scientific reports

OPEN

Unbalanced regularized optimal mass transport with applications to fluid flows in the brain

Xinan Chen[1]✉, Helene Benveniste[2] & Allen R. Tannenbaum[3]

As a generalization of the optimal mass transport (OMT) approach of Benamou and Brenier's, the regularized optimal mass transport (rOMT) formulates a transport problem from an initial mass configuration to another with the optimality defined by the total kinetic energy, but subject to an advection-diffusion constraint equation. Both rOMT and the Benamou and Brenier's formulation require the total initial and final masses to be equal; mass is preserved during the entire transport process. However, for many applications, e.g., in dynamic image tracking, this constraint is rarely if ever satisfied. Therefore, we propose to employ an unbalanced version of rOMT to remove this constraint together with a detailed numerical solution procedure and applications to analyzing fluid flows in the brain.

The optimal mass transport (OMT) problem is concerned with finding a transport mapping from an initial density distribution to a final one, with optimality defined relative to a given cost function[1–4]. A reformulation of the OMT problem in a fluid dynamical framework was proposed by Benamou and Brenier[5] using the L^2 distance as the basis for the cost function, which may be written as follows in the special case of interest to us in the present work.

Given an initial density distribution function $\rho_0(x) \geqslant 0$ and a final one $\rho_1(x) \geqslant 0$ both defined on a bounded region $\Omega \subseteq \mathbb{R}^3$ and with the same total mass (i.e., $\int_\Omega \rho_0(x)\,dx = \int_\Omega \rho_1(x)\,dx$), the dynamic OMT problem by Benamou and Brenier[5] aims to solve

$$\min_{\rho,v} \quad \int_0^T \int_\Omega \|v(t,x)\|^2 \rho(t,x)\,dx\,dt \tag{1a}$$

$$\text{subject to} \quad \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = 0, \tag{1b}$$

$$\rho(0,x) = \rho_0(x), \quad \rho(T,x) = \rho_1(x) \tag{1c}$$

where a temporal dimension $t \in [0, T]$ is added to the transport process. In the above formulation, $\rho(t, x)$ is the dynamic density function, and $v(t, x)$ is the dynamic velocity field defining the fluid flows from ρ_0 to ρ_1. Equation (1b) is called the *continuity equation* in fluid dynamics, and characterizes the advective transport of a conserved quantity in bulk flows. The cost function (1a) is the total kinetic energy of the transport process. The square root of the achieved minimum of (1a), if it exists, is called the L^2-Wasserstein distance between ρ_0 and ρ_1.

As an extension of model (1a)–(1c), a regularized version of OMT (rOMT) has been developed and applied in various places in which one includes a diffusion motion into the transport; see relevant work[6–8] and the many references therein. More precisely, a diffusion term is added into the continuity equation (1b) to make it

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = \sigma \Delta \rho \tag{2}$$

where the constant $\sigma > 0$ is the diffusion coefficient. Equation (2) is thus an *advection-diffusion equation* in fluid dynamics. The rOMT model has been proven useful for a number of important tracking problems in

[1]Department of Medical Physics, Memorial Sloan Kettering Cancer Center, New York 10065, USA. [2]Department of Anesthesiology, Yale School of Medicine, New Haven 06510, USA. [3]Departments of Computer Science and Applied Mathematics & Statistics, Stony Brook University, Stony Brook 11794, USA. ✉email: chenx7@mskcc.org

computational fluid dynamics, such as in quantifying and visualizing the movement of solutes in the brain on dynamic contrast enhanced magnetic resonance imaging (DCE-MRI)[8–12].

As is well-known, both the OMT and rOMT models must satisfy the total mass conservation constraint, namely $\int_\Omega \rho_0(x)\,\mathrm{d}x = \int_\Omega \rho_1(x)\,\mathrm{d}x$. In this case, we usually call the problem as *balanced* to refer to the conservation of total mass. Indeed as formulated in (2), neither advection nor diffusion will change the total mass locally or globally in a given region Ω. However, for applications in dynamic imagery in which either OMT or rOMT is employed as an optical flow tracking method, this is almost never the case. For example, in DCE-MRI data where gadolinium-based tracers are injected and delivered into the body, an early climbing period of the total image intensity is usually observed, since it takes time for the tracers to reach and fill the region of interest. Under these circumstances, if we assume that the intensity of image signal which reflects the concentration of the tracers is proportional to the density in the aforementioned two models, the total mass conservation law is no longer satisfied, and thus the OMT and rOMT models cannot be directly applied. Notably, analyzing the initial accumulating stage of tracers may help uncover interesting and physiologically relevant transport patterns in the brain, and therefore a new model is necessary.

In the present work, we propose an *unbalanced regularized OMT* (urOMT) model for applications to the DCE-MRI fluid flow data, where an independent variable and its indicator function are added as an "invisible" sink or source of mass. The urOMT problem is formulated as follows:

$$\min_{\rho,v,r} \quad \int_0^T \int_\Omega \left(\|v(t,x)\|^2 \rho(t,x) \right. \tag{3a}$$
$$\left. + \alpha \chi(t,x) r(t,x)^2 \rho(t,x) \right) \mathrm{d}x\,\mathrm{d}t$$

$$\text{subject to} \quad \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = \sigma \Delta \rho + \chi \rho r, \tag{3b}$$

$$\rho(0,x) = \rho_0(x), \quad \rho(T,x) = \rho_1(x) \tag{3c}$$

where $r(t,x)$ is the relative source variable, $\chi(t,x)$ is the given indicator function of $r(t,x)$ which takes values either 0 or 1 to constrain r to a certain spatial and temporal location, and $\alpha > 0$ is the weighting parameter of the source term in the cost function. This model takes inputs $\rho_0(x)$, $\rho_1(x)$ and $\chi(t,x)$, and solves for the optimal $\rho(t,x)$, $v(t,x)$ and $r(t,x)$. The added second term in the cost function (3a) is called the *Fisher-Rao* term which arises from the Fisher-Rao metric in information geometry. Our model therefore can be viewed as the interpolation between the L^2-Wasserstein and the Fisher-Rao metrics[13,14].

The partial differential equation (3b) indicates that there are three types of physical phenomena taking place in the dynamic system, advection ($\nabla \cdot (\rho v)$), diffusion ($\sigma \Delta \rho$) and mass creation/destruction ($\chi \rho r$). We should note that for numerical convenience, we prefer to use the relative source r which controls the rate of mass gain ($r > 0$) and loss ($r < 0$), rather than the source $s = \rho r$, as the unbalanced variable in our urOMT formulation. Even though r plays a role as a sink of mass when $r < 0$ and a role as a source of mass when $r > 0$, for simplicity we call r as the relative "source" to broadly refer to both mass gain and loss. In other words, one can imagine that r is the source of both positive and negative mass. The main point however, is that we no longer require the total mass conservation condition for the input images ρ_0 and ρ_1.

The unbalanced OMT problem has been studied both theoretically[14–17] and with various applications in meteorology[18], shape modification[19], image registration[20], image deformation[21,22], image generation[23,24], tumor growth modeling[25,26], population modeling[27] and dynamical tracking[28], etc.

Our work presented here should be considered as an extension of the rOMT algorithm[10] since both share the similar numerical structure and setup. We are specifically interested in applying the urOMT method into DCE-MRI studies to quantify the fluid flows in the rat brain which is our motivation of introducing an unbalanced term to analyze unbalanced data. Although there has been a good amount of work in unbalanced OMT as mentioned above, to the best of our knowledge, this is the first work to incorporate the unbalanced regularized OMT for the quantification of dynamic fluid flows using DCE-MRI imaging.

Briefly summarizing the present paper, in the section "Numerical method", we give the detailed numerical method for solving the urOMT problem (3a)–(3c), and in the section "Results", we test our urOMT model and show results of applications to studying dynamic fluid flows in both synthetic data and DCE-MRI rat brain data. In the section "Discussion", we further discuss and summarize the urOMT method. Some relevant efforts and potential future work are also provided. Lastly in the section "Conclusion", we conclude our present work.

Numerical method

In this section, we elaborate on the numerical method developed for the urOMT problem (3a)–(3c) on 3D images, especially on DCE-MRI-based images in which the signal intensity levels are reflecting the concentration of the gadolinium based tracers, and are therefore proportional to the density in our model. Note that this method can be easily adapted to images in any dimension with minor changes. The numerical method used in this work is largely inspired by and based on the previous work[10].

Model

Given a pair of 3D images, $\rho_0^{img}(x)$ and $\rho_1^{img}(x)$, and an indicator $\chi(x,t)$, in avoidance of the over-matching of the image noise, we consider posing a free end-point condition, so that we can remove the end-point constraint

$\rho(T,x) = \rho_1^{img}(x)$. Another fitting term is consequently added into the cost function. The numerical model we solve is therefore written as:

$$\min_{v,r} \quad \int_0^T \int_\Omega \rho(t,x)\big(\|v(t,x)\|^2$$
$$+ \alpha\chi(t,x)r(t,x)^2\big)\,\mathrm{d}x\,\mathrm{d}t \tag{4a}$$
$$+ \beta\int_\Omega(\rho(T,x) - \rho_1^{img}(x))^2\,\mathrm{d}x$$

$$\text{subject to}\quad \frac{\partial\rho}{\partial t} + \nabla\cdot(\rho v) = \sigma\Delta\rho + \chi\rho r, \tag{4b}$$

$$\rho(0,x) = \rho_0^{img}(x) \tag{4c}$$

where $\beta > 0$ is the weighting parameter for the fitting term in the cost function. The dynamic density function $\rho(t,x)$ can be explicitly derived starting at ρ_0^{img} and following equation (4b) with a velocity field v and a relative source r with its indicator χ, so ρ is removed from the optimized variables. Basically, with this setup ρ becomes a *state variable*. We call this model (4a)–(4c) the *unbalanced regularized OMT with free end-point* (free-urOMT).

Discretization

Since 3D images are typically defined on cubical domains, we divide the cubical space Ω into a cell-centered grid size $n_1 \times n_2 \times n_3$ with uniform spacing Δx, Δy and Δz in x, y and z-direction, respectively. Let $n = n_1 n_2 n_3$ be the total number of voxels. The time interval $[0, T]$ is partitioned into m equal sub-intervals with length $\Delta t = \frac{T}{m}$. Then we have $m + 1$ discrete time steps $t_i = i\Delta t$ for $i = 0, \ldots, m$.

We use a bold font to denote a flattened vector discretized from its corresponding continuous function onto the cell-centered grid defined above. Therefore, the given initial and final images $\rho_0^{img}(x)$ and $\rho_1^{img}(x)$ is discretized into vectors $\boldsymbol{\rho_0^{img}}$ and $\boldsymbol{\rho_1^{img}}$. The density function $\rho(t,x)$ is discretized into $\boldsymbol{\rho_i}$ for $i = 0, \ldots, m$, each denoting the density distribution at t_i, and where $\boldsymbol{\rho_0} = \boldsymbol{\rho_0^{img}}$ denotes the given initial image. The velocity field $v(t,x)$, relative source $r(t,x)$ and its indicator $\chi(t,x)$ may also be discretized into $\boldsymbol{v_i}$, $\boldsymbol{r_i}$ and $\boldsymbol{\chi_i}$ for $i = 0, \ldots, m-1$, each denoting the velocity field, relative source, and the indicator transforming $\boldsymbol{\rho_i}$ to $\boldsymbol{\rho_{i+1}}$, respectively. For each vector $\boldsymbol{v_i}$, its length is three times as that of $\boldsymbol{r_i}$ and $\boldsymbol{\chi_i}$ and is written as the form of $\boldsymbol{v_i} = [\boldsymbol{v_{i,x}}; \boldsymbol{v_{i,y}}; \boldsymbol{v_{i,z}}]$ where the subscript indicates its direction along x, y or z axis. We further denote $\boldsymbol{\rho} = [\boldsymbol{\rho_1}; \cdots; \boldsymbol{\rho_m}]$, $\boldsymbol{v} = [\boldsymbol{v_0}; \cdots; \boldsymbol{v_{m-1}}]$, $\boldsymbol{r} = [\boldsymbol{r_0}; \cdots; \boldsymbol{r_{m-1}}]$ and $\boldsymbol{\chi} = [\boldsymbol{\chi_0}; \cdots; \boldsymbol{\chi_{m-1}}]$.

So far, we have defined the discretized variables on the space and time grids. The cost function equation (4a) may therefore be approximated by

$$\Gamma(\boldsymbol{v},\boldsymbol{r}) = \Gamma_1(\boldsymbol{v},\boldsymbol{r}) + \alpha\Gamma_2(\boldsymbol{v},\boldsymbol{r}) + \beta\Gamma_3(\boldsymbol{v},\boldsymbol{r}) \tag{5}$$

where

$$\Gamma_1(\boldsymbol{v},\boldsymbol{r}) = (\Delta t\Delta x\Delta y\Delta z)\boldsymbol{\rho}^T(I_m \otimes [I_n|I_n|I_n])(\boldsymbol{v}\odot\boldsymbol{v}), \tag{6a}$$

$$\Gamma_2(\boldsymbol{v},\boldsymbol{r}) = (\Delta t\Delta x\Delta y\Delta z)\boldsymbol{\rho}^T(\boldsymbol{r}\odot\boldsymbol{r}\odot\boldsymbol{\chi}), \tag{6b}$$

$$\Gamma_3(\boldsymbol{v},\boldsymbol{r}) = (\Delta x\Delta y\Delta z)\|\boldsymbol{\rho_m} - \boldsymbol{\rho_1^{img}}\|^2. \tag{6c}$$

Here $\otimes$ denotes the Kronecker tensor product, and $\odot$ denotes the Hadamard product. Further, $[\cdot|\cdot]$ represents the block matrix, and $\|\cdot\|$ means taking the L^2 norm of a vector. I_k is the k-dimensional identity matrix for $k \in \mathbb{N}^+$.

Solving the partial differential equation

Next, we deal with the partial differential equation (4b). We place ghost points outside of the boundary and employ the Neumann boundary condition such that the derivative across the boundary is always 0. The Laplacian operator Δ may be approximated with a matrix Q under the aforementioned numerical grid and boundary condition. Similar to the technique employed in the previous work[10], we use the operator-splitting method to numerically solve the equation but in this work we divide the equation into three steps. To be precise, at each time step from t_i to t_{i+1} for $i = 0, \ldots, m-1$, we divide the whole process (4b) into first, mass gain/loss: $\frac{\partial\rho}{\partial t} = \chi\rho r$; second, advection: $\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho v) = 0$; and third, diffusion: $\frac{\partial\rho}{\partial t} = \sigma\Delta\rho$, in total three steps, and then integrate them together.

For the first mass gain/loss step, given an initial condition $\rho(t_i,x) = \rho_i(x)$, the equation may be discretized into

$$\frac{1}{\Delta t}(\boldsymbol{\rho_i^{src}} - \boldsymbol{\rho_i}) = \boldsymbol{\rho_i}\odot\boldsymbol{r_i}\odot\boldsymbol{\chi_i} \tag{7a}$$

3

$$\Rightarrow \quad \boldsymbol{\rho}_i^{src} = (\mathbf{1_n} + \Delta t \cdot \boldsymbol{r_i} \odot \boldsymbol{\chi_i}) \odot \boldsymbol{\rho_i} \tag{7b}$$

where $\mathbf{1_n}$ is a vector of length n consisting of 1's. The second advection step with an initial condition $\rho(t_i, x) = \rho_i^{src}(x)$ can be discretized and solved with

$$\boldsymbol{\rho}_i^{adv} = S(\boldsymbol{v_i})\boldsymbol{\rho}_i^{src}, \tag{8}$$

where $S(\boldsymbol{v_i})$ is the averaging matrix linear to $\boldsymbol{v_i}$ using the particle-in-cell method which redistributes the transported mass to its nearest neighbors by a certain ratio[29]. See Fig. 1 for more details. The (j, k) entry of $S(\boldsymbol{v_i})$ is the ratio of mass allocated from the old location k to the new location j. Multiplying $S(\boldsymbol{v_i})$ by a vector which represents a density distribution, we can derive a new density distribution transported by the velocity $\boldsymbol{v_i}$ under the pre-defined numerical grid. The third diffusion step with an initial condition $\rho(t_i, x) = \rho_i^{adv}(x)$ employs the Euler Backwards scheme in the following manner:

$$\frac{1}{\Delta t}(\boldsymbol{\rho_{i+1}} - \boldsymbol{\rho}_i^{adv}) = \sigma Q\boldsymbol{\rho_{i+1}} \tag{9a}$$

$$\Rightarrow \quad (I_n - \sigma \Delta t \cdot Q)\boldsymbol{\rho_{i+1}} = \boldsymbol{\rho}_i^{adv}. \tag{9b}$$

Combining all three steps (7b), (8) and (9b), we have

$$(I_n - \sigma \Delta t \cdot Q)\boldsymbol{\rho_{i+1}} = S(\boldsymbol{v_i})(\mathbf{1_n} + \Delta t \cdot \boldsymbol{r_i} \odot \boldsymbol{\chi_i}) \odot \boldsymbol{\rho_i}. \tag{10}$$

If we denote

$$L \triangleq I_n - \sigma \Delta t \cdot Q, \tag{11a}$$

$$R(\boldsymbol{r_i}) \triangleq I_n + \Delta t \cdot \mathrm{diag}(\boldsymbol{r_i} \odot \boldsymbol{\chi_i}), \tag{11b}$$

where $\mathrm{diag}(\cdot)$ is the operator turning a vector into a diagonal matrix. Equation (10) may be re-written as
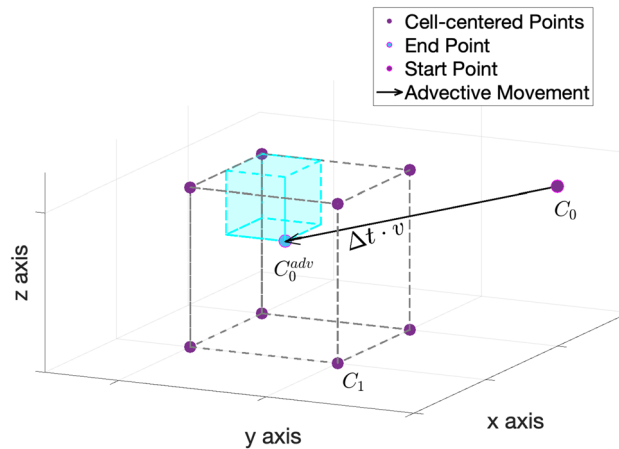
Figure 1. Particle-in-Cell Method. For a cell-centered point C_0, it is transported by a velocity field v over time Δt to arrive at a new location C_0^{adv}. Then the mass at C_0^{adv} will be split to the eight nearest cell-centered neighbors at a ratio. For example, neighbor C_1 will be allocated at a ratio which is the volume colored in cyan over the total voxel volume.
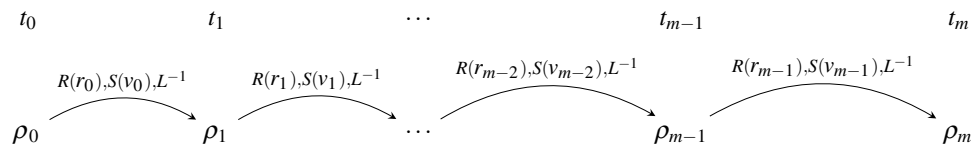
Figure 2. Numerical Pipeline of urOMT. From t_i to t_{i+1} for $i = 0, \ldots, m - 1$, the interpolated image $\boldsymbol{\rho_i}$ is firstly added with mass by applying matrix $R(\boldsymbol{r_i})$, and is secondly advected via the velocity field $\boldsymbol{v_i}$ by applying the averaging matrix $S(\boldsymbol{v_i})$ and is lastly diffused by applying matrix L^{-1}.

$$\boldsymbol{\rho_{i+1}} = L^{-1}S(\boldsymbol{v_i})R(\boldsymbol{r_i})\boldsymbol{\rho_i} \tag{12}$$

for $i = 0, \ldots, m-1$ where $S(\boldsymbol{v_i})$ and $R(\boldsymbol{r_i})$ are n-dimensional matrix in $\boldsymbol{v_i}$ and $\boldsymbol{r_i}$, respectively. See Fig. 2 for the pipeline of the numerical dynamics.

In conclusion, we have discretized the free-urOMT problem (4a)–(4c) as follows:

$$\min_{\boldsymbol{v,r}} \quad \Gamma(\boldsymbol{v,r}) = \Gamma_1(\boldsymbol{v,r}) + \alpha\Gamma_2(\boldsymbol{v,r}) + \beta\Gamma_3(\boldsymbol{v,r}) \tag{13a}$$

$$\text{subject to} \quad \boldsymbol{\rho_{i+1}} = L^{-1}S(\boldsymbol{v_i})R(\boldsymbol{r_i})\boldsymbol{\rho_i}, \quad \text{for } i = 0, \ldots, m-1, \tag{13b}$$

$$\boldsymbol{\rho_0} = \boldsymbol{\rho_0^{img}}, \tag{13c}$$

where Γ_1, Γ_2 and Γ_3 are explicitly given in (6a)–(6c). Given the stable property of the particle-in-cell method used in our work, a small time step Δt is still recommended to avoid error or instability of the entire algorithm[29].

Computing the gradient and the Hessian

Notice that in the discrete problem (13a)–(13c), we can equivalently re-write the constraint (13b) as

$$\boldsymbol{\rho_i} = L^{-1}S(\boldsymbol{v_{i-1}})R(\boldsymbol{r_{i-1}})L^{-1}S(\boldsymbol{v_{i-2}})R(\boldsymbol{r_{i-2}})\cdots L^{-1}S(\boldsymbol{v_0})R(\boldsymbol{r_0})\boldsymbol{\rho_0} \tag{14}$$

for $i = 1, \ldots, m$, which explicitly gives the expression of density distributions at all time steps from a given initial density distribution $\boldsymbol{\rho_0}$, a velocity field $\boldsymbol{v}$ and a relative source $\boldsymbol{r}$. One can prove that $S(\boldsymbol{v_i})$ is linear to $\boldsymbol{v_i}$ and $R(\boldsymbol{v_i})$ is linear to $\boldsymbol{r_i}$ for $i = 0, \ldots, m-1$. Then according to equations (14) and (13c), $\boldsymbol{\rho_m}$ and $\boldsymbol{\rho} = [\boldsymbol{\rho_1}; \cdots; \boldsymbol{\rho_m}]$ in $\Gamma(\boldsymbol{v,r})$ can both be explicitly written in linear to $\boldsymbol{v}$ and $\boldsymbol{r}$. Therefore, the optimization problem (13a)–(13c) can indeed be viewed as an unconstrained minimization problem. Observe that in the cost function, Γ_1 is linear to $\boldsymbol{r}$ and its component $\boldsymbol{v} \odot \boldsymbol{v}$ is quadratic to $\boldsymbol{v}$; Γ_2 is linear to $\boldsymbol{v}$ and its component $\boldsymbol{r} \odot \boldsymbol{r}$ is quadratic to $\boldsymbol{r}$; Γ_3 is quadratic to both $\boldsymbol{v}$ and $\boldsymbol{r}$. It is then natural to employ the Gauss-Newton method to optimize on the problem, which involves computing the gradient and the Hessian matrix of Γ with respect to variables $\boldsymbol{v}$ and $\boldsymbol{r}$.

Next, we focus on calculating the gradient of Γ:

$$g \triangleq [g_{\boldsymbol{v}}; g_{\boldsymbol{r}}], \quad \text{where } g_{\boldsymbol{v}} = \frac{\partial \Gamma}{\partial \boldsymbol{v}}, g_{\boldsymbol{r}} = \frac{\partial \Gamma}{\partial \boldsymbol{r}} \tag{15}$$

and the Hessian matrix of Γ:

$$H \triangleq \begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix}, \quad \text{where } H_{11} = \frac{\partial^2 \Gamma}{\partial \boldsymbol{v}^2}, \quad H_{12} = \frac{\partial^2 \Gamma}{\partial \boldsymbol{v} \partial \boldsymbol{r}}, \quad H_{21} = \frac{\partial^2 \Gamma}{\partial \boldsymbol{r} \partial \boldsymbol{v}}, \quad H_{22} = \frac{\partial^2 \Gamma}{\partial \boldsymbol{r}^2}. \tag{16}$$

Equation (14) shows that $\boldsymbol{\rho_k}$ is determined by $\boldsymbol{v_0}, \ldots, \boldsymbol{v_{k-1}}, \boldsymbol{r_0}, \ldots, \boldsymbol{r_{k-1}}$ and is thus independent of $\boldsymbol{v_j}$ and $\boldsymbol{r_j}$ for $j \geqslant k$. Defining

$$J_{\boldsymbol{v_j}}^k \triangleq \frac{\partial \boldsymbol{\rho_k}}{\partial \boldsymbol{v_j}}, \quad J_{\boldsymbol{r_j}}^k \triangleq \frac{\partial \boldsymbol{\rho_k}}{\partial \boldsymbol{r_j}} \tag{17}$$

for $k = 1, \ldots, m, j = 0, \ldots, m-1$, then $J_{\boldsymbol{v_j}}^k = 0$ and $J_{\boldsymbol{r_j}}^k = 0$ always hold for $j \geqslant k$. If we further denote

$$J_{\boldsymbol{v}} \triangleq \frac{\partial \boldsymbol{\rho}}{\partial \boldsymbol{v}} = (J_{\boldsymbol{v_j}}^k)_{k,j}, \quad J_{\boldsymbol{r}} \triangleq \frac{\partial \boldsymbol{\rho}}{\partial \boldsymbol{r}} = (J_{\boldsymbol{r_j}}^k)_{k,j}, \tag{18}$$

then $J_{\boldsymbol{v}}$ and $J_{\boldsymbol{r}}$ are lower-triangular block matrices of the form

$$J_{\boldsymbol{v}} = \begin{pmatrix} J_{\boldsymbol{v_0}}^1 & & & \\ J_{\boldsymbol{v_0}}^2 & J_{\boldsymbol{v_1}}^2 & & \\ \vdots & \vdots & \ddots & \\ J_{\boldsymbol{v_0}}^m & J_{\boldsymbol{v_1}}^m & \cdots & J_{\boldsymbol{v_{m-1}}}^m \end{pmatrix} \triangleq \begin{pmatrix} J_{\boldsymbol{v}}^1 \\ J_{\boldsymbol{v}}^2 \\ \vdots \\ J_{\boldsymbol{v}}^m \end{pmatrix} \text{ and } J_{\boldsymbol{r}} = \begin{pmatrix} J_{\boldsymbol{r_0}}^1 & & & \\ J_{\boldsymbol{r_0}}^2 & J_{\boldsymbol{r_1}}^2 & & \\ \vdots & \vdots & \ddots & \\ J_{\boldsymbol{r_0}}^m & J_{\boldsymbol{r_1}}^m & \cdots & J_{\boldsymbol{r_{m-1}}}^m \end{pmatrix} \triangleq \begin{pmatrix} J_{\boldsymbol{r}}^1 \\ J_{\boldsymbol{r}}^2 \\ \vdots \\ J_{\boldsymbol{r}}^m \end{pmatrix} \tag{19}$$

where $J_{\boldsymbol{v}}^k = \begin{bmatrix} J_{\boldsymbol{v_0}}^k | J_{\boldsymbol{v_1}}^k | \cdots | J_{\boldsymbol{v_{k-1}}}^k \end{bmatrix}$ denotes the row block of $J_{\boldsymbol{v}}$, and $J_{\boldsymbol{r}}^k$ for that of $J_{\boldsymbol{r}}$ for $k = 1, \ldots, m$. With the notations defined above, then for the gradients we have

$$g_{\boldsymbol{v}} = \frac{\partial \Gamma_1}{\partial \boldsymbol{v}} + \alpha \frac{\partial \Gamma_2}{\partial \boldsymbol{v}} + \beta \frac{\partial \Gamma_3}{\partial \boldsymbol{v}} \tag{20a}$$

$$\begin{aligned} &= (\Delta t \Delta x \Delta y \Delta z)\left(2(M\text{diag}(\boldsymbol{v}))^T \boldsymbol{\rho} + J_{\boldsymbol{v}}^T M(\boldsymbol{v} \odot \boldsymbol{v})\right) \\ &+ \alpha(\Delta t \Delta x \Delta y \Delta z)J_{\boldsymbol{v}}^T(\boldsymbol{r} \odot \boldsymbol{r} \odot \boldsymbol{\chi}) \\ &+ 2\beta(\Delta x \Delta y \Delta z)(J_{\boldsymbol{v}}^m)^T\left(\boldsymbol{\rho_m} - \boldsymbol{\rho_1^{img}}\right) \end{aligned} \tag{20b}$$

and

$$g_r = \frac{\partial \Gamma_1}{\partial r} + \alpha \frac{\partial \Gamma_2}{\partial r} + \beta \frac{\partial \Gamma_3}{\partial r} \tag{21a}$$

$$
\begin{aligned}
= &(\Delta t \Delta x \Delta y \Delta z) J_r^T M(\boldsymbol{v} \odot \boldsymbol{v}) \\
&+ \alpha (\Delta t \Delta x \Delta y \Delta z) \Big(2 \mathrm{diag}(\boldsymbol{r} \odot \boldsymbol{\chi}) \boldsymbol{\rho} + J_r^T (\boldsymbol{r} \odot \boldsymbol{r} \odot \boldsymbol{\chi}) \Big) \\
&+ 2\beta (\Delta x \Delta y \Delta z)(J_r^m)^T \big(\boldsymbol{\rho_m} - \boldsymbol{\rho_1^{img}} \big)
\end{aligned}
\tag{21b}
$$

where we use $M \triangleq I_m \otimes [I_n | I_n | I_n]$ for simplicity of notation. For the Hessian matrix H, we use a function handle which is a MATLAB data type in anticipation of solving the linear system $H\boldsymbol{x} = -g$ in the next step. To be specific, we compute vector $H\boldsymbol{x}$ where $\boldsymbol{x} = [\boldsymbol{x_v}; \boldsymbol{x_r}]$ and $\boldsymbol{x_v} \in \mathbb{R}^{3mn}, \boldsymbol{x_r} \in \mathbb{R}^{mn}$ rather than matrix H. To satisfy the symmetry of the Hessian matrix and to omit some complex second-order terms, we have the approximations

$$H_{11}\boldsymbol{x_v} \approx 2(\Delta t \Delta x \Delta y \Delta z)\mathrm{diag}(\boldsymbol{\rho}^T M) + 2\beta(\Delta x \Delta y \Delta z)(J_v^m)^T J_v^m \boldsymbol{x_v}, \tag{22a}$$

$$H_{22}\boldsymbol{x_r} \approx 2\alpha(\Delta t \Delta x \Delta y \Delta z)\mathrm{diag}(\boldsymbol{\rho} \odot \boldsymbol{\chi}) + 2\beta(\Delta x \Delta y \Delta z)(J_r^m)^T J_r^m \boldsymbol{x_r}, \tag{22b}$$

$$H_{12}\boldsymbol{x_r} \approx 2\beta(\Delta x \Delta y \Delta z)(J_v^m)^T J_r^m \boldsymbol{x_r}, \tag{22c}$$

$$H_{21}\boldsymbol{x_v} \approx 2\beta(\Delta x \Delta y \Delta z)(J_r^m)^T J_v^m \boldsymbol{x_v}. \tag{22d}$$

The motivation of using a function handle $H\boldsymbol{x}$ instead of the matrix H itself is to avoid numerical multiplication of two big matrices which could be time-consuming. With the function handle, for example, in (22c) we can instead multiply a matrix by a vector twice to arrive at the desired computation.

As for the formulation of $J_{v_j}^k$ and $J_{r_j}^k$, by the structure of (13b) we have that for $j = 0, \ldots, m-1$ and $k > j$,

$$J_{v_j}^k = L^{-1} S(\boldsymbol{v_{k-1}}) R(\boldsymbol{r_{k-1}}) \cdots L^{-1} S(\boldsymbol{v_{j+1}}) R(\boldsymbol{r_{j+1}}) L^{-1} B(\boldsymbol{\rho_j}, \boldsymbol{r_j}) \tag{23}$$

where

$$B(\boldsymbol{\rho_j}, \boldsymbol{r_j}) = \frac{\partial}{\partial \boldsymbol{v_j}}(S(\boldsymbol{v_j})R(\boldsymbol{r_j})\boldsymbol{\rho_j}) \tag{24}$$

is dependent only on $\boldsymbol{\rho_j}, \boldsymbol{v_j}$ because $S(\boldsymbol{v_j})$ linear to $\boldsymbol{v_j}$, and

$$J_{r_j}^k = \Delta t \cdot L^{-1} S(\boldsymbol{v_{k-1}}) R(\boldsymbol{r_{k-1}}) \cdots L^{-1} S(\boldsymbol{v_{j+1}}) R(\boldsymbol{r_{j+1}}) L^{-1} S(\boldsymbol{v_j}) \mathrm{diag}(\boldsymbol{\rho_i} \odot \boldsymbol{\chi_i}). \tag{25}$$

With the explicit and recursive expressions in equations (23) and (25), one can compute $J_v^m = [J_{v_0}^m | J_{v_1}^m | \cdots | J_{v_{m-1}}^m]$, $J_r^m = [J_{r_0}^m | J_{r_1}^m | \cdots | J_{r_{m-1}}^m]$ and their transpose multiplied with a vector in an iterative manner. J_v^T and J_r^T multiplied with a vector can also be computed recursively due to their lower-triangularity.

Algorithm

With the analytic formulation of the gradient g and the Hessian handle $H\boldsymbol{x}$ given above, we can then utilize the Gauss-Newton method to find the optimal solution. See Algorithm 1 for the pseudo-code.

If we have more than two successive given images, $\boldsymbol{\rho_0^{img}}, \boldsymbol{\rho_1^{img}}, \ldots, \boldsymbol{\rho_{q-1}^{img}}$ which is inherent to DCE-MRI studies and corresponding given indicator functions $\boldsymbol{\chi_0^{img}}, \boldsymbol{\chi_1^{img}}, \ldots, \boldsymbol{\chi_{q-2}^{img}}$ between adjacent images where $q > 2$ and $q \in \mathbb{N}^+$, we can run the algorithm iteratively between each pair of adjacent images to derive prolonged velocity fields and relative sources. In other words, the process can be graphed as $\boldsymbol{\rho_0^{img}} \xrightarrow[\text{loop 1}]{\text{urOMT}} \boldsymbol{\rho_1^{img}} \xrightarrow[\text{loop 2}]{\text{urOMT}} \cdots \xrightarrow[\text{loop q-2}]{\text{urOMT}} \boldsymbol{\rho_{q-2}^{img}} \xrightarrow[\text{loop q-1}]{\text{urOMT}} \boldsymbol{\rho_{q-1}^{img}}$, where the urOMT algorithm is run for $q-1$ times and therefore $q-1$ successive outputs are returned. If one would like the prolonged velocity fields to be smoother in the temporal dimension, one can put the last interpolated image $\boldsymbol{\rho_m}$ of the previous loop into the next loop as the initial image to avoid constantly introducing new data noise into the system[10].

1: Load in ρ_0^{img}, ρ_1^{img} and χ, and other parameters;
2: $x = [v;r]$ = initial guess (all zeros);
3: **for** $i = 1, 2, \cdots, MaxIter$ **do**
4:     Compute interpolations ρ recursively from the discretized partial differential equation (13b): $\rho = $ SrcAdvDiff$(\rho_0^{img}, \chi, v, r)$;
5:     Compute $S(v_j)$, $R(r_j)$ and $B(\rho_j, r_j)$ for $j = 0, \cdots, m-1$;
6:     Compute gradient g and the Hessian function handle Hx according to equations (20)-(22);
7:     Solve linear system $Hx = -g$ for x;
8:     Do line search to find length l;
9:     **if** line search fails **then**
10:         **return** $[v;r]$;
11:     **end if**
12:     Update $[v;r] = [v;r] + lx$;
13: **end for**
14: **return** $[v;r]$;

Algorithm 1. Gauss-Newton Method.

Results

In this section, we give some examples illustrating the application of the urOMT methodology. Obviously, the model admits both the Eulerian and Lagrangian perspectives for post-processing. The Eulerian formulation focuses on the fluid flows at fixed locations over time, while the Lagrangian formulation enables one to follow the trajectory of a given particle over time, and therefore to analyze the features along the given trajectory.

Specifically, suppose we run the urOMT algorithm on given density images $\boldsymbol{\rho_0^{img}}, \boldsymbol{\rho_1^{img}}, \ldots, \boldsymbol{\rho_{q-1}^{img}}$ and given indicators $\boldsymbol{\chi_0^{img}}, \boldsymbol{\chi_1^{img}}, \ldots, \boldsymbol{\chi_{q-2}^{img}}$ with discretization described before, the algorithm will run $q-1$ successive loops and return the solutions

$$\boldsymbol{v_k^*} = \begin{bmatrix} \boldsymbol{v_{k,0}^*} \\ \vdots \\ \boldsymbol{v_{k,m-1}^*} \end{bmatrix} \text{ and } \boldsymbol{r_k^*} = \begin{bmatrix} \boldsymbol{r_{k,0}^*} \\ \vdots \\ \boldsymbol{r_{k,m-1}^*} \end{bmatrix} \tag{26}$$

for $k = 1, \ldots, q-1$ where the subscript k stands for the k-th loop. By taking the L^2 norm of $\boldsymbol{v_{k,0}^*}, \ldots, \boldsymbol{v_{k,m-1}^*}$, we derive the optimal speed

$$\boldsymbol{s_k^*} = \begin{bmatrix} \boldsymbol{s_{k,0}^*} \\ \vdots \\ \boldsymbol{s_{k,m-1}^*} \end{bmatrix} \triangleq \begin{bmatrix} ||\boldsymbol{v_{k,0}^*}|| \\ \vdots \\ ||\boldsymbol{v_{k,m-1}^*}|| \end{bmatrix}. \tag{27}$$

Therefore, $\boldsymbol{r_{k,i}^*}$ and $\boldsymbol{s_{k,i}^*}$ are called the *Eulerian relative source maps* and *Eulerian speed maps*, respectively, for $k = 1, \ldots, q-1$ and $j = 0, \ldots, m-1$, and they allow us to observe the fluid flows at fixed coordinates at different discrete time steps. For ease of visualization, we define the *time-averaged Eulerian speed map* and *time-averaged Eulerian relative source map* between $\boldsymbol{\rho_{N_0}^{img}}$ and $\boldsymbol{\rho_{N_1}^{img}}$ as

$$\frac{1}{m(N_1 - N_0)} \sum_{k=N_0+1}^{N_1} \sum_{j=0}^{m-1} \boldsymbol{s_{k,j}^*} \text{ and } \frac{1}{m(N_1 - N_0)} \sum_{k=N_0+1}^{N_1} \sum_{j=0}^{m-1} \boldsymbol{r_{k,j}^*}, \tag{28}$$

respectively, where $0 \leqslant N_0 < N_1 \leqslant q-1$ and $N_0, N_1 \in \mathbb{N}^+$.

In contrast, the post-processing framework developed and applied in the previous work[8-10,12] that follows Lagrangian coordinates present data as binary trajectories of the fluid flows, which we refer to as the *pathlines* in our work. By connecting the starting and terminal points of the pathlines, we have the *velocity flux vectors*, which are also called the *displacement fields* in physics. These vectors may be used to visualize the direction and the distance travelled in a compact and interpretable manner. If we endow the pathlines with more information, for example, speed (the L^2 norm of the velocity field) and Péclet (*Pe*) number (i.e., the ratio of the rate of advection to diffusion), we can derive what we call *speed-lines* and *Péclet-lines*, respectively. More details of this Lagrangian method may be found in Koundal et al.[8] and Chen et al.[10]

Now that we have briefly described the two post-processing methods, the Eulerian and the Lagrangian perspectives, we next utilize two datasets to exhibit the results of the urOMT analysis. The first dataset is a synthetic geometric dataset derived from Gaussian spheres as a simple demonstration of the urOMT method. The second is the DCE-MRI data from a rat brain where we elucidate the application of urOMT in in vivo datasets for quantifying and visualizing the fluid flows.
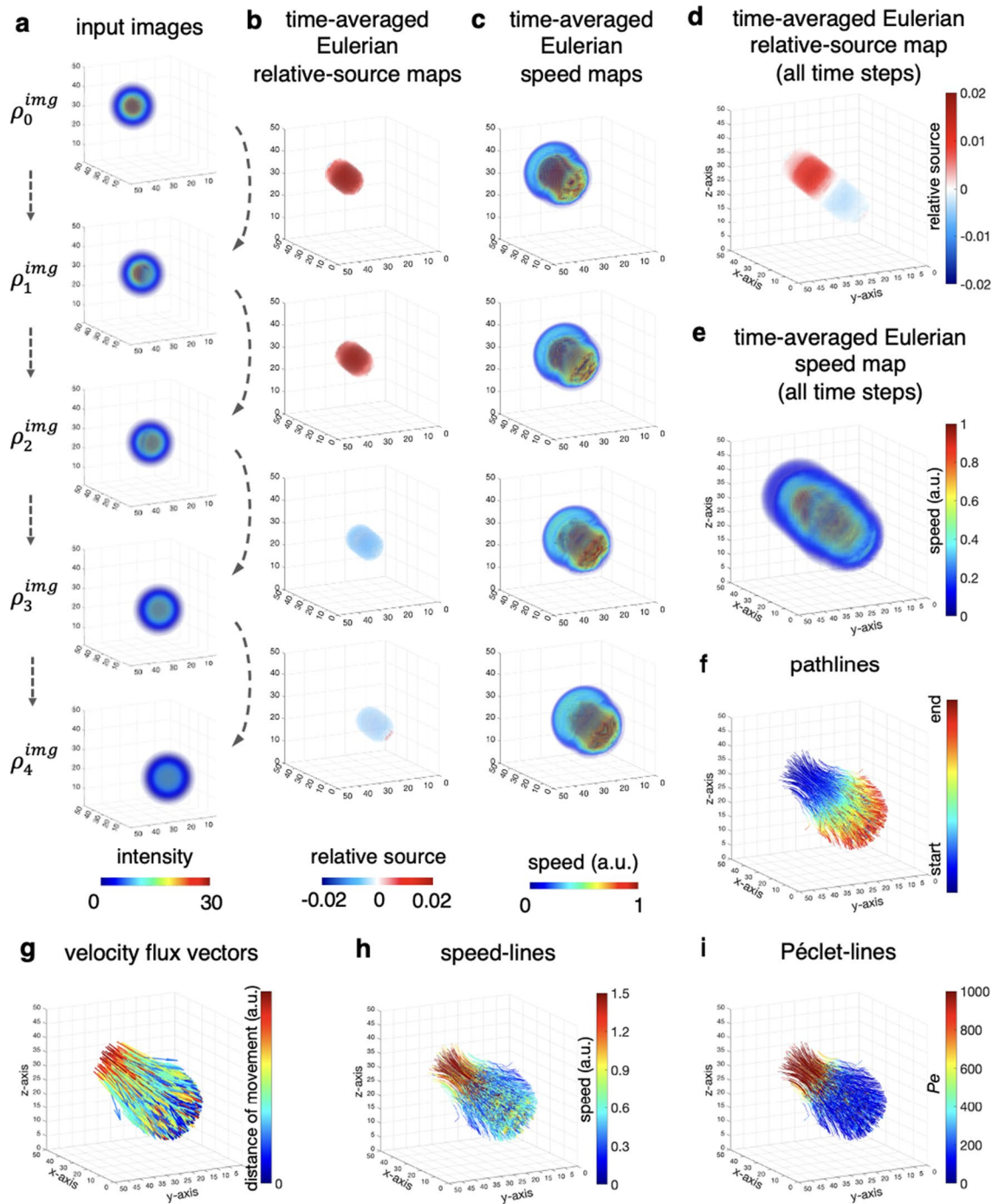
Figure 3. The First Test on 3D Gaussian Spheres. (**a**) Five successive images, shown in 3D rendering, were created from Gaussian spheres as inputs into the urOMT algorithm. In addition to advection (from top-left to bottom-right) and diffusion included in the transport process, mass was gained from ρ_0^{img} to ρ_2^{img} and was lost from ρ_2^{img} to ρ_4^{img} in the center region. (**b–e**) Under Eulerian coordinates, the time-averaged speed maps and relative source maps visualized in 3D indicate the speed and mass gain/loss distribution in the domain, respectively. (**b**) and (**c**) are derived between every pair of input images; (**d**) and (**e**) are derived between ρ_0^{img} and ρ_4^{img}. (**f–i**) Under Lagrangian coordinates, the binary trajectories of the transport are recorded by pathlines color-coded with start and end points. Connecting the start and end points of pathlines, we derive the velocity flux vectors illustrating the direction and distance of the overall movement. By endowing the pathlines with speed and Péclet (*Pe*) number, we derive the speed-lines and Péclet-lines, respectively.
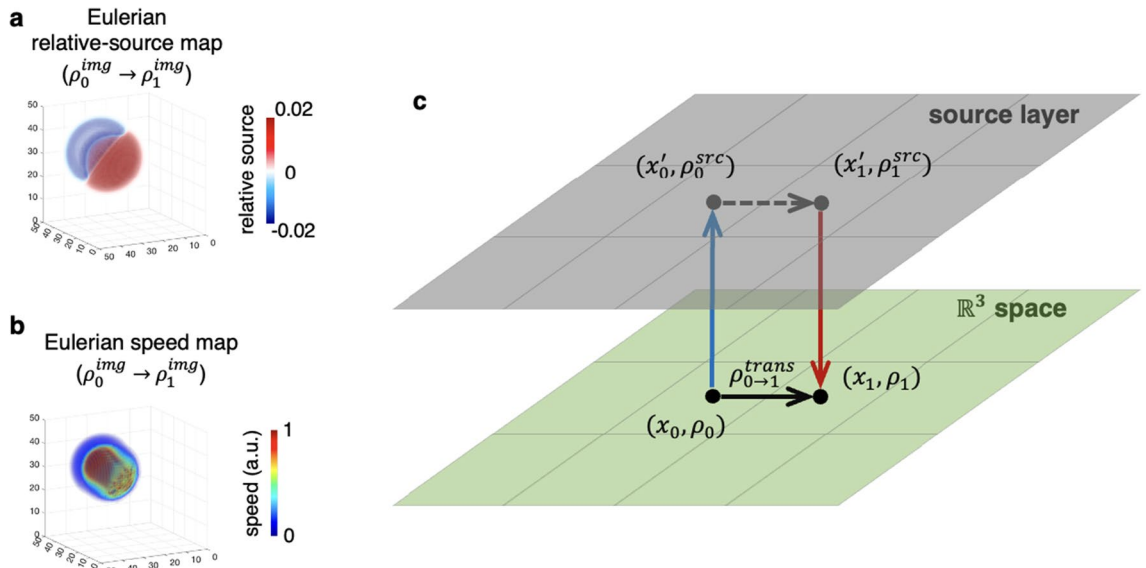
Figure 4. The Second Test on 3D Gaussian Spheres. (**a, b**) Given input images ρ_0^{img} and ρ_1^{img} visualized in Fig. 3a, Eulerian outputs, shown in 3D rendering, were returned from the urOMT analysis. (**c**) The transport in the system can be separated into two channels where the advection and diffusion take place in the $\mathbb{R}^3$ space and the relative source pushes or draws mass between the two channels.

Tests on Gaussian spheres

We first start with creating five successive 3D images from Gaussian spheres denoted as $\rho_0^{img}, \rho_1^{img}, \ldots, \rho_4^{img}$ in order to test the urOMT algorithm. These spheres were created to first gain mass and later lose mass in the center region of the spheres over time, and in addition the spheres are spatially transported forward and are also exhibiting active diffusion over time (Fig. 3a). Specifically, ρ_i was created from a 3D Gaussian function

$$G_i(x, y, z) = \frac{100}{\sqrt{2\pi}} \exp\left(-\frac{(x - 0.8i)^2 + (y - 0.8i)^2 + (z - 0.8i)^2}{2}\right), \tag{29}$$

then advection is naturally included in the whole process to reflect the forward translation of the center of the Gaussian spheres from top-left to bottom-right (Fig. 3a). We define the center region of these Gaussian spheres as the region within a radius of 1.5 of the center $[0.8i, 0.8i, 0.8i]$ which are denoted as $\chi_i(x, y, z)$ for $i = 0, \ldots, 4$. Then we apply χ_i to G_i to imitate mass gain and loss in the center region which are given by $(1 + a_i \chi_i(x, y, z))G_i(x, y, z)$ where $[a_0, a_1, a_2, a_3, a_4] = [0, 0.1, 0.2, 0.1, 0]$. We then discretize these functions by taking a uniform spatial length and fitting them into the same numerical grid of size $50 \times 50 \times 50$. Diffusion was further added to ρ_i by applying a MATLAB inbuilt 3D Gaussian filter *imgaussfilt3* to ρ_i with standard deviation $= (i + 1)\sqrt{0.2}$ for $i = 1, \ldots, 4$.

Parameter	Definition	Value for Gaussian Sphere Data Test 1	Value for Gaussian Sphere Data Test 2	Value for Brain Data
n_1	grid size in x axis	50		56
n_2	grid size in y axis	50		106
n_3	grid size in z axis	50		51
q	number of input images	5	2	15
m	number of time intervals between two input images	10		
Δt	temporal spacing	0.4		
Δx	x-axis spacing	1		
Δy	y-axis spacing	1		
Δz	z-axis spacing	1		
σ	diffusion coefficient	0.002		
α	weighting parameter for the source term	9000	10000	
β	weighting parameter for the fitting term	5000		50
χ	indicator function of the relative source r	1's in the center regions, otherwise 0's	all 1's	

Table 1. Parameters used in the urOMT algorithm.

One can imagine the five successive images as visualizing a "wormhole", which is moving forward (advection) with mass diffusing into its surrounding area. At the same time, at the center region of the wormhole mass is gained (ρ_0^{img} to ρ_2^{img}) or lost (ρ_2^{img} to ρ_4^{img}) from or to another interconnected space.

In our first test on the Gaussian sphere data, we fed all five images, $\rho_0^{img}, \rho_1^{img}, \ldots, \rho_4^{img}$, into the urOMT algorithm. From ρ_i^{img} to ρ_{i+1}^{img}, we utilize the indicators as the linear translation of χ_i to χ_{i+1} (the region within a radius of 1.5 centered at $[0.8(i + \frac{j}{m}), 0.8(i + \frac{j}{m}), 0.8(i + \frac{j}{m})]$ for $j = 0, \ldots, m, i = 0, \ldots, 3$) to only allow mass gain and loss to occur in the center regions. The parameters used in this experiment are listed in Table 1. Computations were run with MATLAB 2018b on the departmental High Performance Computing cluster at Memorial Sloan Kettering Cancer Center with Red Hat Enterprise Linux 7.5 operating system using 3 CPUs and 128GB of memory which took about 3 hours and 20 minutes.

From the Eulerian perspective, we show the time-averaged Eulerian speed maps and relative source maps between every pair of input images (Fig. 3b,c) and between ρ_0^{img} and ρ_4^{img} (Fig. 3d,e). The algorithm recognized the core regions of the spheres as having higher speed. Moreover, it successfully captured the mass gain and loss patterns during the entire process with the red color indicating the initial mass gain from ρ_0^{img} to ρ_2^{img} and the blue color indicating the later mass loss from ρ_2^{img} to ρ_4^{img}. The Eulerian relative source map is restricted in the center region of the spheres because of the indicator functions. With Lagrangian post-processing, we derived the pathlines, velocity flux vectors, speed-lines and Péclet-lines under Lagrangian coordinates (Fig. 3f-i). The pathlines exhibit what trajectories of particles would look like over time if they were placed at given initial points in the system at $t = 0$. The funnel-shape of the pathlines is a reflection of the accumulated effect of diffusion which gradually disperses the mass. The velocity flux vectors are also provided to show the direction and distance of the whole transport process. The speed-lines, i.e., pathlines endowed with speed, indicate that higher speed occurred mainly in the core of the spheres which is in agreement with the Eulerian speed map. The Péclet-lines show that initially the transport was dominated by advection and later by diffusion.

To further understand how indicators affect the results and how mass is transferred in the urOMT system, we performed a second test with only two input images $\rho_0^{img}, \rho_1^{img}$ and indicators as a constant 1 to allow mass gain/loss everywhere. The parameters used in this test are listed in Table 1. The computational runtime was about 30 minutes with the same machine and configuration as the first test. Eulerian relative source map (Fig. 4a) and Eulerian speed map (Fig. 4b) were returned from the urOMT algorithm. We observe that the top-left of the Eulerian relative source map was negative and colored in blue where the input sphere leaves and the bottom-right was positive colored in red where the sphere arrives. In other words, without any constraint on the relative source term, the Eulerian relative source map can provide global information on the transport status in terms of mass loss and arrival.

In this case, mass is freely transferred via two "channels", the real $\mathbb{R}^3$ space and the imaginary source layer (Fig. 4c). Advection and diffusion occur only in the $\mathbb{R}^3$ space. The source layer can generate infinite amount of mass and push it to the corresponding location in the $\mathbb{R}^3$ space; it can also draw a certain amount of the mass from the $\mathbb{R}^3$ space to the corresponding location in the source layer. The activities in both channels happen simultaneously. Suppose in $\mathbb{R}^3$, one would like to move mass $\rho_0 > 0$ at location x_0 to mass $\rho_1 > 0$ at location x_1, i.e., $\begin{bmatrix} (x_0, \rho_0) \\ (x_1, 0) \end{bmatrix} \Rightarrow \begin{bmatrix} (x_0, 0) \\ (x_1, \rho_1) \end{bmatrix}$. We denote x_0' and x_1' as the corresponding locations of x_0 and x_1 in the source payer, respectively; the amount of mass transported in $\mathbb{R}^3$ from x_0 to x_1 as $\rho_{0 \to 1}^{trans} > 0$; the amount of mass drawn from x_0 to x_0' as $\rho_0^{src} > 0$; and the amount of mass pushed from x_1' to x_1 as $\rho_1^{src} > 0$. Therefore, the following equations hold

$$\rho_0 = \rho_{0 \to 1}^{trans} + \rho_0^{src}, \quad \rho_1 = \rho_{0 \to 1}^{trans} + \rho_1^{src}. \tag{30}$$

The weighting parameter α in the urOMT formulation therefore balances the split of ρ_0 into $\rho_{0 \to 1}^{trans}$ and ρ_0^{src} whose effect will be demonstrated in the next section.

In this test, to move ρ_0^{img} to ρ_1^{img}, part of the mass was transported in $\mathbb{R}^3$ and part in the source layer, and these two phenomena facilitate each other. We know a priori from the creation of the data that the sphere should be transported forward and mass gain only occurs in the center region. However, allowing a global relative source, the urOMT algorithm finds an easier way to transform ρ_0^{img} into ρ_1^{img} by making use of the source layer to transfer mass.

Most importantly, removing the indicator constraint the relative source globally compensates the transport in $\mathbb{R}^3$ and is indicative of the leaving and arrival of mass. This can be very useful in the context of applications to some real dataset. As a matter of fact, a priori indicator for the source layer is sometimes unavailable due to the complexity of the system.

Application to rat brain MRI

A very useful application of the urOMT model is to quantify the transport properties of brain fluids with DCE-MRI data. It still remains debated in the scientific community how fluid and solutes are transported in brain parenchyma and how the "dirty" fluid is drained out of the brain to maintain homeostasis[30,31]. The relative source in the urOMT model may be helpful in revealing fluid and solute clearance patterns in the brain.

In this experiment, our urOMT method was applied to 3D DCE-MRI dataset derived from a 3-month-old healthy rat brain. The tracers, gadoteric acid, were injected into the cerebrospinal fluid (CSF) of the rat after the rat was anesthetized. The DCE-MRI data series of the rat brain was collected every 5 minutes and lasted for 140
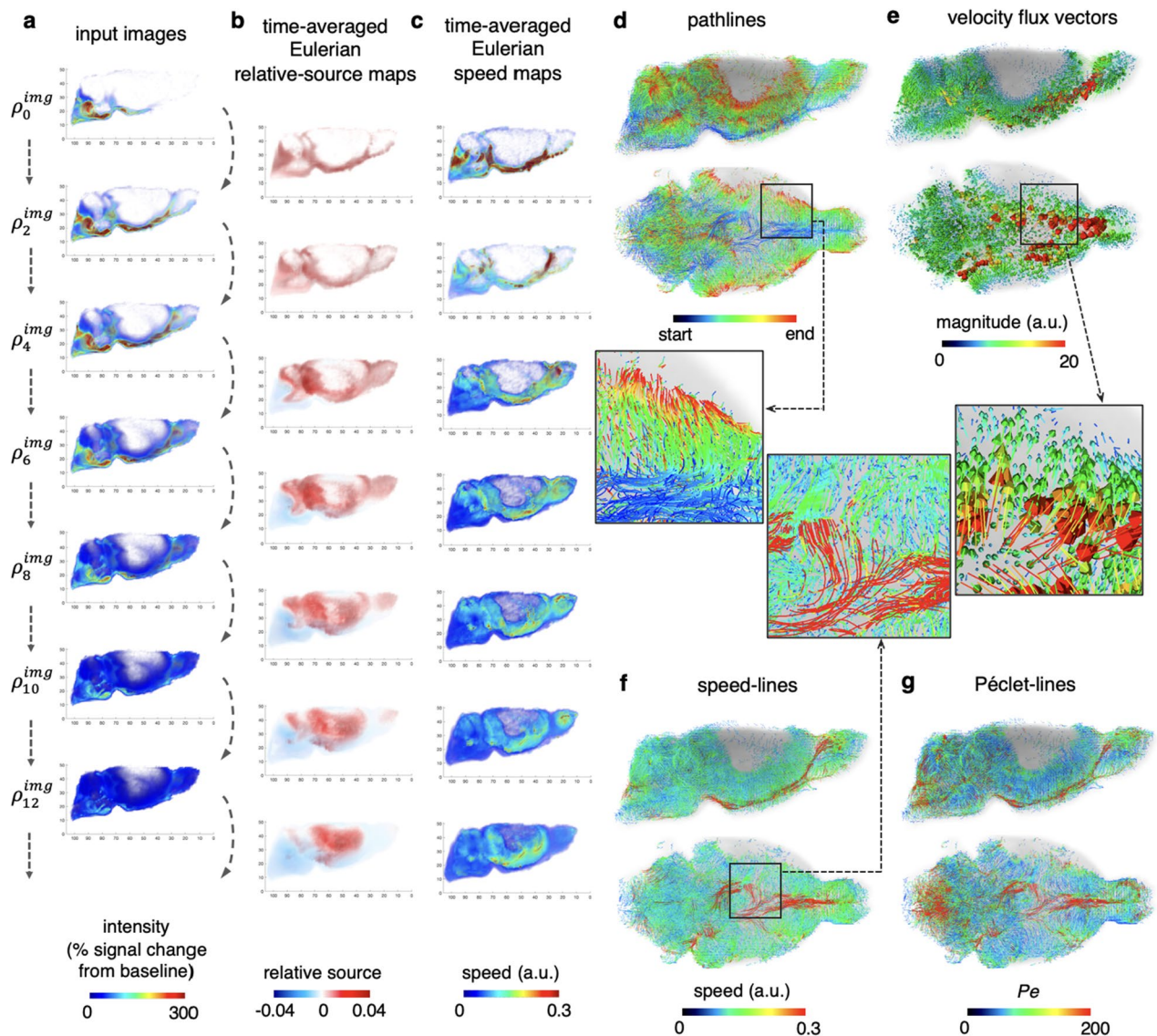
Figure 5. Application to 3D Rat Brain MRI. (**a**) Rat brain MRIs, shown in 3D rendering from the right-lateral view plane, were fed successively into the urOMT algorithm. (**b, c**) As returned outputs, the Eulerian time-averaged relative source maps and speed maps between every other image were plotted, indicating the rate of mass gain/loss and speed distribution over time, respectively. (**d–g**) Under Lagrangian coordinates, pathlines, color-coded with start and end points, show the trajectories of the tracers in brain. The speed-lines show the speed values along pathlines, and similarly Péclet-lines indicate whether the transport is advection or diffusion-dominated along pathlines. The velocity flux vectors show the direction and distance of the transport. All lines and vectors are shown from both the right-lateral and bottom views, and are overlaid on the anatomical data in gray.

minutes, ending up with 29 images in total. The MRI signal images were then processed to derive the %-signal change from the baseline to approximate the concentration images of tracers. More information of the DCE-MRI data may be found in Chen et al.[9]

In this numerical experiment, we assume that the intensity of the DCE-MRI images is proportional to the density in the urOMT model. We used every other image within the brain region as input images in order to save running time, which resulted in a total of 15 images: $\rho_0^{img}, \rho_1^{img}, \ldots, \rho_{14}^{img}$ for the urOMT algorithm (Fig. 5a). In the previous work[9], the rOMT model was applied right before the peak of the total signal intensity of the input images, i.e., $\rho_3^{img}, \ldots, \rho_{14}^{img}$ in current notation. In the present experiment, due to the introduction of the relative source term in the model we were able to include earlier frames when the total intensity was still rapidly increasing (Fig. 6a, the red dashed curve). Since the mechanism of the fluid transport in brains is complex and still under intense investigation, there is no information provided a priori for the relative source term. So we set its indicator function χ to be equal to 1 everywhere with the assumption that the entire brain system is "leaky" to allow the tracers to enter and exit the system through unknown ways. In order to derive smooth prolonged
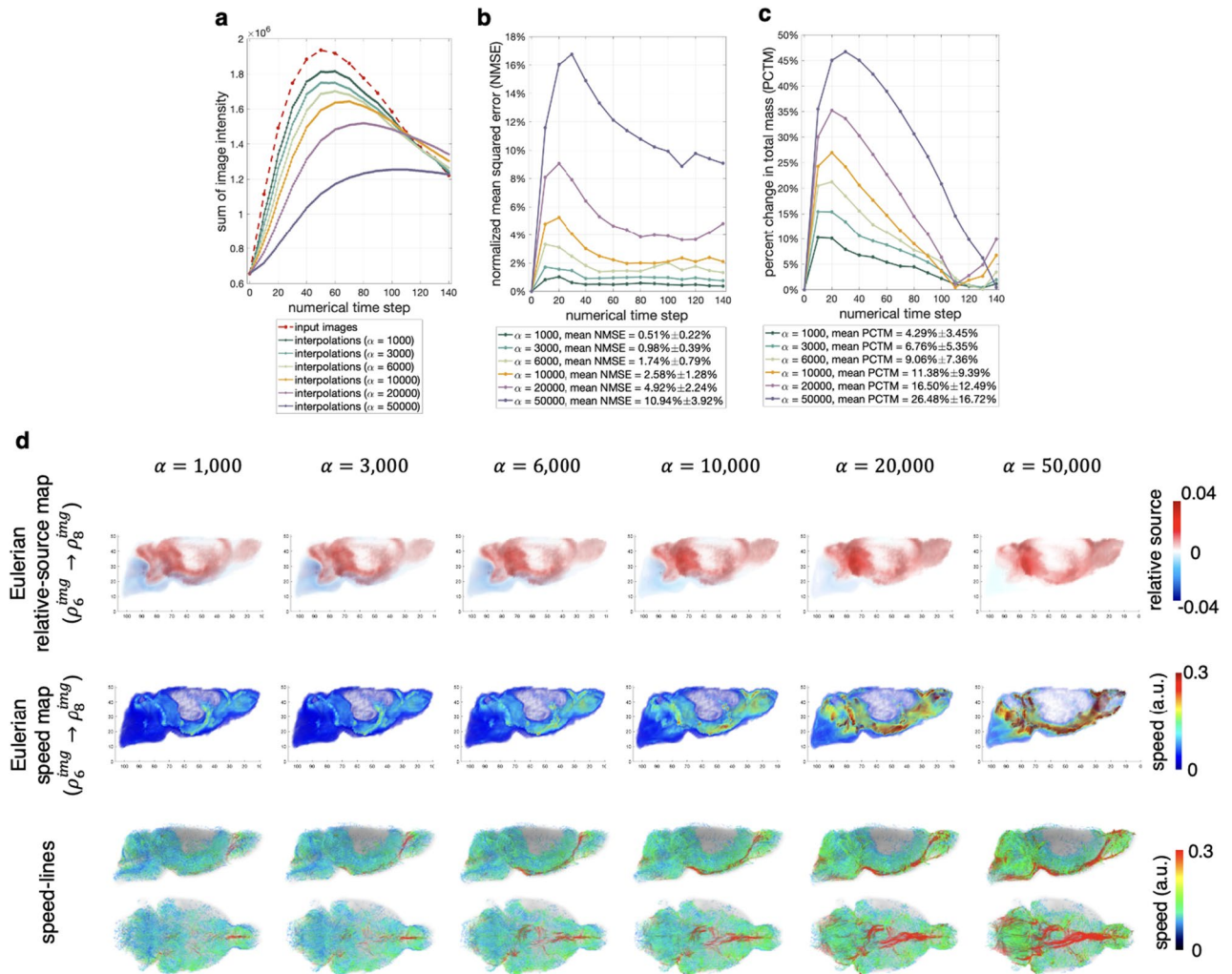
Figure 6. Examination of the Effect of the Parameter α. (**a**) The comparison of the total image intensity curve of the input images and interpolations with different α's. (**b**) The normalized mean squared error curve over the numerical steps with different α's. (**c**) The percent change in total mass curve over the numerical steps with different α's. (**d**) The comparison of the Eulerian relative source maps (first row) and Eulerian speed maps (second row) from ρ_6^{img} to ρ_8^{img} (shown in Fig. 5a) and the speed-lines (the third row) returned from the urOMT and its post-processing algorithm with different α's. The first two rows are from the right-lateral view and the third row is from both the right-lateral and bottom views.

dynamics, we used the last interpolated image from the previous numerical loop as the initial image in the next loop. The parameters used in this experiment are listed in Table 1. The computation was also run with MATLAB 2018b on our departmental High Performance Computing cluster at Memorial Sloan Kettering Cancer Center with Red Hat Enterprise Linux 7.5 operating system using 3 CPUs and 128GB of memory which took about 8 hours and 50 minutes.

We show the Eulerian relative source maps and speed maps between every other input frame (Fig. 5b-c). The Eulerian relative source maps indicate that the tracers first entered the CSF surrounding the brain causing the intensive mass gain noted in the first frames, and then subsequently moved into deeper brain tissue regions resulting in mass loss in CSF and mass gain in the brain tissue. The Eulerian speed maps indicate that the initial speed of tracers when entering the CSF was very high, and slowed when the tracers penetrated deeper, probably due to diffusion dominating the transport in the tissue. From the Lagrangian perspective, we show the path-lines of tracers starting at $t = 0$ as well as those lines endowed with speed and Péclet number (Fig. 5d, f, g). The pathlines signifies the pathways/trajectories of the tracers entering CSF and brain tissue. The speed-lines show that higher speed was mainly in the CSF and along the perivascular space of the large vessels, and further that the transport was identified as advection-dominated according to the high values in the Péclet-lines in those regions. The velocity flux vectors were also derived and demonstrated that the tracers entered the brain tissue in a symmetrical pattern about the the midline of the skull base (Fig. 5e).

Recall that in the urOMT model (3a)–(3c), the weighting parameter $\alpha > 0$ penalizes the source term in the cost function. In theory, as $\alpha \to +\infty$, r gets suppressed and this model approximates the rOMT model where

unbalanced mass gain and loss are not allowed. In the test, we used above $\alpha = 10000$. We ran further tests with different values including $\alpha = 1000, 3000, 6000, 20000$ and 50000 to demonstrate the effect of α on the results.

From the urOMT algorithm, we can compute the final interpolation $\rho_{i,m}$ for $i = 1, \ldots, q - 1$. To test the numerical performance, we define the *normalized mean squared error* (NMSE) between each pair of final interpolation $\rho_{i,m}$ and the ground truth ρ_i^{img} as

$$\frac{\|\rho_{i,m} - \rho_i^{img}\|^2}{\|\rho_i^{img}\|^2} \times 100\%. \tag{31}$$

We also define the *percent change in total mass* (PCTM) between $\rho_{i,m}$ and ρ_i^{img} as

$$\frac{|\text{sum}(\rho_{i,m}) - \text{sum}(\rho_i^{img})|}{\text{sum}(\rho_i^{img})} \times 100\% \tag{32}$$

where $\text{sum}(\cdot)$ denotes taking the sum of all entries in a vector. Both NMSE and PCTM measure the fidelity of the urOMT algorithm and the lower they are, the closer the urOMT results are to the data. However, NMSE measures the closeness of the ground truth and the interpolations in a local manner, while PCTM in a global manner. We emphasize that urOMT is a data-driven method; in other words, the transport between two images is a black box and urOMT is simply giving the most likely evolving path under a pre-defined least cost assumption. Therefore, it is difficult to define the ground truth between two input images for us to compare with.

By fixing the rest of the parameters but using various α values (1000, 3000, 6000, 10000, 20000 and 50000), we ran our novel urOMT algorithm and the post-processing procedure on the same rat brain data. In Fig. 6a, we plot the curves of the total image intensity of both input images and interpolations with different α's. From the curves, the smaller the α is, the closer the curve of interpolations is to the curve of input images. This is in agreement with the theoretical observation that the smaller the α is, the less the source term is penalized, which means more mass gain/loss is allowed in the system in order to adjust the total mass. From Fig. 6b, c, we found that a smaller α may help derive more fidelitous numerical simulations, because for both NMSE and PCTM the trend is that the larger the α is, the higher the NMSE and PCTM values are. This makes sense because when the system is highly unbalanced, and forcefully using a high α produces a nearly balanced environment which contradicts with the data setting. To further examine the effect of α on the quantitative and visual results, we plot the Eulerian relative source maps and Eulerian speed maps from ρ_6^{img} to ρ_8^{img} and speed-lines with different α's in Fig. 6d. They show that when the source term is greatly penalized with a high α, the speed is consequently elevated. For example in Fig. 6d with $\alpha = 50000$ where the model most approximates rOMT, there is very high speed transport color coded in red at the base of the brain, which could be artificial and potentially over-estimated because in such a system mass has to move more quickly in order to match the final input image. In contrast, in a system with low α where instantaneous mass gain/loss is promoted, mass does not need to transport in the same degree to match the final input image, since it can instead pull in (or push out) mass from (or into) the "invisible sink" (the source layer in Fig. 4c) via the relative source r. For example in Fig. 6d with $\alpha = 1000$, the rapid movement of tracers at the base of CSF almost disappeared because the change of the system is mostly accounted by the relative source. In general, the relative source r and the velocity field v compensate each other and their effects in the system are actively controlled by the weighting parameter α.

Discussion

In this work, we introduced an unbalanced version of the rOMT model for studying brain fluid dynamics using DCE-MRI images, which we referred to as urOMT. This method was utilized to make rOMT[8–10] more physically and biologically relevant by removing the mass conservation constraint. Specifically, the urOMT model accounts for the change of the total mass in the system by adding an independent relative source term into the formulation, while the rOMT model requires the total mass to be conserved. As discussed before, both theoretically and numerically the rOMT model can be approximated by the urOMT model when the parameter α goes to infinity. In other words, urOMT "incorporates" rOMT, and one can use urOMT with a large enough α (thereby assuring that the mass conservation condition of input images to be met) in place of the previous version of rOMT. As such the new urOMT introduced in the present work is a more powerful and flexible model for analysis of fluid transport.

The unbalanced regularized OMT problem has been proposed before, and there have been many efforts investigating into its numerical methods[19,20,22,32–35] among which most are scaling algorithms generalized from the Sinkhorn algorithm[6]. The Sinkhorn algorithm solves a balanced entropy-regularized OMT problem, but has limitations in stability and may not be practical for problems with large amount of sample points. The rOMT algorithm, which our urOMT method is built upon, is stable when the regularization term approaches zero and can deal with density distributions of larger scale. The urOMT model generalized from rOMT inherits these advantages compared to the other numerical methods for solving the unbalanced regularized OMT problem. More importantly, for practical application into studying the transport behavior of fluids in brains, both rOMT and urOMT are naturally fitted in a fluid dynamical framework where a partial differential equation in fluid dynamics is directly solved.

The urOMT method may be particularly useful for studying the cross-talk between the glymphatic system and meningeal lymphatics. See relevant work[30,36,37] for more details about how the two systems interplay. With the additional information of the relative source in our urOMT model which reveals mass gain and loss, we are now able to observe and quantify the solute and fluid entering and exiting the two systems. For example, in Fig. 5b,

at first we see mainly mass gain (colored in red) indicating that the tracers are flowing into the CSF, and then slowly we see blue color along the skull base, indicating that the tracers are either redistributing into the tissue bed or exiting via the draining lymphatic vessels. Therefore, our urOMT method has the potential to probe the clearance pattern in more depth at the level of the CSF and tissue compartments.

In the test on the rat brain DCE-MRI, we posed no spatial constraint on the relative source r and used an indicator χ all 1's given that so far there is no agreed upon answer yet on the underlying transport mechanisms and the exact drainage pathways from the brain[30,31]. Indeed from the DCE-MRI data (Fig. 5a), we did not observe a specific anatomical efflux route of the tracers out of the brain but the total intensity curve did decline in later frames (Fig. 6a, the red dashed curve). In this case, we assume that the system is leaky and the tracers can be transferred via unknown tunnels in brain (a correspondence of the source layer in Fig. 4c) to form the given images. Similarly in the widely used Toft's model[38,39] for pharmacokinetic analysis of DCE-MRI studies of tumors where the signal curves are also unbalanced, a leakage between the blood vessels and the tissue is assumed to occur everywhere in the images. The popular parameter K^{trans} returned from the Toft's model is used to quantify the local leakage of gadolinium-based tracers from the blood to the tissue[38,39].

In the urOMT algorithm, the difference between two input images is mainly captured by either the mass gain/loss rate r in the source layer or the velocity field v in $\mathbb{R}^3$, and α is the parameter balancing the two. Indeed as demonstrated by other work[19], with a decreasing α parameter the transport (characterized by the velocity field v) is being compensated by mass gain and loss (characterized by the relative source r). With the rat brain MRI dataset, we demonstrated that a low α value gives higher numerical fidelity, but produces decreased speed by allowing the relative source term to play a greater role in the dynamics. Thus, one needs to be aware of the trade-off between the fidelity and the strength of fluid flows by choosing an appropriate parameter α when applying urOMT. One approach would be to make use of the indicator function χ to restrict the behaviors of the relative source r within a certain region if the entering and exiting information of the fluid is known beforehand. Another approach is to use a time-varying α given that the total intensity curve of the input image is usually not linearly increasing or decreasing. Indeed, we plan to explore this possibility in some future work.

Other than the parameter α, there are also additional parameters worthy of examining in the future, such as the weighting parameter β for the fitting term in the cost function (4a) and the constant diffusion coefficient σ. The parameter β controls the data fidelity and is also coupled with the choice of α. Some preliminary efforts have been made in using a non-linear diffusion term in the partial differential equation (2), given the non-constant diffusion phenomena in brain fluid flows[40]. Future work includes adding a non-linear and spatially dependent diffusion term in the current urOMT formulation.

The urOMT model was largely motivated by the changing pattern of the total signal curve from DCE-MRI experiments. Specifically in rat brains, the temporal signal intensity curve typically peaks at approximately 1 hour after tracer injection, and later on, either keeps decreasing (Fig. 6a, red dashed curve) or reaches a plateau (this depends on the total amount of tracer administered), signifying that the total mass may be highly unbalanced over time in the system. Given that the DCE-MRI protocol has been widely used in cancer imaging in clinics[41–43], the urOMT method also has the potential to be applied to tumor DCE-MRI data in human experiments to investigate the tumor vasculature, and to help pave the way for new medical treatments.

Conclusions
The urOMT methodology incorporates both advection and diffusion motions into the transport process, as well as allowing for mass gain and loss in the dynamic images by introducing a relative source variable. For special cases, it may also constrain the relative source to a given region or time interval. As an extension of the rOMT model[8–10], the urOMT model removes the total mass conservation constraint, while keeping the attractive advection-diffusion framework, making it applicable to modeling the fluid flows in the brain under DCE-MRI protocol and many other real-world modeling problems in computational fluid dynamics.

Data availability
The code for the urOMT algorithm and its post-processings is available at https://github.com/xinan-nancy-chen/urOMT and was also deposited at Zenodo[44] . The synthetic data and the DCE-MRI data used in this work is also provided therein.

References
1. Monge, G. Mémoire sur la théorie des déblais et des remblais. *Hist. Acad. R. Sci. Paris* (1781).
2. Kantorovich, L. V. On a problem of monge. *CR (Doklady) Acad. Sci. URSS (NS)* **3**, 225–226 (1948).
3. Villani, C. *Topics in Optimal Transportation* (American Mathematical Soc., 2003).
4. Villani, C. *Optimal Transport: Old and New* Vol. 338 (Springer, Berlin, 2008).
5. Benamou, J.-D. & Brenier, Y. A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem. *Numer. Math.* **84**, 375–393 (2000).
6. Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, vol. 26 (2013).
7. Chen, Y., Georgiou, T. T. & Pavon, M. On the relation between optimal transport and Schrödinger bridges: A stochastic control viewpoint. *J. Optim. Theory Appl.D* **169**, 671–691 (2016).
8. Koundal, S. *et al.* Optimal mass transport with lagrangian workflow reveals advective and diffusion driven solute transport in the glymphatic system. *Sci. Rep.* **10** (2020).
9. Chen, X. *et al.* Cerebral amyloid angiopathy is associated with glymphatic transport reduction and time-delayed solute drainage along the neck arteries. *Nat. Aging* **2**, 214–223 (2022).

10. Chen, X., Tran, A. P., Elkin, R., Benveniste, H. & Tannenbaum, A. R. Visualizing fluid flows via regularized optimal mass transport with applications to neuroscience. *J. Sci. Comput.* **97**, 26. https://doi.org/10.1007/s10915-023-02337-9 (2023).
11. Robert, S. M. *et al.* The choroid plexus links innate immunity to csf dysregulation in hydrocephalus. *Cell* **186**, 764-785.e21. https://doi.org/10.1016/j.cell.2023.01.017 (2023).
12. Ozturk, B. *et al.* Continuous positive airway pressure increases csf flow and glymphatic transport. *JCI Insight* **8** (2023).
13. Chizat, L., Peyré, G., Schmitzer, B. & Vialard, F.-X. An interpolating distance between optimal transport and Fisher-Rao metrics. *Found. Comput. Math.* **18**, 1–44 (2018).
14. Chen, Y., Georgiou, T. T. & Tannenbaum, A. Interpolation of density matrices and matrix-valued measures: The unbalanced case. *Eur. J. Appl. Math.* **30**, 458–480 (2018).
15. Chizat, L., Peyré, G., Schmitzer, B. & Vialard, F.-X. Unbalanced optimal transport: Dynamic and Kantorovich formulations. *J. Funct. Anal.* **274**, 3090–3123 (2018).
16. Liero, M., Mielke, A. & Savaré, G. Optimal entropy-transport problems and a new Hellinger-Kantorovich distance between positive measures. *Invent. Math.* **211**, 969–1117 (2018).
17. Piccoli, B. & Rossi, F. Generalized Wasserstein distance and its application to transport equations with source. *Arch. Ration. Mech. Anal.* **211**, 335–358 (2014).
18. Benamou, J.-D. Numerical resolution of an unbalanced mass transport problem. *ESAIM Math. Model. Numer. Anal.* **37**, 851–868 (2010).
19. Chizat, L., Peyré, G., Schmitzer, B. & Vialard, F.-X. Scaling algorithms for unbalanced optimal transport problems. *Math. Comput.* **87**, 2563–2609 (2018).
20. Feydy, J., Charlier, B., Vialard, F.-X. & Peyré, G. Optimal transport for diffeomorphic registration. In *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, 291–299 (Springer, 2017).
21. Maas, J., Rumpf, M., Schönlieb, C. & Simon, S. A generalized model for optimal transport of images including dissipation and density modulation (2015). arXiv:1504.01988.
22. Séjourné, T., Peyré, G. & Vialard, F.-X. Unbalanced optimal transport, from theory to numerics. arXiv preprint arXiv:2211.08775 (2022).
23. Zhan, F. *et al.* Unbalanced feature transport for exemplar-based image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15028–15038 (2021).
24. Choi, J., Choi, J. & Kang, M. Generative modeling through the semi-dual formulation of unbalanced optimal transport. arXiv preprint arXiv:2305.14777 (2023).
25. Lombardi, D. & Maitre, E. Eulerian models and algorithms for unbalanced optimal transport. *ESAIM Math. Model. Numer. Anal.* **49**, 1717–1744 (2015).
26. Gallouët, T., Laborde, M. & Monsaingeon, L. An unbalanced optimal transport splitting scheme for general advection-reaction-diffusion problems (2017). arXiv:1704.04541.
27. Yang, K. D. & Uhler, C. Scalable unbalanced optimal transport using generative adversarial networks. arXiv preprint arXiv:1810.11447 (2018).
28. Lee, J., Bertrand, N. P. & Rozell, C. J. Parallel unbalanced optimal transport regularization for large scale imaging problems. arXiv preprint arXiv:1909.00149 (2019).
29. Steklova, K. & Haber, E. Joint hydrogeophysical inversion: state estimation for seawater intrusion models in 3d. *Comput. Geosci.* **21**, 75–94 (2017).
30. Bohr, T. *et al.* The glymphatic system: Current understanding and modeling. *iScience* **25**, 104987 (2022).
31. Zhao, L., Tannenbaum, A., Bakker, E. N. & Benveniste, H. Physiology of glymphatic solute transport and waste clearance from the brain. *Physiology* **37**, 349–362 (2022).
32. Pham, K., Le, K., Ho, N., Pham, T. & Bui, H. On unbalanced optimal transport: An analysis of sinkhorn algorithm. In *International Conference on Machine Learning*, 7673–7682 (PMLR, 2020).
33. Lübeck, F. *et al.* Neural unbalanced optimal transport via cycle-consistent semi-couplings. arXiv preprint arXiv:2209.15621 (2022).
34. Baradat, A. & Lavenant, H. Regularized unbalanced optimal transport as entropy minimization with respect to branching brownian motion. arXiv preprint arXiv:2111.01666 (2021).
35. Lacombe, T. An homogeneous unbalanced regularized optimal transport model with applications to optimal transport with boundary. In *International Conference on Artificial Intelligence and Statistics*, 7311–7330 (PMLR, 2023).
36. Hablitz, L. M. & Nedergaard, M. The glymphatic system. *Curr. Biol.* **31**, R1371–R1375 (2021).
37. Benveniste, H. *et al.* The glymphatic system and waste clearance with brain aging: a review. *Gerontology* **65**, 106–119 (2019).
38. Tofts, P. S. Modeling tracer kinetics in dynamic gd-dtpa mr imaging. *J. Magn. Resonance Imaging* **7**, 91–101 (1997).
39. Tofts, P. S. *et al.* Estimating kinetic parameters from dynamic contrast-enhanced t1-weighted mri of a diffusable tracer: Standardized quantities and symbols. *J. Magn. Resonance Imaging* **10**, 223–232 (1999).
40. Xu, K., Chen, X., Benveniste, H. & Tannenbaum, A. Regularized optimal mass transport with nonlinear diffusion. arXiv **2301.03428** (2023).
41. Yankeelov, T. E. & Gore, J. C. Dynamic contrast enhanced magnetic resonance imaging in oncology: Theory, data acquisition, analysis, and examples. *Curr. Med. Imaging* **3**, 91–107 (2007).
42. Türkbey, B., Thomasson, D., Pang, Y., Bernardo, M. & Choyke, P. L. The role of dynamic contrast-enhanced mri in cancer diagnosis and treatment. *Diagnostic and Interventional Radiology (Ankara, Turkey)* **16**, 186 (2010).
43. Hylton, N. Dynamic contrast-enhanced magnetic resonance imaging as an imaging biomarker. *J. Clin. Oncol.* **24**, 3293–3298 (2006).
44. Chen, X. Unbalanced Regularized Optimal Mass Transport (urOMT). *Zenodo.* https://doi.org/10.5281/zenodo.7839019 (2023).

Acknowledgements

Author contributions

The numerical method and code development were done by X.C. The theory of OMT and other material preparations were contributed by A.R.T. The DCE-MRI data collection and processing were performed by H.B. The first draft of the manuscript was written by X.C. and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to X.C.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.