volume breakage kernel. For both these examples, the initial distribution is given as

$$n(0, L) = 3L^2 \frac{N_0}{\nu_0} e^{-L^3/\nu_0} \quad (21)$$

with $N_0 = 1 \text{ m}^{-3}$ and $\nu_0 = 1 \text{ m}^3$. The moments at $t = 0$ are obtained by analytically calculating $\mu_r(0)$ for $r = 0, 1, \dots, 2N - 1$, where $N = 6$. The weights w and abscissas L at $t = 0$ are obtained by solving the nonlinear algebraic equations numerically. For DAE-QMOM, the DAE system is solved using Matlab routine *ode15s* with relative and absolute tolerances set to 10^{-12} and 10^{-10} , respectively. For AD-QMOM, we set $d = 20$ and initially, $\tau = 0.15$, while the tolerance level δ is reduced until both the methods have similar percentage errors in μ , which is defined as

$$\% \text{ error} = \frac{\mu_{\text{analytical}} - \mu_{\text{calculated}}}{\mu_{\text{analytical}}} \times 100 \quad (22)$$

All computations are carried out on a Windows XP SP2 notebook with an Intel®Core™ Duo Processor T2500 (2.0 GHz, 2MB L2 Cache, 667 MHz FSB) using MATLAB® R2007b.

4.1 Example 1: Diffusion-controlled growth

We first consider a particulate process with growth only, where the growth rate is given as

$$G = G_0/L \quad (23)$$

with $G_0 = 0.01 \text{ m}^2/\text{s}$. For this process, the moment equations can be derived to be:

$$\frac{d\mu_r}{dt} = f_r(w_\ell, L_\ell, G_0) = rG_0 \sum_{\ell=1}^N w_\ell L_\ell^{r-2} \quad (24)$$

The analytical solution is available only for the even moments and is given as (McGraw, 1997)

$$\mu_0(t) = N_0 \quad (25)$$

$$\mu_2(t) = 2G_0\mu_0(t)t + \mu_2(0) \quad (26)$$

$$\mu_4(t) = 4G_0^2\mu_0(t)t^2 + 4G_0\mu_2(0)t + \mu_4(0) \quad (27)$$

Based on (24) and Table 1, the Taylor coefficients of f_r are derived as follows:

$$f_r^{[k]} = rG_0 \sum_{\ell=1}^N \sum_{j=0}^k w_\ell^{[k-j]} (L_\ell^{r-2})^{[j]} \quad (28)$$

where $(L_\ell^{r-2})^{[j]}$ can be found using Identity 4 in Table 1.

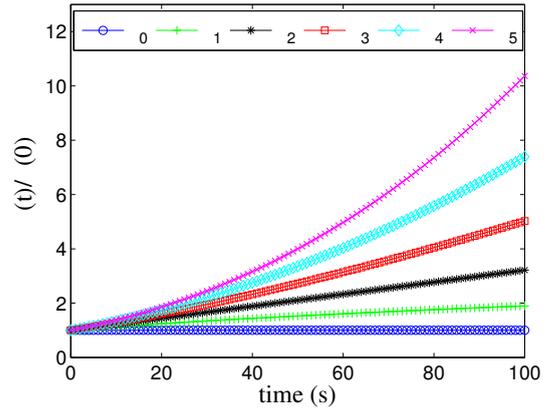


Fig. 1. Evolution of Moments for Example 1

AD-QMOM is applied with $\delta = 10^{-12}$ and the evolution of the moments is shown in Figure 1. The percentage errors for the even moments (for which analytical solution is available) are shown in Figure 2. It can be noted that AD-QMOM provides better accuracy than DAE-QMOM. To solve this problem, the AD-QMOM requires 0.043 sec., while the DAE-QMOM requires 0.477 sec. Thus, the AD-QMOM is able to provide the same level of accuracy as the DAE-QMOM with an order of magnitude smaller solution time. Note that the solution time for AD-QMOM is not very sensitive to the choice of d . For $15 \leq d \leq 25$, the solution time for AD-QMOM differ by only 10%.

4.2 Example 2: Volume Breakage Kernel

Next, we consider a breakage example, where the breakage kernel and probability of breakage are given as

$$a(L) = L^3; \quad b(L, \lambda) = 6L^2/\lambda^3 \quad (29)$$

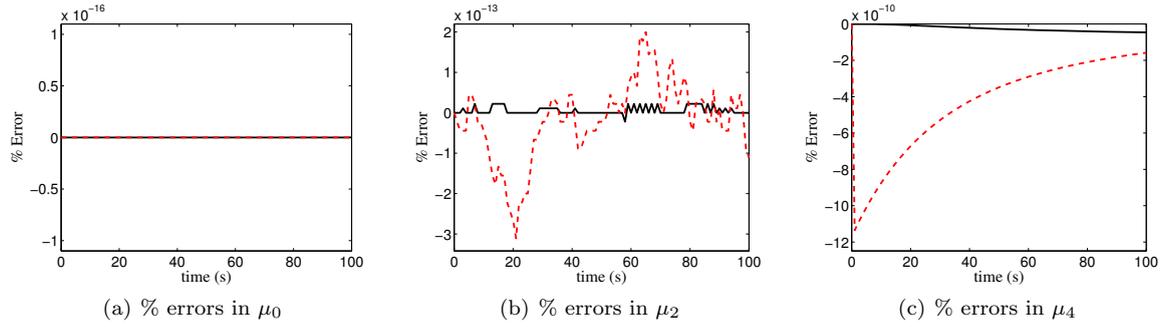


Fig. 2. Comparison of errors for Example 1 using AD-QMOM (continuous lines) and DAE-QMOM (dashed lines)

For $b(L, \lambda)$ in (29), $b(r, L_\ell) = \frac{6L_\ell^r}{r+3}$. Thus, the moment equations can be simplified as

$$\frac{d\mu_r}{dt} = f_r(w_\ell, L_\ell, G_0) = \frac{3-r}{3+r} \sum_{\ell=1}^N w_\ell L_\ell^{r+3} \quad (30)$$

The analytical solution at any time t is given as (Hounslow et al., 2001)

$$n(L, t) = 3L^2(1+t)^2 \exp(-L^3(1+t)) \quad (31)$$

which can be integrated to find the moments analytically.

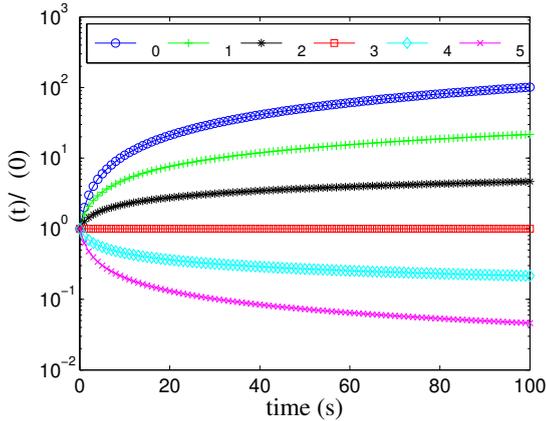


Fig. 3. Evolution of moments for Example 2

For application of the AD-QMOM, the Taylor coefficients of f_r can be found using Table 1 as follows:

$$f_r^{[k]} = \frac{3-r}{3+r} \sum_{\ell=1}^N \sum_{j=0}^k w_\ell^{[k-j]} (L_\ell^{r+3})^{[j]} \quad (32)$$

where $(L_\ell^{r+3})^{[j]}$ can be found using Identity 4 in Table 1.

AD-QMOM is applied with $\delta = 10^{-18}$ and the evolution of the first 6 moments and their percentage errors are shown in Figures 3 and 4, respectively. It can be noted that both the methods conserve μ_3 precisely. The accuracy of the two methods for other moments is nearly identical. To solve this problem, the AD-QMOM requires 0.2 sec., whereas the DAE-QMOM requires 0.936 sec. Thus, the AD-QMOM is able to provide the same level of accuracy as the DAE-QMOM with approximately 5 times smaller solution time.

Interestingly, the errors for both methods are higher than seen for Example 1. The maximum absolute error for the first six moments is 3.675×10^{-3} . When N is increased to 12 with the AD-QMOM (the DAE-QMOM is not able to solve this problem), however, the maximum absolute error for the first 6 moments reduces to 2.754×10^{-6} . This observation highlights that the quadrature approximation is not very good for this example for low values of N . Both methods are still able to keep the numerical error small, however, as expected, have no control over the error arising due to the quadrature approximation.

5. CONCLUSIONS

An automatic differentiation (AD) based approach is proposed to solve population balance equations (PBE) using quadrature method of moments (QMOM). The AD-QMOM uses high-order Taylor expansions to obtain accurate solution of the differential-algebraic equations (DAEs) arising from the quadrature approximation. Examples involving diffusion-controlled growth and volume breakage kernel show that for the same accuracy level, AD-QMOM is 5 – 10 times faster than conventional DAE solvers. Similar results are seen for other phenomena like nucleation and aggregation (not discussed in this paper). The AD-QMOM is also more robust and is able to handle a much higher number of moments than conventional DAE solvers. In summary, this paper establishes AD-QMOM as a promising approach for solving PBEs. The superior efficiency and robustness of AD-QMOM make the approach useful for parameter estimation, process optimization and online model-based control.

ACKNOWLEDGEMENTS

This work was supported by the Royal Society, UK through grant no. IV0871568.

REFERENCES

- Aamir, E., Nagy, Z.K., Rielly, C.D., Kleinert, T., and Judat, B. (2009). Combined quadrature method of moments and method of characteristics approach for efficient solution of population balance models for dynamic modeling and crystal size distribution control of crystallization processes. *Ind. Eng. Chem. Res.*, 48(18), 8575–8584.
- Al-Seyab, R. and Cao, Y. (2008a). Differential recurrent neural network based predictive control. *Comput. Chem. Engng.*, 32(7), 1533–1545.

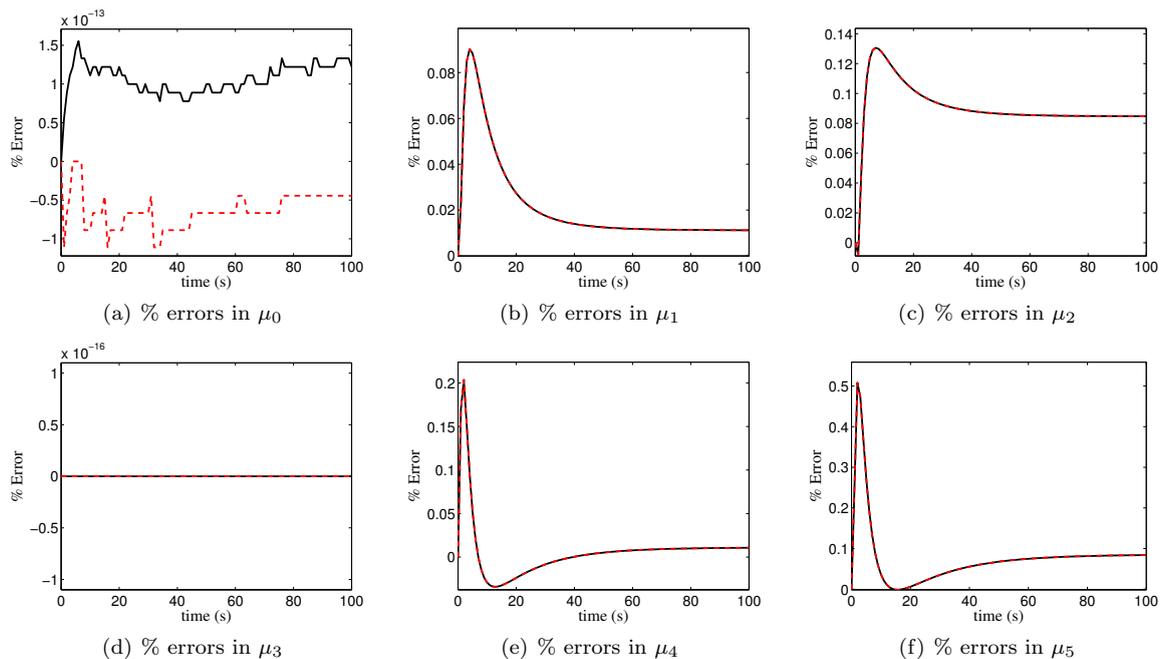


Fig. 4. Comparison of errors for Example 2 using AD-QMOM (continuous line) and DAE-QMOM (dashed line)

- Al-Seyab, R. and Cao, Y. (2008b). Nonlinear system identification for predictive control using continuous time recurrent neural networks and automatic differentiation. *J. Proc. Control*, 18(6), 568–581.
- Alopaeus, V., Laakkonen, M., and Aittamaa, J. (2006). Numerical solution of moment-transformed population balance equation with fixed quadrature points. *Chem. Eng. Sci.*, 61(15), 4919–4929.
- Barrio, R. (2005). Performance of the Taylor series method for ODEs/DAEs. *Applied Mathematics and Computation*, 163, 525–545.
- Cao, Y. (2005). A formulation of nonlinear model predictive control using automatic differentiation. *J. Proc. Control*, 15(8), 851–858.
- Fan, R., Marchisio, D., and Fox, R. (2004). Application of the direct quadrature method of moments to polydisperse gas–solid fluidized beds. *Powder Technol.*, 139(1), 7–20.
- Gimbutu, J., Nagy, Z., and Rielly, C. (2009). Simultaneous quadrature method of moments for the solution of population balance equations, using a differential algebraic equation framework. *Ind. Eng. Chem. Res.*, 48(16), 7798–7812.
- Gordon, R. (1968). Error bounds in equilibrium statistical mechanics. *Journal of Mathematical Physics*, 9, 655.
- Griewank, A. (2000). *Evaluating derivatives: principles and techniques of algorithmic differentiation*. Society for Industrial and Applied Mathematics, Philadelphia, PA.
- Grosch, R., Briesen, H., Marquardt, W., and Wulkow, M. (2006). Generalization and numerical investigation of QMOM. *AIChE J.*, 53(1), 207–227.
- Gunawan, R., Fusman, I., and Braatz, R.D. (2004). High resolution algorithms for multidimensional population balance equations. *AIChE J.*, 50(11), 2738–2749.
- Hounslow, M., Pearson, J., and Instone, T. (2001). Tracer studies of high-shear granulation: II. Population balance modeling. *AIChE J.*, 47(9), 1984–1999.
- Hulburt, H.M. and Katz, S. (1964). Some problems in particle technology: A statistical mechanical formulation. *Chem. Eng. Sci.*, 19(8), 555–574.
- Majumder, A., Kariwala, V., Ansumali, S., and Rajendran, A. (2010). Entropic lattice Boltzmann method for crystallization processes. *Chem. Eng. Sci.*, In press.
- Marchisio, D., Vigil, R., and Fox, R. (2003). Quadrature method of moments for aggregation–breakage processes. *J. colloid interface sci.*, 258(2), 322–334.
- McGraw, R. (1997). Description of aerosol dynamics by the quadrature method of moments. *Aerosol Sci. Technol.*, 27(2), 255–265.
- McGraw, R. and Wright, D. (2003). Chemically resolved aerosol dynamics for internal mixtures by the quadrature method of moments. *J. Aerosol Sci.*, 34(2), 189–209.
- Nicmanis, M. and Hounslow, M. (1998). Finite-element methods for steady-state population balance equations. *AIChE J.*, 44(10), 2258–2272.
- Pryce, J. (1998). Solving high-index DAEs by Taylor series. *Numerical Algorithms*, 19, 195–211.
- Ramkrishna, D. (2000). *Population balances: Theory and applications to particulate systems in engineering*. Academic Press, San Diego, CA.
- Rosner, D., McGraw, R., and Tandon, P. (2003). Multivariate population balances via moment and Monte Carlo simulation methods. *Ind. Eng. Chem. Res.*, 42(12), 2699–2711.
- Rosner, D. and Pyykonen, J. (2002). Bivariate moment simulation of coagulating and sintering nanoparticles in flames. *AIChE J.*, 48(3), 476–491.
- Wright, D., McGraw, R., and Rosner, D. (2001). Bivariate extension of the quadrature method of moments for modeling simultaneous coagulation and sintering of particle populations. *J. colloid interface sci.*, 236(2), 242–251.