Some Challenges in Combinatorial Optimization for High-Performance Computing

Jean-Paul Watson

jwatson@sandia.gov

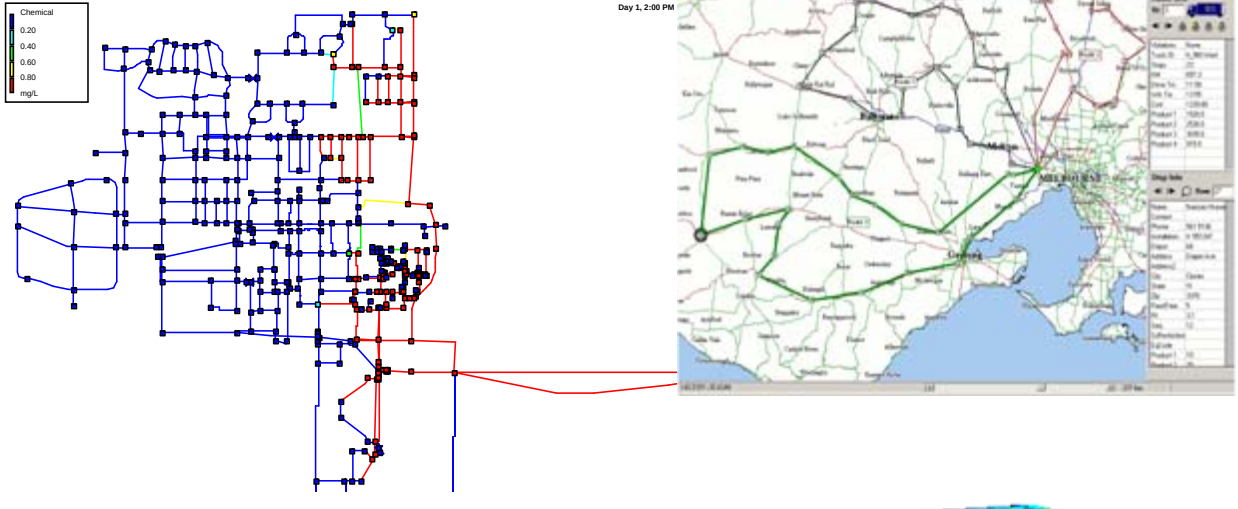
Discrete Mathematics and Complex Systems

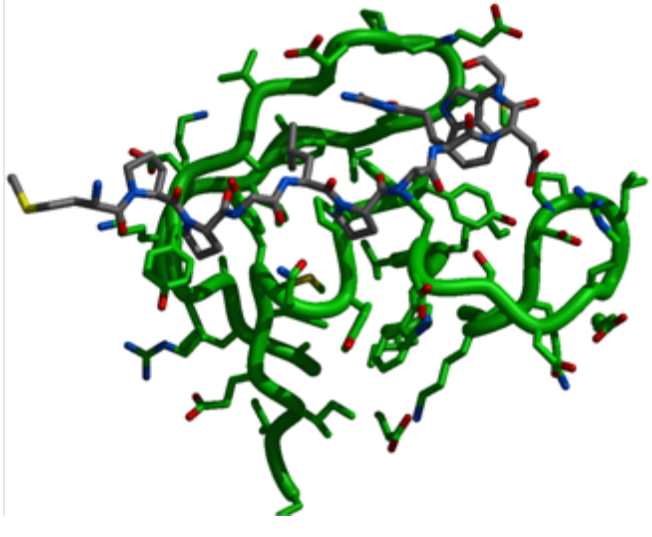
Sandia National Laboratories

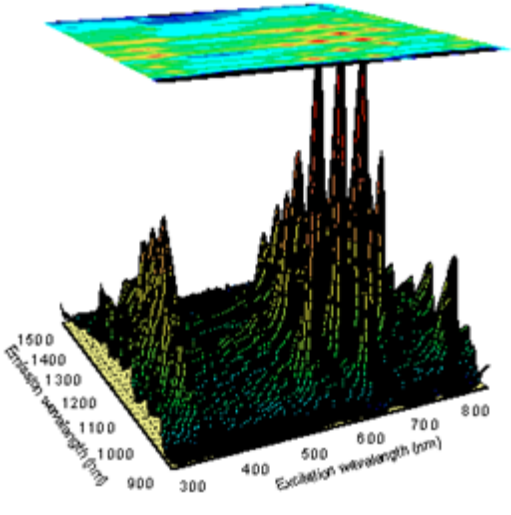
Albuquerque, New Mexico USA

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Examples of (Combinatorial) Optimization Problems at SNL







Sandia National Laboratories

Where is HPC heavily leveraged in optimization?

- Engineering design (DAKOTA)
  - Parallel evaluation of objective function
  - Pattern search, genetic and evolutionary algorithms
- Mixed-Integer Programming (CONDOR, PICO, …)
  - Parallelization of branch-and-bound process
  - Exact solution of large TSP and QAP instances
- Metaheuristic search (tabu search, parallel tempering, etc)
  - Independent iterative search trajectories
  - Periodic communication of solutions among trajectories
- A Paradox (?)
  - Combinatorial optimization problems are pervasive (DWave)
  - Parallelism is not widely used in practical optimization

Parallelization of Heuristically Guided Tree Search

- Tree search basics (100K feet)
  - Select an unexplored node in the search tree
  - Select a *variable* to branch on
  - For each legal *value*, propagate effects and create new node
- Core of SAT and constraint programming solver engines (AI)
  - Pervasive in scheduling and routing applications
  - In contrast to MIP, little computation per node required
- Heuristic “merit of worth” assigned to each node (variable ordering)
  - Significantly faster than traditional tree search strategies
  - Requires heap or pseudo-heap as core data structure
- Cluster / Grid => High communication-to-computation ratio
- SMP => Low numbers of processors, contention in heap update

Parallelization of Very Large Neighborhood Search

- Significant recent trend in iterative search
  - Larger neighborhoods => better chance of escaping optima
- Wide-ranging practical impact
  - Facility location, constraint programming, vehicle routing
- Basic operation, given a starting solution
  - Generate (large) polynomial number of neighboring solutions
  - Compute “worth” of each neighboring solution (fast)
  - Find best neighboring solution
  - Recurse on best neighboring solution
- Cluster / Grid => Synchronization, high comm-to-compute ratio
- SMP => Spin-locking on best neighbor

Parallelization of “Landscape Mapping” in Search

- From “point” solutions to real decision support
  - Identification of the *set* of high-quality solutions
  - Implicit mapping of the search space or fitness landscape
  - Evidence: CPLEX 11
- Algorithmic structure (MD, metaheuristics)
  - Set of high-quality “landmark” solutions
  - Independent set of search trajectories
  - Frequent query of landmark solutions to modify trajectories
  - Frequent update of landmark solution set
- Cluster / Grid => Distributed query / management of landmarks
- SMP => Contention in query / management of landmarks

Summary and Wrap-Up

- In combinatorial optimization, there is no silver bullet technology
  - Or: No over-arching paradigms (contrary to OR belief)
  - Toolbox of techniques, fractional market share for each

- As a consequence, can't afford to expend significant development effort to develop parallel optimization codes
  - Need tools that facilitate rapid prototyping and deployment

- I've tried to show some potentially low-hanging fruit for combinatorial optimization on alternative HPC architectures
  - Symbolic, light-weight computations
  - FLOPS are *not* the main concern

Questions?

- Thanks!