Hydrol. Earth Syst. Sci. Discuss., 6, 2415–2449, 2009
www.hydrol-earth-syst-sci-discuss.net/6/2415/2009/

Hydrology and Earth System Sciences Discussions

Papers published in *Hydrology and Earth System Sciences Discussions* are under open-access review for the journal *Hydrology and Earth System Sciences*

A general real-time formulation for multi-rate mass transfer problems

O. Silva[1], J. Carrera[1], S. Kumar[2], M. Dentz[3], A. Alcolea[4], and M. Willmann[5]

[1]Institute of Environmental Assessment and Water Research (IDÆA-CSIC), Barcelona, Spain
[2]Department of Geology and Geophysics, Texas A&M University, USA
[3]Department of Geotechnical Engineering and Geosciences, Technical University of Catalonia (UPC), Barcelona, Spain
[4]Hydrogeology Centre (CHYN), University of Neuchâtel, Neuchâtel, Switzerland
[5]Institute of Environmental Engineering, ETH Zurich, Switzerland

Received: 20 February 2009 – Accepted: 3 March 2009 – Published: 17 March 2009

Correspondence to: O. Silva (orlando.silva@idaea.csic.es)

Published by Copernicus Publications on behalf of the European Geosciences Union.

Abstract

Many flow and transport phenomena, ranging from delayed storage observed in pumping tests to tailing in river or aquifer tracer breakthrough curves, display non-equilibrium behavior. Usually, they are modeled by non-local in time formulations, such as multi-porosity, multiple processes non equilibrium, continuous time random walk, memory functions, integro-differential equations, fractional derivatives or multi-rate mass transfer (MRMT), among others. We develop a MRMT algorithm that can be used to represent all these formulations. The method is accurate, computationally inexpensive and easy to implement in groundwater or river flow and transport codes. In fact, we present a module that can be linked to existing programs with minimal programming effort. Its accuracy is verified by comparison with existing solutions.

1 Introduction

Solving flow and solute transport phenomena in natural media requires using variables, such as heads and concentrations, that characterize the state of the system at every point. Therefore, they are termed state variables. State variables are assumed representative of a small portion of water around such point. This implies that the traditional flow and transport equations implicitly assume local equilibrium.

Even though local equilibrium is assumed by default, non-equilibrium behavior is frequently observed in water flow and solute transport through water bodies. Numerous causes may explain non-equilibrium. In water flow through permeable media, it has been attributed to delayed storage mobilization, either because of resistance at the aquifer free surface (Boulton, 1955; Neuman and Witherspoon, 1971; Neuman and Tartakovsky, 2008), or because of resistance at low permeability blocks (Warren and Root, 1963). It has also been attributed to heterogeneity (Cortis and Knudby, 2006). In solute transport, it has been attributed to diffusion-limited storage into immobile regions, kinetic sorption or heterogeneity (Brusseau et al., 1989; Valocchi, 1990; Sardin

HESSD

6, 2415–2449, 2009

et al., 1991; Cvetkovic et al., 1992; Toride et al., 1993; Haggerty and Gorelick, 1995; Ray et al., 1997; Carrera et al., 1998; Salamon et al., 2006; Vogel et al., 2006; Zhang et al., 2006; Zhang et al., 2007; Alcolea et al., 2008; Willmann et al., 2008; Kumar, 2008; Gouze et al., 2008). Non-equilibrium also has been observed in solute transport through rivers that are influenced by the exchange of water between the river and the underlying hyporheic zone (Fernald et al., 2001; Boano et al., 2007; Marion et al., 2008) or by aggregated dead-zones (Beer and Young, 1983; Lees et al., 1998, 2000; Davis et al., 2000).

Non-equilibrium is typically modeled by non-local in time formulations. Non-local in time means that the mobilization of storage does not depend solely on heads (or concentration) at the current time, but also on their past history. In practice, this implies that a sink-source term on the past history (e.g. Carrera et al., 1998) or that an additional storage term (e.g. Haggerty and Gorelick, 1995) are added to the mass balance equations. It is the form of such terms what sets different non-local formulations apart. The number of non-local formulations is too long to list. The most widely used have been:

- Multi-rate mass transfer (MRMT) (Haggerty and Gorelick, 1995).
- Fractional derivatives (Barker, 1988; Acuna and Yortsos, 1995; Schumer et al., 2003).
- Continuous time random walk (CTRW) (Berkowitz and Scher, 1998; Dentz and Berkowitz, 2003; Benson and Meerschaert, 2009).
- Memory functions (Carrera et al., 1998; Haggerty et al., 2000; Gouze et al., 2008).

Among these, the MRMT formulation is appealing, first, because it is easy to understand. In MRMT formulations, the domain is assumed to consist of a mobile continuum and several overlapping immobile continua. These exchange mass linearly with the mobile region. In this way, the state of immobile zones can be characterized by heads (or concentrations). That is, the traditional single state variable can be viewed

as substituted by a continuum of state variables. Effectively, these work as local state variables, representing a local equilibrium in each of the immobile regions.

A second virtue of MRMT approaches is that they have been proven to be equivalent to other non-local formulations. Haggerty et al. (2000) showed that the MRMT is equivalent to the memory functions approach of Carrera et al. (1998). The latter is also equivalent to the non-local in time version of CTRW (Dentz and Berkowitz, 2003), which can also represent non-local in space terms. The same can be said about fractional derivatives (Berkowitz et al., 2002).

Arguably, the most widely extended form of MRMT is that of Haggerty and Gorelick (1995). They developed the model to account for small-scale variation in rates and types of mass transfer by using a sequence of first-order exchange terms to represent mass transfer with an equal number of immobile zones. Their work and other similar studies have revealed that multiple-porosity models can also be used to solve more general mass-transfer problems and their combinations. For instance, Valocchi (1985) studied the conditions for which local equilibrium is a valid assumption for modeling the transport of sorbing solutes in homogeneous soils. Gerke and van Genuchten (1993a, b) presented a dual-porosity model to simulate transient water flow and solute transport in unsaturated fractured rock formations and structured soils. Sánchez-Vila and Carrera (2004) analyzed the moments of the breakthrough curves in tracer tests and concluded that macrodispersion can be represented by means of mass transfer terms. Wang et al. (2005) developed a direct integration method to solve dual-domain multi-rate mass transfer coupled with advective-dispersive transport. Through their approach they extended the MRMT formulation of Haggerty et al. (2000) to more general, transient flow fields. El-Zein et al. (2005) developed a numerical method to solve mass-transfer problems in intact soils, soils with non-equilibrium sorption and immobile solutions, fissured clayey soils, and structured topsoils, in addition to their combinations. Zhou et al. (2006) proposed a conceptual model of multi-process matrix diffusion in a single fracture consisting in a sequence of three mechanisms: diffusion into stagnant water and infilling materials in fractures, diffusion into a degraded matrix zone adjacent

to fractures, and further diffusion into an intact matrix zone away from fractures. They validated their model with a field tracer test. In the same way, Liu et al. (2007) showed that the dual-domain mass transfer model can reproduce both the near-source peak and the downstream low-concentration spreading when small-scale high-conductivity preferential flow paths are present in a homogeneous porous matrix.

The problem with MRMT approaches, and with most non-local formulations for that matter, is that they are usually implemented to solve a particular numerical application. They are typically solved through Laplace transformation, which facilitates overcoming the nuisances of fractional derivatives or non-locality. These formulations are numerically efficient, but hard to translate to non-linear phenomena, such as multicomponent reactive transport. As a result, these formulations have been typically linked to specific codes, which hinders their general use. Thus, the objective of the present work is to propose an easy numerical implementation of multi-rate mass transfer that is equivalent to other existing formulations, and that also is able to describe other phenomena distinct from solute transport. The present approach is implemented in a Fortran 90 module that can be quite easily embedded into any existing numerical code for flow and solute transport.

2 Governing equations

Non-local in time formulations can be used to enrich the behavior of either the flow or transport equations, or both. In either case, they can be viewed in two complementary fashions: (i) as a continuum of delayed storage terms, in which case these equations represent the total mass balance in both mobile and immobile regions, or (ii) as a continuum of sink/source terms, which act as linear mass exchange terms between mobile and immobile zones. In practice, the continuum is substituted by a discrete number of

terms. Therefore, using the first view, the flow equation becomes

$$S_m \frac{\partial h_m}{\partial t} + \sum_{j=1}^{N} S_{im,j} \frac{\partial h_{im,j}}{\partial t} = -\nabla \cdot \mathbf{q} + q \tag{1}$$

where t (T) is time, h_m (L) is head in the mobile zone, S_m (L^{-1}) is the specific storage coefficient, $\boldsymbol{q}$ (L/T) is water flux and q (T^{-1}) represents a sink/source (recharge/extraction). $h_{im,j}$ (L) and $S_{im,j}$ (L^{-1}) are head and specific storage coefficients of the jth immobile zone, respectively. Water storage in each immobile region is fed by a linear exchange with the mobile domain

$$S_{im,j} \frac{\partial h_{im,j}}{\partial t} = \sigma_{im,j} \frac{K_{im,j}}{L_{im,j}} \left(h_m - h_{im,j} \right) \tag{2}$$

where $\sigma_{im,j}$ (L^2/L^3) is the specific surface of the jth immobile region, $L_{im,j}$ (L) its distance from the mobile zone and $K_{im,j}$ (L/T) its hydraulic conductivity.

Analogously, the solute transport equation expresses the solute mass balance per unit volume of aquifer

$$\phi_m R_m \frac{\partial c_m}{\partial t} + \sum_{j=1}^{N} \phi_{im,j} R_{im,j} \frac{\partial c_{im,j}}{\partial t} = \nabla \cdot (\boldsymbol{D}_m \cdot \nabla c_m) - \boldsymbol{q} \cdot \nabla c_m \tag{3}$$

where c_m (M/L^3) is the mobile concentration, $\boldsymbol{D}_m$ (L^2/T) is the hydrodynamic dispersion tensor, ϕ_m (L^3/L^3) is the mobile porosity (volume of pores per unit aquifer volume), and R_m (–) is the mobile zone retardation factor. Similarly, $c_{im,j}$ (M/L^3), $\phi_{im,j}$ (L^3/L^3) and $R_{im,j}$ (–) are the concentration, porosity and retardation factor of the jth immobile zone. As in flow phenomena, mass balance in the jth immobile region is given by

$$R_{im,j} \phi_{im,j} \frac{\partial c_{im,j}}{\partial t} = \sigma_{im,j} \frac{\phi'_{im,j} D_{im,j}}{L_{im,j}} \left(c_m - c_{im,j} \right) \tag{4}$$

where $\phi'_{im,j}$ (L^3/L^3) is its porosity (volume of pores per unit volume of immobile region), $D_{im,j}$ (L^2/T) is a molecular diffusion coefficient in the jth immobile region. Equation (4) can be somewhat simplified by writing $\phi'_{im,j}$ as a function of $\phi_{im,j}$ (see e.g., Carrera et al., 1998). However, we must bear in mind that the physical meaning of the other parameters in Eqs. (2) and (4) is somewhat approximative. Bearing this in mind, Eqs. (1)–(4) can be written in general as

$$\beta \frac{\partial u_m}{\partial t} + \sum_{j=1}^{N} \beta_j \frac{\partial u_{im,j}}{\partial t} = L_u(u_m) \tag{5}$$

$$\frac{\partial u_{im,j}}{\partial t} = \alpha_j \left(u_m - u_{im,j}\right) \quad j = 1 \ldots N \tag{6}$$

where u=h, for flow or u=c, for solute transport. The β_j (dimensionless) coefficients are called capacity coefficients ($R_{im,j}\phi_{im,j}$ for transport or $S_{im,j}$ for flow) to account for the distribution of mass in the immobile phases; β (dimensionless) is the capacity coefficient of the mobile phase ($R_m\phi_m$ for transport or S_m for flow); and α_j (T^{-1}) is a first-order mass transfer rate coefficient. The right-hand side of Eqs. (1) and (3) is designated generically by the operator L_u.

Denoting F_j the jth term of the sum in Eq. (5), the governing equations for the immobile zones are given as

$$F_j = \beta_j \frac{\partial u_{im,j}}{\partial t} = \beta_j \alpha_j (u_m - u_{im,j}) \quad j = 1 \ldots N \tag{7a}$$

$$F = \sum_{j=1}^{N} F_j \tag{7b}$$

HESSD
6, 2415–2449, 2009

3 General formulation for MRMT

Spatial and time discretization of either the flow or transport equations under local equilibrium assumptions (i.e., without MRMT) leads to a linear system of equations (e.g., Medina and Carrera, 1996)

$$\mathbf{D}\frac{\Delta \boldsymbol{u}_m}{\Delta t} + \mathbf{A}\boldsymbol{u}_m^{k+\theta} = \boldsymbol{b}^{k+\theta} \tag{8}$$

where $\Delta \boldsymbol{u}_m = \boldsymbol{u}_m^{k+1} - \boldsymbol{u}_m^k$, $\Delta t = t^{k+1} - t^k$ is the time step, θ is a weighting factor and the superscripts stand for the time in which the variable is evaluated.

Accounting for MRMT can be achieved in two ways: (a) using an appropriate mesh with nodes representing the immobile zones (e.g., Neuman, 1982), or (b) by eliminating the unknown in the immobile region as an explicit state variable, i.e. expressing $u_{im,j}$ as a function of u_m (e.g., Carrera et al., 1998). Here, we have adopted the later approach because: first, it maintains the number of unknowns unchanged and, second, it is actually simpler to implement into existing generic flow and transport simulation codes.

Figure 1 displays a schematic representation of a hypothetical numerical mesh that includes both the mobile and immobile domains. We assume that each node m of the mobile zone is connected to all adjacent nodes of the mesh and to all the immobile blocks. Node im,j of the immobile region is only connected to node m. Geometrically, node im,j overlaps with node m. We show below that the variable u at node im,j (i.e., $u_{im,j}$) can be solved explicitly as a function of u_m. Therefore, node im,j need not be an "uncertain" node, but can be considered as a zero-D node.

We first solve the N first-order ordinary differential Eq. (6) in terms of Δu_m, while assuming that u_m varies linearly during each time increment. That is, $u_m = u_m^k + (\Delta u_m/\Delta t)(t - t^k)$. This leads to N first-order linear differential equations, whose solution is

$$u_{im,j}(t) = u_{im,j}^k e^{-\alpha_j(t-t^k)} + u_m^k\left(1 - e^{-\alpha_j(t-t^k)}\right) + \frac{\Delta u_m}{\Delta t}\left[(t - t^k) - \frac{1}{\alpha_j}\left(1 - e^{-\alpha_j(t-t^k)}\right)\right] \tag{9}$$

HESSD
6, 2415–2449, 2009

Combining Eqs. (6), (7a) and (9), the flux F_j evaluated at time $t^{k+\theta}$ will be

$$F_j^{k+\theta} = \beta_j \alpha_j \left(u_m^{k+\theta} - u_{im,j}^{k+\theta} \right)$$
$$= \beta_j \alpha_j \left(u_m^k - u_{im,j}^k \right) e^{-\alpha_j \theta \Delta t} + \frac{\Delta u_m}{\Delta t} \beta_j \left(1 - e^{-\alpha_j \theta \Delta t} \right) \quad (10)$$

Notice that this flux is only a function of u at the previous time step and Δu_m. The total mass flux, $F^{k+\theta}$, is given by Eq. (7b). Substituting the resulting expression into Eq. (8) leads to an identical system, except that the storage matrix and sink/source term are modified according to

$$(\mathbf{D}^*)_{ii} = (\mathbf{D})_{ii} + v_i \sum_{j=1}^{N} \beta_j \left(1 - e^{-\alpha_j \theta \Delta t} \right) \quad (11a)$$

$$(\boldsymbol{b}^*)_i^{k+\theta} = (\boldsymbol{b})_i^{k+\theta} - v_i \sum_{j=1}^{N} \beta_j \alpha_j \left[u_{m,i}^k - \left(u_{im,j}^k \right)_i \right] e^{-\alpha_j \theta \Delta t} \quad (11b)$$

where $u_{m,i}^k$ is the value of u at node i of the mobile region and time t^k, and $\left(u_{im,j}^k \right)_i$ the corresponding value in the jth immobile block, and v_i is the volume of cell i in volume integrated formulations (e.g., finite element) and is equal to 1 in discretized formulations (e.g., finite differences). Finally, it is necessary to update $u_{im,j}$ at the end of each time step using Eq. (9). This approach is quite simple to program and should lead to accurate solutions at a very low computational cost. As with the integro-differential approach, the number of nodes/elements is not altered by the addition of the MRMT terms.

4 Equivalence with other similar approaches

As mentioned in the introduction, a large number of non-local in time schemes have been presented by different authors (Beer and Young, 1983; Sudicky, 1989; Haggerty and Gorelick, 1995; Carrera et al., 1998; Lees et al., 2000; Fernald et al., 2001; Dentz and Berkowitz, 2003; Schumer et al., 2003; Boano et al., 2007; Marion et al., 2008). This section is devoted to discussing such equivalence so as to facilitate using them in the proposed formulation. Haggerty et al. (2000) provided a comparison table with different MRMT formulations considering governing equations similar to Eqs. (5) and (6). The present approach is essentially identical to that of Haggerty and Gorelick (1995). The main difference is that they formulated their equations per unit volume of water. Therefore, their capacity coefficients are equal to the coefficients β_j of Eqs. (5) and (6), but divided by the mobile capacity, β. Denoting β_{jHG} and α_{jHG} the capacity and first-order mass transfer coefficients considered by Haggerty and Gorelick (1995), we have the following equivalence relationship

$$\beta_{jHG} = \beta_j/\beta \tag{12a}$$

$$\alpha_{jHG} = \alpha_j \tag{12b}$$

We have preferred to use capacity coefficients as defined in Eq. (5) to keep the physical meaning and consistence of the governing equations as mass balances per unit volume of aquifer.

Many schemes approximate the effect of the immobile region by a continuous memory function. The governing equations are then solved in the Laplace domain. These solutions can be approximated by expanding the memory function as a sum of exponentials (Carrera et al., 1998). Each summand can then be solved as explained in Sect. 3. For the purposes of comparison with our approach, the important issue is to acknowledge that such approaches are typically defined in terms of overall parameters

for the whole immobile region (as opposed to independent α_j and β_j). For the works of Carrera et al. (1998) and Salamon et al. (2006), the equivalence is given by:

$$\alpha_j = \gamma_j^2 \frac{D_{im}}{R_{im} L_{im}^2} \tag{13a}$$

$$\beta_j = \frac{a_j}{\gamma_j^2} R_{im} \phi_{im} \tag{13b}$$

where ϕ_{im} (L^3/L^3), R_{im}, D_{im} (L^2/T) and L_{im} (L) are characteristic parameters of the entire immobile domain. Coefficients a_j and γ_j can be found in the literature (e.g., Haggerty and Gorelick, 1995; Carrera et al., 1998; Haggerty et al., 2000; Salamon et al., 2006) for diffusion into different geometries (layered, cylindrical, spherical and veins) and the standard first-order model. These formulations result from the analytical solution of the diffusion equation. The coefficients in Eqs. (13) result from an infinite series expansion that needs to be truncated. Salamon et al. (2006) provide a table with the term required to cope with the truncation error. A large number of first-order mass transfer rate coefficients and their distributions estimated from field and laboratory test results can be obtained from the works of Cosler (2004) and Haggerty et al. (2004).

The mass flux, F, into the immobile region in memory function based approaches is given by

$$F(x,t) = \int_0^t g(t-\tau) \frac{\partial u_m(x,\tau)}{\partial \tau} d\tau = g * \frac{\partial u_m}{\partial t} + g(t) u_m(x,0) \tag{14}$$

where g is the memory function and $*$ denotes the convolution product. Carrera et al. (1998) approximate this product using the integro-differential approach of Herrera and Rodarte (1973) and Herrera and Yates (1977). An equivalent alternative is to

HESSD
6, 2415–2449, 2009

approximate g by

$$g(t) = \int_0^\infty \alpha b(\alpha) e^{-\alpha t} d\alpha \tag{15}$$

where $b(\alpha)$ (T) is a density function of first-order rate coefficients. Haggerty et al. (2000) provide explicit expressions for the density and memory functions for various models or geometries. To use the approach of Sect. 3, we need to express the memory function as

$$g(t) = \sum_{j=1}^{\infty} \alpha_j \beta_j e^{-\alpha_j t} \tag{16}$$

Note that Haggerty et al. (2000) included the factor $\alpha_j \beta_j$ on the memory function, unlike Carrera et al. (1998) who placed it on flux F. However, both approaches are equivalent. We calculate the convolution product in Eq. (14), truncating the memory function at Nth term and following the same algebraic analysis described in the Appendix 1 of Carrera et al. (1998). Thus, we can express $F^{k+\theta}$ as

$$F^{k+\theta} = \sum_{j=1}^{N} \beta_j \alpha_j e^{-\alpha_j \theta \Delta t} I_j^k + \frac{\Delta u_m}{\Delta t} \sum_{j=1}^{N} \beta_j \left(1 - e^{-\alpha_j \theta \Delta t}\right) \tag{17a}$$

$$I_j^{k+1} = \int_0^{t^{k+1}} e^{-\alpha_j \left(t^{k+1} - \tau\right)} \frac{\partial u_m}{\partial \tau} d\tau = e^{-\alpha_j \Delta t} I_j^k + \frac{\left(1 - e^{-\alpha_j \Delta t}\right)}{\alpha_j} \frac{\Delta u_m}{\Delta t} \tag{17b}$$

The equivalence between our approach and integro-differential approach becomes evident by comparing Eqs. (17a) and (10). Also note that, from Eq. (9) we obtain the recursive relationship

$$u_m^{k+1} - u_{im,j}^{k+1} = e^{-\alpha_j \Delta t} \left(u_m^k - u_{im,j}^k\right) + \frac{\left(1 - e^{-\alpha_j \Delta t}\right)}{\alpha_j} \frac{\Delta u_m}{\Delta t} \tag{18}$$

HESSD
6, 2415–2449, 2009

which is similar to Eq. (17b). Therefore, we arrive at Eqs. (7b) and (10) by imposing the condition

$$I_j^k = u_m^k - u_{im,j}^k \tag{19}$$

Truncation criteria and expression for the final terms of truncated multi-rate series can be found in Haggerty and Gorelick (1995) and Salamon et al. (2006). In the case of diffusion into different geometries, they proposed the same criteria to evaluate β_N. However, while Haggerty and Gorelick (1995) suggested writing α_N as the rest of the α_j coefficients (i.e., Eq. 13a), Salamon et al. (2006) proposed the following expressions

$$\beta_N = \left(1 - \sum_{j=1}^{N-1} \frac{a_j}{\gamma_j^2}\right) R_{im}\phi_{im} \tag{20a}$$

$$\alpha_N = \lambda \frac{\left(1 - \sum_{j=1}^{N-1} \frac{a_j}{\gamma_j^2}\right)}{\left(1 - \lambda \sum_{j=1}^{N-1} \frac{a_j}{\gamma_j^4}\right)} \frac{D_{im}}{R_{im} L_{im}^2} \tag{20b}$$

where λ for layers, spheres, cylinders are given by Salamon et al. (2006).

Dentz and Berkowitz (2003) found a mathematical equivalence between MRMT and the CTRW model (Berkowitz and Scher, 1998; Berkowitz et al., 2006; Margolin et al., 2003; Salamon et al., 2006; Benson and Meerschaert, 2009). They formulated a CTRW approach which is formally equivalent to the integro-differential formulation of MRMT presented in this paper. They present a map between the memory function defined in the context of MRMT and the transition time distribution $\psi(t)$

$$g^*(s) = \frac{1 + \psi^*(s)(1 + s\tau_0)}{s\tau_0 \psi^*(s)}, \tag{21}$$

where τ_0 defines which part of the medium is mobile or immobile and as such is related to the mobile and immobile volume fractions of the medium (see Dentz and Berkowitz, 2003). The Laplace transform of the memory function, g^*, can be expanded into a series in s according to

$$g^*(s) = \sum_{k=1}^{\infty} (-1)^k a_k s^k, \tag{22}$$

where explicit expressions for the a_k are given in Dentz and Berkowitz (2003). For $g^*(s)$ given by the Laplace transform of Eq. (16), we obtain

$$g^*(s) = \sum_{j=1}^{\infty} \frac{\alpha_j \beta_j}{\alpha_j + s} = \sum_{j=1}^{\infty} \beta_j \sum_{k=1}^{\infty} (-1)^k \alpha_j^{-k} s^k = \sum_{k=1}^{\infty} (-1)^k s^k \left[\sum_{j=1}^{\infty} \alpha_j^{-k} \beta_j \right] \tag{23}$$

By comparison of Eqs. (22) and (23), we obtain relations between the β_j and the a_k for a given series of rates α_j

$$a_k = \left[\sum_{j=1}^{\infty} \alpha_j^{-k} \beta_j \right] \tag{24}$$

The latter expression can be inverted (numerically) in order to obtain explicit expressions for the weights β_j and thus for the memory function $g(t)$ that simulates the transport behavior in a CTRW.

Dentz and Berkowitz (2003) also proposed the use of the truncated power law memory function, which has become widely used because breakthrough curves often display a power law behavior at late times (see, e.g., Zhang et al., 2007; Willmann et al., 2008). The late time behavior of the breakthrough curve can be related to the memory function (Haggerty et al., 2000). This memory function only requires specifying the slope of the memory function in log-log scale, m_g, and the interval of time (t_1, t_N) on which this function displays a power-law behavior. A practical method to calculate the

HESSD
6, 2415–2449, 2009

distribution coefficients β_j consists of, first, calculate the α_j values assuming they are evenly distributed on a logarithmic scale while fixing $\alpha_1=t_N^{-1}$ and $\alpha_N=t_1^{-1}$. Secondly, we obtain a recursive relationship for β_j values by approximating the memory function with expressions of successive increasing orders, i.e.

$$\log\left(\sum_{i=j}^{N}\beta_i\alpha_i\right)-\log\left(\sum_{i=j+1}^{N}\beta_i\alpha_i\right)=m_g(\log t_j-\log t_{j+1}) \tag{25}$$

where $t_j=\alpha_j^{-1}$. This leads to

$$\beta_j=\frac{\sum_{i=j+1}^{N}\beta_i\alpha_i}{\alpha_j}\left[\left(\frac{\alpha_j}{\alpha_{j+1}}\right)^{m_g}-1\right]\quad j=1\ldots N-1 \tag{26}$$

To get the values of β_j, we first assign an arbitrary value to β_N (e.g., $\beta_N=1$). Then we apply Eq. (26) and finally scale these values imposing the condition $\sum_{j=1}^{N}\beta_j=1$.

5 Numerical implementation

5.1 Module structure, main attributes and subroutines

The equations described in the previous sections have been implemented in a Fortran 90 module called `mod_process_MRMT.f90`, which is structured following the coding guidelines and rules proposed by Slooten et al. (2008). The module defines MRMT objects by means of type `t_immobile` and provides services to solve the equation described here. The main attributes of the module contained in the type `t_immobile` are detailed in Fig. 2. The main services of `mod_process_MRMT.f90` are described in Table 1. Auxiliary subroutines, private arguments and attributes are commented within `mod_process_MRMT.f90`.

5.2 Input files

Input data of `mod_process_MRMT.f90` are entered in XML format. Reading of input files is achieved by the use of the modules `flib_sax.f90` and `flib_xpath.f90`. These modules were obtained from García (2004) and basically are libraries to handle the reading of xml files. These libraries depends on other Fortran modules, which are stored in folders `xmlreader` and `xpath`, respectively. Based on the mathematical equivalence of the different approaches explained in Sect. 4, four types of input files are available: `Parameters.xml`, `Coefficients.xml`, `ExpansionSeries.xml`, and `PowerLaw.xml`.

5.3 Embedding `mod_process_MRMT.f90` into a standard numerical code

The structure and arguments of `mod_process_MRMT.f90` were designed so that state variables and parameters characterizing the immobile domain are only accessible from the module. This guarantees a minimal information exchange with the program units of any host code to which the user would hope to link `mod_process_MRMT.f90`. That feature helps to get a straightforward implementation of the present MRMT approach on a general advection-dispersion transport simulator programmed in Fortran, as shown in Fig. 3. We assume that the user (programmer) has got any standard numerical code for flow and solute transport that will be modified to include the multi-rate mass transfer equations described in Sects. 2 and 3, according to the following steps:

- Add the file `mod_process_MRMT.f90` and folders `xmlreader` and `xpath` containing all the `XML` supporting material into the source files of the host code. Also copy the `XML` input file to the work directory, depending on the user's application and options given in Sect. 5.2.

- Include the "`Use mod_process_MRMT`" statement into the source code and declare as many variables of type `t_immobile` as required by the user's application. We suggest an `optional` declaration because the user might want to use logical

variables indicating whether MRMT is or not included in the simulations. Therefore, any variable declared as `t_immobile` that could be an argument of host code subroutines will be properly passed whether they are or not present. To create these variables, immediately after declaration in the main transport code include as many callings to the `Create_` subroutine as `t_immobile` variables have been declared. Initialization also requires reading input files in the corresponding section of the standard transport code, so include a calling to the `Read_XML_` subroutine to read the mass-transfer parameters of each immobile zone. After initialization in the host code one has to initialize all variables of `type t_immobile`, by calling the subroutine `Initialize_`.

– After calling the host code subroutines, modules or program units that construct the system matrices **A**, **D** and **b** (Eq. 8), one has to modify **D** and **b** according to Eqs. (11a and 11b). This action is shown in Fig. 3 for the case of including MRMT into the solute transport problem. The subroutine `ContriToMatrices_` calculates de contribution of MRMT to **D** according to Eq. (11a). For each variable declared as `t_immobile`, the user has to include a calling statements to `ContriToMatrices_` and `ContriToSink_` subroutines. The user will need to add the corrections due to MRMT. For instance

```
DO I=1,NUMNP
  D(I) = D(I) + Dcorr(I)
  B(I) = B(I) + Bcorr(I)
END DO
```

where `D` and `B` would be arrays storing the matrices elements of system (8), `Dcorr` and `Bcorr` arrays storing the corrections given by Eqs. (11a and 11b), and `NUMNP` an integer variable representing the number of mobile nodes. Note that subroutines `ContriToMatrices_` and `ContriToSink_` must be called at every time step. They can be called within the subroutines that form matrices **D**

HESSD
6, 2415–2449, 2009

and **b** or after them, following the structure of the host code and user's preferences, but before execution of the unit program that solve the system of Eqs. (8).

– After solving a time step of the transport problem (Fig. 3), the state variable of the immobile domain, $u_{im,j}$, must be updated according to Eq. (9). This is done by calling the subroutine `UpdateConc_`. If required, the total amount and the average value of u in the immobile domains can be computed calling subroutine `TotSolMass_`. Note that those quantities are saved within the internal structure of the variable `ImmbReg`. However, they can be obtained by calling the subroutine `GetSolMass_`. Output subroutines are available to follow the evolution of the state variable u at immobile regions (`WriteConc_`), as well as the total and average of u (`WriteSolMass_`). If required, these subroutines can be called within the program unit for writing output results.

– Finally, it may be necessary and advisable to deallocate all the attributes of all `type t_immobile` variables used in the simulation. This can be achieved by including a call to subroutine `Destroy_` for each of these variables at the end of the transport simulation.

The module supports some characteristics of object-oriented programming. In fact, the module was designed such that its types and operations are available from outside but the details of the implementation are hidden from the user, i.e., the "black box" principle (Gorelik, 2004). Moreover, the present module lies within the definition of objects belonging to the "Process" class of the object-oriented framework tool PROOST (Slooten et al., 2008). These functionalities would also permit to link other Fortran modules or objects to the present module. For instance, reactive transport may be included in both the mobile and immobile regions, by properly linking the object-oriented tool CHEPROO (Bea et al., 2009) to both any conservative transport code and the module `mod_process_MRMT.f90`.

`mod_process_MRMT.f90` can be downloaded from http://www.h2ogeo.upc.es/English/English/software.htm#Mod_process_MRMT.

6 Applications

In order to assess the accuracy of the present approach, we embedded the module into TRACONF (Carrera et al., 1993), a Fortran program for the simulation of water flow and solute transport through porous media. The testing exercise is divided into three parts. We first compare the present formulation with an analytical solution for delayed yield from storage (Boulton, 1955), to test its applicability to flow problem. Next, we compare our approach with the integro-differential approach, simulating the hypothetical radially convergent tracer test described by Alcolea et al. (2001). Finally, we apply the present approach to solve two problems of radial flow to a pumping well (Haggerty and Gorelick, 1995).

6.1 Verification for a flow problem

Boulton (1955) developed an analytical solution for unsteady radial flow allowing delayed yield from storage. This problem can be described by Eq. (1) with N=1, assuming q=0. We model a hypothetical pumping test, considering a transmissivity of T=0.01 m^2/s, S_m=0.001 and S_{im}=0.1, and a pumping rate of 0.04 π m^3/s. Initial head equals zero. We compare our non-local in time approach with the Boulton's solution for three values of the rate coefficient α=2.5×10^{-6}, 10^{-5} and 5×10^{-5} s^{-1}. Boulton (1955) referred to α as an empirical constant. We considered a mesh of 208 nodes with a spacing size Δr increasing geometrically with a factor of 1.08. The integration scheme in time was semi-implicit with variable Δt.

Figure 4 displays the evolution of heads at a distance of R=51.6 m from the well. The dimensionless time t_D=Tt/S_mR^2 was used. We can see that the numerical solution (solid line) obtained with the present approach matches the analytical solution (circles) obtained by Boulton (1955). The figure shows clearly the influence of the rate coefficient α on the system behavior. As α decreases, the system approaches the case in which there is no delayed yield. Again, as α increases the system behaves as it is constituted only by one domain under a mobile storage dominated regime. This is ex-

pected because for very large values of α, the water mass transfer between mobile and immobile zones occurs nearly instantaneously: mobile and immobile domains tend to be at equilibrium.

6.2 Verification for a transport problem

We model the convergent tracer test described by Alcolea et al. (2001). A tracer mass of 7.88 g is injected 8 m away from a well pumping 150 m^3/d. The radius of the pumping well and the aquifer thickness are 0.2 m and 5 m, respectively. Porosity of the mobile domain is ϕ_m=0.1. The immobile zone consists of layers of length L_{im}=0.05 m and porosity ϕ_{im}=0.045. The diffusion coefficient in the immobile zone was set to D_{im}=0.001 m^2/d. TRANSIN code uses the integro-differential approach to solve matrix diffusion problems. Accordingly, we have chosen the ExpansionSeries option of our module, which used Eq. (13a and b) to define α's and β's with N=50. A uniform mesh with a grid size Δr=0.005 m (93 nodes) and a fully implicit integration scheme in time was used in the simulations.

Figure 5 shows the breakthrough curves at the pumping well simulated with the present approach and with TRANSIN, which compare quite well. The maximum relative error was 1%.

6.3 Verification for a field problem

Haggerty and Gorelick (1995) presented a case of radial flow to a pumping well, in the context of PCE removal from the Borden sand aquifer under realistic pumping rates. To solve the governing equations, they expressed the MRMT model in dimensionless form and used a semianalytic method. Here we only give the main characteristics concerning with our approach, as the specifications of the problem are well described in their work (Haggerty and Gorelick, 1995). Two hypothetical case studies were considered: the remediation of a homogeneous aquifer (Borden sand) and the remediation of a hypothetical heterogeneous aquifer with a mixture of mass transfer processes. In

HESSD
6, 2415–2449, 2009

both cases we used a mesh of 102 nodes with a spacing size Δr decreasing in such a way that all cells have the same volume and $\Delta r_1 = 1.39$ m. The integration scheme in time was semi-implicit with variable Δt.

The first case study simulates the cleanup history of a homogeneous aquifer where seven immobile zones are associated to a distribution of grain sizes. The grains are assumed to be spherical; the α_j and β_j coefficients are given in Table 2 of Haggerty and Gorelick (1995). As in their work, we have considered 50 exponential terms to adequately describe each immobile domain. Figure 6a displays the evolution of the PCE mass fraction remaining through an scenario of 500 d. We can see that the numerical solution (solid line) obtained with the present approach match very well the semi-analytical solution (dashed line) obtained by Haggerty and Gorelick (1995).

In the second numerical experiment, we simulate the removal of PCE from an heterogeneous aquifer. Here, heterogeneity arises from four immobile zones of different geometry (porous grains, grain aggregates, clay layers and clay pods) and two immobile zones characterized by a surface reaction (slow and fast reactions). The mass-transfer parameters are those appearing in Table 3 of Haggerty and Gorelick (1995). Note that these parameters are assumed locally heterogeneous, but they have the same distribution at all points in space. Once again, 50 terms were used to describe each geometry and only a single term for each reaction. Figure 6b shows predictions of the mass fraction of PCE remaining in the aquifer during a remediation scenario of 20 000 d. Again, the evolution of the mass fraction remaining obtained through the present approach (solid line) fit well the results predicted by the semi-analytical solution (dashed line). Therefore, the present model can reproduce the behavior of heterogeneous media characterized by different types and rates of mass transfer.

7 Conclusions

We have developed an easy-to-use numerical implementation of multi-rate mass transfer, which embeds existing formulations for MRMT. We developed a simple numerical

method, which is physically consistent with and mathematically equivalent to other formulations such as integro-differential formulation. The present approach avoids the spatial discretization of the immobile domain, because it solves state variables of that zone as explicit function of the state variables in the mobile domain. The numerical method is accurate, as it involves an analytical solution for the mass balance equations at immobile zones. We also have implemented the method into a Fortran 90 module that can be easily embedded into standard numerical codes for flow and solute transport to model multi-rate mass transfer. Portability of the module has been achieved because of its object-oriented structure. The module has been tested by comparison with published solutions and is publicly available at http://www.h2ogeo.upc.es/English/English/software.htm#Mod_process_MRMT.

Acknowledgement. We gratefully acknowledge the financial support received from CICYT and ENRESA of Spain, and EC projects FUNMIG (contract number 516514) and MUSTANG (contract number 227286). We also acknowledge the support received from the MICINN and the European Social Fund.

References

Acuna, J. A. and Yortsos, Y. C.: Application of fractal geometry to the study of networks of fractures and their pressure transient, Water Resour. Res., 31(3), 527–540, 1995.

Alcolea, A. and Carrera, J.: Difusión en la matriz. Formulación integro-diferencial, in: VII Simposio de Hidrogeología, Hidrogeología y Recursos Hidráulicos 24, Asociación española de hidrogeólogos, Murcia, Spain, 703–714, 2001.

Alcolea, A., Carrera, J., and Medina, A.: Regularized pilot points method for reproducing the effect of small scale variability: Application to simulations of contaminant transport, J. Hydrol., in press, doi:10.1016/j.jhydrol.2008.03.004, 2008.

Barker, J. A.: A generalized radial flow model for hydraulic tests in fractured rock, Water Resour. Res., 24(11), 1796–1804, 1988.

Bea, S. A., Carrera, J., Ayora, C., Batlle, F., and Saaltink, M. W.: CHEPROO: A Fortran 90 object-oriented module to solve chemical processes in Earth Science models, Comput. Geosci., doi:10.1016/j.cageo.2008.08.010, in press, 2009.

Beer, T. and Young, P. C.: Longitudinal dispersion in natural streams, J. Environ. Eng.-ASCE, 109(5), 1049–1067, 1983.
Benson, D. and Meerschaert, M. M.: A simple and efficient random walk solution of multi-rate mobile/immobile mass transport equations, Adv. Water Resour., doi:10.1016/j.advwatres.2009.01.002, in press, 2009.
Berkowitz, B. and Scher, H.: Theory of anomalous chemical transport in random fracture networks, Phys. Rev. E, 57(5), 5858–5869, 1998.
Berkowitz, B., Klafter, J., Metzler, R., and Scher, H.: Physical pictures of transport in heterogeneous media: advection-dispersion, random-walk, and fractional derivative formulations, Water Resour. Res., 38(11), 1191, doi:10.1029/2001WR001030, 2002.
Berkowitz, B., Cortis, A., Dentz, M., and Scher, H.: Modeling non-Fickian transport in geological formations as a continuous time random walk, Rev. Geophys., 44, RG2003, doi:10.1029/2005RG000178, 2006.
Boano, F., Packman, A. I., Cortis, A., Revelli, R., and Rodolfi, L.: A continuous time random walk approach to the stream transport of solutes, Water Resour. Res., 43, W10425, doi:10.1029/2007WR006062, 2007.
Boulton, N. S.: Unsteady radial flow to a pumped well allowing for delayed yield from storage, in: Publication N°37 de l'Association Internationale d'Hydrologie, Assemblée générale de Rome. Tome II, 472–477, 1955.
Brusseau, M. L., Jessup, R. E., and Rao, P. S. C.: Modeling the transport of solutes influenced by multiprocess nonequilibrium, Water Resour. Res., 25(10), 1971–1988, 1989.
Carrera, J., Galarza, G., and Medina, A.: TRACONF, Programa de elementos finitos para la solución de las ecuaciones de flujo y transporte en acuíferos confinados, E. T. S.I. Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Barcelona, 53 pp., 1993.
Carrera, J., Sánchez-Vila, X., Benet, I., Medina, A., Galarza, G., and Guimerà, J.: On matrix diffusion: formulations, solution methods and qualitative effects, Hydrogeol. J., 6, 178–190, 1998.
Cortis, A. and Knudby, C.: A continuous time random walk approach to transient flow in heterogeneous media, Water Resour. Res., 42, W10201, doi:10.1029/2006WR005227, 2006.
Cosler, D. J.: Effects of rate-limited mass transfer on water sampling with partially penetrating wells, Ground Water, 42(2), 203–222, 2004.
Cvetkovic, V., Shapiro, A., and Dagan, G.: A solute flux approach to transport in heterogeneous formations 2. Uncertainty analysis, Water Resour. Res., 28(5), 1377–1388, 1992.

Davis, P. M, Atkinson, T. C., and Wigley, T. M. L.: Longitudinal dispersion in natural channels: 2. The role of shear flow dispersion and dead zones in the River Severn, UK, Hydrol. Earth Syst. Sci, 4, 355–371, 2000.

Dentz, M. and Berkowitz, B.: Transport behavior of a passive solute in continuous time random walks and multirate mass transfer, Water Resour. Res., 39(5), 1111, doi:10.1029/2001WR001163, 2003.

El-Zein, A. H., Carter, J. P., and Airey, D. W.: Multiple-porosity contaminant transport by finite-element method, Int. J. Geomech., 5(1), 24–34, 2005.

Fernald, A. G., Wigington Jr., P. J., and Landers, D.: Transient storage and hyporheic flow along the Willamette River, Oregon: Field measurements and model estimates, Water Resour. Res., 37(6), 1681–1694, 2001.

García, A.: Xmlf90: A parser for XML in Fortran90, http://lcdx00.wm.lc.ehu.es/ag/xml/index.html, 2004.

Gerke, H. H. and van Genuchten, M. T.: A dual-porosity model for simulating the preferential movement of water and solutes in structured media, Water Resour. Res., 29(2), 305–319, 1993a.

Gerke, H. H. and van Genuchten, M. T.: Evaluation of a first-order water transfer term for variably saturated dual-porosity flow models, Water Resour. Res., 29(4), 1225–1238, 1993b.

Gorelik, A. M.: Object-oriented programming in modern Fortran, Program. Comput. Softw., 30(3), 173–179, 2004.

Gouze, P., Melean, Y., Le Borgne, T., Dentz, M., and Carrera, J.: Non-Fickian dispersion in porous media explained by heterogeneous microscale matrix diffusion, Water Resour. Res., 44, W11416, doi:10.1029/2007WR006690, 2008.

Haggerty, R. and Gorelick, S. M.: Multiple-rate mass transfer for modeling diffusion and surface reactions in media with pore-scale heterogeneity, Water Resour. Res., 31(11), 2383–2400, 1995.

Haggerty, R., McKenna, S. A., and Meigs, L. C.: On the late-time behavior of tracer test breakthrough curves, Water Resour. Res., 36(12), 3467–3479, 2000.

Haggerty, R., Harvey, C. F., von Schwerin, C. F., and Meigs, L.: What controls the apparent timescale of solute mass transfer in aquifers and soils? A comparison of experimental results, Water Resour. Res., 40, W01510, doi:10.1029/2002WR001716, 2004.

Herrera, I. and Rodarte, L.: Integrodifferential equations for system of leaky aquifers and applications. 1. The nature of approximate theories, Water Resour. Res., 9(4), 995–1005, 1973.

Herrera, I. and Yates, R.: Integrodifferential equations for system of leaky aquifers and applications. 3. A numerical method of unlimited applicability, Water Resour. Res., 13(4), 725–732, 1977.

Kumar, S.: Effect of sorption intensities on dispersivity and macro-dispersion coefficient in a single fracture with matrix diffusion, Hydrogeol. J., 16, 235–249, 2008.

Lees, M. J., Camacho, L., and Whitehead, P.: Extension of the QUASAR river water quality model to incorporate dead-zone mixing, Hydrol. Earth Syst. Sci., 2, 353–365, 1998, http://www.hydrol-earth-syst-sci.net/2/353/1998/.

Lees, M. J., Camacho, L. A., and Chapra, S.: On the relationship of transient storage and aggregated dead zone models of longitudinal solute transport in streams, Water Resour. Res., 36(1), 213–224, 2000.

Liu, G., Zheng, C., and Gorelick, S. M.: Evaluation of the applicability of the dual-domain mass transfer model in porous media containing connected high-conductivity channels, Water Resour. Res., 43, 725–732, 2007.

Margolin, G., Dentz, M., and Berkowitz, B.: Continuous time random walk and multirate mass transfer modeling of sorption, Chem. Phys., 295(1), 71–80, 2003.

Marion, A., Zaramella, M., and Bottacin-Busolin, A.: Solute transport in rivers with multiple storage zones: the STIR model, Water Resour. Res., 44, W10406, doi:10.1029/2008WR007037, 2008.

Medina, A. and Carrera, J.: Coupled estimation of flow and solute transport parameters, Water Resour. Res., 32(11), 3063–3076, 1996.

Neuman, S. P. and Witherspoon, P. A.: Analysis of non-steady flow with a free surface using the finite element method, Water Resour. Res., 7(3), 611–623, 1971.

Neuman, S. P., Preller, C., and Narasimhan, T. N.: Adaptive explicit-implicit quasi-three-dimensional finite element model of flow and subsidence in multiaquifer systems, Water Resour. Res., 18(5), 1551–1562, 1982.

Neuman, S. P. and Tartakovsky, D. M.: Perspective on theories of non-Fickian transport in heterogeneous media, Adv. Water Resour., doi:10.1016/j.advwatres.2008.08.005, in press, 2008.

Ray, C., Ellsworth, T. R., Valocchi, A. J. and Boast, C. W.: An improved dual porosity model for chemical transport in macroporous soils, J. Hydrol., 193(1–4), 270–292, 1997.

Salamon, P., Fernàndez-Garcia, D., and Gómez-Hernández, J. J.: Modeling mass transfer processes using random walk particle tracking, Water Resour. Res., 42, W11417,

doi:10.1029/2006WR004927, 2006.
Sánchez-Vila, X. and Carrera, J.: On the striking similarity between the moments of breakthrough curves for a heterogeneous medium and a homogeneous medium with a matrix diffusion term, J. Hydrol., 294(1–3), 164–175, 2004.
Sardin, M., Schweich, D., Leij, F. J., and van Genuchten, M. T.: Modeling the nonequilibrium transport of linearly interacting solutes in porous media, A review, Water Resour. Res., 27(10), 2287–2307, 1991.
Schumer, R., Benson, D. A., Meerschaert, M. M., and Baeumer, B.: Fractal mobile/immobile solute transport, Water Resour. Res., 39(11), 1296, doi:10.1029/2003WR002141, 2003.
Slooten, L. J., Batlle, F., and Carrera, J.: An object oriented design for hydrological problems, Adv.Water Resour., submitted, 2008.
Sudicky, E. A.: The Laplace transform Galerkin technique: a time-continuous finite element theory and application to mass transport in groundwater, Water Resour. Res., 25(9), 1833–1846, 1989.
Toride, N., Leij, F., and van Genuchten, M. T.: A comprehensive set of analytical solutions for nonequilibrium solute transport with first-order decay and zero-order production, Water Resour. Res., 29(7), 2167–2182, 1993.
Valocchi, A. J.: Validity of the local equilibrium assumption for modeling sorbing solute transport through homogeneous media, Water Resour. Res., 21(6), 808–820, 1985.
Valocchi, A. J.: Use of temporal moment analysis to study reactive solute transport in aggregated porous media, Geoderma, 46(1–3), 233–247, 1990.
Villermaux, J.: Chemical engineering approach to dynamic modeling of linear chromatography. A flexible method for representing complex phenomena from simple concepts, J. Chromatogr., 406, 11–26, 1987.
Vogel, H.-J., Cousin, I., Ippisch, O., and Bastian, P.: The dominant role of structure for solute transport in soil: experimental evidence and modelling of structure and transport in a field experiment, Hydrol. Earth Syst. Sci., 10, 495–506, 2006, http://www.hydrol-earth-syst-sci.net/10/495/2006/.
Wang, P. P., Zheng, C., and Gorelick, S. M.: A general approach to advective-dispersive transport with multirate mass transfer, Adv. Water Resour., 28(1), 33–42, 2005.
Warren, J. E. and Root, P. J.: The behavior of naturally fractured reservoirs, T. Soc. Petrol. En. AIME, 228, 245–255, 1963.
Willmann, M., Carrera, J., and Sánchez-Vila, X.: Transport upscaling in heterogeneous

aquifers: what physical parameters control memory functions?, Water Resour. Res., 44, W12437, doi:10.1029/2007WR006531, 2008.

Zhang, Y., Liu, H.-h., Zhou, Q., and Finsterle, S.: Effects of diffusive property heterogeneity on effective matrix diffusion coefficient for fractured rock, Water Resour. Res., 42, W04405, doi:10.1029/2005WR004513, 2006.

Zhang, Y., Benson, D. A., and Baeumer, B.: Predicting the tails of breakthrough curves in regional-scale alluvial systems, Ground Water, 45(4), 473–484, 2007.

Zhou, Q., Liu, H.-H., Gudmundur, S., Bodvarsson, S., and Molz, F.: Evidence of multi-process matrix diffusion in a single fracture from a field tracer test, Transport Porous Med., 63, 473–487, 2006.

Table 1. Description of the main subroutines of `mod_process_MRMT.f90`.

Subroutine	Task description	Arguments
`Create_`	It is a constructor function called at the beginning of the program. It nullifies all the pointer parameters and sets all the other parameters to zero if they are integer and real.	– `this`: type `t_immobile`
`Destroy_`	Destructor function is called in the end as it deallocates all the pointers allocated.	– `this`: type `t_immobile`
`Read_XML_`	It reads immobile attributes from *.xml files. Checks if there is any error in opening the `XML` file. The subroutine `read_xml_loc_` is called twice. First, without `optional` attribute, so it reads the parameters of the immobile zones from the XML file. Later, when it is called with the `optional` attribute `this`, the parameters are allocated with space and values.	– `this`: type `t_immobile` – `namefile`: type `character` for input file name
`Initialize_`	Allocates the variables and takes the initial condition of the state variable in the mobile region to set an initial condition in the immobile domain. This subroutine is called after reading values from the input `XML` file.	– `this`: type `t_immobile` – `frac_nodes`: type `integer` for number of nodes in mobile region – `conc_imm`: real array with dimension `frac_nodes` to set the initial condition of the state variable in the immobile region. This attribute is an optional variable with zero default value.
`ContriToMatrices_`	Computes the contribution to the storage matrix **D** of the system Eq. (8) according to Eq. (11a). This is one of the main computational subroutines and should be called whenever there is a change in time step.	– `this`: type `t_immobile` – `delt`: type `real` for time step – `theta`: type `real` for time integration factor – `contri_stor`: `real` array with dimension `this%frac_nodes` for contribution to storage matrix
`ContriToSink_`	Computes the contribution to the source/sink term ***b*** of equations system (8) according to Eq. (11b). This subroutine should be called at every time step in conjunction with `ContriToMatrices_` in order to solve governing equations with MRMT.	– `this`: type `t_immobile` – `delt`: type `real` for time step – `theta`: type `real` for time integration factor – `prevConc_mob`: `real` array with dimension `this%frac_nodes` for the state variable in mobile domain at previous time step – `contri_sink`: `real` array with dimension `this%frac_nodes` for contribution to sink/source term
`UpdateConc_`	Updates state variable of immobile region according to Eq. (9). The subroutine takes as arguments the current and previous time step values of the state variable in mobile region. `UpdateConc_` should be called only after calling the contributions subroutines as they require previous time step state variables.	– `this`: type `t_immobile` – `delt`: type `real` for time step – `theta`: type `real` for time integration factor – `prevConc_mob`: `real` array with dimension `this%frac_nodes` for the state variable in mobile domain at previous time step – `CurrConc_mob`: `real` array with dimension `this%frac_nodes` for the state variable in mobile domain at the end of the current time step

HESSD
6, 2415–2449, 2009

Table 1. Continued.

Subroutine	Task description	Arguments
`TotSolMass_`	Computes the total amount of a state variable u (e.g., total mass of solute) in all the immobile zones for a particular mobile node, the total amount and the average value of u in the immobile region.	– `this`: type `t_immobile`
`WriteConc_`	Writes the state variable (e.g., concentration) in the immobile region in a `DAT` file.	– `this`: type `t_immobile` – `time`: type `real` for time at which the state variable is stored – `fileID`: type `integer` and `optional`. If present, output file will be a unit number; if not present output will be written to file `conc_imm.out`
`WriteSolMass_`	Writes the values of solute mass in a separate `DAT` output file.	– `this`: type `t_immobile` – `time`: type `real` indicating the time at which results are stored – `fileID`: type `integer` and `optional`. If present, output file will be a unit number; if not present output will be written to file `Sol_mass.out`
`GetSolMass_`	Get the values of total mass of solute and the average concentration in the immobile zones	– `this`: type `t_immobile` – `tomasim`: `real` array with dimension `this%frac_nodes` returning the total amount of u in the immobile domain – `avcon`: `real` array with dimension `this%frac_nodes` returning an average value of the state variable in the immobile domain

HESSD
6, 2415–2449, 2009

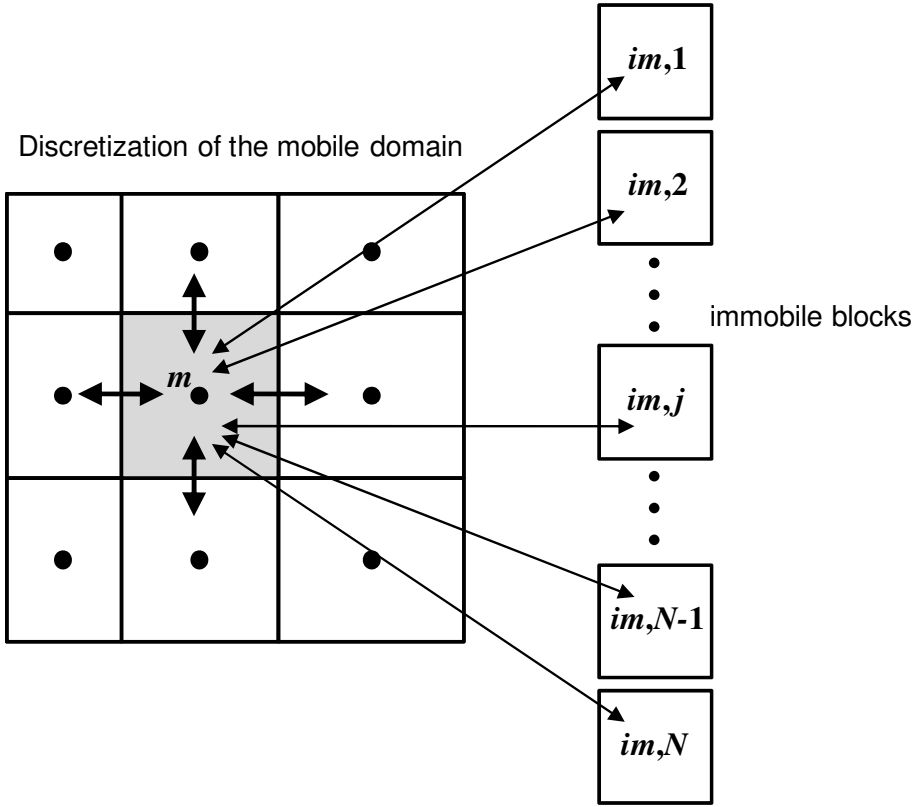

Fig. 1. Hypothetical numerical discretization of the mobile and immobile domains.

HESSD
6, 2415–2449, 2009

```
type, public:: t_immobile
    private
    character(len=100):: name                 !Descriptor of the immobile region
    integer:: nimz                            !Number of types of immobile zones
    integer:: nimztot                         !Total number of immobile zones
    integer:: frac_nodes                      !Number of nodes in the mobile region
    double precision, pointer:: por(:)        !Porosity of the immobile region (Eqs. 3 and 4)
    double precision, pointer:: vol_aqui(:)   !Volume of each cell of the mesh used to solve the mobile region
                                               (Eq. 11)
    double precision:: vol_mob                !Parameter for volume of aquifer
    double precision, pointer:: len_imm(:)    !Average size of matrix blocks or characteristic length of immobile
                                               region (Eqs. 2 and 4)
    double precision, pointer:: diff(:)       !Diffusion or conductivity coefficient in the immobile zones
                                               (Eqs. 2 and 4)
    double precision, pointer:: rtd_fac(:)    !Retardation factor (Eqs. 3 and 4)
    double precision, pointer:: conc_mat(:,:) !Concentration in the immobile region with dimension one representing
                                               n° of immobile zones and two representing n° of mobile nodes, in
                                               case of solute transport. Pressure head in the immobile region for
                                               flow equation.
    double precision, pointer:: totmass_imm(:)!Total "mass" in the immobile region at the mobile nodes
    double precision, pointer:: avg_conc(:)   !Average "concentration" in the immobile zone at the mobile nodes
    double precision, pointer:: alpha(:)      !First-order rate coefficients appearing in Eq. (6)
    double precision, pointer:: betha(:)      !Capacity coefficients of immobile phases appearing in Eq. (5)
    double precision, pointer:: alphaset(:)   !To store all alpha coefficients in the Expansion Series and Power
                                               Law options (e.g., Eq. 14a)
    double precision, pointer:: alphabetha(:) !Product of alpha and betha coefficients (Eqs. 7a, 11b, 16)
    character(len=100), pointer:: geometry(:) !Geometry or name of the process in the immobile regions
    integer, pointer:: nexpterms(:)           !Number of expansion terms in integro-differential, memory function
                                               and CTRW formulations (Eqs. 18-26)
    double precision:: mg                     !Slope of the memory function for power-law behavior (Eqs. 25 and 26)
    double precision:: taui                   !Initial time of power-law behavior (Eqs. 25 and 26)
    double precision:: tauf                   !Final time of power-law behavior (Eqs. 25 and 26)
    character(len=100):: namemodel            !In the power law approach, namemodel = WCSV for Willman et al.
                                               (2008) method, namemodel = SCK for the present method (Eq. 26).
end type t_immobile
```

Fig. 2. Definition of the data type `t_immobile`.

HESSD

6, 2415–2449, 2009

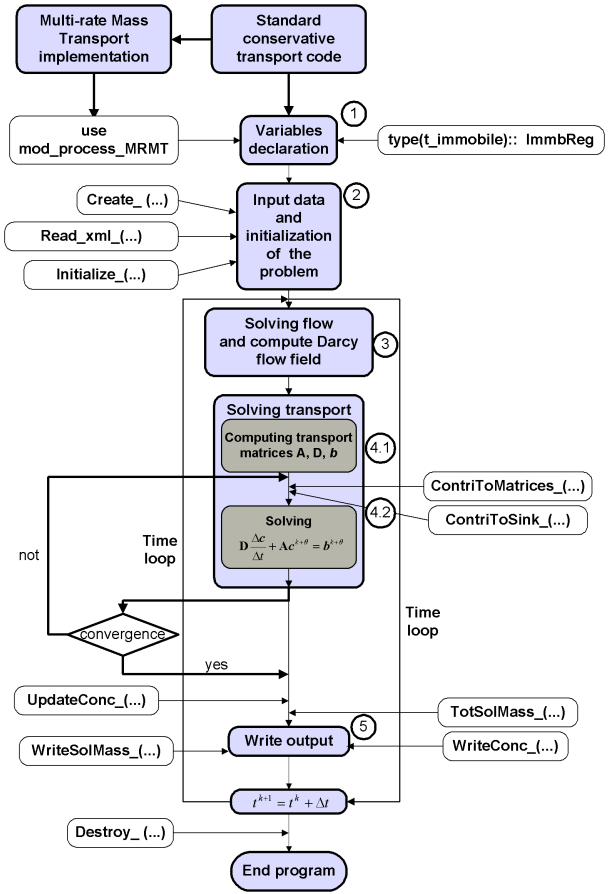

Fig. 3. Linking `mod_process_MRMT.f90` to a conventional numerical code for flow and transport.

HESSD
6, 2415–2449, 2009

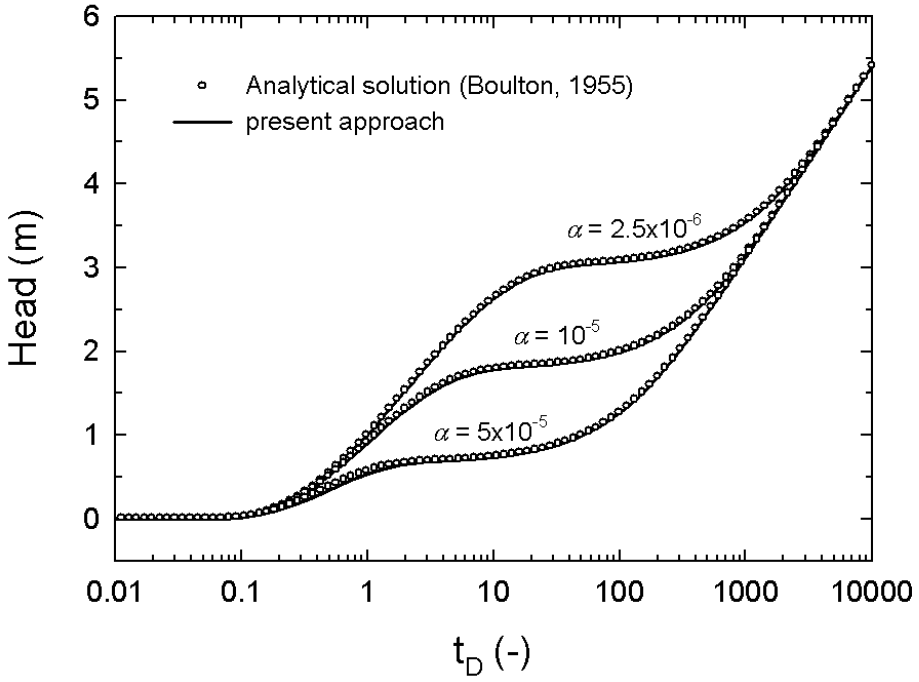

Fig. 4. Comparison between the present approach and an analytic solution (Boulton, 1955) for delayed yield from storage.

HESSD
6, 2415–2449, 2009

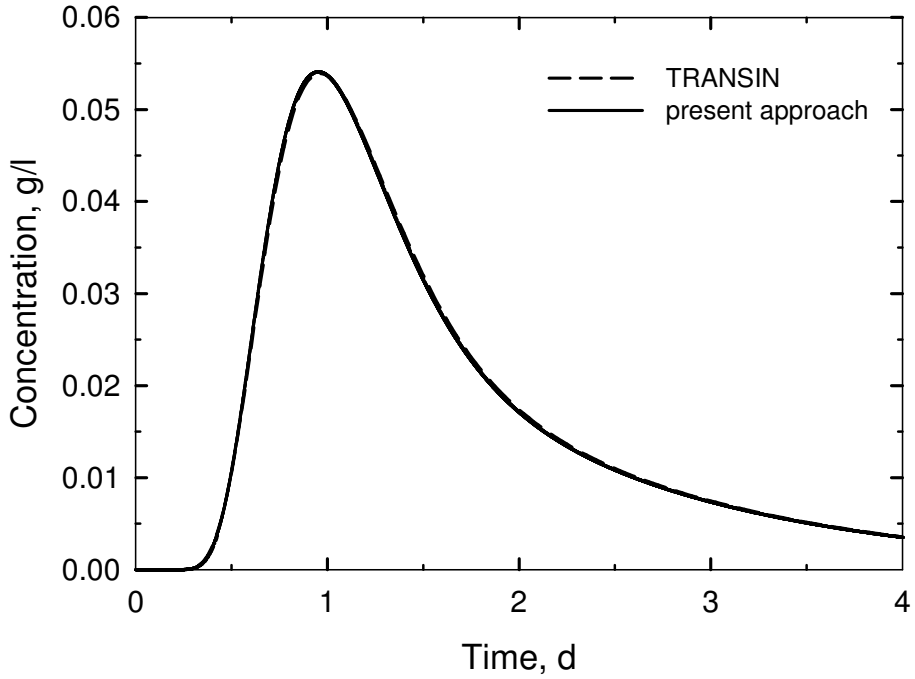

Fig. 5. Verification test with TRANSIN code. Breakthrough curve of tracer in a radial convergent test.

HESSD
6, 2415–2449, 2009

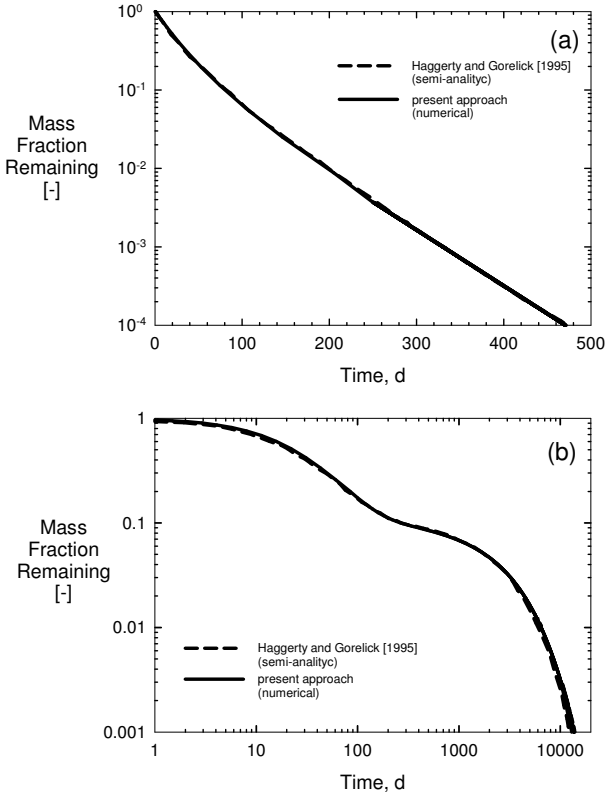

Fig. 6. Comparison between the present approach and a semi-analytic solution (Haggerty and Gorelick, 1995). **(a)** Homogeneous aquifer. **(b)** Mixture of mass transfer processes (heterogeneous aquifer).

HESSD
6, 2415–2449, 2009

Hydrol. Earth Syst. Sci. Discuss., 6, 2415–2449, 2009
www.hydrol-earth-syst-sci-discuss.net/6/2415/2009/

Hydrology and Earth System Sciences Discussions

Papers published in *Hydrology and Earth System Sciences Discussions* are under open-access review for the journal *Hydrology and Earth System Sciences*

A general real-time formulation for multi-rate mass transfer problems

O. Silva[1], J. Carrera[1], S. Kumar[2], M. Dentz[3], A. Alcolea[4], and M. Willmann[5]

[1]Institute of Environmental Assessment and Water Research (IDÆA-CSIC), Barcelona, Spain
[2]Department of Geology and Geophysics, Texas A&M University, USA
[3]Department of Geotechnical Engineering and Geosciences, Technical University of Catalonia (UPC), Barcelona, Spain
[4]Hydrogeology Centre (CHYN), University of Neuchâtel, Neuchâtel, Switzerland
[5]Institute of Environmental Engineering, ETH Zurich, Switzerland

Received: 20 February 2009 – Accepted: 3 March 2009 – Published: 17 March 2009

Correspondence to: O. Silva (orlando.silva@idaea.csic.es)

Published by Copernicus Publications on behalf of the European Geosciences Union.

Abstract

Many flow and transport phenomena, ranging from delayed storage observed in pumping tests to tailing in river or aquifer tracer breakthrough curves, display non-equilibrium behavior. Usually, they are modeled by non-local in time formulations, such as multi-porosity, multiple processes non equilibrium, continuous time random walk, memory functions, integro-differential equations, fractional derivatives or multi-rate mass transfer (MRMT), among others. We develop a MRMT algorithm that can be used to represent all these formulations. The method is accurate, computationally inexpensive and easy to implement in groundwater or river flow and transport codes. In fact, we present a module that can be linked to existing programs with minimal programming effort. Its accuracy is verified by comparison with existing solutions.

1 Introduction

Solving flow and solute transport phenomena in natural media requires using variables, such as heads and concentrations, that characterize the state of the system at every point. Therefore, they are termed state variables. State variables are assumed representative of a small portion of water around such point. This implies that the traditional flow and transport equations implicitly assume local equilibrium.

Even though local equilibrium is assumed by default, non-equilibrium behavior is frequently observed in water flow and solute transport through water bodies. Numerous causes may explain non-equilibrium. In water flow through permeable media, it has been attributed to delayed storage mobilization, either because of resistance at the aquifer free surface (Boulton, 1955; Neuman and Witherspoon, 1971; Neuman and Tartakovsky, 2008), or because of resistance at low permeability blocks (Warren and Root, 1963). It has also been attributed to heterogeneity (Cortis and Knudby, 2006). In solute transport, it has been attributed to diffusion-limited storage into immobile regions, kinetic sorption or heterogeneity (Brusseau et al., 1989; Valocchi, 1990; Sardin

et al., 1991; Cvetkovic et al., 1992; Toride et al., 1993; Haggerty and Gorelick, 1995; Ray et al., 1997; Carrera et al., 1998; Salamon et al., 2006; Vogel et al., 2006; Zhang et al., 2006; Zhang et al., 2007; Alcolea et al., 2008; Willmann et al., 2008; Kumar, 2008; Gouze et al., 2008). Non-equilibrium also has been observed in solute transport through rivers that are influenced by the exchange of water between the river and the underlying hyporheic zone (Fernald et al., 2001; Boano et al., 2007; Marion et al., 2008) or by aggregated dead-zones (Beer and Young, 1983; Lees et al., 1998, 2000; Davis et al., 2000).

Non-equilibrium is typically modeled by non-local in time formulations. Non-local in time means that the mobilization of storage does not depend solely on heads (or concentration) at the current time, but also on their past history. In practice, this implies that a sink-source term on the past history (e.g. Carrera et al., 1998) or that an additional storage term (e.g. Haggerty and Gorelick, 1995) are added to the mass balance equations. It is the form of such terms what sets different non-local formulations apart. The number of non-local formulations is too long to list. The most widely used have been:

- Multi-rate mass transfer (MRMT) (Haggerty and Gorelick, 1995).
- Fractional derivatives (Barker, 1988; Acuna and Yortsos, 1995; Schumer et al., 2003).
- Continuous time random walk (CTRW) (Berkowitz and Scher, 1998; Dentz and Berkowitz, 2003; Benson and Meerschaert, 2009).
- Memory functions (Carrera et al., 1998; Haggerty et al., 2000; Gouze et al., 2008).

Among these, the MRMT formulation is appealing, first, because it is easy to understand. In MRMT formulations, the domain is assumed to consist of a mobile continuum and several overlapping immobile continua. These exchange mass linearly with the mobile region. In this way, the state of immobile zones can be characterized by heads (or concentrations). That is, the traditional single state variable can be viewed

as substituted by a continuum of state variables. Effectively, these work as local state variables, representing a local equilibrium in each of the immobile regions.

A second virtue of MRMT approaches is that they have been proven to be equivalent to other non-local formulations. Haggerty et al. (2000) showed that the MRMT is equivalent to the memory functions approach of Carrera et al. (1998). The latter is also equivalent to the non-local in time version of CTRW (Dentz and Berkowitz, 2003), which can also represent non-local in space terms. The same can be said about fractional derivatives (Berkowitz et al., 2002).

Arguably, the most widely extended form of MRMT is that of Haggerty and Gorelick (1995). They developed the model to account for small-scale variation in rates and types of mass transfer by using a sequence of first-order exchange terms to represent mass transfer with an equal number of immobile zones. Their work and other similar studies have revealed that multiple-porosity models can also be used to solve more general mass-transfer problems and their combinations. For instance, Valocchi (1985) studied the conditions for which local equilibrium is a valid assumption for modeling the transport of sorbing solutes in homogeneous soils. Gerke and van Genuchten (1993a, b) presented a dual-porosity model to simulate transient water flow and solute transport in unsaturated fractured rock formations and structured soils. Sánchez-Vila and Carrera (2004) analyzed the moments of the breakthrough curves in tracer tests and concluded that macrodispersion can be represented by means of mass transfer terms. Wang et al. (2005) developed a direct integration method to solve dual-domain multi-rate mass transfer coupled with advective-dispersive transport. Through their approach they extended the MRMT formulation of Haggerty et al. (2000) to more general, transient flow fields. El-Zein et al. (2005) developed a numerical method to solve mass-transfer problems in intact soils, soils with non-equilibrium sorption and immobile solutions, fissured clayey soils, and structured topsoils, in addition to their combinations. Zhou et al. (2006) proposed a conceptual model of multi-process matrix diffusion in a single fracture consisting in a sequence of three mechanisms: diffusion into stagnant water and infilling materials in fractures, diffusion into a degraded matrix zone adjacent

to fractures, and further diffusion into an intact matrix zone away from fractures. They validated their model with a field tracer test. In the same way, Liu et al. (2007) showed that the dual-domain mass transfer model can reproduce both the near-source peak and the downstream low-concentration spreading when small-scale high-conductivity preferential flow paths are present in a homogeneous porous matrix.

The problem with MRMT approaches, and with most non-local formulations for that matter, is that they are usually implemented to solve a particular numerical application. They are typically solved through Laplace transformation, which facilitates overcoming the nuisances of fractional derivatives or non-locality. These formulations are numerically efficient, but hard to translate to non-linear phenomena, such as multicomponent reactive transport. As a result, these formulations have been typically linked to specific codes, which hinders their general use. Thus, the objective of the present work is to propose an easy numerical implementation of multi-rate mass transfer that is equivalent to other existing formulations, and that also is able to describe other phenomena distinct from solute transport. The present approach is implemented in a Fortran 90 module that can be quite easily embedded into any existing numerical code for flow and solute transport.

2 Governing equations

Non-local in time formulations can be used to enrich the behavior of either the flow or transport equations, or both. In either case, they can be viewed in two complementary fashions: (i) as a continuum of delayed storage terms, in which case these equations represent the total mass balance in both mobile and immobile regions, or (ii) as a continuum of sink/source terms, which act as linear mass exchange terms between mobile and immobile zones. In practice, the continuum is substituted by a discrete number of

terms. Therefore, using the first view, the flow equation becomes

$$S_m \frac{\partial h_m}{\partial t} + \sum_{j=1}^{N} S_{im,j} \frac{\partial h_{im,j}}{\partial t} = -\nabla \cdot \mathbf{q} + q \tag{1}$$

where t (T) is time, h_m (L) is head in the mobile zone, S_m (L^{-1}) is the specific storage coefficient, $\boldsymbol{q}$ (L/T) is water flux and q (T^{-1}) represents a sink/source (recharge/extraction). $h_{im,j}$ (L) and $S_{im,j}$ (L^{-1}) are head and specific storage coefficients of the jth immobile zone, respectively. Water storage in each immobile region is fed by a linear exchange with the mobile domain

$$S_{im,j} \frac{\partial h_{im,j}}{\partial t} = \sigma_{im,j} \frac{K_{im,j}}{L_{im,j}} \left(h_m - h_{im,j} \right) \tag{2}$$

where $\sigma_{im,j}$ (L^2/L^3) is the specific surface of the jth immobile region, $L_{im,j}$ (L) its distance from the mobile zone and $K_{im,j}$ (L/T) its hydraulic conductivity.

Analogously, the solute transport equation expresses the solute mass balance per unit volume of aquifer

$$\phi_m R_m \frac{\partial c_m}{\partial t} + \sum_{j=1}^{N} \phi_{im,j} R_{im,j} \frac{\partial c_{im,j}}{\partial t} = \nabla \cdot (\boldsymbol{D}_m \cdot \nabla c_m) - \boldsymbol{q} \cdot \nabla c_m \tag{3}$$

where c_m (M/L^3) is the mobile concentration, $\boldsymbol{D}_m$ (L^2/T) is the hydrodynamic dispersion tensor, ϕ_m (L^3/L^3) is the mobile porosity (volume of pores per unit aquifer volume), and R_m (–) is the mobile zone retardation factor. Similarly, $c_{im,j}$ (M/L^3), $\phi_{im,j}$ (L^3/L^3) and $R_{im,j}$ (–) are the concentration, porosity and retardation factor of the jth immobile zone. As in flow phenomena, mass balance in the jth immobile region is given by

$$R_{im,j} \phi_{im,j} \frac{\partial c_{im,j}}{\partial t} = \sigma_{im,j} \frac{\phi'_{im,j} D_{im,j}}{L_{im,j}} \left(c_m - c_{im,j} \right) \tag{4}$$

where $\phi'_{im,j}$ (L^3/L^3) is its porosity (volume of pores per unit volume of immobile region), $D_{im,j}$ (L^2/T) is a molecular diffusion coefficient in the jth immobile region. Equation (4) can be somewhat simplified by writing $\phi'_{im,j}$ as a function of $\phi_{im,j}$ (see e.g., Carrera et al., 1998). However, we must bear in mind that the physical meaning of the other parameters in Eqs. (2) and (4) is somewhat approximative. Bearing this in mind, Eqs. (1)–(4) can be written in general as

$$\beta \frac{\partial u_m}{\partial t} + \sum_{j=1}^{N} \beta_j \frac{\partial u_{im,j}}{\partial t} = L_u(u_m) \tag{5}$$

$$\frac{\partial u_{im,j}}{\partial t} = \alpha_j \left(u_m - u_{im,j}\right) \quad j = 1 \ldots N \tag{6}$$

where $u=h$, for flow or $u=c$, for solute transport. The β_j (dimensionless) coefficients are called capacity coefficients ($R_{im,j}\phi_{im,j}$ for transport or $S_{im,j}$ for flow) to account for the distribution of mass in the immobile phases; β (dimensionless) is the capacity coefficient of the mobile phase ($R_m\phi_m$ for transport or S_m for flow); and α_j (T^{-1}) is a first-order mass transfer rate coefficient. The right-hand side of Eqs. (1) and (3) is designated generically by the operator L_u.

Denoting F_j the jth term of the sum in Eq. (5), the governing equations for the immobile zones are given as

$$F_j = \beta_j \frac{\partial u_{im,j}}{\partial t} = \beta_j \alpha_j (u_m - u_{im,j}) \quad j = 1 \ldots N \tag{7a}$$

$$F = \sum_{j=1}^{N} F_j \tag{7b}$$

3 General formulation for MRMT

Spatial and time discretization of either the flow or transport equations under local equilibrium assumptions (i.e., without MRMT) leads to a linear system of equations (e.g., Medina and Carrera, 1996)

$$\mathbf{D}\frac{\Delta \boldsymbol{u}_m}{\Delta t} + \mathbf{A}\boldsymbol{u}_m^{k+\theta} = \boldsymbol{b}^{k+\theta} \tag{8}$$

where $\Delta \boldsymbol{u}_m = \boldsymbol{u}_m^{k+1} - \boldsymbol{u}_m^k$, $\Delta t = t^{k+1} - t^k$ is the time step, θ is a weighting factor and the superscripts stand for the time in which the variable is evaluated.

Accounting for MRMT can be achieved in two ways: (a) using an appropriate mesh with nodes representing the immobile zones (e.g., Neuman, 1982), or (b) by eliminating the unknown in the immobile region as an explicit state variable, i.e. expressing $u_{im,j}$ as a function of u_m (e.g., Carrera et al., 1998). Here, we have adopted the later approach because: first, it maintains the number of unknowns unchanged and, second, it is actually simpler to implement into existing generic flow and transport simulation codes.

Figure 1 displays a schematic representation of a hypothetical numerical mesh that includes both the mobile and immobile domains. We assume that each node m of the mobile zone is connected to all adjacent nodes of the mesh and to all the immobile blocks. Node im,j of the immobile region is only connected to node m. Geometrically, node im,j overlaps with node m. We show below that the variable u at node im,j (i.e., $u_{im,j}$) can be solved explicitly as a function of u_m. Therefore, node im,j need not be an "uncertain" node, but can be considered as a zero-D node.

We first solve the N first-order ordinary differential Eq. (6) in terms of Δu_m, while assuming that u_m varies linearly during each time increment. That is, $u_m = u_m^k + (\Delta u_m/\Delta t)(t - t^k)$. This leads to N first-order linear differential equations, whose solution is

$$u_{im,j}(t) = u_{im,j}^k e^{-\alpha_j(t-t^k)} + u_m^k \left(1 - e^{-\alpha_j(t-t^k)}\right) + \frac{\Delta u_m}{\Delta t}\left[(t - t^k) - \frac{1}{\alpha_j}\left(1 - e^{-\alpha_j(t-t^k)}\right)\right] \tag{9}$$

Combining Eqs. (6), (7a) and (9), the flux F_j evaluated at time $t^{k+\theta}$ will be

$$F_j^{k+\theta} = \beta_j \alpha_j \left(u_m^{k+\theta} - u_{im,j}^{k+\theta} \right)$$
$$= \beta_j \alpha_j \left(u_m^k - u_{im,j}^k \right) e^{-\alpha_j \theta \Delta t} + \frac{\Delta u_m}{\Delta t} \beta_j \left(1 - e^{-\alpha_j \theta \Delta t} \right) \quad (10)$$

Notice that this flux is only a function of u at the previous time step and Δu_m. The total mass flux, $F^{k+\theta}$, is given by Eq. (7b). Substituting the resulting expression into Eq. (8) leads to an identical system, except that the storage matrix and sink/source term are modified according to

$$(\mathbf{D}^*)_{ii} = (\mathbf{D})_{ii} + v_i \sum_{j=1}^{N} \beta_j \left(1 - e^{-\alpha_j \theta \Delta t} \right) \quad (11a)$$

$$(\boldsymbol{b}^*)_i^{k+\theta} = (\boldsymbol{b})_i^{k+\theta} - v_i \sum_{j=1}^{N} \beta_j \alpha_j \left[u_{m,i}^k - \left(u_{im,j}^k \right)_i \right] e^{-\alpha_j \theta \Delta t} \quad (11b)$$

where $u_{m,i}^k$ is the value of u at node i of the mobile region and time t^k, and $\left(u_{im,j}^k \right)_i$ the corresponding value in the jth immobile block, and v_i is the volume of cell i in volume integrated formulations (e.g., finite element) and is equal to 1 in discretized formulations (e.g., finite differences). Finally, it is necessary to update $u_{im,j}$ at the end of each time step using Eq. (9). This approach is quite simple to program and should lead to accurate solutions at a very low computational cost. As with the integro-differential approach, the number of nodes/elements is not altered by the addition of the MRMT terms.

4 Equivalence with other similar approaches

As mentioned in the introduction, a large number of non-local in time schemes have been presented by different authors (Beer and Young, 1983; Sudicky, 1989; Haggerty and Gorelick, 1995; Carrera et al., 1998; Lees et al., 2000; Fernald et al., 2001; Dentz and Berkowitz, 2003; Schumer et al., 2003; Boano et al., 2007; Marion et al., 2008). This section is devoted to discussing such equivalence so as to facilitate using them in the proposed formulation. Haggerty et al. (2000) provided a comparison table with different MRMT formulations considering governing equations similar to Eqs. (5) and (6). The present approach is essentially identical to that of Haggerty and Gorelick (1995). The main difference is that they formulated their equations per unit volume of water. Therefore, their capacity coefficients are equal to the coefficients β_j of Eqs. (5) and (6), but divided by the mobile capacity, β. Denoting β_{jHG} and α_{jHG} the capacity and first-order mass transfer coefficients considered by Haggerty and Gorelick (1995), we have the following equivalence relationship

$$\beta_{jHG} = \beta_j / \beta \quad (12a)$$

$$\alpha_{jHG} = \alpha_j \quad (12b)$$

We have preferred to use capacity coefficients as defined in Eq. (5) to keep the physical meaning and consistence of the governing equations as mass balances per unit volume of aquifer.

Many schemes approximate the effect of the immobile region by a continuous memory function. The governing equations are then solved in the Laplace domain. These solutions can be approximated by expanding the memory function as a sum of exponentials (Carrera et al., 1998). Each summand can then be solved as explained in Sect. 3. For the purposes of comparison with our approach, the important issue is to acknowledge that such approaches are typically defined in terms of overall parameters

for the whole immobile region (as opposed to independent α_j and β_j). For the works of Carrera et al. (1998) and Salamon et al. (2006), the equivalence is given by:

$$\alpha_j = \gamma_j^2 \frac{D_{im}}{R_{im} L_{im}^2} \tag{13a}$$

$$\beta_j = \frac{a_j}{\gamma_j^2} R_{im} \phi_{im} \tag{13b}$$

where ϕ_{im} (L^3/L^3), R_{im}, D_{im} (L^2/T) and L_{im} (L) are characteristic parameters of the entire immobile domain. Coefficients a_j and γ_j can be found in the literature (e.g., Haggerty and Gorelick, 1995; Carrera et al., 1998; Haggerty et al., 2000; Salamon et al., 2006) for diffusion into different geometries (layered, cylindrical, spherical and veins) and the standard first-order model. These formulations result from the analytical solution of the diffusion equation. The coefficients in Eqs. (13) result from an infinite series expansion that needs to be truncated. Salamon et al. (2006) provide a table with the term required to cope with the truncation error. A large number of first-order mass transfer rate coefficients and their distributions estimated from field and laboratory test results can be obtained from the works of Cosler (2004) and Haggerty et al. (2004).

The mass flux, F, into the immobile region in memory function based approaches is given by

$$F(x,t) = \int_0^t g(t-\tau) \frac{\partial u_m(x,\tau)}{\partial \tau} d\tau = g * \frac{\partial u_m}{\partial t} + g(t) u_m(x,0) \tag{14}$$

where g is the memory function and $*$ denotes the convolution product. Carrera et al. (1998) approximate this product using the integro-differential approach of Herrera and Rodarte (1973) and Herrera and Yates (1977). An equivalent alternative is to

approximate g by

$$g(t) = \int_0^\infty \alpha b(\alpha) e^{-\alpha t} d\alpha \tag{15}$$

where $b(\alpha)$ (T) is a density function of first-order rate coefficients. Haggerty et al. (2000) provide explicit expressions for the density and memory functions for various models or geometries. To use the approach of Sect. 3, we need to express the memory function as

$$g(t) = \sum_{j=1}^{\infty} \alpha_j \beta_j e^{-\alpha_j t} \tag{16}$$

Note that Haggerty et al. (2000) included the factor $\alpha_j \beta_j$ on the memory function, unlike Carrera et al. (1998) who placed it on flux F. However, both approaches are equivalent. We calculate the convolution product in Eq. (14), truncating the memory function at Nth term and following the same algebraic analysis described in the Appendix 1 of Carrera et al. (1998). Thus, we can express $F^{k+\theta}$ as

$$F^{k+\theta} = \sum_{j=1}^{N} \beta_j \alpha_j e^{-\alpha_j \theta \Delta t} I_j^k + \frac{\Delta u_m}{\Delta t} \sum_{j=1}^{N} \beta_j \left(1 - e^{-\alpha_j \theta \Delta t}\right) \tag{17a}$$

$$I_j^{k+1} = \int_0^{t^{k+1}} e^{-\alpha_j \left(t^{k+1} - \tau\right)} \frac{\partial u_m}{\partial \tau} d\tau = e^{-\alpha_j \Delta t} I_j^k + \frac{\left(1 - e^{-\alpha_j \Delta t}\right)}{\alpha_j} \frac{\Delta u_m}{\Delta t} \tag{17b}$$

The equivalence between our approach and integro-differential approach becomes evident by comparing Eqs. (17a) and (10). Also note that, from Eq. (9) we obtain the recursive relationship

$$u_m^{k+1} - u_{im,j}^{k+1} = e^{-\alpha_j \Delta t} \left(u_m^k - u_{im,j}^k\right) + \frac{\left(1 - e^{-\alpha_j \Delta t}\right)}{\alpha_j} \frac{\Delta u_m}{\Delta t} \tag{18}$$

which is similar to Eq. (17b). Therefore, we arrive at Eqs. (7b) and (10) by imposing the condition

$$I_j^k = u_m^k - u_{im,j}^k \tag{19}$$

Truncation criteria and expression for the final terms of truncated multi-rate series can be found in Haggerty and Gorelick (1995) and Salamon et al. (2006). In the case of diffusion into different geometries, they proposed the same criteria to evaluate β_N. However, while Haggerty and Gorelick (1995) suggested writing α_N as the rest of the α_j coefficients (i.e., Eq. 13a), Salamon et al. (2006) proposed the following expressions

$$\beta_N = \left(1 - \sum_{j=1}^{N-1} \frac{a_j}{\gamma_j^2}\right) R_{im}\phi_{im} \tag{20a}$$

$$\alpha_N = \lambda \frac{\left(1 - \sum_{j=1}^{N-1} \frac{a_j}{\gamma_j^2}\right)}{\left(1 - \lambda \sum_{j=1}^{N-1} \frac{a_j}{\gamma_j^4}\right)} \frac{D_{im}}{R_{im} L_{im}^2} \tag{20b}$$

where λ for layers, spheres, cylinders are given by Salamon et al. (2006).

Dentz and Berkowitz (2003) found a mathematical equivalence between MRMT and the CTRW model (Berkowitz and Scher, 1998; Berkowitz et al., 2006; Margolin et al., 2003; Salamon et al., 2006; Benson and Meerschaert, 2009). They formulated a CTRW approach which is formally equivalent to the integro-differential formulation of MRMT presented in this paper. They present a map between the memory function defined in the context of MRMT and the transition time distribution $\psi(t)$

$$g^*(s) = \frac{1 + \psi^*(s)(1 + s\tau_0)}{s\tau_0 \psi^*(s)}, \tag{21}$$

where τ_0 defines which part of the medium is mobile or immobile and as such is related to the mobile and immobile volume fractions of the medium (see Dentz and Berkowitz, 2003). The Laplace transform of the memory function, g^*, can be expanded into a series in s according to

$$g^*(s) = \sum_{k=1}^{\infty} (-1)^k a_k s^k, \tag{22}$$

where explicit expressions for the a_k are given in Dentz and Berkowitz (2003). For $g^*(s)$ given by the Laplace transform of Eq. (16), we obtain

$$g^*(s) = \sum_{j=1}^{\infty} \frac{\alpha_j \beta_j}{\alpha_j + s} = \sum_{j=1}^{\infty} \beta_j \sum_{k=1}^{\infty} (-1)^k \alpha_j^{-k} s^k = \sum_{k=1}^{\infty} (-1)^k s^k \left[\sum_{j=1}^{\infty} \alpha_j^{-k} \beta_j\right] \tag{23}$$

By comparison of Eqs. (22) and (23), we obtain relations between the β_j and the a_k for a given series of rates α_j

$$a_k = \left[\sum_{j=1}^{\infty} \alpha_j^{-k} \beta_j\right] \tag{24}$$

The latter expression can be inverted (numerically) in order to obtain explicit expressions for the weights β_j and thus for the memory function $g(t)$ that simulates the transport behavior in a CTRW.

Dentz and Berkowitz (2003) also proposed the use of the truncated power law memory function, which has become widely used because breakthrough curves often display a power law behavior at late times (see, e.g., Zhang et al., 2007; Willmann et al., 2008). The late time behavior of the breakthrough curve can be related to the memory function (Haggerty et al., 2000). This memory function only requires specifying the slope of the memory function in log-log scale, m_g, and the interval of time (t_1, t_N) on which this function displays a power-law behavior. A practical method to calculate the

distribution coefficients β_j consists of, first, calculate the α_j values assuming they are evenly distributed on a logarithmic scale while fixing $\alpha_1=t_N^{-1}$ and $\alpha_N=t_1^{-1}$. Secondly, we obtain a recursive relationship for β_j values by approximating the memory function with expressions of successive increasing orders, i.e.

$$\log\left(\sum_{i=j}^{N}\beta_i\alpha_i\right)-\log\left(\sum_{i=j+1}^{N}\beta_i\alpha_i\right)=m_g(\log t_j-\log t_{j+1}) \tag{25}$$

where $t_j=\alpha_j^{-1}$. This leads to

$$\beta_j=\frac{\sum_{i=j+1}^{N}\beta_i\alpha_i}{\alpha_j}\left[\left(\frac{\alpha_j}{\alpha_{j+1}}\right)^{m_g}-1\right]\quad j=1\ldots N-1 \tag{26}$$

To get the values of β_j, we first assign an arbitrary value to β_N (e.g., β_N=1). Then we apply Eq. (26) and finally scale these values imposing the condition $\sum_{j=1}^{N}\beta_j=1$.

5 Numerical implementation

5.1 Module structure, main attributes and subroutines

The equations described in the previous sections have been implemented in a Fortran 90 module called `mod_process_MRMT.f90`, which is structured following the coding guidelines and rules proposed by Slooten et al. (2008). The module defines MRMT objects by means of type `t_immobile` and provides services to solve the equation described here. The main attributes of the module contained in the type `t_immobile` are detailed in Fig. 2. The main services of `mod_process_MRMT.f90` are described in Table 1. Auxiliary subroutines, private arguments and attributes are commented within `mod_process_MRMT.f90`.

5.2 Input files

Input data of `mod_process_MRMT.f90` are entered in XML format. Reading of input files is achieved by the use of the modules `flib_sax.f90` and `flib_xpath.f90`. These modules were obtained from García (2004) and basically are libraries to handle the reading of xml files. These libraries depends on other Fortran modules, which are stored in folders `xmlreader` and `xpath`, respectively. Based on the mathematical equivalence of the different approaches explained in Sect. 4, four types of input files are available: `Parameters.xml`, `Coefficients.xml`, `ExpansionSeries.xml`, and `PowerLaw.xml`.

5.3 Embedding `mod_process_MRMT.f90` into a standard numerical code

The structure and arguments of `mod_process_MRMT.f90` were designed so that state variables and parameters characterizing the immobile domain are only accessible from the module. This guarantees a minimal information exchange with the program units of any host code to which the user would hope to link `mod_process_MRMT.f90`. That feature helps to get a straightforward implementation of the present MRMT approach on a general advection-dispersion transport simulator programmed in Fortran, as shown in Fig. 3. We assume that the user (programmer) has got any standard numerical code for flow and solute transport that will be modified to include the multi-rate mass transfer equations described in Sects. 2 and 3, according to the following steps:

- Add the file `mod_process_MRMT.f90` and folders `xmlreader` and `xpath` containing all the `XML` supporting material into the source files of the host code. Also copy the `XML` input file to the work directory, depending on the user's application and options given in Sect. 5.2.
- Include the "`Use mod_process_MRMT`" statement into the source code and declare as many variables of type `t_immobile` as required by the user's application. We suggest an `optional` declaration because the user might want to use logical

variables indicating whether MRMT is or not included in the simulations. Therefore, any variable declared as `t_immobile` that could be an argument of host code subroutines will be properly passed whether they are or not present. To create these variables, immediately after declaration in the main transport code include as many callings to the `Create_` subroutine as `t_immobile` variables have been declared. Initialization also requires reading input files in the corresponding section of the standard transport code, so include a calling to the `Read_XML_` subroutine to read the mass-transfer parameters of each immobile zone. After initialization in the host code one has to initialize all variables of `type t_immobile`, by calling the subroutine `Initialize_`.

– After calling the host code subroutines, modules or program units that construct the system matrices **A**, **D** and **b** (Eq. 8), one has to modify **D** and **b** according to Eqs. (11a and 11b). This action is shown in Fig. 3 for the case of including MRMT into the solute transport problem. The subroutine `ContriToMatrices_` calculates de contribution of MRMT to **D** according to Eq. (11a). For each variable declared as `t_immobile`, the user has to include a calling statements to `ContriToMatrices_` and `ContriToSink_` subroutines. The user will need to add the corrections due to MRMT. For instance

```
DO I=1,NUMNP
  D(I) = D(I) + Dcorr(I)
  B(I) = B(I) + Bcorr(I)
END DO
```

where `D` and `B` would be arrays storing the matrices elements of system (8), `Dcorr` and `Bcorr` arrays storing the corrections given by Eqs. (11a and 11b), and `NUMNP` an integer variable representing the number of mobile nodes. Note that subroutines `ContriToMatrices_` and `ContriToSink_` must be called at every time step. They can be called within the subroutines that form matrices **D**

and **b** or after them, following the structure of the host code and user's preferences, but before execution of the unit program that solve the system of Eqs. (8).

– After solving a time step of the transport problem (Fig. 3), the state variable of the immobile domain, $u_{im,j}$, must be updated according to Eq. (9). This is done by calling the subroutine `UpdateConc_`. If required, the total amount and the average value of u in the immobile domains can be computed calling subroutine `TotSolMass_`. Note that those quantities are saved within the internal structure of the variable `ImmbReg`. However, they can be obtained by calling the subroutine `GetSolMass_`. Output subroutines are available to follow the evolution of the state variable u at immobile regions (`WriteConc_`), as well as the total and average of u (`WriteSolMass_`). If required, these subroutines can be called within the program unit for writing output results.

– Finally, it may be necessary and advisable to deallocate all the attributes of all `type t_immobile` variables used in the simulation. This can be achieved by including a call to subroutine `Destroy_` for each of these variables at the end of the transport simulation.

The module supports some characteristics of object-oriented programming. In fact, the module was designed such that its types and operations are available from outside but the details of the implementation are hidden from the user, i.e., the "black box" principle (Gorelik, 2004). Moreover, the present module lies within the definition of objects belonging to the "Process" class of the object-oriented framework tool PROOST (Slooten et al., 2008). These functionalities would also permit to link other Fortran modules or objects to the present module. For instance, reactive transport may be included in both the mobile and immobile regions, by properly linking the object-oriented tool CHEPROO (Bea et al., 2009) to both any conservative transport code and the module `mod_process_MRMT.f90`.

`mod_process_MRMT.f90` can be downloaded from http://www.h2ogeo.upc.es/English/English/software.htm#Mod_process_MRMT.

6 Applications

In order to assess the accuracy of the present approach, we embedded the module into TRACONF (Carrera et al., 1993), a Fortran program for the simulation of water flow and solute transport through porous media. The testing exercise is divided into three parts. We first compare the present formulation with an analytical solution for delayed yield from storage (Boulton, 1955), to test its applicability to flow problem. Next, we compare our approach with the integro-differential approach, simulating the hypothetical radially convergent tracer test described by Alcolea et al. (2001). Finally, we apply the present approach to solve two problems of radial flow to a pumping well (Haggerty and Gorelick, 1995).

6.1 Verification for a flow problem

Boulton (1955) developed an analytical solution for unsteady radial flow allowing delayed yield from storage. This problem can be described by Eq. (1) with N=1, assuming q=0. We model a hypothetical pumping test, considering a transmissivity of T=0.01 m^2/s, S_m=0.001 and S_{im}=0.1, and a pumping rate of 0.04 π m^3/s. Initial head equals zero. We compare our non-local in time approach with the Boulton's solution for three values of the rate coefficient α=2.5×10^{-6}, 10^{-5} and 5×10^{-5} s^{-1}. Boulton (1955) referred to α as an empirical constant. We considered a mesh of 208 nodes with a spacing size Δr increasing geometrically with a factor of 1.08. The integration scheme in time was semi-implicit with variable Δt.

Figure 4 displays the evolution of heads at a distance of R=51.6 m from the well. The dimensionless time t_D=Tt/S_mR^2 was used. We can see that the numerical solution (solid line) obtained with the present approach matches the analytical solution (circles) obtained by Boulton (1955). The figure shows clearly the influence of the rate coefficient α on the system behavior. As α decreases, the system approaches the case in which there is no delayed yield. Again, as α increases the system behaves as it is constituted only by one domain under a mobile storage dominated regime. This is expected because for very large values of α, the water mass transfer between mobile and immobile zones occurs nearly instantaneously: mobile and immobile domains tend to be at equilibrium.

6.2 Verification for a transport problem

We model the convergent tracer test described by Alcolea et al. (2001). A tracer mass of 7.88 g is injected 8 m away from a well pumping 150 m^3/d. The radius of the pumping well and the aquifer thickness are 0.2 m and 5 m, respectively. Porosity of the mobile domain is ϕ_m=0.1. The immobile zone consists of layers of length L_{im}=0.05 m and porosity ϕ_{im}=0.045. The diffusion coefficient in the immobile zone was set to D_{im}=0.001 m^2/d. TRANSIN code uses the integro-differential approach to solve matrix diffusion problems. Accordingly, we have chosen the ExpansionSeries option of our module, which used Eq. (13a and b) to define α's and β's with N=50. A uniform mesh with a grid size Δr=0.005 m (93 nodes) and a fully implicit integration scheme in time was used in the simulations.

Figure 5 shows the breakthrough curves at the pumping well simulated with the present approach and with TRANSIN, which compare quite well. The maximum relative error was 1%.

6.3 Verification for a field problem

Haggerty and Gorelick (1995) presented a case of radial flow to a pumping well, in the context of PCE removal from the Borden sand aquifer under realistic pumping rates. To solve the governing equations, they expressed the MRMT model in dimensionless form and used a semianalytic method. Here we only give the main characteristics concerning with our approach, as the specifications of the problem are well described in their work (Haggerty and Gorelick, 1995). Two hypothetical case studies were considered: the remediation of a homogeneous aquifer (Borden sand) and the remediation of a hypothetical heterogeneous aquifer with a mixture of mass transfer processes. In

both cases we used a mesh of 102 nodes with a spacing size Δr decreasing in such a way that all cells have the same volume and $\Delta r_1 = 1.39$ m. The integration scheme in time was semi-implicit with variable Δt.

The first case study simulates the cleanup history of a homogeneous aquifer where seven immobile zones are associated to a distribution of grain sizes. The grains are assumed to be spherical; the α_j and β_j coefficients are given in Table 2 of Haggerty and Gorelick (1995). As in their work, we have considered 50 exponential terms to adequately describe each immobile domain. Figure 6a displays the evolution of the PCE mass fraction remaining through an scenario of 500 d. We can see that the numerical solution (solid line) obtained with the present approach match very well the semi-analytical solution (dashed line) obtained by Haggerty and Gorelick (1995).

In the second numerical experiment, we simulate the removal of PCE from an heterogeneous aquifer. Here, heterogeneity arises from four immobile zones of different geometry (porous grains, grain aggregates, clay layers and clay pods) and two immobile zones characterized by a surface reaction (slow and fast reactions). The mass-transfer parameters are those appearing in Table 3 of Haggerty and Gorelick (1995). Note that these parameters are assumed locally heterogeneous, but they have the same distribution at all points in space. Once again, 50 terms were used to describe each geometry and only a single term for each reaction. Figure 6b shows predictions of the mass fraction of PCE remaining in the aquifer during a remediation scenario of 20 000 d. Again, the evolution of the mass fraction remaining obtained through the present approach (solid line) fit well the results predicted by the semi-analytical solution (dashed line). Therefore, the present model can reproduce the behavior of heterogeneous media characterized by different types and rates of mass transfer.

7 Conclusions

We have developed an easy-to-use numerical implementation of multi-rate mass transfer, which embeds existing formulations for MRMT. We developed a simple numerical

method, which is physically consistent with and mathematically equivalent to other formulations such as integro-differential formulation. The present approach avoids the spatial discretization of the immobile domain, because it solves state variables of that zone as explicit function of the state variables in the mobile domain. The numerical method is accurate, as it involves an analytical solution for the mass balance equations at immobile zones. We also have implemented the method into a Fortran 90 module that can be easily embedded into standard numerical codes for flow and solute transport to model multi-rate mass transfer. Portability of the module has been achieved because of its object-oriented structure. The module has been tested by comparison with published solutions and is publicly available at http://www.h2ogeo.upc.es/English/English/software.htm#Mod_process_MRMT.

Acknowledgement. We gratefully acknowledge the financial support received from CICYT and ENRESA of Spain, and EC projects FUNMIG (contract number 516514) and MUSTANG (contract number 227286). We also acknowledge the support received from the MICINN and the European Social Fund.

References

Acuna, J. A. and Yortsos, Y. C.: Application of fractal geometry to the study of networks of fractures and their pressure transient, Water Resour. Res., 31(3), 527–540, 1995.

Alcolea, A. and Carrera, J.: Difusión en la matriz. Formulación integro-diferencial, in: VII Simposio de Hidrogeología, Hidrogeología y Recursos Hidráulicos 24, Asociación española de hidrogeólogos, Murcia, Spain, 703–714, 2001.

Alcolea, A., Carrera, J., and Medina, A.: Regularized pilot points method for reproducing the effect of small scale variability: Application to simulations of contaminant transport, J. Hydrol., in press, doi:10.1016/j.jhydrol.2008.03.004, 2008.

Barker, J. A.: A generalized radial flow model for hydraulic tests in fractured rock, Water Resour. Res., 24(11), 1796–1804, 1988.

Bea, S. A., Carrera, J., Ayora, C., Batlle, F., and Saaltink, M. W.: CHEPROO: A Fortran 90 object-oriented module to solve chemical processes in Earth Science models, Comput. Geosci., doi:10.1016/j.cageo.2008.08.010, in press, 2009.

Beer, T. and Young, P. C.: Longitudinal dispersion in natural streams, J. Environ. Eng.-ASCE, 109(5), 1049–1067, 1983.

Benson, D. and Meerschaert, M. M.: A simple and efficient random walk solution of multi-rate mobile/immobile mass transport equations, Adv. Water Resour., doi:10.1016/j.advwatres.2009.01.002, in press, 2009.

Berkowitz, B. and Scher, H.: Theory of anomalous chemical transport in random fracture networks, Phys. Rev. E, 57(5), 5858–5869, 1998.

Berkowitz, B., Klafter, J., Metzler, R., and Scher, H.: Physical pictures of transport in heterogeneous media: advection-dispersion, random-walk, and fractional derivative formulations, Water Resour. Res., 38(11), 1191, doi:10.1029/2001WR001030, 2002.

Berkowitz, B., Cortis, A., Dentz, M., and Scher, H.: Modeling non-Fickian transport in geological formations as a continuous time random walk, Rev. Geophys., 44, RG2003, doi:10.1029/2005RG000178, 2006.

Boano, F., Packman, A. I., Cortis, A., Revelli, R., and Rodolfi, L.: A continuous time random walk approach to the stream transport of solutes, Water Resour. Res., 43, W10425, doi:10.1029/2007WR006062, 2007.

Boulton, N. S.: Unsteady radial flow to a pumped well allowing for delayed yield from storage, in: Publication N°37 de l'Association Internationale d'Hydrologie, Assemblée générale de Rome. Tome II, 472–477, 1955.

Brusseau, M. L., Jessup, R. E., and Rao, P. S. C.: Modeling the transport of solutes influenced by multiprocess nonequilibrium, Water Resour. Res., 25(10), 1971–1988, 1989.

Carrera, J., Galarza, G., and Medina, A.: TRACONF, Programa de elementos finitos para la solución de las ecuaciones de flujo y transporte en acuíferos confinados, E. T. S.I. Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Barcelona, 53 pp., 1993.

Carrera, J., Sánchez-Vila, X., Benet, I., Medina, A., Galarza, G., and Guimerà, J.: On matrix diffusion: formulations, solution methods and qualitative effects, Hydrogeol. J., 6, 178–190, 1998.

Cortis, A. and Knudby, C.: A continuous time random walk approach to transient flow in heterogeneous media, Water Resour. Res., 42, W10201, doi:10.1029/2006WR005227, 2006.

Cosler, D. J.: Effects of rate-limited mass transfer on water sampling with partially penetrating wells, Ground Water, 42(2), 203–222, 2004.

Cvetkovic, V., Shapiro, A., and Dagan, G.: A solute flux approach to transport in heterogeneous formations 2. Uncertainty analysis, Water Resour. Res., 28(5), 1377–1388, 1992.

Davis, P. M, Atkinson, T. C., and Wigley, T. M. L.: Longitudinal dispersion in natural channels: 2. The role of shear flow dispersion and dead zones in the River Severn, UK, Hydrol. Earth Syst. Sci, 4, 355–371, 2000.

Dentz, M. and Berkowitz, B.: Transport behavior of a passive solute in continuous time random walks and multirate mass transfer, Water Resour. Res., 39(5), 1111, doi:10.1029/2001WR001163, 2003.

El-Zein, A. H., Carter, J. P., and Airey, D. W.: Multiple-porosity contaminant transport by finite-element method, Int. J. Geomech., 5(1), 24–34, 2005.

Fernald, A. G., Wigington Jr., P. J., and Landers, D.: Transient storage and hyporheic flow along the Willamette River, Oregon: Field measurements and model estimates, Water Resour. Res., 37(6), 1681–1694, 2001.

García, A.: Xmlf90: A parser for XML in Fortran90, http://lcdx00.wm.lc.ehu.es/ag/xml/index.html, 2004.

Gerke, H. H. and van Genuchten, M. T.: A dual-porosity model for simulating the preferential movement of water and solutes in structured media, Water Resour. Res., 29(2), 305–319, 1993a.

Gerke, H. H. and van Genuchten, M. T.: Evaluation of a first-order water transfer term for variably saturated dual-porosity flow models, Water Resour. Res., 29(4), 1225–1238, 1993b.

Gorelik, A. M.: Object-oriented programming in modern Fortran, Program. Comput. Softw., 30(3), 173–179, 2004.

Gouze, P., Melean, Y., Le Borgne, T., Dentz, M., and Carrera, J.: Non-Fickian dispersion in porous media explained by heterogeneous microscale matrix diffusion, Water Resour. Res., 44, W11416, doi:10.1029/2007WR006690, 2008.

Haggerty, R. and Gorelick, S. M.: Multiple-rate mass transfer for modeling diffusion and surface reactions in media with pore-scale heterogeneity, Water Resour. Res., 31(11), 2383–2400, 1995.

Haggerty, R., McKenna, S. A., and Meigs, L. C.: On the late-time behavior of tracer test breakthrough curves, Water Resour. Res., 36(12), 3467–3479, 2000.

Haggerty, R., Harvey, C. F., von Schwerin, C. F., and Meigs, L.: What controls the apparent timescale of solute mass transfer in aquifers and soils? A comparison of experimental results, Water Resour. Res., 40, W01510, doi:10.1029/2002WR001716, 2004.

Herrera, I. and Rodarte, L.: Integrodifferential equations for system of leaky aquifers and applications. 1. The nature of approximate theories, Water Resour. Res., 9(4), 995–1005, 1973.

Herrera, I. and Yates, R.: Integrodifferential equations for system of leaky aquifers and applications. 3. A numerical method of unlimited applicability, Water Resour. Res., 13(4), 725–732, 1977.

Kumar, S.: Effect of sorption intensities on dispersivity and macro-dispersion coefficient in a single fracture with matrix diffusion, Hydrogeol. J., 16, 235–249, 2008.

Lees, M. J., Camacho, L., and Whitehead, P.: Extension of the QUASAR river water quality model to incorporate dead-zone mixing, Hydrol. Earth Syst. Sci., 2, 353–365, 1998, http://www.hydrol-earth-syst-sci.net/2/353/1998/.

Lees, M. J., Camacho, L. A., and Chapra, S.: On the relationship of transient storage and aggregated dead zone models of longitudinal solute transport in streams, Water Resour. Res., 36(1), 213–224, 2000.

Liu, G., Zheng, C., and Gorelick, S. M.: Evaluation of the applicability of the dual-domain mass transfer model in porous media containing connected high-conductivity channels, Water Resour. Res., 43, 725–732, 2007.

Margolin, G., Dentz, M., and Berkowitz, B.: Continuous time random walk and multirate mass transfer modeling of sorption, Chem. Phys., 295(1), 71–80, 2003.

Marion, A., Zaramella, M., and Bottacin-Busolin, A.: Solute transport in rivers with multiple storage zones: the STIR model, Water Resour. Res., 44, W10406, doi:10.1029/2008WR007037, 2008.

Medina, A. and Carrera, J.: Coupled estimation of flow and solute transport parameters, Water Resour. Res., 32(11), 3063–3076, 1996.

Neuman, S. P. and Witherspoon, P. A.: Analysis of non-steady flow with a free surface using the finite element method, Water Resour. Res., 7(3), 611–623, 1971.

Neuman, S. P., Preller, C., and Narasimhan, T. N.: Adaptive explicit-implicit quasi-three-dimensional finite element model of flow and subsidence in multiaquifer systems, Water Resour. Res., 18(5), 1551–1562, 1982.

Neuman, S. P. and Tartakovsky, D. M.: Perspective on theories of non-Fickian transport in heterogeneous media, Adv. Water Resour., doi:10.1016/j.advwatres.2008.08.005, in press, 2008.

Ray, C., Ellsworth, T. R., Valocchi, A. J. and Boast, C. W.: An improved dual porosity model for chemical transport in macroporous soils, J. Hydrol., 193(1–4), 270–292, 1997.

Salamon, P., Fernàndez-Garcia, D., and Gómez-Hernández, J. J.: Modeling mass transfer processes using random walk particle tracking, Water Resour. Res., 42, W11417,

doi:10.1029/2006WR004927, 2006.

Sánchez-Vila, X. and Carrera, J.: On the striking similarity between the moments of breakthrough curves for a heterogeneous medium and a homogeneous medium with a matrix diffusion term, J. Hydrol., 294(1–3), 164–175, 2004.

Sardin, M., Schweich, D., Leij, F. J., and van Genuchten, M. T.: Modeling the nonequilibrium transport of linearly interacting solutes in porous media, A review, Water Resour. Res., 27(10), 2287–2307, 1991.

Schumer, R., Benson, D. A., Meerschaert, M. M., and Baeumer, B.: Fractal mobile/immobile solute transport, Water Resour. Res., 39(11), 1296, doi:10.1029/2003WR002141, 2003.

Slooten, L. J., Batlle, F., and Carrera, J.: An object oriented design for hydrological problems, Adv.Water Resour., submitted, 2008.

Sudicky, E. A.: The Laplace transform Galerkin technique: a time-continuous finite element theory and application to mass transport in groundwater, Water Resour. Res., 25(9), 1833–1846, 1989.

Toride, N., Leij, F., and van Genuchten, M. T.: A comprehensive set of analytical solutions for nonequilibrium solute transport with first-order decay and zero-order production, Water Resour. Res., 29(7), 2167–2182, 1993.

Valocchi, A. J.: Validity of the local equilibrium assumption for modeling sorbing solute transport through homogeneous media, Water Resour. Res., 21(6), 808–820, 1985.

Valocchi, A. J.: Use of temporal moment analysis to study reactive solute transport in aggregated porous media, Geoderma, 46(1–3), 233–247, 1990.

Villermaux, J.: Chemical engineering approach to dynamic modeling of linear chromatography. A flexible method for representing complex phenomena from simple concepts, J. Chromatogr., 406, 11–26, 1987.

Vogel, H.-J., Cousin, I., Ippisch, O., and Bastian, P.: The dominant role of structure for solute transport in soil: experimental evidence and modelling of structure and transport in a field experiment, Hydrol. Earth Syst. Sci., 10, 495–506, 2006, http://www.hydrol-earth-syst-sci.net/10/495/2006/.

Wang, P. P., Zheng, C., and Gorelick, S. M.: A general approach to advective-dispersive transport with multirate mass transfer, Adv. Water Resour., 28(1), 33–42, 2005.

Warren, J. E. and Root, P. J.: The behavior of naturally fractured reservoirs, T. Soc. Petrol. En. AIME, 228, 245–255, 1963.

Willmann, M., Carrera, J., and Sánchez-Vila, X.: Transport upscaling in heterogeneous

aquifers: what physical parameters control memory functions?, Water Resour. Res., 44, W12437, doi:10.1029/2007WR006531, 2008.

Zhang, Y., Liu, H.-h., Zhou, Q., and Finsterle, S.: Effects of diffusive property heterogeneity on effective matrix diffusion coefficient for fractured rock, Water Resour. Res., 42, W04405, doi:10.1029/2005WR004513, 2006.

Zhang, Y., Benson, D. A., and Baeumer, B.: Predicting the tails of breakthrough curves in regional-scale alluvial systems, Ground Water, 45(4), 473–484, 2007.

Zhou, Q., Liu, H.-H., Gudmundur, S., Bodvarsson, S., and Molz, F.: Evidence of multi-process matrix diffusion in a single fracture from a field tracer test, Transport Porous Med., 63, 473–487, 2006.

Table 1. Description of the main subroutines of `mod_process_MRMT.f90`.

Subroutine	Task description	Arguments
`Create_`	It is a constructor function called at the beginning of the program. It nullifies all the pointer parameters and sets all the other parameters to zero if they are integer and real.	– `this`: `type t_immobile`
`Destroy_`	Destructor function is called in the end as it deallocates all the pointers allocated.	– `this`: `type t_immobile`
`Read_XML_`	It reads immobile attributes from *.xml files. Checks if there is any error in opening the `XML` file. The subroutine `read_xml_loc_` is called twice. First, without `optional` attribute, so it reads the parameters of the immobile zones from the XML file. Later, when it is called with the `optional` attribute `this`, the parameters are allocated with space and values.	– `this`: type `t_immobile` – `namefile`: type `character` for input file name
`Initialize_`	Allocates the variables and takes the initial condition of the state variable in the mobile region to set an initial condition in the immobile domain. This subroutine is called after reading values from the input `XML` file.	– `this`: type `t_immobile` – `frac_nodes`: type `integer` for number of nodes in mobile region – `conc_imm`: real array with dimension `frac_nodes` to set the initial condition of the state variable in the immobile region. This attribute is an optional variable with zero default value.
`ContriToMatrices_`	Computes the contribution to the storage matrix **D** of the system Eq. (8) according to Eq. (11a). This is one of the main computational subroutines and should be called whenever there is a change in time step.	– `this`: type `t_immobile` – `delt`: type `real` for time step – `theta`: type `real` for time integration factor – `contri_stor`: `real` array with dimension `this%frac_nodes` for contribution to storage matrix
`ContriToSink_`	Computes the contribution to the source/sink term $\boldsymbol{b}$ of equations system (8) according to Eq. (11b). This subroutine should be called at every time step in conjunction with `ContriToMatrices_` in order to solve governing equations with MRMT.	– `this`: type `t_immobile` – `delt`: type `real` for time step – `theta`: type `real` for time integration factor – `prevConc_mob`: `real` array with dimension `this%frac_nodes` for the state variable in mobile domain at previous time step – `contri_sink`: `real` array with dimension `this%frac_nodes` for contribution to sink/source term
`UpdateConc_`	Updates state variable of immobile region according to Eq. (9). The subroutine takes as arguments the current and previous time step values of the state variable in mobile region. `UpdateConc_` should be called only after calling the contributions subroutines as they require previous time step state variables.	– `this`: type `t_immobile` – `delt`: type `real` for time step – `theta`: type `real` for time integration factor – `prevConc_mob`: `real` array with dimension `this%frac_nodes` for the state variable in mobile domain at previous time step – `CurrConc_mob`: `real` array with dimension `this%frac_nodes` for the state variable in mobile domain at the end of the current time step

Table 1. Continued.

Subroutine	Task description	Arguments
`TotSolMass_`	Computes the total amount of a state variable u (e.g., total mass of solute) in all the immobile zones for a particular mobile node, the total amount and the average value of u in the immobile region.	– `this`: type `t_immobile`
`WriteConc_`	Writes the state variable (e.g., concentration) in the immobile region in a `DAT` file.	– `this`: type `t_immobile` – `time`: type `real` for time at which the state variable is stored – `fileID`: type `integer` and `optional`. If present, output file will be a unit number; if not present output will be written to file `conc_imm.out`
`WriteSolMass_`	Writes the values of solute mass in a separate `DAT` output file.	– `this`: type `t_immobile` – `time`: type `real` indicating the time at which results are stored – `fileID`: type `integer` and `optional`. If present, output file will be a unit number; if not present output will be written to file `Sol_mass.out`
`GetSolMass_`	Get the values of total mass of solute and the average concentration in the immobile zones	– `this`: type `t_immobile` – `tomasim`: `real` array with dimension `this%frac_nodes` returning the total amount of u in the immobile domain – `avcon`: `real` array with dimension `this%frac_nodes` returning an average value of the state variable in the immobile domain

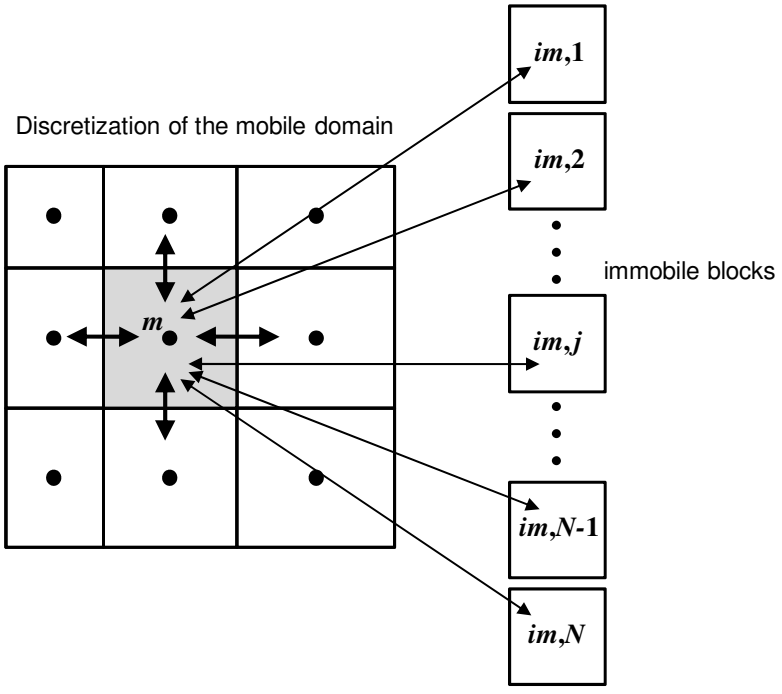

Fig. 1. Hypothetical numerical discretization of the mobile and immobile domains.

```
type, public:: t_immobile
    private
    character(len=100):: name                !Descriptor of the immobile region
    integer:: nimz                           !Number of types of immobile zones
    integer:: nimztot                        !Total number of immobile zones
    integer:: frac_nodes                     !Number of nodes in the mobile region
    double precision, pointer:: por(:)       !Porosity of the immobile region (Eqs. 3 and 4)
    double precision, pointer:: vol_aqui(:)  !Volume of each cell of the mesh used to solve the mobile region
                                              (Eq. 11)
    double precision:: vol_mob               !Parameter for volume of aquifer
    double precision, pointer:: len_imm(:)   !Average size of matrix blocks or characteristic length of immobile
                                              region (Eqs. 2 and 4)
    double precision, pointer:: diff(:)      !Diffusion or conductivity coefficient in the immobile zones
                                              (Eqs. 2 and 4)
    double precision, pointer:: rtd_fac(:)   !Retardation factor (Eqs. 3 and 4)
    double precision, pointer:: conc_mat(:,:) !Concentration in the immobile region with dimension one representing
                                              n° of immobile zones and two representing n° of mobile nodes, in
                                              case of solute transport. Pressure head in the immobile region for
                                              flow equation.
    double precision, pointer:: totmass_imm(:)!Total "mass" in the immobile region at the mobile nodes
    double precision, pointer:: avg_conc(:)  !Average "concentration" in the immobile zone at the mobile nodes
    double precision, pointer:: alpha(:)     !First-order rate coefficients appearing in Eq. (6)
    double precision, pointer:: betha(:)     !Capacity coefficients of immobile phases appearing in Eq. (5)
    double precision, pointer:: alphaset(:)  !To store all alpha coefficients in the Expansion Series and Power
                                              Law options (e.g., Eq. 14a)
    double precision, pointer:: alphabetha(:) !Product of alpha and betha coefficients (Eqs. 7a, 11b, 16)
    character(len=100), pointer:: geometry(:) !Geometry or name of the process in the immobile regions
    integer, pointer:: nexpterms(:)          !Number of expansion terms in integro-differential, memory function
                                              and CTRW formulations (Eqs. 18-26)
    double precision:: mg                    !Slope of the memory function for power-law behavior (Eqs. 25 and 26)
    double precision:: taui                  !Initial time of power-law behavior (Eqs. 25 and 26)
    double precision:: tauf                  !Final time of power-law behavior (Eqs. 25 and 26)
    character(len=100):: namemodel           !In the power law approach, namemodel = WCSV for Willman et al.
                                              (2008) method, namemodel = SCK for the present method (Eq. 26).
end type t_immobile
```

Fig. 2. Definition of the data type `t_immobile`.

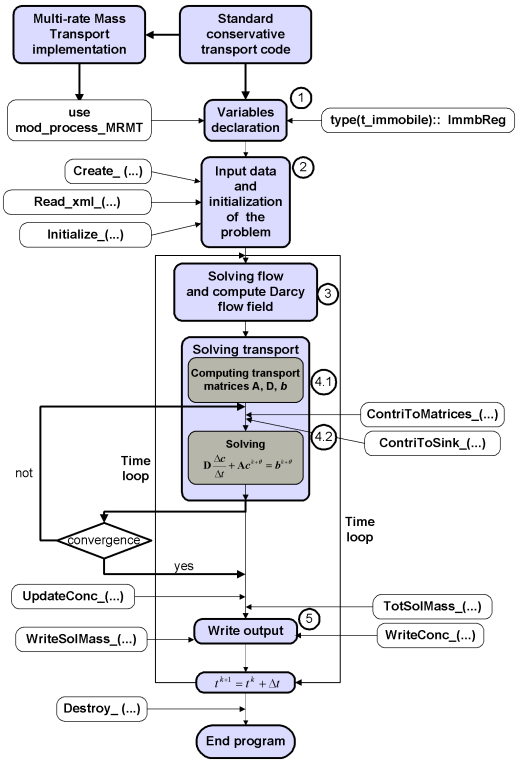

Fig. 3. Linking `mod_process_MRMT.f90` to a conventional numerical code for flow and transport.

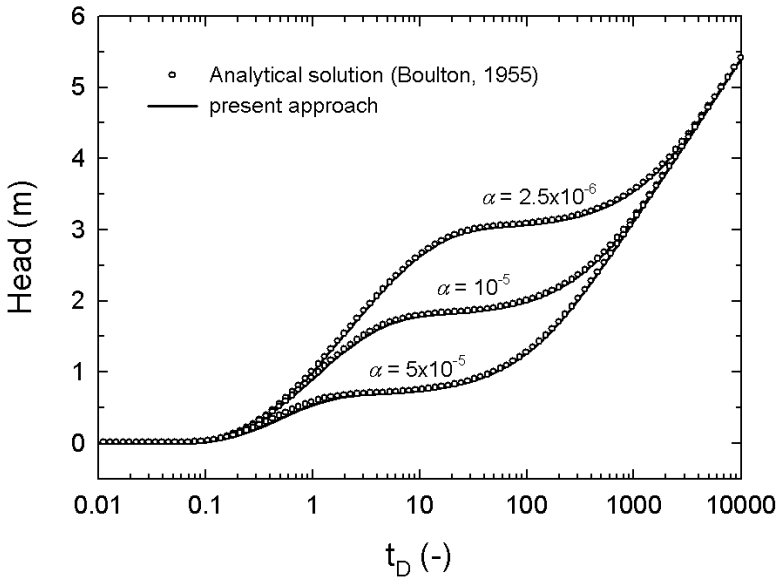

Fig. 4. Comparison between the present approach and an analytic solution (Boulton, 1955) for delayed yield from storage.

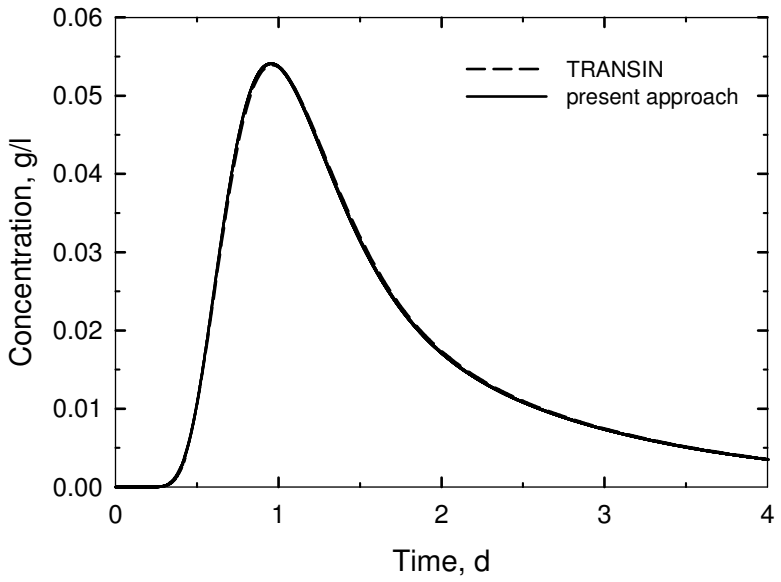

Fig. 5. Verification test with TRANSIN code. Breakthrough curve of tracer in a radial convergent test.

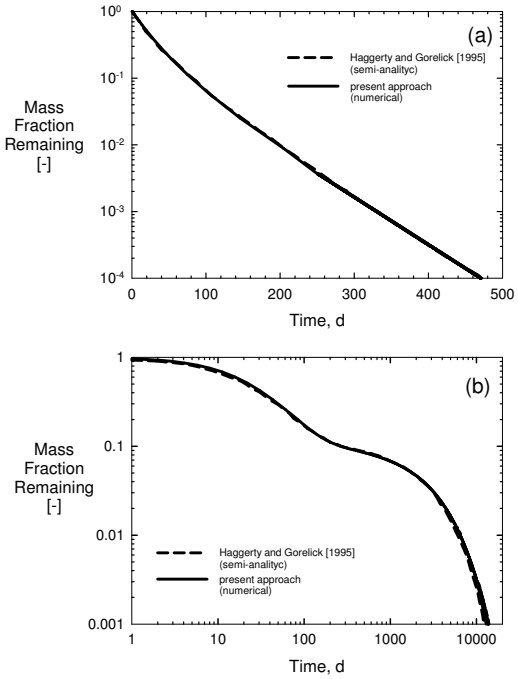

Fig. 6. Comparison between the present approach and a semi-analytic solution (Haggerty and Gorelick, 1995). **(a)** Homogeneous aquifer. **(b)** Mixture of mass transfer processes (heterogeneous aquifer).