High Performance FORTRAN Motivating Applications and User Feedback

Paul Havlak
Ken Havwick

CRPC-TR95612
January 1995

Center for Research on Parallel Computation
Rice University
6100 South Main Street
CRPC - MS 41
Houston, TX 77005

Also available as *SCCS-692* from the Northeast Parallel Architectures Center at Syracuse University.

HPF Motivating Applications and User Feedback

Ken Hawick

Northeast Parallel Architectures Center

Syracuse University

and

Paul Havlak

University of Maryland

Overview:

- Projects
  - HPF-2 Motivating Applications collection effort (Maryland)
  - High Performance Fortran Applications (HPFA) project (Syracuse)
- FTP/WWW sites at Maryland and Syracuse
- Features
  - Education and Training Material
  - Source Code Collection and Documentation
  - User Feedback

Education, Training and Outreach

- Premise: The best way to accelerate usage of HPF is to provide good example codes.
- *Good* = short enough to understand ($<$ 1000 lines?) and long enough to illustrate something interesting ($>$ 25 lines?)
- Need volunteers for:
  - Existing codes or new ones written to illustrate particular features
  - Other explanatory and training material
  - Reviewing and editing existing material

Application Categories

- Kernels: fragments demonstrating use of some feature or idiom
- Mini-Applications: simplified programs with similar algorithms and tradeoffs to full applications
- Full Applications: true test; may be too complicated for non-experts

Motivating Applications Overview

- Goal: guide the extension *or non-extension* of HPF
- Are programmers leaving HPF behind?
  If so, where are they going?
- Collect representative mini-applications
  - computationally demanding problem domains
  - programming idioms representative of full-size codes
- Clarify the choices of what to support with HPF-2

Example Applications

- Neighbor-list computations
  - computational chemistry, irregular meshes
  - require irregular mapping of data and computations
- Particle-in-Cell
  - vortices, direct-simulation Monte Carlo gases
  - irregular mapping and data motion as generalized reduction
- Task parallel codes: CMU Fx project examples of non-scalable subproblems.
- Scalable I/O: out-of-core arrays; checkpointing.
- Dynamic linked structures
  - e.g., Barnes-Hut tree for long-range N-body problems
  - required primitives not well understood

Significance

- Underneath the laundry list of desired HPF-2 features are genuine application requirements!
- The goal of high-level, portable high performance for these applications should guide the HPF-2 process.

Mini-Applications and Applications Collected

- Elliptic Equation Solution by ADI and Gauss-Seidel
- N-Body Molecular Dynamics
- Fast Fourier Transform in 2d
- NAS EP Benchmark – Tabulation of Random Numbers
- NAS IS Benchmark – Integer Sorting
- NAS MG Benchmark – Regular Multigrid Solver
- NAS CG Benchmark – Conjugate Gradient Solver
- Convolution in 2d
- Accept/Reject for Gaussian Random Number Generation
- Monte Carlo Integration for Stock Pricing
- Spanning Percolation Cluster Generation in 2d
- CFD Pipe Flow Simulation in 2d
- 2d Potts Model Simulation using Metropolis Heatbath

- 2d Binary Phase Quenching of Cahn Hilliard Cook Equation
- NAS BT Benchmark – Block Tridiagonal
- NAS FT Benchmark – Fourier Transform
- NAS SP Benchmark – Scalar Pentadiagonal
- NAS LU Benchmark – LU Decomposition
- Electromagnetic TE Code
- Electromagnetic TM Code
- FFT NPAC Benchmark
- Gaussian Elimination NPAC Benchmark
- Laplace Solver NPAC Benchmark
- Integration Computation of Pi NPAC Benchmark
- N-Body NPAC Benchmark
- Simplex NPAC Benchmark
- Modified Simplex NPAC Benchmark
- Monte Carlo Simulation of Dynamically Triangulated Random Surfaces

- Monte Carlo Simulation of Fixed Triangulation Random Surfaces
- Shallow Water Climate/Weather Model
- Block LU using GAXPY
- Block LU using SAXPY
- Block LU using SDOT
- Block Cholesky Decomposition
- Block QR
- Metropolis Monte Carlo Simulation for Spin Models
- Swendsen-Wang Monte Carlo Simulation for Spin Models
- Wolff Monte Carlo Simulation for Spin Models
- Segmented Bitonic Sort
- Van Leer/ Prather Advection for Atmospheric Transport
- Barnes-Hut
- ASA – Accessible Surface Area calculation
- Molecular Dynamics (MolDyn)/Non-bonded Force with Cut-off

- EULER: A Multimaterial, Multidiscipline, 3-D Hydrodynamics Code
- Multigrid (MG)
- Binz – Vortex Dynamics
- DSMC (Direct Simulation Monte Carlo) method
- Sparse Cholesky Factorization
- Flame Simulation
- Fock Matrix Construction
- Task parallel (these four codes from CMU/Fx)
- FFT: Fast Fourier Transform (TASK version)
- Narrowband tracking radar
- Multibaseline stereo
- Airshed simulation
- Out-of-Core Matrix Transposition
- FFT: Fast Fourier Transform (VIEWAS version)
- SpLU – Sparse LU Factorization

Thanks to

- Maryland
- Syracuse
- LONG LIST of contributors online... (including much of the HPFF)

User Experience

- How can this be quantified/qualified?
- Example codes? (full applications don’t actually help as they tend to be far too big to digest by non-domain expert)
- Anecdotes and “war stories”? (tend not to be specific enough)

Some (Industrial) Potential-User Comments:

- “planning to port to HPF when available” (UK Meteorological Office)
- “want to use HPF, but only if it does everything F90 does”, Aerospace Company.
- “want to use it, but where is a real compiler?”, Defense Contractor.
- “misgivings about Fortran 90, let alone HPF - we remember the pain of going from 66 to 77”, major Oil Reservoir Simulation Code ISV.
- “will it become a real standard? Then we will use it.”, CFD ISV.
- “We believe we will want to use HPF when we get an actual compiler on our workstation clusters, but we are concerned it is limited to regular (load balanced) problems and that we will need to hack PVM code into the application.” (Other Aerospace Company)

Some (Academic) User Comments

- “Quicker to get simulation code written”, Syracuse student
- “Easier to learn and remember than message passing”, Edinburgh Student
- “wish it was more widely available”,
- lots of FAQ suggests that it is easy to learn the ideas but not the syntax (problem of several ways of expressing the same thing)

Contents of the “HPFA” Web Package:

- (Incomplete) List of Available Compilers;
- Links to HPF material on the Web at other sites — (contributed links welcome)
- List of Industrial and Academic application areas with an indication of appropriate software including suitability of High Performance Fortran.
- List of generic exemplar applications codes with discussion of issues of relevance to HPF and HPF+.
- List of papers and books on HPF, Fortran90 and associated parallel computing issues;
- Talks and Lectures on HPF and an On-line HPF Tutorial;
- (http://www.npac.syr.edu/hpfa/index.html)

Conclusions:

- Deficiencies in HPF, being addressed by HPF-2...
- Can we identify any obvious priorities from user requirements?
- Increase the outreach process...
- more example codes...(any volunteers?)
- Demand and potential demand for real compilers exists!

More Information & Online Internet Resources:

- **http://www.npac.syr.edu/hpfa/index.html**
  HPFA Project material at NPAC (codes, documents, training material, as well as links to other sites, including Maryland)
- Ken Hawick (hawick@npac.syr.edu)
- **ftp://hpsl.cs.umd.edu/pub/hpf_bench/index.html**
  Maryland "HPF Motivating Applications"
- Paul Havlak (havlak@cs.umd.edu)