



Multiplicative Operator Splittings in Nonlinear Diffusion: From Spatial Splitting to Multiple Timesteps

DANNY BARASH* AND TAMAR SCHLICK

*Department of Chemistry and the Courant Institute of Mathematical Sciences, New York University,
New York 10012, USA
barash@biomath.nyu.edu.*

MOSHE ISRAELI AND RON KIMMEL

Computer Science Department, Technion, Israel Institute of Technology, Haifa 32000, Israel

Abstract. Operator splitting is a powerful concept used in many diversified fields of applied mathematics for the design of effective numerical schemes. Following the success of the additive operator splitting (AOS) in performing an efficient nonlinear diffusion filtering on digital images, we analyze the possibility of using multiplicative operator splittings to process images from different perspectives.

We start by examining the potential of using fractional step methods to design a multiplicative operator splitting as an alternative to AOS schemes. By means of a Strang splitting, we attempt to use numerical schemes that are known to be more accurate in linear diffusion processes and apply them on images. Initially we implement the Crank-Nicolson and DuFort-Frankel schemes to diffuse noisy signals in one dimension and devise a simple extrapolation that enables the Crank-Nicolson to be used with high accuracy on these signals. We then combine the Crank-Nicolson in 1D with various multiplicative operator splittings to process images. Based on these ideas we obtain some interesting results. However, from the practical standpoint, due to the computational expenses associated with these schemes and the questionable benefits in applying them to perform nonlinear diffusion filtering when using long timesteps, we conclude that AOS schemes are simple and efficient compared to these alternatives.

We then examine the potential utility of using multiple timestep methods combined with AOS schemes, as means to expedite the diffusion process. These methods were developed for molecular dynamics applications and are used efficiently in biomolecular simulations. The idea is to split the forces exerted on atoms into different classes according to their behavior in time, and assign longer timesteps to nonlocal, slowly-varying forces such as the Coulomb and van der Waals interactions, whereas the local forces like bond and angle are treated with smaller timesteps. Multiple timestep integrators can be derived from the Trotter factorization, a decomposition that bears a strong resemblance to a Strang splitting. Both formulations decompose the time propagator into trilateral products to construct multiplicative operator splittings which are second order in time, with the possibility of extending the factorization to higher order expansions. While a Strang splitting is a decomposition across spatial dimensions, where each dimension is subsequently treated with a fractional step, the multiple timestep method is a decomposition across scales. Thus, multiple timestep methods are a realization of the multiplicative operator splitting idea. For certain nonlinear diffusion coefficients with favorable properties, we show that a simple multiple timestep method can improve the diffusion process.

Keywords: nonlinear diffusion, multiplicative operator splittings, additive operator splittings, multiple timestep methods

*To whom correspondence should be addressed.

1. Introduction

There are various applications of nonlinear diffusion filtering [16, 31] in image processing. Such ‘filters’ can be used for denoising, gap completion and computer aided quality control among many other tasks. These kind of applications demand high processing capabilities. The balance between accuracy, stability and computational efficiency, as well as splittings across different dimensions and scales to guide the diffusion process are therefore important issues in the design of such filters. As processing capabilities advance, these considerations are expected to play an increasing role in future applications.

In this paper, we first present numerical schemes for diffusion processes that have been used in other areas of application and examine them as alternative extensions to Weickert-Romeny-Viergever’s additive operator splitting (AOS) schemes [32] to process images. Historically, AOS schemes were developed for the Navier-Stokes equations [12, 13]. In image processing applications AOS schemes are efficient and reliable, in the sense that they permit the use of larger time steps, whereas the straight-forward explicit schemes that were proposed originally in Perona and Malik’s classical paper [16] are restricted to small time steps in order to ensure stability. However, the AOS schemes are limited in their accuracy to first order in time even for the linear case. We therefore examine the possibility of increasing the accuracy in one-dimension, along with preserving this increase in accuracy by a suitable split-operator scheme. The splittings are across dimensions and are multiplicative in nature, although a combined additive-multiplicative operator splitting (AMOS) is suggested since the additivity is essential to make the splitting symmetric. Our approach closely resembles the use of alternating direction implicit (ADI) type schemes [15], which are second order in time for the linear case. We show that as we increase the time steps the gain in accuracy can be visualized, with an efficiency tradeoff. Although the immediate advantages over the use of simple and efficient AOS schemes are not apparent, future applications may benefit from these alternative schemes.

Next, we consider a different type of multiplicative operator splittings, across scales. These can be used in conjunction with AOS schemes or any other splitting schemes across dimensions. These type of splittings, known as multiple timestep (MTS) methods, have been developed extensively for use in the area of biomolec-

ular simulations (see [20] for a general discussion). They were introduced [25, 26] in an effort to reduce the computational cost of molecular simulations and have been actively pursued since the reversible multiple timestep methods [29] were developed. It was shown in [29] and subsequent work that a Trotter expansion of the Liouville propagator can lead to designing MTS integrators with favorable properties. The Trotter factorization is essentially a multiplicative operator splitting and therefore all ideas discussed in this paper, either in splitting operators across spatial dimensions or by splitting operators across time scales, belong to the general class of multiplicative operator splittings.

Nonlinear diffusion filtering is a continuous filter, formulated as a partial-differential-equation (PDE). The filter operation is practically performed by solving the nonlinear PDE numerically. Related PDE approaches can be found [17, 19, 21], as well as connections to certain nonlinear digital filters that offer noniterative ways of performing edge-preserving smoothing [6, 27]. For example, the bilateral filter [27], is closely related to the geometrical framework in [10, 22], as explained in [1]. For illustration, Fig. 2 demonstrates two approaches of performing edge-preserving smoothing on the original image in Fig. 1. The result of using several iterations of nonlinear diffusion filtering with a 3×3 kernel size and the result of the noniterative bilateral filtering procedure with an extended kernel performed on a test image is similar but not identical [1]. In some applications one might prefer to use a middle-way approach (i.e., a single or very few long timesteps with a 3×3 kernel) for integrating the nonlinear diffusion PDE. The middle way approach is considered in this paper and leads



Figure 1. Original image: Laplace.



Figure 2. Edge-preserving smoothing: Anisotropic diffusion with 20 time-steps of $\tau = 1.0$ (left) and Gaussian bilateral filtering with a 30×30 window size, $\sigma_D = 5.0$ and $\sigma_R = 30.0$ (right). σ_D and σ_R are bilateral filtering parameters, see [27] for details. The one-step bilateral filtering can be viewed as an extreme example of using a nonlinear diffusion-like process with a very long timestep, by extending the size of the kernel [1].

us to examine how various numerical schemes perform on the nonlinear diffusion PDE when using long timesteps.

The outline of the paper is as follows. Section 2 presents the continuous model used throughout the paper for applying nonlinear diffusion as a filter. Section 3 illustrates one-dimensional schemes that are the building blocks for higher dimensions, by splitting the evolution operator across dimensions. In Section 4, extensions of these one-dimensional schemes to higher dimensions are discussed and the motivation for using operator splitting schemes is given. Section 5 provides the operator splitting schemes which have been proposed by Weickert et al. [32], all of which are accurate to first order in time for the linear case. Motivation for examining alternative methods by constructing multiplicative operator splitting schemes is given. Consequently, in Section 6, two operator splitting schemes that preserve second-order for the linear case are introduced. The performances of all operator splitting schemes for nonlinear image diffusion are compared. In Section 7 the idea of multiple timestep (MTS) methods, a realization of multiplicative operator splitting across scales that is borrowed from the field of molecular dynamics, is introduced. In molecular dynamics, these type of methods have been developed on the basis of splitting the forces into classes according to their range of interaction. We examine the possibility of using MTS methods in nonlinear diffusion, by providing an example and suggesting other strategies to exploit their use. Section 8 concludes this paper.

2. Nonlinear Diffusion Filtering

Let us first provide a model for nonlinear diffusion in image filtering. We briefly describe the filter proposed by Catté et al. [5]. The CLMC filter leads to well-posed, mathematically correct approach for image selective smoothing that was used in [32] as a benchmark for studying various numerical schemes. The basic equation which governs nonlinear diffusion filtering is

$$\frac{\partial u}{\partial t} = \nabla \cdot (g(|\nabla u_\sigma|^2) \nabla u), \quad (1)$$

where $u(x, t)$ is a filtered version of the original image. The original image $f(x)$ is given as the initial condition

$$u(x, 0) = f(x), \quad (2)$$

and reflecting boundary conditions are used

$$\frac{\partial u}{\partial n} = 0 \text{ on } \partial\Omega, \quad (3)$$

where n is the normal to the image boundary $\partial\Omega$.

The goal of selective smoothing in edge-preserving applications is to reduce smoothing across edges. In order to achieve this goal, the diffusivity g is chosen as a rapidly decreasing function of the gradient magnitude (edge indicator). Specifically, the following form for the diffusivity is suggested in the CLMC filter

$$g(s) = \begin{cases} 1 & (s \leq 0) \\ 1 - \exp\left(\frac{-3.315}{(s/\lambda)^4}\right) & (s > 0), \end{cases} \quad (4)$$

where $\lambda = 10.0$ throughout this paper. In addition, CLMC suggest at each time step a presmoothing mechanism, in which the image u is convolved with a Gaussian of standard deviation σ to obtain u_σ . This can be achieved by solving the linear diffusion filtering ($g \equiv 1$)

$$\frac{\partial u_\sigma}{\partial t} = \nabla \cdot (\nabla u_\sigma), \quad (5)$$

for a very small time step of size $T = \sigma^2/2$. This step is called regularization, or presmoothing, and can be approximated by any of the splitting schemes that will be mentioned in the paper (as well as a direct convolution of u with a Gaussian kernel). For example, a simple locally one-dimensional (LOD) scheme is a convenient choice. In the remaining of this paper, $\sigma = 0.25$ is chosen for the presmoothing, except when quantitative comparisons are performed and presmoothing is excluded.

The continuous image $f(x)$ can be considered as a discrete image, in particular a vector $f \in \mathbb{R}^N$ whose

components f_i display the grey values at each pixel. Pixel i corresponds to the location x_i and h is the spatial grid spacing. Discrete times $t_k = k\tau$ are considered, where τ is the time step size. u_i^k denotes an approximation to $u(x_i, t_k)$. Having obtained u_σ , which from now on is referred to as u_i^k , the gradient magnitude can be approximated (in one-dimension) by a central difference scheme

$$|\nabla u_i^k|^2 = \frac{1}{2} \sum_{p,q \in \mathcal{N}(i)} \left(\frac{u_p^k - u_q^k}{2h} \right)^2, \quad (6)$$

where $\mathcal{N}(i)$ is the set of all neighbors of a pixel i . Boundary pixels have only inner pixels as neighbors, as a result of the boundary conditions. In addition, closely following the notation used in [32], the diffusivities in their discrete form will be denoted by $g_i^k = g(u_i^k)$. In the next sections, numerical schemes are presented for the implementation of nonlinear diffusion filtering.

3. One-Dimensional Schemes

This section briefly describes the one-dimensional explicit and semi-implicit schemes, before we explore the Crank-Nicolson and DuFort-Frankel schemes. It is mentioned how all these schemes satisfy discrete nonlinear diffusion scale-spaces criteria, and in particular the accuracy of these schemes is discussed. For more details and theoretical considerations regarding the framework for discrete nonlinear diffusion scale-spaces, the reader is referred to [31, 32].

It follows from (1) that the basic equation which governs one-dimensional nonlinear diffusion filtering is

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left(g(|\nabla u_\sigma|^2) \frac{\partial u}{\partial x} \right). \quad (7)$$

A simple numerical scheme for solving this equation numerically was suggested by Perona and Malik [16] (an independent pioneering work that originated this approach, based on functional minimization, can be found in [17]). The simple scheme uses the following discretization

$$\frac{u_i^{k+1} - u_i^k}{\tau} = \sum_{j \in \mathcal{N}(i)} \frac{g_j^k + g_i^k}{2h^2} (u_j^k - u_i^k), \quad (8)$$

where $\mathcal{N}(i)$ is the set of two neighbors of i , one neighbor for the boundary pixels. A compact way of

writing this scheme is

$$\frac{u^{k+1} - u^k}{\tau} = A(u^k)u^k, \quad (9)$$

where u^k is a signal vector of size N and $A(u^k) = (a_{ij}(u^k))$ is an $N \times N$ matrix whose elements are given by

$$a_{ij}(u^k) = \begin{cases} \frac{g_i^k + g_j^k}{2h^2} & j \in \mathcal{N}(i), \\ -\sum_{n \in \mathcal{N}(i)} \frac{g_i^k + g_n^k}{2h^2} & j = i, \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

Isolating u^{k+1} on the left hand side, we obtain

$$u^{k+1} = (I + \tau A(u^k))u^k. \quad (11)$$

This scheme is known as an *explicit scheme*, since u^{k+1} is obtained explicitly from u^k without a matrix inversion. This scheme is simple, straight-forward, and computationally cheap because only matrix-vector multiplications are required. However, it is conditionally stable and therefore limited to small time steps. A way to analyze numerical schemes in our context is to verify that they satisfy six criteria in order to create a discrete scale-space [32]. The matrix on the right hand side (in this case, $I + \tau A(u^k)$) needs to satisfy continuity in its argument, symmetry, unit row sum, nonnegativity, positive diagonal, and irreducibility. It follows that these conditions are satisfied for (11) if $\tau < \frac{1}{2}$ (in one-dimension, for $g \leq 1$), assuming $h = 1$ (see [32] for exact details). This means that implementation of (11) is restricted by small time steps, and even though each iteration by itself is computationally cheap, as a whole, the efficiency for applying the filter can be improved. The improvement comes by a different numerical scheme

$$\frac{u^{k+1} - u^k}{\tau} = A(u^k)u^{k+1}, \quad (12)$$

Rearranging terms, so that u^{k+1} is on the left hand side and u^k is on the right hand side, we obtain

$$(I - \tau A(u^k))u^{k+1} = u^k. \quad (13)$$

This scheme is known as a *semi-implicit scheme*, since u^{k+1} is obtained implicitly from u^k by inverting a

matrix. Although a matrix inversion is in general an expensive $O(N^3)$ operation, the matrix in Eq. (13) is tridiagonal, which can be inverted efficiently using the Thomas algorithm which is $O(N)$. Furthermore, the scheme is unconditionally stable, and in the discrete scale-space framework one can verify that it satisfies all six criteria. However, both the explicit scheme and the semi-implicit scheme are only first order in time. A scheme which is a combination of (11) and (13) and is second order in time for the linear case is the Crank-Nicolson scheme

$$\left(I - \frac{\tau}{2}A(u^k)\right)u^{k+1} = \left(I + \frac{\tau}{2}A(u^k)\right)u^k. \quad (14)$$

Another candidate scheme to try and achieve higher accuracy is the DuFort-Frankel method [8]. However, its inconsistency results (see Fig. 5) in a scheme that is not reliable from a certain time step onwards. Nevertheless, an extended DuFort-Frankel in higher dimensions that averages the fluctuations at higher time steps might perform well in the anisotropic cases like the Beltrami framework [22], or coherence enhancement [31] as well as for the Perona-Malik [16] or TV [17] (which are simplifications of the Beltrami). Experimental results with all schemes are shown in Figs. 3–7, in which a 1D cross-section of a natural image was taken (Fig. 3) and edge-preserving smoothing was applied using small and large time steps. It is seen in Fig. 4 (right) that the Forward-Euler scheme becomes unstable for larger time steps. Reducing the time-step by two orders of magnitude can recover an edge-preserved smoothed signal (Fig. 4 left), but this is

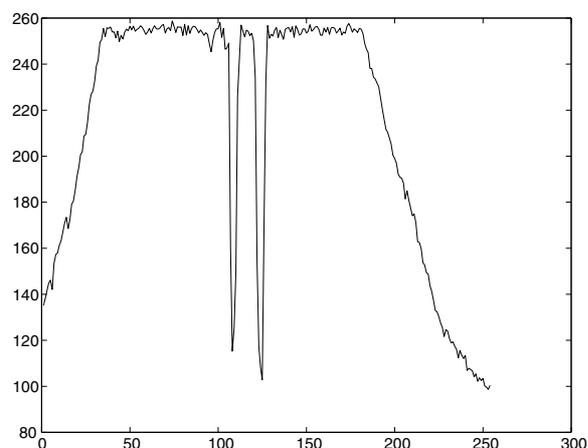


Figure 3. Original noisy signal.

Table 1. l_2 norm error estimation.

τ	Linear CN	Nonlin CN	Extrapolation
0.4	0.0079	0.052	0.0262
0.2	0.002	0.024	0.0068
0.1	$4.94 \cdot 10^{-4}$	0.0113	0.0024
0.05	$1.22 \cdot 10^{-4}$	0.0052	$5.6 \cdot 10^{-4}$
0.025	$2.9 \cdot 10^{-5}$	0.0022	$1.18 \cdot 10^{-4}$
0.0125	$5.81 \cdot 10^{-6}$	$7.34 \cdot 10^{-4}$	$2.37 \cdot 10^{-5}$
0.00625	0	0	0

inefficient. We are left with Backward-Euler and Crank-Nicolson for obtaining a robust nonlinear diffused signal at large time steps. An l_2 norm error comparison between the different output signals (see Section 6 on how this is calculated for images) reveals that in the nonlinear case, the 1D Crank-Nicolson scheme without extrapolation remains first-order accurate in time. This is because the nonlinear diffusivity term, calculated at a specific time step, interferes with achieving higher order accuracy in time. In order to retain second-order accuracy, extrapolation is needed such that the diffusivity is calculated according to two levels of time step. Table 1 indicates that a simple extrapolation along with the Crank-Nicolson, in the form of $g_i^{\text{new}} = 2 \cdot g_i^{\text{new}} - g_i^{\text{old}}$ for each time step, can boost the accuracy. We will refer to some more involved extrapolation procedures, such as the Douglas Jones predictor-corrector method proposed in [31], in Section 6.

4. Higher-Dimensional Schemes

This section builds upon the one-dimensional semi-implicit scheme (13) and the Crank-Nicolson scheme to construct schemes for higher dimensions. It follows from (1) that the basic equation which governs m -dimensional nonlinear diffusion filters is

$$\frac{\partial u}{\partial t} = \sum_{l=1}^m \frac{\partial}{\partial x_l} \left(g(|\nabla u_\sigma|^2) \frac{\partial u}{\partial x_l} \right). \quad (15)$$

And, a straight forward extension to the one-dimensional semi-implicit scheme (13) is

$$\left(I - \tau \sum_{l=1}^m A_l(u^k)\right)u^{k+1} = u^k, \quad (16)$$

where the matrix $A_l(u^k)$ corresponds to the derivatives along the l -th coordinate axis. It is shown in [32] that

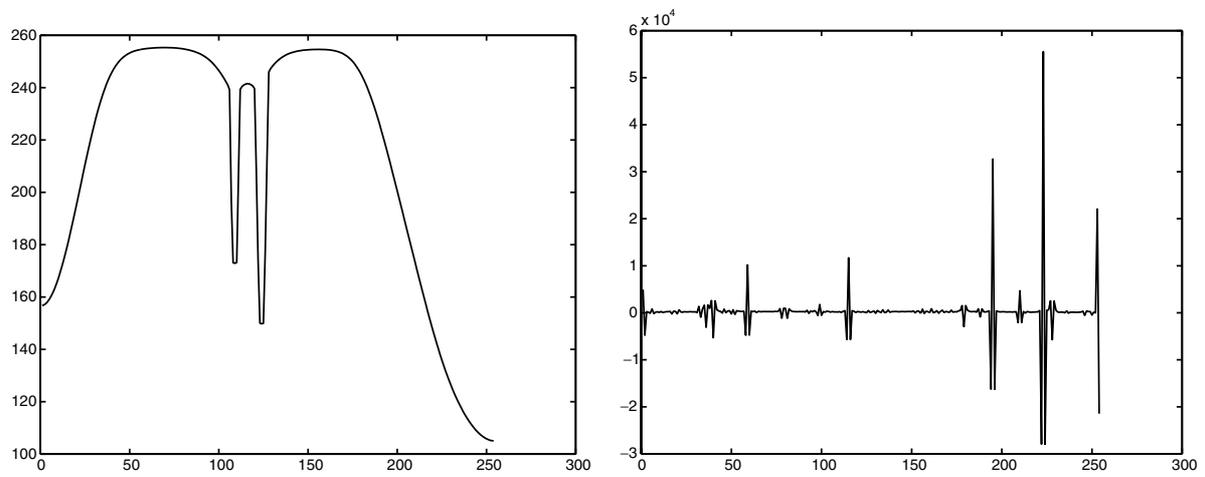


Figure 4. Explicit scheme (Forward-Euler). Left: 100 time steps of $\tau = 0.5$. Right: 5 time steps of $\tau = 10.0$.

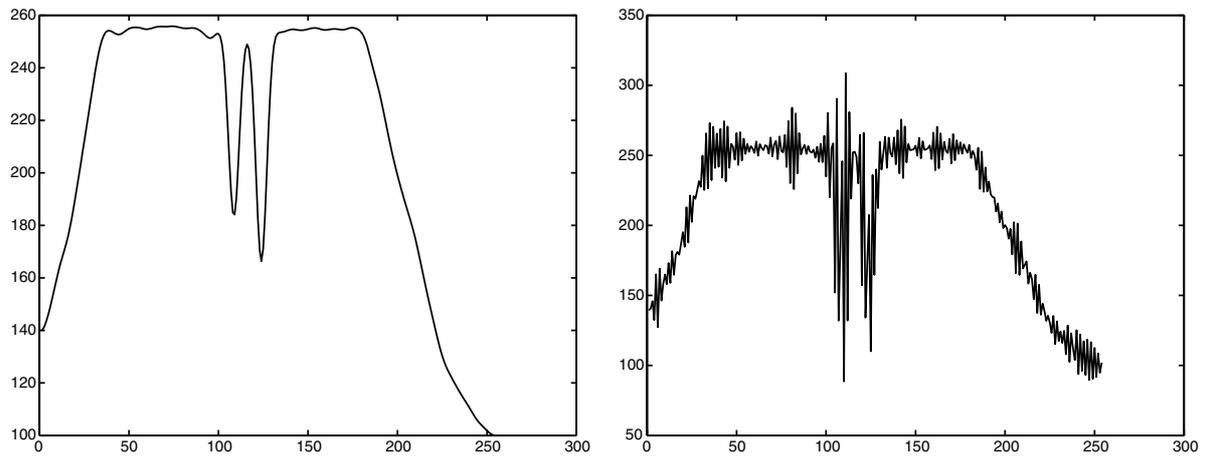


Figure 5. DuFort-Frankel. Left: 8 time steps of $\tau = 0.4$. Right: 4 time steps of $\tau = 0.8$.

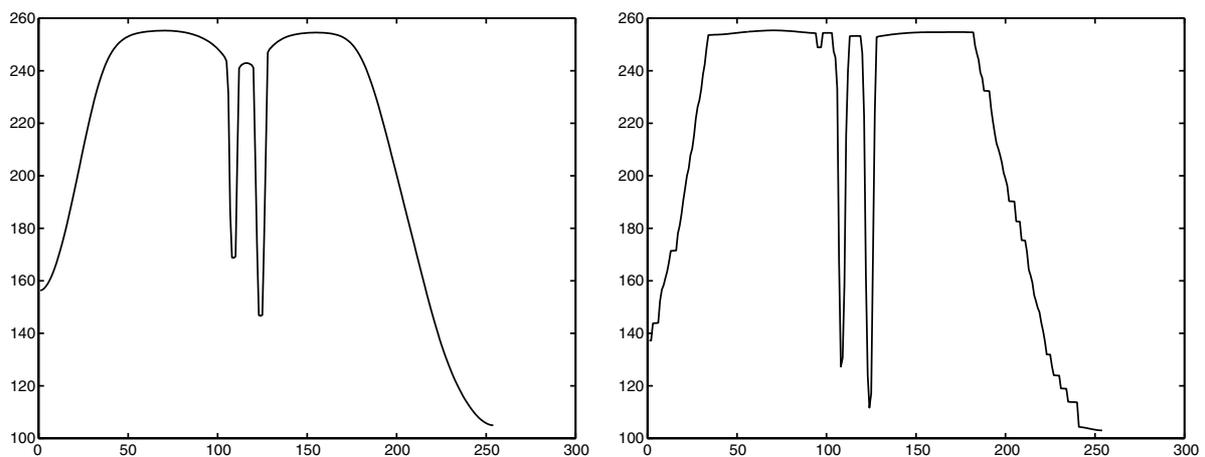


Figure 6. Semi-implicit scheme (Backward Euler). Left: 5 time steps of $\tau = 10.0$. Right: same as left, except the diffusivity $g(s) = (1 + 3s^8)^{-1}$ ($s > 0$) was used.

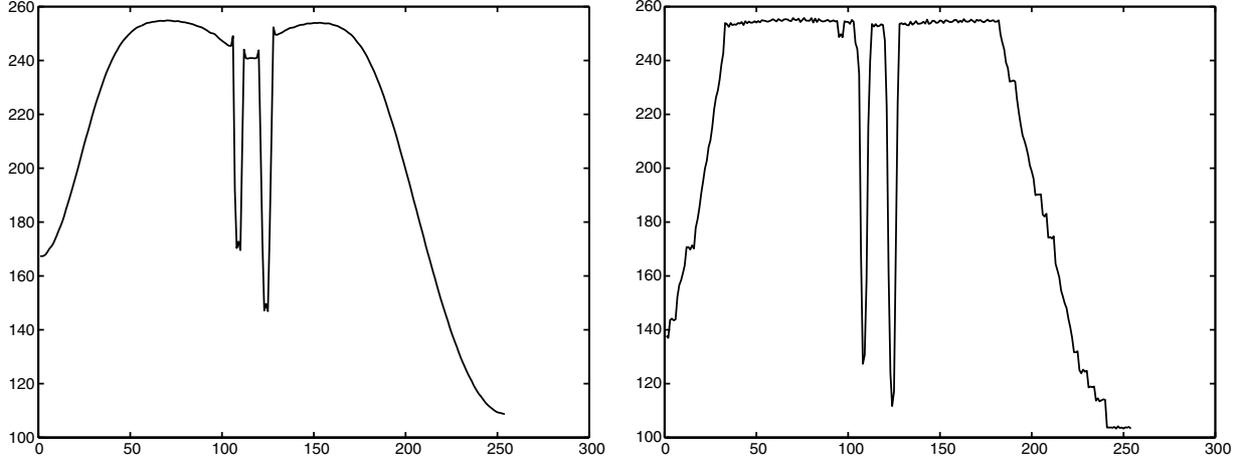


Figure 7. Crank-Nicolson. Left: 5 time steps of $\tau = 10.0$. Right: same as left, except the diffusivity $g(s) = (1 + 3s^8)^{-1}$ ($s > 0$) was used.

the m -dimensional semi-implicit scheme unconditionally satisfies all the requirements of the discrete scale-space. However, the accuracy of the scheme in (16) is limited because it is built upon a one-dimensional scheme that is only first order in time. An extension to second-order in the linear case, can be built upon the Crank-Nicolson

$$\left(I - \frac{\tau}{2} \sum_{l=1}^m A_l(u^k) \right) u^{k+1} = \left(I + \frac{\tau}{2} \sum_{l=1}^m A_l(u^k) \right) u^k \quad (17)$$

It is worthwhile noticing that a drawback of implicit schemes when moving to higher dimensions is in the efficiency of (17): the matrix $\sum_{l=1}^m A_l(u^k)$ is no longer tridiagonal and therefore the matrix inversion at each time step is costly. This occurrence in higher-dimensional diffusion equations has been known since the early days of numerical solutions to parabolic PDEs. The work of Peaceman and Rachford [15] is a famous example for overcoming this problem by *splitting methods* [11, 14, 35]. For simplicity, let us consider $m = 2$ (two-dimensions) for the time being, noting that it is possible to extend splitting methods to three and higher dimensions. In addition, let us assume the case of a linear diffusion equation, $g = \alpha$, where α is constant. We start from the two-dimensional linear diffusion equation

$$\frac{\partial u}{\partial t} = \alpha \left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} \right). \quad (18)$$

The scheme in (17) now reads

$$\begin{aligned} & \left(I - \frac{\alpha\tau}{2} \left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} \right) \right) u^{k+1} \\ & = \left(I + \frac{\alpha\tau}{2} \left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} \right) \right) u^k. \end{aligned} \quad (19)$$

However, this scheme amounts to inverting a non-tridiagonal matrix at each time step, which is inefficient. The *alternating direction implicit* (ADI) scheme [15] suggests approximating the scheme in (19) the following way

$$\begin{aligned} & \left(I - \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_1^2} \right) \left(I - \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_2^2} \right) u^{k+1} \\ & = \left(I + \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_1^2} \right) \left(I + \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_2^2} \right) u^k. \end{aligned} \quad (20)$$

It is now possible to perform two half time steps, splitting the two dimensions such that in each half time step one of the two dimensions is treated implicitly

$$\begin{aligned} & \left(I - \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_1^2} \right) u^{k*} = \left(I + \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_2^2} \right) u^k \\ & \left(I - \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_2^2} \right) u^{k+1} = \left(I + \frac{\alpha\tau}{2} \frac{\partial^2 u}{\partial x_1^2} \right) u^{k*}, \end{aligned} \quad (21)$$

where k^* is an intermediate time step. The ADI scheme is both efficient, since at each time step a tridiagonal matrix inversion is performed, and accurate to second order in time. Our goal is to seek a splitting scheme for

the nonlinear case (15), that will be as good as the ADI scheme for the linear case. More precisely, it should amount to inverting tridiagonal matrices, unconditionally satisfy all discrete scale-space requirements, and retain the time accuracy which was achieved before the splitting by starting from accurate one-dimensional schemes.

5. Operator Splitting Schemes

Before we introduce more accurate splitting schemes for solving (15), let us review the first-order accurate splitting schemes which have been proposed in [32]. The simplest splitting scheme that might be considered is the *locally one-dimensional* (LOD) scheme

$$\prod_{l=1}^m (I - \tau A_l(u^k)) u^{k+1} = u^k, \quad (22)$$

which belongs to the general class of *multiplicative operator splitting* schemes. It is the most efficient and straight-forward for implementation. However, the main drawback of the LOD scheme is that the system matrix in (22) is non-symmetric, which violates one of the criteria for discrete diffusion scale-spaces as proposed in [32]. Because of the non-commutativity of the operators A_l , the order of applying these operators can affect the final result. For example, the filtered two-dimensional image will not be the same after a rotation by 90 degrees. Figure 8 illustrates this disadvantage and motivates the search for a symmetric splitting which

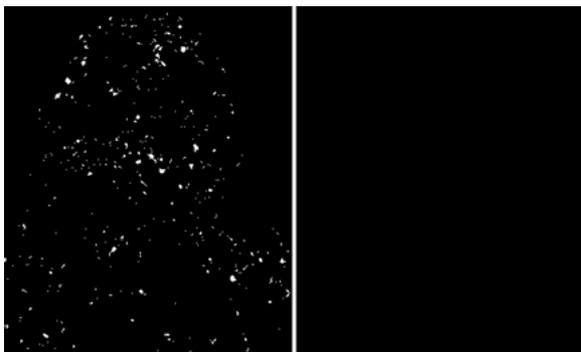


Figure 8. The difference between applying first the operator which corresponds to the x -axis and then the operator which corresponds to the y -axis and vice versa, on the original image in Fig. 1. The LOD scheme (left) is sensitive to the order, whereas the AOS scheme (right) is independent of the order. Nonlinear diffusion filtering was performed with 20 time-steps of $\tau = 1.0$.

does not suffer from this deficiency. It is worthwhile noticing that the splitting suggested in (22), when applied to any of the one-dimensional schemes discussed in Section 3, results in a multi-dimensional scheme that is first-order accurate in time.

The splitting operator scheme proposed in [32] as the method of choice, the *additive operator splitting* (AOS), is

$$u^{k+1} = \frac{1}{m} \sum_{l=1}^m (I - m\tau A_l(u^k))^{-1} u^k. \quad (23)$$

Unlike the LOD scheme the AOS scheme is symmetric, see Fig. 8, and unconditionally satisfies all discrete diffusion scale-space requirements. It is almost as efficient as the LOD scheme; instead of applying the operators in a pipeline, one calculates the operators independently and then sums them up at each time step. It is therefore a reliable and efficient scheme. However, similar to the LOD scheme it is first-order accurate in time. Moreover, it is less accurate than the LOD scheme since operators of type $(I - m\tau A_l)^{-1}$ that are used in the AOS scheme, represent one-dimensional diffusions with a step size $m\tau$, whereas operators of the type $(I - \tau A_l)^{-1}$ that are used in the LOD scheme possess smaller time steps when stepping in one dimension.

Let us illustrate how in some potential applications, the better accuracy of the LOD scheme can be noticeable. We compare the AOS and the LOD schemes' performances on the veneer image in Fig. 9. Figure 10 is the reference image, after applying nonlinear diffusion filtering with 256 time steps of $\tau = 0.78$ each. We now keep the time constant ($T = 200$) and decrease the number of iterations while increasing the duration τ of each time step accordingly. In the reference

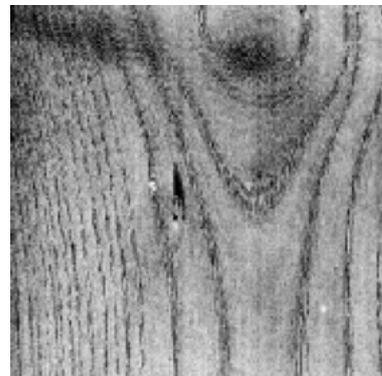


Figure 9. Original image of a veneer, taken from [34] with permission.

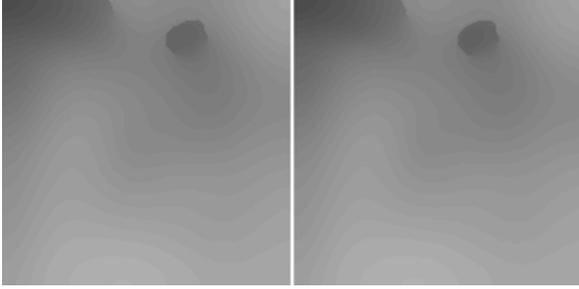


Figure 10. Reference image: LOD vs. AOS, nonlinear diffusion filtering with 256 time-steps of $\tau = 0.78$.

image, Fig. 10, the LOD and the AOS schemes results are practically identical. As we increase the time steps, the results start to deviate from the reference by a certain amount which is related to the accuracy of the scheme. Figures 11 and 12 demonstrate nonlinear diffusion filtering approximated by two time steps of $\tau = 100$ and eventually one time step of $\tau = 200$. The LOD scheme is found to be more accurate than the AOS scheme, as it is closer to the reference filtered image. These results motivate us to look for more accurate schemes, as well as symmetric accurate ones,

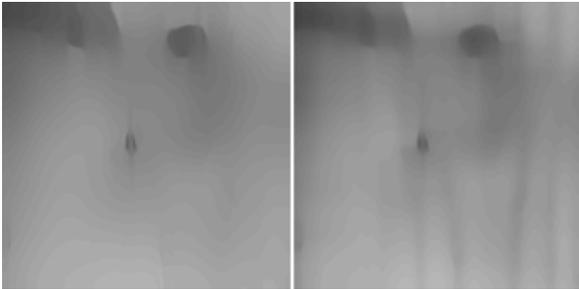


Figure 11. Checking visual accuracy: LOD vs. AOS, nonlinear diffusion filtering with two time-steps of $\tau = 100.0$.

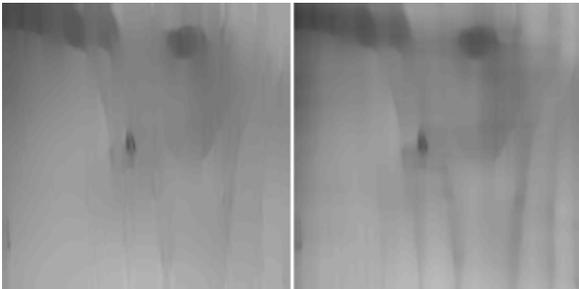


Figure 12. Checking visual accuracy: LOD vs. AOS, nonlinear diffusion filtering with one time-step of $\tau = 200.0$.

that will lead to higher accuracy compared to the AOS schemes.

6. Accurate Operator Splitting Schemes

In this section, we propose accurate operator splitting schemes. For simplicity, we will choose the two dimensional case which corresponds to images, noting that these schemes can easily be extended to three and higher dimensions. We then compare the performance of the additive-multiplicative operator splitting (AMOS) schemes and the AOS schemes.

In order to increase the accuracy achieved by the LOD scheme (22), we may use a device of Strang [11, 23] who alternates two steps of a LOD scheme. In the linear case, if we start with a second order in time one-dimensional scheme, the order of accuracy will be kept by the so-called ‘‘Strang Splitting’’. If we use the first order semi-implicit scheme in each one of the dimensions, for simplicity in writing, the proposed scheme is given by

$$\begin{aligned} \left(I - \frac{\tau}{2}A_1(u^k)\right)(I - \tau A_2(u^k))\left(I - \frac{\tau}{2}A_1(u^k)\right)u^{k+1} \\ = u^k. \end{aligned} \quad (24)$$

This scheme for the linear case preserves second order accuracy [11] provided the Crank-Nicolson scheme is used in each one of the dimensions, instead of the semi-implicit scheme as in (24) which is only first-order accurate. This is an advantage over the LOD and AOS schemes, which do not preserve the second order accuracy of the Crank-Nicolson. However, the splitting is not symmetric and the non-symmetry which was also a deficiency in the LOD scheme has not been recovered. In practice, results after a rotation by 90 degrees (not shown) appear cleaner with this scheme compared to the LOD scheme, but there is no guarantee the symmetry criterion in the list of criteria for discrete diffusion scale-spaces will be satisfied. Therefore, after several trials in which the rotation by 90 degrees was still noticeable, we abandoned the Strang splitting suggested in (24). A split operator scheme which unconditionally meets the scale-spaces criteria and offers an improvement in accuracy is desired.

Motivated by ADI [15] which was mentioned in Section 4 as a favorable splitting scheme for the linear diffusion equation, we wish to combine the merits of the AOS scheme as a symmetric scheme, together with the family of multiplicative operator splittings (to

which the ADI belongs, as well as the first order LOD and the Strang splitting suggested in (24)). Multiplicative operator splittings are known in general to be more accurate than the AOS schemes. We therefore propose another scheme, also mentioned by Strang in [23, 24], which is both additive and multiplicative operator splitting (AMOS)

$$u^{k+1} = \frac{1}{2} \left[(I - \tau A_1(u^k))^{-1} (I - \tau A_2(u^k))^{-1} + (I - \tau A_2(u^k))^{-1} (I - \tau A_1(u^k))^{-1} \right] u^k \quad (25)$$

As in (24), Eq. (25) applies the AMOS scheme to the semi-implicit scheme. Such a combination is known in the literature [8] as the approximate factorization implicit (AFI) scheme, which is first order accurate in time. However, even in the case where it is built upon the semi-implicit scheme, the AMOS scheme is expected to be more accurate than the AOS scheme while preserving symmetry. Furthermore, it is possible to try to achieve better accuracy by applying the AMOS scheme on the Crank-Nicolson scheme. At each time step, two calculations are performed

$$\begin{aligned} \left(I - \frac{\tau}{2} A_1(u^k) \right) u^{k*} &= \left(I + \frac{\tau}{2} A_1(u^k) \right) u^k \\ \left(I - \frac{\tau}{2} A_2(u^k) \right) u^{k+1} &= \left(I + \frac{\tau}{2} A_2(u^k) \right) u^{k*}, \end{aligned} \quad (26)$$

and

$$\begin{aligned} \left(I - \frac{\tau}{2} A_2(u^k) \right) u^{k*} &= \left(I + \frac{\tau}{2} A_2(u^k) \right) u^k \\ \left(I - \frac{\tau}{2} A_1(u^k) \right) u^{k+1} &= \left(I + \frac{\tau}{2} A_1(u^k) \right) u^{k*}. \end{aligned} \quad (27)$$

After the time step is completed, the two results are averaged together which ensures a symmetric splitting. Although the directions are not alternating in each of the two calculations, i.e. the forward and backward Euler are performed on the same direction, in effect this scheme belongs to the family of alternating direction implicit (ADI) type methods. In our experiments, alternating the directions as in the classical ADI, produced no better results when applied to nonlinear diffusion filtering. Therefore, we refer to (26) and (27) as ADI, whereas (25) is AFI. We also note that adding the extrapolation suggested in the one-dimensional case, as in Table 1, did not increase the order of accuracy to exactly second when performing quantitative calculations with very small time steps in two dimensions. At

the expense of more computations, one can try to improve the extrapolation procedure by using the Wynn extrapolation or predictor-corrector methods, such as Adams Bashforth [8] or Douglas Jones [31], in which the Crank-Nicolson is the corrector. While these more complicated procedures are costly, it is not obvious how much accuracy will be gained as a consequence of larger time steps and whether this will be justifiable. However, practical use in applications requires mostly large time steps to perform the filtering, and it turns out the ADI scheme in (26) and (27) leads to visually better results for such time steps as can be seen in Figs. 13–15. We take 512 time steps of 0.05 as a reference, then decrease the number of iterations to check the deviation from the reference. First, we observe that the ADI scheme acts as a slightly better filter than the AOS scheme already in the reference image calculation, Fig. 13. As we decrease the number of iterations, we observe that the deviation from the converged result is smaller with the ADI scheme than with the AOS scheme. Filtering effect becomes stronger in the ADI scheme, while preserving fine details, which is an indication that the ADI scheme is visually more accurate than the AOS scheme, Figs. 14 and 15.

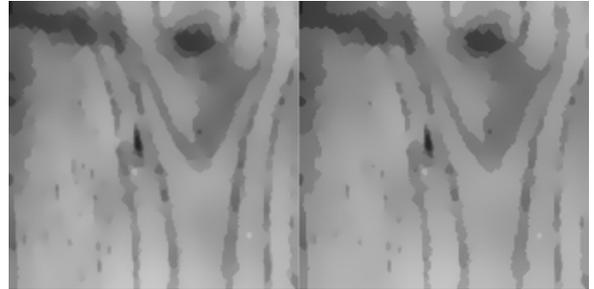


Figure 13. Reference: AOS vs. ADI, nonlinear diffusion filtering with 512 time steps of $\tau = 0.05$.

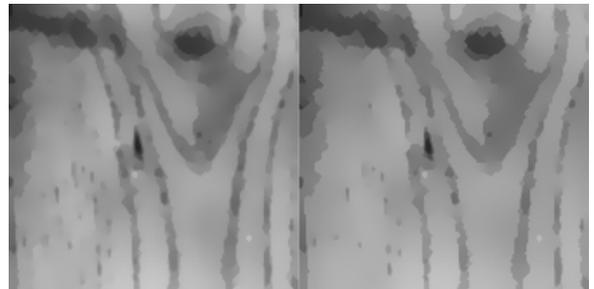


Figure 14. Checking visual accuracy: AOS vs. ADI, nonlinear diffusion filtering with 32 time steps of $\tau = 0.875$.

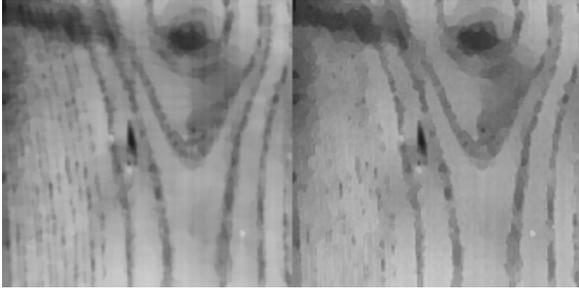


Figure 15. Checking visual accuracy: AOS vs. ADI, nonlinear diffusion filtering with one time step of $\tau = 28.0$.

Quantitative examination of the deviations from the reference is calculated as follows. We start from the original image in Fig. 16, which is a texture image taken from a neutron diffraction experiment. Figs. 17–19 show the comparison in terms of accuracy between the AOS, AFI and ADI schemes, which are discussed next. In terms of speed, the AOS and AFI schemes in actual simulations indicate that the AFI scheme takes roughly 1.5 the time it takes the AOS scheme to perform the filtering. The ADI scheme is

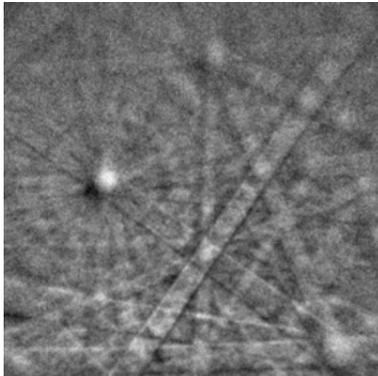


Figure 16. Original texture image.

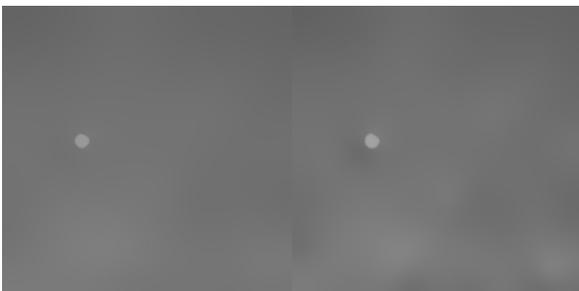


Figure 17. Reference: ADI vs. AOS, nonlinear diffusion filtering with 2000 time steps of $\tau = 0.1$.

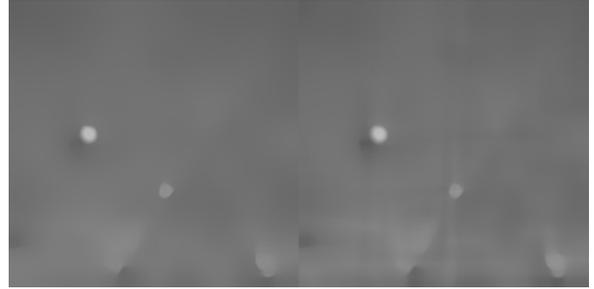


Figure 18. AFI vs. AOS, nonlinear diffusion filtering with four time step of $\tau = 50.0$.

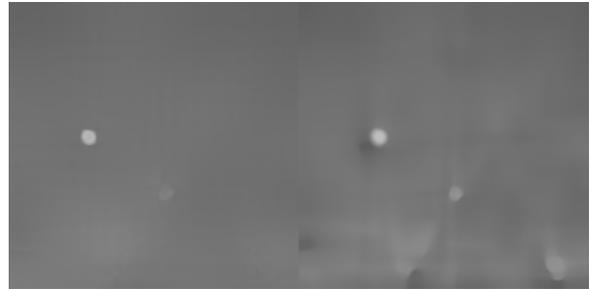


Figure 19. ADI vs. AOS, nonlinear diffusion filtering with four time step of $\tau = 50.0$.

roughly a factor of 2 to 3 longer in processing the images in Figs. 9 and 16, relative to the AOS scheme. We note that simply increasing the time step with the AOS scheme by this ratio does not produce the fine filtering that is achieved with the ADI scheme. This fact can be visually observed in practice and will not be reflected in the results of Table 2, as will be explained in the next paragraph.

In Table 2, the relative l_2 norm errors are calculated for the example in Figs. 17–19 as follows. Let v denote

Table 2. l_2 norm error estimation.

τ	AOS	AFI (%)	ADI (%)
0.25	0.09	0.06	0.08
0.5	0.13	0.1	0.11
1.0	0.17	0.14	0.13
2.0	0.22	0.17	0.17
5.0	0.29	0.24	0.19
10	0.36%	0.27	0.21
20	0.47%	0.32	0.23
50	0.79%	0.41	0.47
100	1.3%	0.54	1.25
200	2.07%	0.81	3.14

the reference solution: AOS, $\tau = 0.1$, in the case of the AOS and AFI schemes, and ADI, $\tau = 0.1$, in the case of the ADI scheme. Let u denote the approximate solution in each of the schemes. The relative error percentages are calculated by

$$\frac{\|u - v\|_2}{\|v\|_2}. \quad (28)$$

Note that the small relative error percentage values do not completely reflect the strength of the deviations and accuracies, since large propagation times produce smooth images, where the differences between the schemes appear only in small regions near prominent features within the original image. Moreover, the comparison with the ADI scheme is done for a separate reference frame, since even with a small time step the ADI scheme acts as a better filter, see Figs. 13 and 17, and hence its reference to measure deviations should be different. Therefore, Table 2 and the plot in Fig. 20 should be analyzed with caution, especially with respect to the comparison between the ADI and the AOS/AFI. From Table 2 and Fig. 20 it can be observed that up to a time step of $\tau = 50.0$, the ADI scheme is the most accurate, which is expected because the Crank-Nicolson is used as its building block. With very large time steps of more than $\tau = 50.0$, the AFI scheme is the most balanced scheme in deviations from the corresponding references, probably because the higher order error terms affect the closeness of the ADI scheme to its reference in Fig. 17. Among

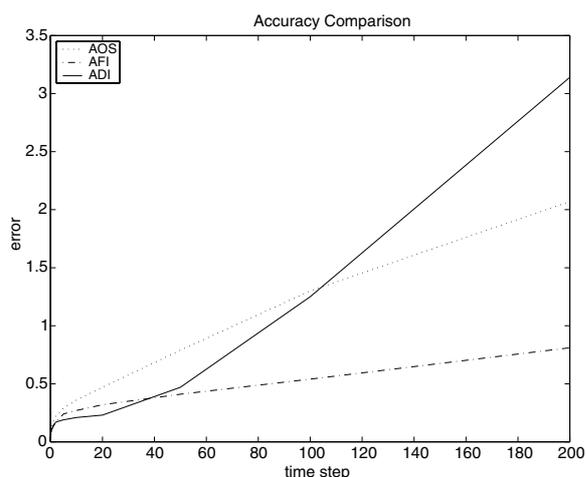


Figure 20. Comparison of error estimation for different time steps based on Table 2: AOS, AFI, ADI. However, note that for the ADI, a different reference image was used.

the schemes which are based on the semi-implicit scheme as their building block, the AFI scheme will produce more accurate results than the AOS scheme since the AMOS scheme is a more accurate splitting scheme than the AOS scheme at the expense of some increase in computations. We also note that in [32] (Figs. 1 and 2 of that reference) an illustrative comparison between implicit and explicit schemes was performed. It was shown that AOS schemes will not affect discontinuous structures in the processed images nor introduce distortion artifacts because the scheme is non-explicit. The ADI scheme contains both the implicit component of the AOS and an explicit component in addition. Therefore, as expected, we have not noticed any undesirable effects when examining discontinuous structures while smoothing out noise in additional experiments. Finally, we tried to obtain better accuracy out of the results in Fig. 15 by using Richardson's extrapolation [8] for our case

$$R_I(\tau/2) = \frac{4R(\tau/2) - R(\tau)}{3}, \quad (29)$$

where $R_I(\tau/2)$ denotes an improved result, using a time grid with a spacing of $\tau/2$ or coarser. $R(\tau/2)$ and $R(\tau)$ are the results of applying nonlinear diffusion filtering for time steps $\tau/2$ and τ , respectively. Our trials (with $\tau = 28.0$) failed to show an improvement of $R_I(\tau/2)$ relative to $R(\tau/2)$. An improvement is not guaranteed to begin with, since our equation is nonlinear and the solution is non-smooth.

7. Multiple Timestep Methods

Since their introduction in the 1970s, multiple timestep (MTS) methods have been developed extensively to reduce the cost of molecular dynamics simulations. In deriving the classical equations of motion for a molecular system, the total force consists of long-range terms (e.g., electrostatic and van der Waals interactions) and short-range terms (e.g., bond-length, bond-angle and torsion). The basic idea is to split the total force into components that allow more efficient integration of the equations of motion by resolving the slowly-varying long-range components with a large timestep and the fast components with a small timestep. Thus, calculations of the most time-consuming part (the nonbonded terms) can be enhanced significantly. The early MTS variants [25, 26] suffered from instabilities until symplectic and time-reversible MTS methods [7, 29] were

formulated, the latter based on the Trotter factorization [28] of the Liouville operator. These methods were extensively investigated and applied to biomolecular simulations [3, 9, 30]. A general overview on the development and applications of MTS methods in the field of biomolecular simulations can be found in [20].

To examine the use of MTS methods in performing nonlinear diffusion, we follow their derivation using the formulation outlined in [29]. Although for most modern MTS implementations in molecular dynamics packages it is customary to define three classes (fast, medium, and slow) and assign corresponding forces according to their range of interaction [2, 20], we will use a two-level mode for simplicity. We briefly describe the method using the standard notation used in molecular dynamics that appears in [29], and refer the interested reader to reference [29] for more discussion. First, let us define the Liouville operator \mathcal{L} for a system of N degrees of freedom in Cartesian coordinates:

$$i\mathcal{L} = \sum_{i=1}^N \left[\dot{X}_i \frac{\partial}{\partial X_i} + F_i \frac{\partial}{\partial P_i} \right], \quad (30)$$

where X_i and P_i are the position and conjugate momenta components for coordinate i , \dot{X}_i is the time derivative of X_i , and F_i is the force acting on the i th independent variable. The state of the system at a time t , $\Gamma(t)$, is defined as the collective set of positions and conjugate momenta ($X(t)$, $P(t)$). The state of the system at time t is given by applying the classical time evolution propagator, $\exp(i\mathcal{L}t)$, to the initial state of the system:

$$\Gamma(t) = \exp(i\mathcal{L}t)\Gamma(0). \quad (31)$$

Second, for systems with two different time scales, we factor the propagator $\exp(i\mathcal{L}t)$ into a propagator with a smaller timestep δt combined with a propagator with a larger timestep Δt . Let us split the Liouville operator into two distinct components:

$$i\mathcal{L} = i\mathcal{L}_1 + i\mathcal{L}_2. \quad (32)$$

After simplifications [29], the Trotter factorization [28] of the split Liouville propagator becomes:

$$\begin{aligned} \exp(i\mathcal{L}_1 + i\mathcal{L}_2)\Delta t \\ = \exp\left(i\mathcal{L}_1 \frac{\Delta t}{2}\right) \exp(i\mathcal{L}_2\Delta t) \exp\left(i\mathcal{L}_1 \frac{\Delta t}{2}\right) \end{aligned}$$

$$\begin{aligned} + O(\Delta t^3) = \exp\left(i\mathcal{L}_1 \frac{\Delta t}{2}\right) [\exp(i\mathcal{L}_2\delta t)]^n \\ \times \exp\left(i\mathcal{L}_1 \frac{\Delta t}{2}\right) + O(\Delta t^3), \end{aligned} \quad (33)$$

where n is the number of steps taken with the propagator associated with \mathcal{L}_2 to complete a timestep Δt of the propagator associated with \mathcal{L}_1 . Note the similarity to the Strang splitting in (24), since we are using closely related formulations to construct second-order multiplicative operator splittings. Here, our goal is to further examine the multiple timestep idea without accuracy considerations that were prioritized in previous sections. Thus, we examine the least expensive multiplicative operator splitting (and hence only first-order accurate in time) for a decomposition across scales:

$$\begin{aligned} \exp(i\mathcal{L}_1 + i\mathcal{L}_2)\Delta t \\ = \exp(i\mathcal{L}_1\Delta t) [\exp(i\mathcal{L}_2\delta t)]^n + O(\Delta t^2). \end{aligned} \quad (34)$$

Note that in this type of splitting symmetry need not be preserved, since rotation between scales is of no concern. Furthermore, each of the two propagators, namely the one corresponding to a timestep δt and the other corresponding to a timestep Δt can be treated using the AOS scheme for splitting the spatial coordinates which is independent from the splitting to distinct time scales.

For nonlinear diffusion, several ways can be considered to make use of the multiplicative operator splitting into different scales, suggested in (33) and (34). As an example, one may think of accelerating the calculation in the same manner as in molecular dynamics simulations, except that the force splitting is replaced by a partitioning of the nonlinear diffusion coefficient into smooth and non-smooth regions. Continuing with this analogy, it may be advantageous to use a large timestep Δt in smooth regions whereas the non-smooth regions, separated by a threshold, can be treated with small timesteps δt such that $\Delta t = n\delta t$ (if n is an integer, the two timesteps are synchronized after the larger timestep Δt , otherwise we obtain non-synchronized propagations which may have certain advantages). Thus, image pixels belonging to smooth regions need not be processed for a whole duration Δt though pixels in non-smooth regions are processed every δt . In molecular dynamics applications, the timesteps are synchronized (i.e., n is an integer) and the Verlet integration is commonly used [20]. We may do the same with the AOS schemes. The idea is to skip iteration steps (i.e., no need to set up the matrix $a_{ij}(u^k)$)

of Eq. (10) followed by matrix inversion) each δt , when pixels in smooth regions are encountered, since these pixels can be updated each Δt without loss of accuracy. Thus, we can attempt to accelerate the calculation when the added work for splitting pays off overall (by skipping iteration steps for selective pixels). However, because of the way the AOS schemes are structured, in each iteration a fixed amount of pixels are processed at once regardless of their separation to smooth and non-smooth regions. Matrix inversions are performed each iteration using the Thomas algorithm, which requires as input four one-dimensional vectors. Each of these vectors is of fixed length, corresponding to the number of pixels in a row (or column) of the image, with elements ordered according to the location of the pixels within the row (or column). It is possible to avoid calculating off-diagonal elements of the pixels belonging to non-smooth regions before calls to the Thomas algorithm are made. Still, standard implementation of the Thomas algorithm will process them regardless of their values. It is therefore challenging to devise strategies to speed up this process, by modifying the Thomas algorithm to perform more efficiently in places where the off-diagonals are zeroes. Various other strategies are possible, such as constructing a function or a transformation between the diffusion coefficient and the corresponding timestep size. In that way, many timesteps can be performed in parallel during the integration, without necessarily worrying about a synchronization of the different propagations. As a compromise between the two extremes, from the one side a non-synchronized multi-level breakup strategy and from the other side a synchronized (i.e., $\Delta t = n\delta t$) two-level breakup as suggested in (34), a synchronized three-level breakup strategy such as used in (33) and in modern molecular dynamics simulations [20] may prove optimal for some applications.

Here, we implement the simplest strategy, namely a non-synchronized two-level breakup. The goal is to demonstrate potential improvement in the diffusion process. This idea is similar to algebraic multigrid methods [4]. We note that scale-based diffusion has recently been tried in [18] by an ad hoc procedure of altering the diffusion coefficient, without the mathematical framework of multiple timestep methods given here. Instead of evolving the nonlinear diffusion equation with the same timestep for all spatial regions, we specifically double (or multiply by a desired factor) the timestep for non-smooth regions while preserving the same timestep for smooth regions. Thus, regions with

texture and edges will be given preference in the diffusion process. This is performed with a minimal added effort (by adding a few selection statements that contribute negligibly to the execution time) for a certain diffusion coefficient suggested in [36]. In [36], a behavioral analysis was performed in detail for several diffusion coefficients. Specifically, it was found that the following diffusion coefficient leads to well-posed nonlinear diffusion possessing good behavior and distinguishing between smooth and non-smooth regions by using a threshold T :

$$c(x) = \begin{cases} \frac{1}{T} + \frac{p(T + \epsilon)^{p-1}}{T}, & x < T \\ \frac{1}{x} + \frac{p(x + \epsilon)^{p-1}}{x}, & x \geq T, \end{cases} \quad (35)$$

where $\epsilon > 0$ and $0 < p < 1$. It was explained in [36] how the diffusion coefficient (35) was constructed in order to avoid “blocky effects” [36] and achieve backward diffusion [16, 36] at the same time. Furthermore, it was noted in [36] that staircasing effects will eventually disappear during the diffusion process when using the above diffusion coefficient.

We apply the diffusion coefficient (35) on the original image of Fig. 9. Figure 21 (left) shows the result of applying a single timestep $\Delta t = 0.5$ for 18 iterations with the values $T = 10$, $\epsilon = 1$, and $p = 0.5$ for the parameters of the diffusivity in (35). We observe that regions in the original image containing rich texture and edges have shrunk to blotted dots. If we increase the timestep as in Fig. 21 (right) to $\Delta t = 2.0$, we achieve over-smoothing. As a compromise, a timestep $\Delta t = 1.25$ is taken in Fig. 22 (right) and succeeds in achieving less smoothing. Still, there are noticeable tradeoffs (the



Figure 21. AOS, nonlinear diffusion filtering with 18 single timesteps of $\Delta t = 0.5$ (left) and $\Delta t = 2.0$ (right). Diffusion was performed using the diffusion coefficient (35) with parameter values $T = 10$, $\epsilon = 1$, $p = 0.5$.

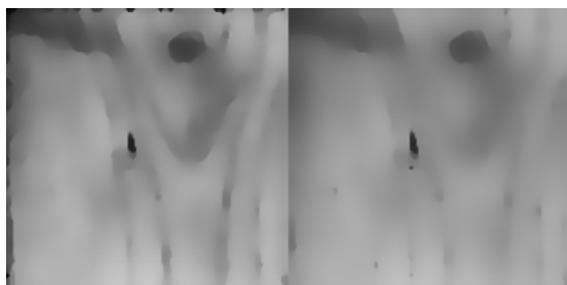


Figure 22. AOS with multiple timesteps [$\delta t = 0.5$, $\Delta t = 2.0$] (left) vs. AOS with a single timestep $\Delta t = 1.25$ (right). In both cases, 18 iterations were performed using the diffusion coefficient (35) with parameter values $T = 10$, $\epsilon = 1$, $p = 0.5$.

blotted dot near the center remains, and some bands are lost). However, by using two timesteps (e.g., $\delta t = 0.5$, $\Delta t = 2.0$) it is possible to reach a state in which all blotted dots disappear and possibly desired band features of the original image remain. By tuning all other parameters, we have tried reaching the same state with a single timestep approach (examining Δt values between 0.5 and 2.0) without success. Such a selective state can only be reached by using multiple timesteps.

8. Conclusions

In this paper, multiplicative operator splitting schemes across dimensions and scales are examined for designing nonlinear diffusion integrators. Multiplicative splitting schemes across dimensions are gradually constructed step by step, starting from one dimension and the linear case, by reviewing various schemes which are relevant and have been suggested in this context to other applications. These are presented as alternatives to the very efficient additive operator splitting (AOS) scheme. Subsequently, multiplicative operator splitting schemes across scales (i.e., multiple timestep methods) are introduced and discussed. These can be combined with Weickert et al's [12, 32] AOS scheme.

For the splitting schemes across dimensions, it is found that better accuracy can be visually inspected and might become a desirable feature in some future applications. The two splitting methods which unconditionally satisfy all discrete scale-space criteria are Weickert et al's AOS scheme and our proposed scheme, the AMOS scheme. Both are reliable, simple and parallelizable [33] splittings for implementation. The AOS scheme is more efficient than the AMOS with Backward-Euler scheme, the AFI scheme,

by approximately a factor of 1.5, and the AMOS with Crank-Nicolson scheme, the ADI scheme, by a factor of 2 to 3, depending on the efficiency of the implementation. Multiplicative operator schemes are in general more accurate than their additive counterparts, and the combination of the two in the AMOS schemes ensures both symmetry and better accuracy at the expense of an increase in execution time. In the arsenal of numerical schemes for performing nonlinear diffusion filtering the AMOS scheme can be considered as an extension to the AOS scheme for applications that require high accuracy. However, the advantage of constructing accurate numerical schemes for the nonlinear diffusion of images with long timesteps is not clear at present, and the AOS remains the simplest and most efficient choice for implementation.

Consequently, multiple timestep methods are introduced for examining multiplicative operator splittings across scales. Following a discussion of their use in molecular dynamics, possible ways to incorporate them in nonlinear diffusion are suggested. An example is given to illustrate how multiple timestep methods can be used to improve the diffusion process. Additional work on targeting selective scales in regions of interest by processing them individually is a natural continuation of these ideas.

Acknowledgments

Part of this work was performed while D.B. was with Hewlett-Packard Laboratories Israel. The continuation of the work was supported by NSF Award ASC-9318159, NIH Award R01 GM55164, and a John Simon Guggenheim fellowship to T.S.

References

1. D. Barash, "A fundamental relationship between bilateral filtering, adaptive smoothing and the nonlinear diffusion equation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 7, 2002.
2. D. Barash, L. Yang, X. Qian, and T. Schlick, "Inherent speedup limitations in multiple timestep/particle mesh ewald algorithms," *Journal of Computational Chemistry*, Vol. 24, No. 1, p. 77, 2002.
3. E. Barth, M. Mandziuk, and T. Schlick, "A separating framework for increasing the timestep in molecular dynamics," *Computer Simulation of Biomolecular Systems: Theoretical and Experimental Applications*, Vol. 3, W.F. van Gunsteren, P.K. Weiner, and A.J. Wilkinson (Eds.), p. 97, 1997.
4. A. Brandt, S. McCormick, and J. Ruge, "Algebraic multigrid (AMG) for automatic multigrid solution with application to

- geodesic computations," Technical Report, Institute for Computational Studies, Fort Collins, CO, 1982.
5. F. Catté, P.L. Lions, J.M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Numer. Anal.*, Vol. 29, No. 1, p. 182, 1992.
 6. T.F. Chan, S. Osher, and J. Shen, "The digital filter and nonlinear denoising," Technical Report CAM 99-34, UCLA Computational and Applied Mathematics, 1999.
 7. H. Grubmüller, H. Heller, A. Windemuth, and K. Schulten, "Generalized verlet algorithm for efficient molecular simulations with long-range interactions," *Mol. Sim.*, Vol. 6, p. 121, 1991.
 8. J.D. Hoffman, *Numerical Methods for Engineers and Scientists*, McGraw-Hill, Inc., 1992.
 9. J. Izaguirre, S. Reich, and R.D. Skeel, "Longer time steps for molecular dynamics," *Journal of Chemical Physics*, Vol. 110, p. 9853, 1999.
 10. R. Kimmel, R. Malladi, and N. Sochen, "Images as embedding maps and minimal surfaces: Movies, color, and volumetric medical images," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Puerto Rico, 1997.
 11. R. LeVeque, *Numerical Methods for Conservation Laws*, Birkhuser Verlag, Basel, 1990.
 12. T. Lu, P. Neittaanmäki, and X.-C. Tai, "A parallel splitting up method and its application to Navier-Stokes equations," *Applied Mathematics Letters*, Vol. 4, No. 2, p. 25, 1991.
 13. T. Lu, P. Neittaanmäki, and X.-C. Tai, "A parallel splitting up method for partial differential equations and its application to Navier-Stokes equations," *RAIRO Mathematical Modelling and Numerical Analysis*, Vol. 26, No. 6, p. 673, 1992.
 14. G.I. Marchuk, "Splitting and alternating direction methods," in *Handbook of Numerical Analysis*, P.G. Ciarlet and J.L. Lions (Eds.), Vol. 1, p. 197, 1990.
 15. D.W. Peaceman and H.H. Rachford, "The numerical solution of parabolic and elliptic differential equations," *Journal Soc. Ind. Appl. Math.*, Vol. 3, p. 28, 1955.
 16. P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 7, p. 629, 1990.
 17. L.I. Rudin, S. Osher, and F. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D.*, Vol. 60, p. 259, 1992.
 18. P.K. Saha and J.K. Udupa, "Scale-based diffusive image filtering preserving boundary sharpness and fine structures," *IEEE Transactions on Medical Imaging*, Vol. 20, No. 11, p. 1140, 2001.
 19. G. Sapiro, *Geometric Partial Differential Equations and Image Processing*, Cambridge University Press, 2001, <http://us.cambridge.org/titles/catalogue.asp?isbn=0521790751>.
 20. T. Schlick, *Molecular Modeling and Simulation: An Interdisciplinary Guide*, Springer-Verlag, New York, 2002, <http://www.springer-ny.com/detail.tpl?isbn=038795404X>.
 21. J.A. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, 1999, <http://us.cambridge.org/titles/catalogue.asp?isbn=0521645573>.
 22. N. Sochen, R. Kimmel, and R. Malladi, "A geometrical framework for low level vision," *IEEE Transactions on Image Processing*, Vol. 7, No. 3, p. 310, 1998.
 23. G. Strang, "On the construction and comparison of difference schemes," *SIAM J. Numer. Anal.*, Vol. 5, No. 3, p. 506, 1968.
 24. G. Strang, "Accurate partial difference methods I: Linear cauchy problems," *Arch. Rational Mech. Anal.*, Vol. 12, p. 392, 1963.
 25. W.B. Streett, D.J. Tildesley, and G. Saville, "Multiple time step methods in molecular dynamics," *Mol. Phys.*, Vol. 35, p. 639, 1978.
 26. R.D. Swindoll and J.M. Haile, "A multiple time-step method for molecular dynamics simulations of fluids of chain molecules," *J. Chem. Phys.*, Vol. 53, p. 289, 1984.
 27. C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the 1998 IEEE International Conference on Computer Vision*, Bombay, India, 1998.
 28. H.F. Trotter, "On the product of semi-groups of operators," in *Proceedings of the American Mathematical Society*, Vol. 10, p. 545, 1959.
 29. M.E. Tuckerman, B.J. Berne, and G.J. Martyna, "Reversible multiple time scale molecular dynamics," *J. Chem. Phys.*, Vol. 97, p. 1990, 1992.
 30. M. Watanabe and M. Karplus, "Simulation of macromolecules by multiple-timestep methods," *J. Phys. Chem.*, Vol. 99, No. 15, p. 5680, 1995.
 31. J. Weickert, *Anisotropic Diffusion in Image Processing*, Tuebner, Stuttgart, 1998.
 32. J. Weickert, B.M. ter Haar Romeny, and M. Viergever, "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Transactions on Image Processing*, Vol. 7, No. 3, p. 398, 1998.
 33. J. Weickert, K.J. Zuiderveld, B.M. ter Haar Romeny, and W.J. Niessen, "Parallel implementations of AOS schemes: A fast way of nonlinear diffusion filtering," in *Proceedings of the 1997 IEEE International Conference on Image Processing*, Vol. 3, p. 396, Santa Barbara, CA, 1997.
 34. J. Weickert, "Anisotropic diffusion filters for image processing based quality control," in *Proc. Seventh European Conf. on Mathematics in Industry*, A. Fasano and M. Primicerio (Eds.), Teubner, Stuttgart, p. 355, 1994.
 35. N.N. Yanenko, *The Method of Fractional Steps: The Solution of Problems of Mathematical Physics in Several Variables*, Springer, New York, 1971.
 36. Y. You, W. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image enhancement," *IEEE Transactions on Image Processing*, Vol. 5, No. 11, p. 1539, 1996.