A Novel Anisotropic Fast Marching Method and its Application to Blood Flow Computation in Phase-contrast MRI

M. Schwenke1; A. Hennemuth1; B. Fischer2,3; O. Friman1
1Fraunhofer MEVIS – Project Group Image Registration, Lübeck, Germany;
2Fraunhofer MEVIS – Institute for Medical Image Computing, Bremen, Germany;
3Institute of Mathematics and Image Computing – University of Lübeck, Lübeck, Germany

1. Introduction

Fast Marching (FM) methods are widely applied to segmentation and analysis in the field of medical imagery [1]. Example applications are centerline computations of tubular structures, such as vessels, and image segmentation methods based on minimal paths and front evolution [2, 3].

The original FM method is a numerical scheme for solving the so-called Eikonal differential equation, whose solution can be interpreted as the monotone evolution of a front on a domain with a spatially varying speed function. The speed function in this case does not depend on the direction of front evolution. A generalization of the original FM method is obtained by considering anisotropic speed functions, meaning that the speed at each point is direction dependent. In cases of vessel centerline detection, an anisotropic speed function could propagate the front faster along the vessel direction than orthogonally [4].

In medical imaging, anisotropic FM methods have been previously applied to brain connectivity analysis [5] and tumor growth modeling [6]. In [7], we presented a novel application of anisotropic FM to compute blood flow trajectories in Phase-Contrast-MRI (PC MRI).

PC MRI is able to measure blood flow, and the acquired images can be reconstructed into velocity vector fields of the blood flow. Spatiotemporal image analysis can then be used to investigate the dynamic processes and characteristics of blood flow [8]. There is an increasing clinical interest in applying PC MRI for assessing stenoses, aneurysms, and heart valves, and for surgical planning for congenital heart disease. Streamline methods [9] are traditionally used to trace blood flow trajectories in the PC MRI data. These methods ignore the intrinsic uncertainty due to noise in the measured vector field, and may give the user a false impression of accuracy. Previously, a sequential Monte Carlo method was proposed to investigate the probability distribution of flow trajectories [10]. In [7], we proposed modeling blood flow trajectories as minimal paths with respect to an anisotropic speed function. Based on an estimation of the PC MRI measurement uncertainty, the method computes a bundle of trajectories that indicates the uncertainty about the path that a particle would follow in the given flow vector field.

This contribution presents a fast novel numerical scheme for anisotropic FM. In our prior publication [7], Recursive FM [6] was utilized to solve the anisotropic minimal path problem. Recursive FM is well suited for problems with small anisotropy ratios, i.e., the ratio of the largest to the smallest eigenvalue of the tensor, meaning the fastest...
to slowest directional travel speed. However, it is not as well suited for the blood flow trajectory application in which strong anisotropy ratios are present. The novel numerical scheme is shown to perform much better in this setting. The presentation of the anisotropic minimal path methods in Section 3.1 and the modeling of blood flow trajectories as minimal paths in Section 3.2 follows the description in our prior publication [7]. The results section shows a new comparison of flow patterns in aneurysms to flow patterns of a healthy subject.

2. Methods and Materials

2.1 Minimal Paths

Let $\gamma(s): \mathbb{R} \rightarrow \mathbb{R}^3$ be a path connecting two points in 3D space and $\gamma'(s)$ its tangent. Let $\mathcal{M}(x)$ denote a metric tensor represented by a symmetric positive definite matrix that defines a spatially and directionally varying traveling cost, i.e., the inverse of the speed. The total cost of a path connecting two points $a$ and $b$ is

$$J(\gamma) = \int_{s_0}^{s_f} F(\gamma(s), \gamma'(s))ds,$$  \hspace{1cm} (1)

where the local cost of traveling at a certain position $\gamma(s)$ along the path is defined as

$$F(\gamma(s), \gamma'(s)) = \sqrt{\gamma'(s)^T \mathcal{M}(\gamma(s)) \gamma'(s)}.$$  \hspace{1cm} (2)

The path having the minimum cost among all possible paths $\Gamma_{ab}$ between $a$ and $b$ is the minimal path. Minimal paths from end points are reconstructed following the tangent direction using standard numerical methods such as Euler’s, Heun’s or Runge-Kutta’s method.

2.2 Modeling Blood Flow Trajectories as Minimal Paths

A 3D vector field of blood flow velocities can be reconstructed from the PC MRI images (Fig. 2a). Traditionally, vector field visualization methods, such as streamlines, are used to compute blood flow trajectories [9]. These methods, however, do not consider the uncertainty in the measured flow vectors due to image noise, and the streamlines may therefore give a false impression of precision. Friman et al. [10] showed that each component of a measured flow vector $v$ is well approximated by a multivariate Gaussian distribution around a true velocity $\mu$:

$$v \in \mathcal{N}(\mu, \sigma)$$

and this constitutes the foundation of Dijkstra’s shortest path and FM algorithms that construct $U$ from smaller to greater values. The minimal path tangents satisfy $\gamma' = \mathcal{M}^{-1} \nabla U$ and the minimal paths from end points are reconstructed following the tangent direction using standard numerical methods such as Euler’s, Heun’s or Runge-Kutta’s method.

where $v_{\text{enc}}$ is a known velocity-encoding sequence parameter and SNR the signal-to-noise-ratio of the PC MRI images within vessels. Typical values for the parameters are $v_{\text{enc}} = 1500$ mm/s and SNR = 10, resulting in a standard deviation of about $\sigma = 70$ mm/s. Normal flow velocities, as a reference, range between 0 mm/s and at peak systole in the aorta. A PC MRI flow vector $\gamma = [v_x, v_y, v_z]^T$ can be considered as drawn from a multivariate Gaussian distribution.

To capture uncertainty in flow trajectories traced in PC MRI data, we view the data as a field of 3D multivariate Gaussian distributions instead of considering deterministic vectors. We incorporate the uncertainty of the measured velocities into a metric tensor for the anisotropic FM and then solve the minimal path problem in the ensuing metric space to find flow trajectories. The inverse of the metric tensor is constructed as

$$\mathcal{M}^{-1} = vv^T + \sigma^2 I,$$  \hspace{1cm} (6)

(see Fig. 1 for a schematic illustration). The steps for calculating flow connectivity maps and flow trajectories as minimal paths based on PC MRI data are shown in Fig. 2. Figure 2b shows the tensor field constructed based on a PC MRI data set of an aorta (Fig. 2a). Figure 2c shows the value function $U$ from a starting point in the aortic arch. From every point in the aorta, a minimal path that represents the flow trajectory to that point can be found by backtracking on $U$ to the starting point. To differentiate more likely paths from less likely paths, an approach similar to [5] in the context of DTI is adopted, in which the alignment of the minimal path tangent with the major eigenvector of the metric tensor $\mathcal{M}$ is evaluated. In the current case, the alignment can be measured by $C(\gamma, \gamma') = |v(\gamma')^T \gamma|$ and the mean alignment $\mu(x)$ along the minimal path to the point $x$ is

$$\mu(x) = \frac{1}{U(x)} \int_0^{U(x)} C(\gamma(s), \gamma'(s))ds.$$  \hspace{1cm} (7)

Fig. 1  Construction of a metric tensor from a given PC-MRI velocity vector. Illustration taken from [11].
distribution of possible flow trajectories under the influence of image noise. Figure 2e shows the minimal paths with highest connectivity, representing the most likely paths under the influence of noise in the PC MRI images.

In a flow application, it may be of interest to trace trajectories either forwards or backwards. The direction of the FM propagation is restricted as follows: After estimating the $U$ value for a certain point, an approximation to the minimal path tangent at that point is known, and the propagation is restricted to points for which the dot product with the flow direction is strictly positive or negative, respectively.

### 2.3 Iterative Anisotropic FM

Applying the standard Dijkstra-like FM solver to anisotropic problems introduces errors beyond the discretization error. The error is dependent on the anisotropy, i.e., the ratio of the largest to the smallest eigenvalue of the tensor. In applications like brain connectivity analysis based on Diffusion Tensor Imaging [5], anisotropy in the data is physically bounded and comparably low with ratios of about 10. To compute trajectories in PC MRI, anisotropy ratios are much larger, ranging from 30 to 100, depending on the noise present in the data. The standard FM solver is not applicable in this case. Anisotropic FM solvers can be found in literature, e.g., Recursive FM [6], and the Buffered FM [12]. Recursive FM [6] was used in our prior publications [7, 11]. Fast Sweeping [13] is another method to compute the solution of the anisotropic Eikonal equation. The method uses a Gauss-Seidel iteration with alternating sweeping directions. Fast Sweeping is not a FM method and is used here as a means of ground truth and computing time reference. In contrast to the FM solver, it solves the problem on the whole domain, whereas the FM solver introduces a means to restrict computations automatically to the set of points needed for the given problem (e.g., the computation of a path between two points or a path from a source with a certain maximal path length), resulting in faster computations. In [14], the ordered upwind method was proposed for aniso-

```plaintext
IterativeFM(T, ε, U_max, itermax):
/* Initialization */
T₁ ← Ø, T₂ ← Ø
U(x) ← 0 for x ∈ T, ∞ otherwise
/* Standard FM loop */
while T ≠ Ø do
    x* ← argmin U(x)
    T ← T \ {x*}
    A ← A ∪ {x*}
    if U(x*) > U_max then
        break loop
end if
UpdateNeighbors(x*)
T₁ ← T₁ \ {x*} \ U(x*)
/* Improvement loop */
iter = iter + 1
while not converged and iter < itermax do
    converged ← TRUE
    while T₁ ≠ Ø do
        x* ← argmin U(x)
        T₁ ← T₁ \ {x*}
        u ← U(x*)
        if u ≤ U(x*) then
            T₂ ← T₂ \ {x*} \ U(x*)
            converged ← FALSE
        end if
        Update(x*)
    end while
end if
/* Advance trial front */
if x* ∈ T and U(x*) ≤ U_max then
    T ← T \ {x*}
    A ← A ∪ {x*}
    T₁ ← T₁ \ U(x*)
end if
end while
/* Update step */
for all y ∈ N(x) 
    U(y) ← Update(y)
if y ∈ T and y ∉ A then
    T ← T \ {y}
end if
end for
Update(x): returning u
u = \min \left\{ \sum_{i=1}^{n} \alpha_i \right\}_{\alpha_i \in S(x)} \ U(x, \alpha),
where n = 3, S is the set of all 3-groups of pairwise
neighborhood neighbors of x and
U(x, \alpha) ≈ U(\bar{g}) + \sum_{i=1}^{n} \alpha_i U(x_i) \approx U(\bar{g}) + \int_{\gamma} F(\gamma, \gamma') ds
```

Fig. 3 Iterative FM
Anisotropic speed functions in the context of optimal control problems. It is basically a discretization with which the standard FM solver can be applied for anisotropic speed functions. However, the accuracy of the discretization decreases with increasing anisotropy.

The main idea of our Iterative FM is to build an iterative method based on the standard FM. Because standard FM is able to resolve some anisotropy, applying it iteratively resolves more and more anisotropy in each iteration (see Fig. 4 for an example). As convergence sets in from smaller to larger U values, the number of points that need recomputation in an iteration decreases rapidly with the number of iterations. We use a local discretization similar to [15]. In contrast to [15] we also use non-alive points, i.e., points not yet with a known final value but with a first estimate. This results in faster convergence because of higher anisotropy that can be resolved with this approach. The discretization is based on the entire 26-voxel neighborhood in 3D, which results in considerably reduced directional error in comparison to a 6-voxel neighborhood discretization used, e.g., in [5] for brain connectivity analysis. The Iterative FM algorithm is outlined in Figure 3. Like standard FM, the TRIAL points, i.e., the points for which the final U value had not yet been found, are managed in a sorted heap denoted by T and the set of ALIVE points with final value, denoted by A. The method consists of two parts: First, the standard FM is run until the minimal value of U in the TRIAL set has a value larger than $U_{\text{max}}$, while all TRIAL and ALIVE points are collected in another sorted heap structure $I_1$. Next, an iteration is started in which the U-values of all points in $I_1$ are recomputed, starting from small values to big values of U. If U at a point changes in comparison to the last iteration, the point plus its neighbors are moved to a heap $I_2$ to ensure that U is recomputed for these points in the next iteration. As the values at a point strictly decrease monotonically in each iteration, the sub-domain for which the U value is less than the specified $U_{\text{max}}$ needs to be expanded during the iteration: If the iteration reaches a point with neighbors that have yet not been visited, the values for these neighbors are computed, and U values are checked to determine if they are smaller than the specified $U_{\text{max}}$. If the value at a point is smaller than $U_{\text{max}}$, the expansion of the domain is also done for the neighbors of this point. This is done until all points having unvisited neighbors have a value greater than $U_{\text{max}}$.

Figure 4 gives an example of the result of the method after a certain number of iterations. In the case of an isotropic tensor field, Iterative FM only needs one additional computation of U for each point compared to the standard FM solver.

2.4 Data

Artificial 3D test tensor fields with constant anisotropy were constructed to validate and compare the different anisotropic FM methods. Anisotropy ratios, i.e., the factor between the major and smaller eigenvalues of the metric tensor, of 100 and 1000 were used. The three PC MRI data sets, used in [7], demonstrate the blood flow computation using anisotropic FM: a data set covering the aorta of a healthy subject (P1), a patient with an aortic aneurysm (P2), and a data set covering the carotids of a patient with an aneurysm in the carotids (P3) (Fig. 6a–c). P1 and P2 have a spatial resolution of $1.7 \times 1.7 \times 3$ mm$^3$, and P3 has a spatial resolution of $0.86 \times 0.086 \times 1.1$ mm$^3$.

3. Results

Iterative FM, Recursive FM and Fast Sweeping were implemented in C++ and experiments were carried out on a PC with 2.4 GHz CPU and 2 GB memory.
3.1 Computing Time of Iterative FM

This section compares Fast Sweeping [13], Recursive FM [6] and the novel Iterative FM method using the artificial test data. The methods converge to exactly the same solution (when using the same local discretization), so the ground truth can be computed given enough iterations, and the focus is on comparing computing time to attain a certain accuracy. The Buffered FM [12] method was not included because of the lack of a publicly available implementation. To control the accuracy of the methods, a local error parameter $\varepsilon$ was introduced to control whether a local change of value is propagated to neighbors. For a fair comparison, Fast Sweeping was implemented to recompute $U$ only if a neighbor changed value. The first experiment used two artificial test data sets of size $25 \times 25 \times 25$ and with anisotropy of 100 and 1000, respectively. A point in the center of the image was selected as the starting point, and the value function $U$ was calculated for all points. Time-versus-accuracy plots are shown in Figure 5. First, one can note that the computing time depends on the anisotropy. Fast Sweeping and Iterative FM show comparable computing times although Iterative FM is slightly faster than Fast Sweeping. Because of deep recursion, Recursive FM becomes very time-consuming when increasing the desired accuracy (decreasing $\varepsilon$). Furthermore, the experiments show that Iterative FM can also be used to resolve even stronger anisotropy (see Figure 5b for results on a test data set with anisotropy ratio of 1000). The main application area for FM is applications on a domain of unknown extent, such as a volume between two endpoints of a path. The minimal sub-domain to solve the path problem between two points is found automatically with the FM approach. Nevertheless, as Figure 5 shows, the novel Iterative FM performs well in solving for $U$ on the entire image.

3.2 Blood Flow Trajectory Computations

Figure 6 shows a visualization of trajectory bundles for the PC MRI data sets. The computations of the value function took about five seconds using Iterative FM. The computation of the trajectory bundle took about a second. The first row shows the connectivity distribution for a single particle of interest for each data set. For the first and second column, the interest lies in the origin of the particle, i.e. the computed trajectories are directed backwards in time. A comparison of healthy flow to pathologic flow patterns can be done using these two columns. For a connectivity interval of 0.9–1.0, the flow pattern do not differ severely, comparing Figure 6d to Figure 6e. The same holds to some extent for the connectivity interval of 0.8–0.9 (Figs. 6g and 6h). At a connectivity interval of 0.7–0.8, the flow patterns differ significantly: the aortic aneurysm in the second column introduces possible backward flow of blood from the descending aorta up to the supra aortal arteries, which is not present in the healthy aorta (compare Figure 6k to Figure 6j). A traditional streamlining approach to compute a single trajectory for the particle of interest would not have revealed this difference, because the trajectory would lie inside the trajectory bundles.
shown in Figures 6d and 6e – it would follow the path of the most likely trajectory computed with the presented method. The third column shows the results for the cardiac data set. The trajectories for a connectivity interval of 0.9–1.0 show that the main direction of flow is straight upwards. Despite the relatively large radius for the vessel on the right, the inflow is rather low (Fig. 6f). The trajectories for a connectivity interval of 0.8–0.9 reveal a disturbed flow at the beginning of the rightmost vessel which is introduced by the aneurysm (Fig. 6i). Figure 6f finally shows trajectories following the right vessel at a connectivity interval of 0.7–0.8.

4. Discussion and Future Work

A novel anisotropic FM scheme is presented and applied to a method to model blood flow trajectories as minimal paths. Anisotropic FM methods are generally more computationally complex than the isotropic counterpart, and computing time depends on the anisotropy ratio in the tensor field. In the presented application, because of strong anisotropy, computing time can be critical. Therefore, the proposed Iterative FM was developed. The method has an advantage in terms of computational speed compared to previous algorithms. It was furthermore shown that the Iterative FM scheme is able to solve highly anisotropic front propagation and thereby minimal path problems in our tests even faster than Fast Sweeping – a state-of-the-art solver for such problems utilizing a Gauß-Seidel iteration scheme with alternating sweeping direction. Whether this holds in general needs to be investigated more formally in future work. Furthermore, the Iterative FM may allow parallel computations, and we will investigate this in future work, because FM methods suffer from the inherently sequential computations that are not well suited to make use of today’s parallel hardware. The Iterative FM method is applicable to any problem that reduces to solving an anisotropic Eikonal equation, for example wave-front propagation through anisotropic media.

The presented blood flow analysis method intends to make the clinician aware of the uncertainty in the data and to consider it in the process of decision making. The trajectory bundles and flow connectivity maps give a static visualization of dynamic flow processes that might facilitate interpretation and recording of a patient’s flow patterns. Future work will include the extension for 4D data, because PC MRI images frequently also have a temporal dimension. More experimental work, such as with physical flow phantoms or flow simulations, to further validate the method is also required. Example clinical applications include investigating the probability for different stroke embolization pathways, to investigate flow in aneurysms and through heart valves, and to provide visualizations for surgical planning in congenital heart disease.

Acknowledgment

The authors would like to thank Dr. Michael Markl, Department of Radiology, Medical Physics at University Medical Center Freiburg, Germany, for providing the PC MRI data.

References


