

Proposal Presentation Notes

Title Slide

Good afternoon, everyone. Thank you for coming to my proposal. This is joint work with my co-advisors Leszek Demkowicz and Bob Moser developing a space-time DPG finite element method for transient fluid dynamics problems.

Motivation

Here's an overview of the talk. I'll be discussing the motivation behind why we want to apply DPG for computational fluid dynamics and review some of the popular methods in this field. Then I'll jump into a quick overview and derivation of the DPG method followed by my preliminary work on locally conservative and space-time formulations. Then I would like to propose future work to be included in my thesis.

Navier-Stokes Equations

The Navier-Stokes equations of fluid motion play a critical role in many parts of engineering design and analysis, but their robust simulation remains a challenging issue.

- Numerical methods need to resolve a wide variety of flow features from shocks to boundary layers to turbulence.
- Nonlinearity presents a challenge, and uniqueness of solutions is an open question.
- On top of that, most numerical methods have minimum mesh resolution requirements before convergence can be guaranteed. Meshes that are too coarse for the numerical method to converge on are termed to be in the pre-asymptotic regime. This means that in some sense, the mesh for the simulation to the phenomena on the right needs to anticipate the flow

features before they are known.

Initial Mesh Design

This requirement can present additional challenges to the CFD practitioner designing a mesh. At the minimum level, the surface mesh needs to adequately represent the physical geometry of the problem under consideration. But the stability of the numerical methods presents additional requirements on the volume mesh surrounding the surface mesh. To throw another wrench into the problem, turbulence models have their own requirements about the size of the first layer of volume elements next to the surface. This often leads to a trial and error process for the CFD engineer. When I was doing my undergrad in Aerospace Engineering, I had a lot of friends on the Formula SAE team doing CFD simulations of Formula 1 cars like the one shown here (in Fluent, coincidentally). I would watch the process as they spent hours generating a mesh before attempting a simulation, which would then crash due to some inadequate features in the mesh. They would attempt to fix those areas, try the simulation again, and iterate until everything appeared to be working correctly. This kind of trial and error process is not sustainable as you start ramping up to tens of thousands of processors. Now, Fluent is great software, our professors deemed it a great introduction to CFD for undergrads, but the underlying finite volume method has its limitations.

DPG on Coarse Meshes

My goal with this research is to remove one of those constraints on CFD practitioners. In contrast to most other methods out there, the DGP method does not suffer convergence issues on coarse meshes. We are able to start on the coarsest mesh possible which sufficiently resolves the geometry and adaptively refine towards a resolved solution. These are results from Jesse Chan's thesis on supersonic flow over a flat plate. This is not a remarkably difficult problem, but what is notable is that we were able to start the simulation on a mesh of only two elements, sit back, and let the method take over as it solved, refined, and converged to a resolved solution after 11 adaptive steps.

Stabilized Finite Elements for CFD

SUPG 1

Before I jumped into our work, I wanted to discuss a couple of similar methods that have successfully been applied to CFD. Tom Hughes originally developed the streamline upwind Petrov-Galerkin method which became the first stabilized finite element method to successfully solve fluid dynamics problems. Originally inspired by finite difference techniques of upwinding, SUPG effectively introduces an upwind bias into the test functions which introduced the idea that an appropriate change of test space may produce a more stable method. The rich field of stabilized finite elements grew out of this work with major contributions from Hughes and his collaborators, Franca, Johnson, Codina, Tezduyar and many more than could fit on this slide. A particularly successful framework for stabilized methods has been the variational multiscale method.

SUPG 2

There are two equivalent views on how the method works. The first is that we can modify the bilinear form with additional stabilizing terms weighted by the residual. The residual weighting ensures that we maintain a consistent scheme. This can also be interpreted as a Petrov-Galerkin scheme where test functions are upwinded.

DG 1

The discontinuous Galerkin method has been increasingly popular for CFD simulations. Originally developed for neutron transport and developed for hyperbolic conservation laws, DG combines attractive features of finite elements and finite volumes. Early mathematical work on DG was done by Babuska, Lions, Nitsche, and Zlamal, but the development for CFD can largely be attributed to Cockburn and Shu with further extension to elliptic problems by Arnold, Brezzi, Cockburn, and Marini. DG utilizes a discontinuous basis defined element-wise and provides a natural high order extension to many of the concepts from finite

volume methods. The field of DG methods is similarly rich that there is no way I could fit all major contributors on one slide.

DG 2

The basic idea is to develop the bilinear form element by element and then glue the contributions together. Since basis functions are multi-valued at element boundaries, a numerical flux is introduced which provides stabilization. A recent advance is the development of the hybridized DG method which introduces trace unknowns as a means of reducing coupling between unknowns. This facilitates static condensation and reduces the global solve to a similar complexity as standard continuous Galerkin methods.

Space-Time Finite Element Methods

I also wanted to present a brief look at some of the work that has been done on space-time finite element methods. The basic idea is to treat time similar to another spatial dimension to be meshed and discretized. Early proponents of the idea include Kaczkowski and Oden. Lesoinne and Farhat analyzed several different methods for moving boundary problems in the context of geometric conservation laws and found space-time techniques to be superior to the other considerations. One challenge with space-time techniques is developing a discretization that is stable in both space and time. Tezduyar and his colleagues opt for a Galerkin/least-squares approach, while the camp of Van der Vegt and Van der Ven opt for discontinuous Galerkin. Ungor's tent pitcher algorithm is an effective way to decouple element solves, but only works for hyperbolic equations.

Overview of DPG

Now I'm going to present a brief mathematical overview of the DPG method. For more on the method, I would recommend Dr. Demkowicz's DPG Overview paper.

Dual Minimum Residual Method

We start with an abstract variational form and apply some basic functional analysis to it.

- So we are looking for a solution u in some trial space, big U .
- We've multiplied both sides by some test function v and somehow come out with a bilinear form on u and v .
- This bilinear form defines an operator which maps values from the trial space to the dual of the test space.
- We can then write the operator equation in V'
- Now we wish to find the solution u_h which minimizes the residual for some finite dimensional subset of big U
- The dual space norm is not especially convenient to work with computationally, but another functional analysis construct makes this more accessible.
- The inverse Riesz map, which is an isometry, turns this into a standard norm.
- Before I move on with the derivation of the method, I wanted to point out that really, this is a generalization of the least squares finite element method. If we chose the L_2 topology for V , we would get the standard least squares method. But now we are free to define the test space topology however we want.

Optimal Petrov-Galerkin Methods

- Since we have a minimization problem, we want to find the critical points, so we take the Gateaux derivative.
- We can then reverse one of the Riesz maps to give this duality pairing, which if you squint your eyes, kind of looks like a new bilinear form with a very special test function defined through the inverse Riesz map and operator B .
- The one complication is that the Riesz map is defined through the test space inner product, so evaluation of the optimal test functions requires an auxiliary problem with the test space inner product on the LHS and the bilinear form operator on the RHS.

The Most Stable Petrov-Galerkin Method

Babuska's theorem guarantees that discrete stability and approximability imply convergence. So given a bilinear form with discrete inf-sup constant γ_h , then the Galerkin error will be bounded by M , γ_h , and the best approximation error. But the optimal test functions are defined such that they realize the supremum and the discrete γ_h is greater than or equal to the continuous constant γ . If continuous problem is well-posed, OPG will be as well. Furthermore, using the energy norm in the above estimates gives $M = \gamma = 1$, and the optimal Petrov-Galerkin method is the most stable Petrov-Galerkin method possible.

Other Features

A few other interesting features of the DPG method.

The test space so far has not been defined, but if we were to use continuous test functions, then the auxiliary problem would turn into a global solve of similar complexity to the original problem. Introducing a broken test space localizes these solves making it embarrassingly parallel. The downside is that this introduces additional interface unknowns, which I will illustrate later with an example.

As a minimum residual method, the method always produces a hermitian positive definite stiffness matrix (SPD for real valued problems). We haven't really explored the implications for iterative solvers, but this seems like it would be a positive feature.

We can also evaluate the residual error without knowing the exact solution. This can be used to robustly drive adaptivity.

High Performance Computing

DPG has a lot of attractive features for high performance computing. Since I don't have infinite time, I'm not going to enumerate them, but we believe the method has a lot of promise for future HPC simulations. Parallel simulations are going to be a key element to this research, and we plan on running test problems on the Stampede supercomputer at TACC as well as Mira at Argonne National Lab.

Local Conservation

DPG for Convection-Diffusion

Now I'll go over some of my work developing a locally conservative DPG formulation using convection-diffusion as a model problem. We prefer to work with a system of first order PDEs, but it's possible to do DPG directly on the second order problem. The first equation represents our conservation law, and the second a constitutive law for stress σ .

Proceed by multiplying by test functions v and τ , then integrating by parts over each element K . Note that the test functions are discontinuous between elements. We seek field variables u and σ in L^2 , but this leaves their traces between elements undefined. So we introduce new flux and trace unknowns \hat{t} and \hat{u} . Putting it all together we get our new bilinear form.

Local Conservation for Convection-Diffusion

What does local conservation mean in the context of convection-diffusion? We essentially just want the net flux through element edges to be balanced by any source terms internal. If we were to look at the bilinear form on the previous slide, this is equivalent to having a particular test function in the test space, namely $v = 1$ and $\tau = 0$. Unfortunately, due to the dynamically computed nature of DPG, we can't guarantee that this will happen with standard DPG. However, we can explicitly augment our test space via Lagrange multipliers. The Lagrangian for our newly conservative scheme appears below.

Locally Conservative Saddle Point System

Proceeding as before, we find the critical points, but now we end up with a saddle point system rather than the symmetric positive definite stiffness matrix (which still resides in the upper left hand corner). We've also added an additional unknown for each element. An additional nicety is that since we've explicitly added constants to the test space, we only need to search for optimal test functions in a

space orthogonal to constants.

Optimal Test Functions

Here is how we do our optimal test function solve. For each trial function u , we need to find its complementary optimal test function v_u such that the inner product of v_u with some w equals the bilinear form acting on u and w for all w in V . Technically space V is infinite dimensional, but in order to make this computationally tractable, we use an enriched space to approximate the optimal test functions. This approximation can be troublesome when the inner product contains boundary layers as it can with convection-diffusion. Fortunately, the topology of the V space is our choice, and we can choose the test norm that we want. We've found that the following one works well for convection-diffusion.

Stability and Robustness Analysis

We were able to extend our stability and robustness theory for standard DPG to the locally conservative version via a Brezzi style analysis. I'm not going to reproduce the proof here in this presentation since it can get a little tedious, but you can refer to our ICES report for details.

Erickson-Johnson Problem

This is one of the few convection-diffusion problems with an exact solution, shown below. On the left we see our energy and L2 error for standard DG vs the conservative formulation. You can see that the conservative one lies nearly on top of the standard, showing at the least that it is not hurting us. You can also see plots of two different measures of conservation for the two methods. You can see that standard DPG isn't too bad and gets better with refinements, while the conservative one bumps around near machine precision. Both methods perform nearly identically, but the locally conservative one has a much smaller flux imbalance.

Stokes Cylinder

That was an easy problem, but let's see how the method does on a notoriously bad one. This is Stokes flow over a cylinder with a narrow channel on either side. Least squares methods will lose nearly 100% of the mass flux around the cylinder, and since DPG can be viewed as a kind of generalized least squares, we might expect a similar issue. On coarse meshes, standard DPG does indeed produce a poor solution, but converges with resolution. The locally conservative one produces a decent solution on even very coarse meshes.

Here you can see measures of mass loss at different points in the domain, and indeed you do see the standard DPG method getting better with refinement. Conservative DPG on the other hand bumps around at almost exactly zero mass loss.

Stokes Step

The reentrant corner in this problem again causes problems for least squares methods, but the conservative DPG method handles in beautifully. It's not until the standard method resolves the singularity at the corner that it really gets the flow profile correct.

And we see the same kind of story with the mass loss profiles.

Space-Time DPG

Motivation

Now I'm going to get into some of the details of my recent work on a space-time DPG formulation for transient problems. The work of Nate Roberts and Jesse Chan on incompressible and compressible Navier-Stokes focused entirely on steady-state problems, but that limits possible applications considerably.

We did explore the possibility of tacking on a finite difference based time stepping algorithm like most of the other methods out there, but ultimately decided it was not the best course of action. Standard time stepping methods require the entire mesh to move forward at the pace of the most restrictive element. For adaptive

meshes we may have several orders of magnitude between the largest and smallest elements, which results in a very inefficient technique. There are techniques out there to allow different elements to march at different time steps such as asynchronous variational integrators, but that means bolting on another complicated technology to DPG. At the same time, we would lose a lot of the attractive stability and adaptivity features of DPG in the time domain.

- The advantages of doing DPG in full space-time is that we preserve the exceptional stability and robustness properties of DPG.
- We treat time just like another dimension and get a unified theory for the whole thing
- Local spatial adaptivity usually requires temporal adaptivity as well, and this happens naturally in this case. We also have the option of doing anisotropic refinements if needed.
- Space-time has also become a popular framework for moving with moving meshes
- The downsides are that this is significantly more difficult to implement
- We now have to deal with meshes of one higher dimension. For 3D simulations, you would need a 4D mesh, which is a bit harder to think about. We only plan on doing 2D, but it is still a challenge.
- Equations also tend to have different characteristics in space vs time. For example, Navier-Stokes is second order in space with a diffusion operator, but only first in time. This means that certain traces are defined on spatial boundaries but not temporal ones.
- Each solve will be more expensive since it has a higher dimensionality, but hopefully we can make up for that by advancing larger elements at larger effective time steps. Also, time slabs should cut down on the overall cost.

Heat Equation

The heat equation is the simplest nontrivial space-time PDE. As we will see later, it actually serves as a decent model problem for something as complicated as compressible Navier-Stokes. Just like Navier-Stokes, it is parabolic in space-time.

So as we decompose this into a system of first order equations, we get a Fourier's law in space and conservation of energy. The key insight is that we can rewrite our conservation law as a space-time divergence operator.

DPG Formulation

We then proceed as usual, multiplying by test functions τ and v , and integrating by parts over each element. IBP of Fourier's law introduces a trace unknown \hat{u} , while conservation of energy introduces flux \hat{t} . The trace only exists on spatial boundaries Fourier's law is only integrated by parts over spatial dimensions. Note that the flux changes nature depending on whether it is on a temporal or spatial boundary or a mix of the two.

Pulsed Source Problem

We have one simple proof of concept numerical test.

Exam Notes

Boundary Layer Equations

1. Nondimensionalize variables
2. Substitute into incompressible Navier-Stokes equations
3. Following Prandtl, assume thickness of boundary layers small relative to characteristic length in streamwise direction
4. Estimate magnitude of terms
5. Drop smaller magnitude terms

Turbulence Modeling

Nonlinear convection-diffusion-reaction with positivity requirements

TODO

- Proof outline
- Study
 - Boundary layer equations
 - Rankine-Hugoniot conditions
 - Static Condensation
 - Derivation of interface unknowns
- Suit
- Food