Contemporary Mathematics
Volume 180, 1994

ELLAM-Based Domain Decomposition and Local Refinement Techniques for Advection-Diffusion Equations with Interfaces

H. WANG, H.K. DAHLE, R.E. EWING, T. LIN, AND J.E. VÅG

ABSTRACT. We combine Eulerian-Lagrangian localized adjoint methods (ELLAM) with domain decomposition and local refinement techniques to develop two nonoverlapping iterative schemes for advection-diffusion equations with various physical/numerical interfaces.

1. Introduction

Advection-diffusion equations arise from many important applications and often present serious numerical difficulties. Most numerical methods exhibit some combination of excessive numerical dispersion or nonphysical oscillation. Moreover, practical advection-diffusion problems often have various interfaces that introduce extra difficulties. Physical interfaces arise from the modeling of transport processes in composite media and lead to advection-diffusion equations with discontinuous coefficients. Numerical interfaces arise from the application of domain decomposition and local refinement techniques. An identifying feature of groundwater contaminant transport and many other applications is the presence of large scale fluid flows coupled with transient transport of physical quantities such as pollutants, chemical species, radionuclides, and temperature, which are generally smooth outside some small regions and may have sharp fronts inside where important chemistry and physics take place. An extremely fine global mesh in both space and time is impossible due to the excessive computational

1991 *Mathematics Subject Classification.* 65M50, 65M55, 65M60.

Key words and phrases. Lagrangian methods, domain decomposition, grid refinement.

This research was supported in part by DOE, DE-AC05-840R21400, Martin Marietta, Subcontract, SK965C and SK966V, by NSF Grant No. DMS-8922865, by funding from the Institute for Scientific Computation at Texas A&M University, and by funding from the Norwegian Research Council, RV/412.92/003.

This paper is in final form and no version of it will be submitted for publication elsewhere.

cost. A feasible approach is to apply domain decomposition and local refinement techniques by partitioning the global domain into a number of subdomains and solving the problems with fine meshes in both space and time within the sharp front regions (subdomains) and coarse meshes outside (other subdomains). This way, both accuracy and efficiency can be guaranteed, but at a cost of introducing numerical interfaces between different subdomains.

Many domain decomposition and local refinement techniques have been developed for elliptic and parabolic equations, but it is more difficult to develop these techniques for advection-dominated equations. In this case, locally generated errors at the interfaces can be propagated into the domain so that the overall accuracy is decreased; improper treatment of the interfaces might destroy the stability of the numerical methods. Most existing methods for interface problems for advection-dominated equations employ the Eulerian approach and often yield numerical solutions with some combination of excessive numerical dispersion or oscillation. Extremely small time steps have to be used to maintain the accuracy and stability of these methods. While Eulerian-Lagrangian methods can overcome these problems to some extent, they cannot treat general boundary conditions and are therefore difficult to implement for interface problems for advection-dominated equations.

Eulerian-Lagrangian localized adjoint methods (ELLAM) [**1, 2**] (and the references cited there) have been successfully applied to solve advection-dominated equations and have yielded numerical solutions free of oscillations or numerical dispersion. ELLAM maintain mass conservation and treat boundary conditions systematically. In this paper we present two types of ELLAM-based decomposition and local refinement (in both space and time) techniques for the interface problems for the following one-dimensional model equation

$$\mathcal{L}u \equiv u_t + (V(x,t)u - D(x,t)u_x)_x = f(x,t), \; x \in (a,b), \; t \in (0,T]. \tag{1}$$

The boundary conditions at $x = a$ and $x = b$ can be Dirichlet, Neumann, or flux conditions, and different types of boundary conditions may be specified at $x = a$ and $x = b$. In addition, an initial condition is needed to close the system.

In a physical interface case, $V(x,t)$ and $D(x,t)$ are smooth except for the interface $x = d$ where either $V(x,t)$ or $D(x,t)$ or both have the first type of discontinuity with respect to x. Then, (1) is closed by the interface conditions:

$$\begin{aligned} u(d-,t) &= u(d+,t), & t \in [0,T], \\ (Vu - Du_x)(d-,t) &= (Vu - Du_x)(d+,t), & t \in [0,T]. \end{aligned} \tag{2}$$

Numerical interfaces arise when different meshes are imposed over different subdomains. We may have both physical and numerical interfaces at the same locations. For simplicity, only one interface $x = d$ has been assumed to be present; generalization to several interfaces is straightforward. Also, $V(x,t)$ is assumed to be positive, so $x = a$ and $x = b$ are always the inflow and outflow boundaries, respectively.

2. An ELLAM Scheme

In this section we present an ELLAM scheme for equation (1) with smooth coefficients. Let I and N be two positive integers. We define the partitions of space and time as $x_i = a + i\Delta x$ $(i = 0, 1, \ldots, I)$ and $t^n = n\Delta t$ $(n = 0, 1, \ldots, N)$. In the numerical scheme, we use a time-marching algorithm to solve (1). We consider space-time test functions w that vanish outside of $[a, b] \times (t^n, t^{n+1}]$ and are discontinuous in time at each time level t^n. With these test functions, we can write the weak form for equation (1) as

$$
\begin{aligned}
&\int_a^b u(x, t^{n+1}) w(x, t^{n+1}) dx + \int_{t^n}^{t^{n+1}} \int_a^b D u_x w_x dx dt \\
&+ \int_{t^n}^{t^{n+1}} (Vu - Du_x) w|_a^b dt - \int_{t^n}^{t^{n+1}} \int_a^b u(w_t + V w_x) dx dt \\
&= \int_a^b u(x, t^n) w(x, t_+^n) dx + \int_{t^n}^{t^{n+1}} \int_a^b f w dx dt,
\end{aligned} \tag{3}
$$

where $w(x, t_+^n) = \lim_{t \to t_+^n} w(x, t)$.

Based on the ideas of the localized adjoint method and the Lagrangian nature of equation (1), we define the test functions w to be the standard hat functions at the time t^{n+1} (or at the outflow boundary) and to be constant along the approximate characteristics from t^{n+1} (or the outflow boundary) to the time t^n (or the inflow boundary), and to be discontinuous in time at time level t^n **[1, 2]**.

Putting the test functions w above into the variational form (3), we can derive an ELLAM formulation. The first terms on both the left-hand and the right-hand sides of equation (3) are already defined at time levels t^{n+1} and t^n, respectively. The last term on the right-hand side of (3) is the source term that can be computed directly. The last term on the left-hand side of (3) measures the errors of the characteristic tracking and is negligible. (In fact, it vanishes when we track the characteristics exactly.) The inflow and outflow boundary conditions are naturally incorporated into (3) by the third term on the left-hand side of (3) when the test functions are not zero at the boundaries. Applying a one-point backward Euler quadrature to the second term on the left-hand side of (3) reduces this term to a term at the time t^{n+1} as well as terms at the inflow and outflow boundaries. Thus, with the known solution at time t^n as well as the inflow and outflow boundary conditions, our ELLAM scheme yields the numerical solution at time t^{n+1}, and the numerical solution at the outflow boundary for outflow Neumann/flux boundary conditions, or the total flux at the outflow boundary for the outflow Dirichlet boundary condition **[1]**.

3. Generalized ELLAM Schemes for Interface Problems

3.1. Overview. In this section, we present two types of ELLAM-based domain decomposition and local refinement techniques to solve equation (1) with physical or numerical interfaces or both.

We first demonstrate the ideas by recalling the ELLAM-based scheme for the interface problems for first-order advection-reaction equations [**2**], which is an extreme case of equation (1) in that the diffusion coefficient $D(x,t)$ vanishes. Therefore, only an inflow Dirichlet boundary condition is needed at $x = a$, and no outflow boundary condition should be specified at $x = b$. In contrast to many existing methods that have difficulties in solving these problems, ELLAM can naturally be applied to do so. In fact, the ELLAM scheme presented in the last section is valid when $D(x,t)$ vanishes and yields the numerical solution at time t^{n+1} and at the outflow boundary $x = b$ with the given inflow boundary condition at $x = a$ and the known numerical solution at time t^n. The corresponding interface condition reduces to the second equation in (2), which imposes the continuity requirement on the advective flux across the interface $x = d$. Applying the ELLAM scheme to the advection-reaction equation on (a,d) yields the solution at time t^{n+1} and the left-limit of the solution at the interface $x = d$. Then, the interface condition gives the right-limit of the solution at the interface $x = d$. With this right-limit as the inflow boundary condition at $x = d$ and the known solution at time t^n, apply the ELLAM scheme to solve the advection-reaction equation on (d,b). Moreover, the spatial and temporal meshes used in $(a,d) \times [0,T]$ and $(d,b) \times [0,T]$ are independent of each other. Therefore, a noniterative and nonoverlapping ELLAM-based domain decomposition and local refinement technique is naturally derived, which treats various (physical/numerical) interface problems for advection-reaction equations in a universal way, and fully utilizes the intrinsic physics behind them.

3.2. A Dirichlet-Flux Algorithm. In this section we develop ELLAM-based domain decomposition and local refinement techniques for the interface problems for equation (1). Due to the effect of the diffusion term, the downstream values of the solutions also affect their upstream values. Thus, an iterative procedure should be used. Among other questions in the development of the schemes are the following: Which types of outflow and inflow boundary conditions should be imposed at the interface $x = d$ to close (1) over the subdomains (a,d) and (d,b)? What values should be chosen for these boundary conditions? Our studies [**1**] show the following observations. The numerical solutions with inflow flux/Dirichlet boundary conditions are very accurate, while an inflow Neumann condition is not physically reasonable and the corresponding solutions are not so accurate as those with flux/Dirichlet conditions. An outflow Neumann boundary condition is physically reasonable and the corresponding numerical solutions are accurate. For an outflow Dirichlet condition, the numerical solutions are accurate if a "right" value is specified, and may not be accurate otherwise because boundary layers will arise in this case. An outflow flux condition is not numerically stable and should be avoided.

Based on these observations, we propose a Dirichlet-flux iterative scheme for interface problems of equation (1). With the known solution $u(x,t^n)$ at time t^n as well as the inflow and outflow boundary conditions at $x = a$ and $x = b$, we compute the solution $u(x, t^{n+1})$ by the following procedure:

(i) Choose an initial guess $u(d_-, t) = u(x_d^*(t), t^n)$ for $t \in [t^n, t^{n+1}]$, where $x_d^*(t)$ is the foot of the (approximate) characteristics emanating backward from (d, t) at the interface.

(ii) With the given inflow boundary condition at $x = a$ and $u(d_-, t)$ as the outflow Dirichlet boundary condition at $x = d$, use the ELLAM scheme to solve (1) on (a, d) and obtain the solution $u(x, t^{n+1})$ at time t^{n+1} and the total flux $(Vu - Du_x)(d_-, t)$ for $t \in [t^n, t^{n+1}]$ at $x = d$.

(iii) The second equation in (2) gives $(Vu - Du_x)(d_+, t) = (Vu - Du_x)(d_-, t)$, $t \in [t^n, t^{n+1}]$.

(iv) With $(Vu - Du_x)(d_+, t)$ as the inflow flux condition at $x = d$ and the given outflow boundary condition at $x = b$, use the ELLAM scheme to solve (1) on (d, b) and obtain the solution $u(x, t^{n+1})$ at t^{n+1} as well as $u(b, t)$ or the total flux at $x = b$ depending on which type of boundary condition was specified at $x = b$.

(v) Compute $u(d_+, t)$ by projecting $u(x, t^{n+1})$ back along the (approximate) characteristics or by a linear interpolation of the solution u with its values at t^n and t^{n+1} along the (approximate) characteristics.

(vi) The first equation in (2) yields $u(d_-, t) = u(d_+, t)$. Go back to Step (ii) and repeat the process until the algorithm converges.

Since the exact solution of equation (1) is smooth along the characteristics, the initial guess chosen in Step (i) provides a "right" value for the Dirichlet condition at the interface $x = d$. Then, the ELLAM scheme yields the total flux $(Vu - Du_x)(d_-, t)$. Applying the second equation in (2) generates a most desirable inflow flux condition $(Vu - Du_x)(d_+, t)$ (Step (iii)) at $x = d$ for equation (1) on (d, b), which guarantees a full mass conservation when we move from (a, d) to (d, b). The continuity of the solutions is imposed in Step (vi) when we move back from (d, b) to (a, d).

3.3. A Neumann-Dirichlet Algorithm. A major concern for the Dirichlet-flux algorithm presented in the last part is that Step (ii) in the algorithm might introduce some potential error because we computed the flux out of a Dirichlet condition. In this section, we propose an alternative Neumann-Dirichlet iterative scheme. Since a Neumann condition is most appropriate at the outflow boundary, we impose an outflow Neumann condition at $x = d$ for equation (1) on (a, d). While an initial guess $u_x(d_-, t) = u_x(x_d^*(t), t^n)$ for $t \in [t^n, t^{n+1}]$ can be chosen, it is one-order less accurate than the solution itself due to the numerical differentiation involved. Some post-processing techniques can be used to enhance the accuracy. That is, choose $u_x(d_-, t) = (\mathcal{P}u)_x(x_d^*(t), t^n)$ for $t \in [t^n, t^{n+1}]$, where $\mathcal{P}u$ is a post-processed solution obtained from u. Our Neumann-Dirichlet algorithm can be presented as follows:

(i) Choose an initial guess $u_x(d_-, t) = (\mathcal{P}u)_x(x_d^*(t), t^n)$ for $t \in [t^n, t^{n+1}]$.

(ii) With the prescribed inflow boundary condition at $x = a$ and $u_x(d_-, t)$ as the outflow boundary condition at $x = d$, apply the ELLAM scheme to solve (1) on (a, d) and obtain the solution $u(x, t^{n+1})$ at t^{n+1} and $u(d_-, t)$ at the interface $x = d$.

(iii) The first equation in (2) gives $u(d_+, t) = u(d_-, t)$.

(iv) With $u(d_+, t)$ as the inflow Dirichlet boundary condition at $x = d$ and the prescribed outflow boundary condition at $x = b$, use the ELLAM to solve (1) on (d, b) and obtain the solution $u(x, t^{n+1})$ at t^{n+1} as well as the solution or the total flux at the outflow boundary depending on which type of outflow boundary condition was specified at $x = b$.

(v) Compute $u(d_+, t)$ and $(Vu - Du_x)(d_+, t)$ by projecting $u(x, t^{n+1})$ and $(Vu - D\mathcal{P}u_x)(x, t^{n+1})$ back along the approximate characteristics or by a linear interpolation at t^n and t^{n+1}, respectively.

(vi) Applying both equations in (2) yields $u(d_-, t) = u(d_+, t)$ and $(Vu - Du_x)(d_-, t) = (Vu - Du_x)(d_+, t)$. Then obtain the diffusive flux $-Du_x(d_-, t) = (Vu - Du_x)(d_-, t) - Vu(d_-, t)$. Go back to Step (ii) and repeat the process until the algorithm converges.

This algorithm uses an outflow Neumann condition at $x = d$ for equation (1) on (a, d) and avoids some potential numerical difficulties from a possibly improperly specified outflow Dirichlet condition at $x = d$ and from the numerical flux computed out of a Dirichlet condition. Then, imposing $u(d_+, t) = u(d_-, t)$ (the first equation in (2) yields a continuous numerical solution across the interface $x = d$. The continuity of mass is imposed when we move back from (d, b) to (a, d) to maintain mass conservation.

References

1. R.E. Ewing and H. Wang, *Eulerian-Lagrangian localized adjoint methods for variable-coefficient advective-diffusive-reactive equations in groundwater contaminant transport*, Proceedings of Sixth IIMAS-UNAM Workshop on Numerical Analysis and Optimization, Oaxaca, Mexico, 1992, (To appear).
2. H. Wang, T. Lin, and R.E. Ewing, *ELLAM with domain decomposition and local refinement techniques for advection-reaction problems with discontinuous coefficients*, Computational Methods in Water Resources IX. Vol. I: (Russell et al., eds.), Computational Mechanics Publications and Elsevier Applied Science, London and New York, 1992, 17–24.

Department of Mathematics, University of South Carolina, Columbia, SC 29208
E-mail address: hwang@math.scarolina.edu

Department of Mathematics, University of Bergen, Bergen, Norway
E-mail address: Helge.Dahle@mi.uib.no, Jan-Einar.Vag@mi.uib.no

Institute for Scientific Computation, Texas A&M University, College Station, TX 77843-3404
E-mail address: ewing@ewing.tamu.edu

Department of Mathematics, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061
E-mail address: tlin@mthiris.math.vt.edu