A Very Brief History of Hydrodynamic Codes (i.e. Hydrocodes)

Bill Rider
Dept. 1431
wjrider@sandia.gov

Von Neumann

A few thoughts to start us off.

“Progress, far from consisting in change, depends on retentiveness. Those who cannot remember the past are condemned to repeat it.”
- George Santayana

“An expert is someone who knows some of the worst mistakes that can be made in his subject, and how to avoid them.”
- Werner Heisenberg
Talk Outline

- Introduction and key references
- The basic equations that are solved...
- Very early history - the 1st Lagrangian hydrocodes
- The basic logic regarding frame of reference
- Early history of Eulerian hydrocodes
- The Modern Era
  - New methods and their significance
  - The importance of computing power
  - ASC codes (ALE & AMR)

Key References

- Patrick Roache, Fundamentals of Computational Fluid Dynamics (1973)
- Methods in Computational Physics, Volume 3 1964 (Wilkins’ Springer Book)
- Elaine Oran and Jay Boris, Numerical Simulation of Reactive Flow
There are a host of other books available these days to read about CFD methods.

One can't be well versed without reading the literature available in journals.
Quote by Peter Lax: The American Mathematical Monthly, February 1965:

"...who may regard using finite differences as the last resort of a scoundrel that the theory of difference equations is a rather sophisticated affair, more sophisticated than the corresponding theory of partial differential equations."

He goes on to make two points:

1. The proves that an approximation converges is analogous to the estimates of the soln’s to the PDEs (points to the CFL paper in 1928)*
2. These proofs are harder to construct than for the PDEs

*CFL=Courant, Friedrichs, Lewy which used numerics to prove the existence of soln’s to PDE and gives us the term CFL condition.

Lax’s contributions have recently received a great honor - the 2005 Abel Prize

• The Abel prize was created to make up for the lack of a Nobel prize for mathematicians.
• Honored for defining basic numerical methods, fundamental theory, and detailed mathematical analysis in CFD, and more!
• Some of the work he was honored for started at Los Alamos and continued while at NYU’s Courant Institute. It forms much of the theoretical foundation for CFD.

Came to Los Alamos in WWII as a teen!
Its at the heart of really cool problems... These appeared in the London Underground in 2000.

Lagrangian and Eulerian descriptions of fluid dynamics

- The Lagrangian description moves with the fluid at the fluid velocity

- The Eulerian description has the fluid moving through it.
A historical footnote about Lagrangian and Eulerian coordinates.

- Eulerian coordinates are named for Euler, this terminology is due to d’Alembert
- Lagrangian coordinates are named for Lagrange, this terminology is due to Dirichlet

Why Lagrangian Hydrocodes?

- The Lagrangian equations are in many respects the most natural form to solve complex hydrodynamic problems.
- Lagrangian coordinates are superior to Eulerian coordinates for coupling to other physics (especially strength & radiation)
- Pure materials remain pure when treated numerically.
- The mesh provides a natural and cheap form of solution adaptivity.
Why Eulerian Hydrocodes

- Vorticity! The non-symmetric deformation of a material causes a material following a Lagrangian grid to distort without bound and become uncomputable.
- Many problems are more easily expressed in Eulerian coordinates.
- Many problems will compute without undue human intervention (Russians call this “crash-proof”)
- It provides an alternative to the Lagrangian based analysis of a hydrodynamic system.

Nature and technology are full of flows where Eulerian techniques are necessary for simulation.

- Lagrangian techniques are far more limited in applicability.
Why not use both Lagrangian and Eulerian (i.e. ALE)?

- ALE = “Arbitrary” Lagrangian-Eulerian
  - Invented at LANL by Hirt, Amsden & Cook
  - “perfected” at LLNL by Bob Tipton
  - Still an active area of research at all Labs
- It gives you the best of both worlds, and the worst,
  - All the science and all the art.
  - Unfortunately, computation remains an artful endeavor in many respects. This is especially true in shock physics (and more so when other physics are coupled).

The initial ALE paper in reprinted form.
Hirt, Amsden & Cook.

An Arbitrary Lagrangian–Eulerian Computing Method for All Flow Speeds*

C. W. Hirt, A. A. Amsden, and J. L. Cook†
University of California, Los Alamos Scientific Laboratory, Los Alamos, New Mexico 87545

Received November 10, 1972

A new numerical technique is presented that has many advantages for obtaining solutions to a wide variety of transient three-dimensional fluid dynamics problems. The method uses finite difference mesh with vertices that may be moved with the fluid (Lagrangian), held fixed (Eulerian), or be moved in any other prescribed manner, as in the Arbitrary Lagrangian–Eulerian (ALE) technique. In addition, it employs an implicit formulation similar to that of the Implicit Continuum-Eulerian (ICE) technique, making it equally appropriate for all problems.

This paper does not describe the methodological procedures for the different approximations, and discussions with heneas as stability, accuracy, and tuning. In addition, illustrations are included to form a number of representative calculations to the source text (Eulerian), or be moved in any other) of J. Comp. Phys. because of this flexibility the method Arbitrary Lagrangian–Eulerian (ALE) scheme of this nature has previously been developed by Trefil [5] for compressible flow problems. This new technique, however, may be applied to flows at any speed, since it has an implicit formulation similar to that in the Implicit Continuum-Eulerian (ICE) method [4]. In particular, in the limit of infinite sound speed, the difference equations reduce to a generalization of the Navier–Stokes equations for the incompressible Navier–Stokes equations [5]. The advantages of the ICEDALE method include its non-

*30th anniversary of J. Comp. Phys.
Here are the basic governing equations for hydrocodes.

- **Lagrangian**
  \[
  \frac{d\mathbf{x}}{dt} = \mathbf{u}; \quad \frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\rho} \nabla \sigma = 0; \quad \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0
  \]

- **Eulerian (conservation form)**
  \[
  \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0; \quad \frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) + \nabla \sigma = 0; \quad \frac{\partial \rho E}{\partial t} + \nabla \cdot (\rho \mathbf{u} E + \rho \sigma \mathbf{u}) = 0
  \]

- **Eulerian (non-conservation form)**
  \[
  \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0; \quad \frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = 0; \quad \frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho \mathbf{u} e + \frac{1}{2} \rho \sigma \mathbf{u}) = 0
  \]

- **ALE**
  \[
  \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho (\mathbf{u} - \mathbf{u}_s)) = 0; \quad \frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} (\mathbf{u} - \mathbf{u}_s)) \cdot \nabla \mathbf{u} + \frac{1}{\rho} \nabla \sigma = 0; \quad \frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho \mathbf{u} e + \frac{1}{2} \rho \sigma \mathbf{u}) = 0
  \]

Because we deal with shocks, the jump conditions are essential.

- **The jump conditions are called the Rankine-Hugoniot equations.**

  - **Lagrangian**
    \[
    [\Phi] = \Phi_i - \Phi_o \quad \bar{\Phi} = (\Phi_i + \Phi_o) / 2
    \]
    \[
    W[\mathbf{v}] = -[u]; W[u] = [p]; W[E] = [pu]
    \]
    \[
    W[\mathbf{v}] = -[u]; W[u] = [p]; W[E] = \bar{p}[u]
    \]
    \[
    W = (-p c, 0, \rho c)^T \quad c^2 = \left. \frac{\partial p}{\partial \rho} \right|_s
    \]

  - **Eulerian**
    \[
    w[\rho] = [\rho u]; w[\rho u] = [\rho u^2 + p]; w[\rho E] = [\rho u E + pu]
    \]
    \[
    w = (u - c, u, u + c)^T
    \]
Another thought courtesy of Peter Lax

"How fortunate that in this best of all possible worlds the equations of ideal flow are nonlinear!"

SIAM Review, January 1969

Of course how could it be otherwise?

Shocks are treated in a couple of ways by codes either explicit or implicit.

- Tracking uses the information from solving the R-H equations to track the evolution of discontinuities.
  - The complexity of this approach increases w/o bound especially in 2- or 3-D (also called fitting).
- Capturing applies a numerical technique to marry a discontinuity to a finite grid using numerical dissipation to mimic the entropy creation by shocks.
  - Shock width is proportional to $\Delta x$

$\frac{d\bar{x}}{dt} = \bar{u}; \frac{d\bar{u}}{dt} + \frac{1}{\rho} \nabla (\sigma + Q) = 0; \frac{de}{dt} + \frac{1}{\rho} (\sigma + Q) \nabla \bar{u} = 0$

- Tracking is quite complex for shocks/rarefactions, but it is still used quite a bit for material discontinuities (for giving Eulerian codes a Lagrangian character)
**A thought about thermodynamics!**

In this house, we **OBEY** the laws of thermodynamics!*

*It is an actual Homer quote!

---

**The origin of hydrodynamic calculations**

- The first hydro calculation was reported in a Los Alamos report on June 20, 1944
  - lead author Hans Bethe
  - Feynmann was the calculational lead
  - Still classified!
- The first codes were 1-D and Lagrangian, shocks were tracked (no viscosity, finite differences failed completely as of 1945).
- Artificial viscosity was developed by Von Neumann and Richtmyer (Richtmyer published a report in 1948)
  - Q's made codes much easier to write and use!
The artificial viscosity paper by Von Neumann and Richtmyer, J. Appl. Phys. 1950

A Method for the Numerical Calculation of Hydrodynamic Shocks

J. Von Neumann and R. D. Richtmyer
Institute for Advanced Study, Princeton, New Jersey
(Received September 26, 1949)

The equations of hydrodynamics are modified by the inclusion of additional terms which greatly simplify the procedures needed for stepwise numerical solution of the equations in problems involving shocks. The quantitative influence of these terms can be made as small as one wishes by choice of a sufficiently fine mesh for the numerical integrations. A set of difference equations suitable for the numerical work is given, and the condition that must be satisfied to ensure their stability is derived.

I. INTRODUCTION

In the investigation of phenomena arising in the flow of a compressible fluid, it is frequently desirable to solve the equations of fluid motion by stepwise numerical procedures, but the work is usually severely complicated by the presence of shocks. The shocks manifest themselves mathematically as surfaces on which density, fluid velocity, temperature, entropy and the like have discontinuities; and clearly the partial differential equations governing the motion require boundary conditions connecting the values of these quantities on the two sides of each such surface. The necessary boundary conditions (but preferably somewhat larger than) the spacing of the points of the network. Then the differential equations (more accurately, the corresponding difference equations) may be used for the entire calculation, just as though there were no shocks at all. In the numerical results obtained, the shocks are immediately evident as non-discontinuities that move through the fluid with very nearly the correct speed and across which pressure, temperature, etc. have very nearly the correct jumps. It will be seen that for the assumed form of discontinuity (and, indeed, for many others as well), the Rankine-Hugoniot equations are satisfied, provided the thick-

LA-671 a precursor to the Von Neumann-Richtmyer paper. By Richtmyer (only!)

LA-671

March 1949

This document contains 36 pages.

PROPOSED FUNDAMENTAL METHOD FOR CALCULATION OF SHOCKS

Mark Done By:
R. D. Richtmyer

Report Written By:
R. D. Richtmyer

The new content in the J. Appl. Phys. Paper is Von Neumann Stability analysis

Classified till 8/26/93!

UNCLASSIFIED
How have Lagrangian codes and artificial viscosity progressed over time?

1944 - Von Neumann's attempt to compute shocks fails
1948 - Richtmyer's description of artificial viscosity.
1950 - The Von Neumann-Richtmyer paper
1955 - Landshoff's linear viscosity developed
1955 - Rosenbluth's suggestion to turn off Q's when the flow is in expansion
1964 - Schulz's 2-D method and tensor viscosity
1970 - Kurapatenko's analysis relating Q's to the analytical behavior of the Hugoniot.
1990 - Tipton's CALE code
1992 - Christenson's nonlinear "limited" Q
1997 - The publication of compatible Lagrangian hydro

Family tree of Lagrangian hydrocodes by Gene Hertel (SAND 97-1015C)
Eulerian hydrocodes are natural for some problems.

- Eulerian codes may have got their start in numerical weather prediction (Von Neumann at IAS—more later)
- At the Weapons Labs – particle-in-cell codes appeared in the mid-1950’s at LASL
- Grid based Eulerian codes – Noh’s CEL code in Methods in Computational Physics, Volume 3 1964
- The key to Eulerian methods are high-resolution schemes, high-order with monotonicity preservation

The first “real” Eulerian calculation was for weather forecasting.
It took place at IAS (Princeton) in 1950 involving, among others, John Von Neumann.

The connection between weather modeling, John Von Neumann and Large Eddy Simulation

- In 1956 a simulation by Norm Phillips of weather over the eastern half of the US for a month was completed and the subject of a meeting at IAS.
- Late in the simulation the solution began to experience instability (ringing)
- It was suggested by Charney that "Von Neumann’s viscosity" might control this ringing.
- Smagorinsky completed the follow on simulation including this technique
  - This technique became the first Large Eddy Simulation (LES) subgrid turbulence model
The evolution of computers is hard to separate from the history of codes

Of course there are Crays from the 70's-90's
...and the modern ASCI era with room filling machines again!

We’ve been gliding along with Moore’s law for 40 years, will it continue?

- Recently, IBM, Sony & Toshiba has started putting parallel processors on a chip. It’s called Cell (9 proc.)
More recently the whole World has played in this field

One important aspect is the development of operating systems and languages

• In the early days of computers programming was much more challenging, even involving the physical modification of the computer in order to implement programs.
• This difficulty limited the complexity of algorithms that one would place on a machine.
One important aspect is the development of operating systems and languages

- The arrival of Fortran was a big advance.
- The next big event was the placement of programs in memory.
- The operating systems were constantly changing, especially at LASL.
- How did you begin programming?

Most of the Eulerian codes at the NNSA Labs use staggered grids.

- This is a strong consequence of history - early methods at the Labs were based on Lagrangian codes.
- Some codes are basically ALE codes with a full remap every time step
- There are important exceptions which use Godunov-type methods, and AMR
- These days AMR is almost synonymous with Eulerian hydrocodes.
The choice of grid type influences methods heavily

- Collocated or Cell-centered
- Staggered

In 2-D the choices are more complex
- Collocated
- Panel-staggered
- Vertex-staggered

The grid selection effects pathologies associated with methods.

- In 1-D there are two modes of physical motion translation and dilation

\[ \frac{\partial u}{\partial x} \]

- Solutions are impacted by checkboarding, or grid decoupling seen as hourglassing in Lagrangian calculations.
  - A staggered grid has two DOF exactly spanning the space.
  - A cell-centered grid w/centered differencing has 3 DOF, an extra. This instability is cited as a key reason for avoiding cell-centered grids.
Upwinding gets rid of one of these three degrees of freedom!

- With two degrees of freedom, the extra checkerboarding mode is gone.
- This explains why upwinding is used extensively with cell-centered grids.
- These arguments go over to two and three dimensions.

In 2-D there are six physical modes of motion.

- For a vertex-centered grid there are two extra modes of motion, the checkerboard modes.
- For the cell-centered grid there are four extra modes.
- Face-centered grids are under-determined (no checkerboard at all!)
Particle-in-cell codes were the first semi-Eulerian methods.

Developed in 1955, PIC codes were used for hydro until Eulerian hydrocodes matured. PIC is still used extensively in plasma physics.

PIC uses particles to represent material motion and an Eulerian grid for other physics, the problem is how to connect the particles to the grid.

The solution technique used in PIC is still used today.

- The PIC method is a split method:
  1. Solve the equations without material motion.
  2. Compute the material motion using the particles.
- This basic procedure was used in the first Eulerian codes.
- Marvin Rich wrote the first Eulerian code in this manner in 1962.
- The OIL series of codes was written based on PIC codes, by Wallace Johnson (OILER too?).
There is basically a split in the history of Eulerian hydrocodes.

- Staggered codes come from a line of codes starting in LASL and LLNL,
  - Several early codes were derived from PIC codes, replacing particles with a continuous field.
  - Other codes were based on extending the Von Neumann-Richtmyer code in the most straightforward way possible.
- Cell-centered codes come from outside the Lab (sorta), with Lax-Wendroff and Godunov methods.

An important aspect of PIC codes was that they served as the basis for Eulerian codes.

- At least at Los Alamos this was true.
- Marvin Rich (1962) extended the basic PIC framework by removing the particles and replacing it with a continuous field.
- Gentry, Martin and Daly (1966) extended this work with the FLIC (Fluid-in-Cell) method.
Review

Fluid dynamics in Group T-3
Los Alamos National Laboratory (LA-UR-03-3852)

Francis H. Harlow *
Los Alamos National Laboratory, MS-3075, Los Alamos, NM 87544, USA
Received 12 June 2003; revised 30 August 2003; accepted 3 September 2003

Abstract

The development of computer fluid dynamics has been closely associated with the evolution of large high-speed computers. At first the principal incentive was to produce numerical techniques for solving problems related to national

Family tree of Eulerian hydrocodes by Gene Hertel (SAND 97-1015C)

1990
MESA/PAGOS

1975
TRIDORF
DORP9

1965
OIL
SPEAR

1955
PIC

Lecture 1/27: Introduction to Methods for Eulerian Hydrodynamics, LA-UR-03-7300, Bill Rider, wjrider@sandia.gov
Grid based Eulerian codes started appearing in the late 1960's at LASL & LLL (GA too).

- 1962 - Marvin Rich’s Los Alamos report on Eulerian hydro
- 1961 or 1962 - The OILER code at LLNL
- 1964 - William Noh’s CEL method
- 1965 - Wallace Johnson develops the OIL code based on a PIC method
- 1966-1968 - Richtmyer’s L-W method plus Emery’s paper in JCP, FLIC by Gentry at LASL
- 1968 DeBar at LLL develops KRAKEN

Grid based Eulerian codes underwent major developments outside the Labs ~1970

- 1969 - MacCormack’s method developed at Stanford
- 1969 - Van Leer’s thesis work
- 1971 - FCT invented by Boris & Book at NRL
- 1972 - Van Leer’s nonlinear method
- 1972 - Kolgan’s high-order Godunov method in USSR
- 1974 - Harten’s methods from NYU and Israel
- 1975-1982 - These methods start to appear at the NNSA labs, and reinvigorate Eulerian code development.
Grid-based codes were too dissipative until the early 1970’s when...

- ... a revolution began in computational physics.
- Within the span of one or two years four researchers independently developed a key idea that made Eulerian codes viable.
- All four developed “high-resolution” methods:
  - Boris (NRL) Flux Corrected Transport
  - Van Leer (Leiden, Netherlands) limiters
  - Kolgan (USSR, Taiga) high-order Godunov
  - Harten (NYU/Israel) self-adjusting hybrid

Monotonicity is a desirable property to maintain numerically leading to the suppression of oscillations without too much dissipation.

The first high-resolution methods invoked a geometric definition of monotonicity.

For an interpolation the reconstruction of a function in a cell should not exceed the values of its neighboring cells.
Godunov’s Theorem relating high-order and monotonicity

• Godunov’s theorem says that a high-order linear methods (2nd or higher) cannot be monotone.
• Restated: only 1st order linear methods are monotone
• A linear method uses the same differencing stencil for all zones.

• Godunov also developed a method that we’ll talk about in a later lecture.

Godunov’s account of the creation of his method and theorem.


SPECIAL ARTICLE

Reminiscences about Difference Schemes

Sergei Konstantinovich Godunov

Stecklov Institute of Mathematics, Russian Academy of Sciences, St.-Petersburg Branch 650400 Novosibirsk, Russia

Received August 31, 1999

FOREWORD

In these notes, I will tell how in 1953–1954 the first version of the “Godunov’s scheme” was invented and how it was modified in the subsequent works by myself (until 1969) and by others at the Institute of Applied Mathematics in Moscow (now named after its founder, Academician M. V. Keldish).
Overcoming Godunov’s Theorem with nonlinear methods

- The key to overcoming Godunov’s theorem is using nonlinear methods – using different stencils dependent on the local solution.
- Developed independently by four men in 1971-1972
  - Jay Boris (NRL)
  - Bram Van Leer (U. Leiden)
  - Kolgan (USSR)
  - Harten (Israel)

What the heck was going on in 1971 and 1972? Was something in the water?

- Eulerian hydrocodes became mainstream with the publication of the Journal of Computational Physics,
  ...
- ... and the availability of large scale computing like the CDC machines.
- The Lax-Wendroff method was the mainstay of computations outside the Lab and provided the basis to work from.
- Aerodynamics, combustion and astrophysics communities all started computing (more naturally Eulerian in how the problems were cast).
For Eulerian codes the appearance of the CDC 6600 appears to be meaningful (mid 60’s).

- The combination of new machines and new methods make useful Eulerian codes possible.

The most obvious aspect is the raw performance of the machines.

The LLNL Plot

Follows Moore’s Law (approx.)

The advent of Eulerian hydrocodes

Year introduced


Peak speed: operations per second (circuits)

10^6 10^7 10^8 10^9 10^10

Next range is exaflops
The AWE had a similar picture of gains in Computer power...

The advent of Eulerian hydrocodes

LANL' curve in computing power 1945-1994

The advent of Eulerian hydrocodes

Eulerian codes Appear at about the 1 MFLOP rate
Grid based Eulerian codes spread across the world in the 1970’s:

- 1972 Hirt’s ALE method published at LASL
- 1974 KRAKEN code at LLL (DeBar’s paper)
- 1979 Woodward works with Van Leer in Leiden using BBC as a template - MUSCL
- 1981-1984 Woodward & Colella develop PPM
- 1982 Harten’s TVD method
- 1987 UNO & ENO developed by Harten with the UCLA crowd
- 1994 & 1996 - WENO developed first by Liu, Osher & Chan then refined by Jiang & Shu.

Here are some of the other key players in the development of high-resolution methods:

Paul Woodward  
Phil Colella  
Phil Roe

Ami Harten  
Chi-Wang Shu & Stan Osher
The development of these nonlinear methods have allowed Eulerian hydrocodes to be useful.

- The numerical diffusion associated with material interfaces limited the first Eulerian methods in two ways:
  - Until the mesh density could be increased, and/or
  - The methods used were less diffusive
  - Effective interface tracking.
- This also includes ALE methods which suffer from the same issues albeit to a lesser degree

What about the present and future? What will be the legacy of the ASC(I) era?
What about the future?

“But the only way of discovering the limits of the possible is to venture a little way past them into the impossible.”

Arthur C. Clarke [Clarke’s Second Law]

“Nothing is destroyed until it is replaced.”

–Auguste Compte

Acknowledgements

• I’d like to especially thank Paul Whalen, Frank Harlow, Paul Woodward, Len Margolin, Jay Boris, David Youngs and Bram van Leer for discussions that have been essential to this material.
Questions?