Efficient Atomic Cluster Optimizer

Kumara Sastry
Illinois Genetic Algorithms Laboratory
University of Illinois at Urbana-Champaign
Urbana, IL 61801
http://www-illigal.ge.uiuc.edu

Genetic and Evolutionary Computation Conference
(GECCO-2001)
July 7-11, 2001
San Francisco, CA

Overview

- Background & Motivation
- Objective
- Overview of ECGA
- Algorithm Description
- Results & Conclusions

Background: GA Design

- Design of *competent* GAs: A key challenge
  - Solve hard problems Quickly, Reliably and Accurately
- Much progress made (Goldberg, 1999)
- Existing competent GAs:
  - Render intractable problems tractable
  - Require subquadratic function evaluations
- ECGA is a competent GA (Harik, 1999)

Background: Cluster Optimization

- Used in Surface & atomic simulations
- Simplest problem is NP hard
- Local minima grows as $\exp\left(n^2\right)$
- GAs for cluster optimization:
  - Hartke (1993,1995); Zeiri et al, 1995; Deaven & Ho, 1995; Gregurick & Alexander, 1996; Niesse & Mayne, 1996; Zeiri, 1997; Iwamatsu, 2000
- They use "not-so-good" operators

Motivation

- ECGA: $\mathcal{O}(\ell^k)$ function Evals
  - for small clusters, $k \approx 8.2$ (Sastry& Xiao, 2001)
- Clusters with large no. of atoms
  - function evaluations is high
  - Need *Efficiency Enhancement Techniques* (EET)
- *Hybridization* and *Seeding* are EETs

Objective

- Employ ECGA to optimize atomic clusters
  - Hybridize with a local search
    - Nelder-Mead simplex used as local search
  - Seed initial population
- Obtain better scale-up
- Solve larger clusters
- Silicon clusters used as test case

Overview of ECGA

- Probability distribution $\equiv$ *Linkage learning*
- Prob. dist.: *Marginal Product Models*
  - Maps models of good data as linkage groups
  - Groups linked variables as a single variable
  - Eg. $[1],[2,5,9],[3,8],[4,6],[7],[10]$
- Quantified by *Minimum Description Length*
  - Penalize inaccurate distributions
  - Penalize complex distributions

Encoding & Fitness Function

- Variables: Fixed-space Cartesian coords
  - Each atom is coded by three variables
  - Each coordinate is encoded by 5-bit binary
- Fitness Function: Cluster potential energy
- Silicon Potential:
  - Gong, X.G. *Phys. Rev. B* **47**, 2329 (1993)
  - Empirical two & three body potential
  - Also includes angular terms
  - Accurate for predicting structural properties

Seeding Initial Population

- Initial population generated through *seeding*
  - Hoare (1979), Niesse & Mayne (1986)
  - Use optimal structure of $n-1$ atom cluster
  - Insert an atom to the $n-1$ atom cluster
  - Randomly generate its position
- Considerably reduces the population size
- Initial structures have better fitness

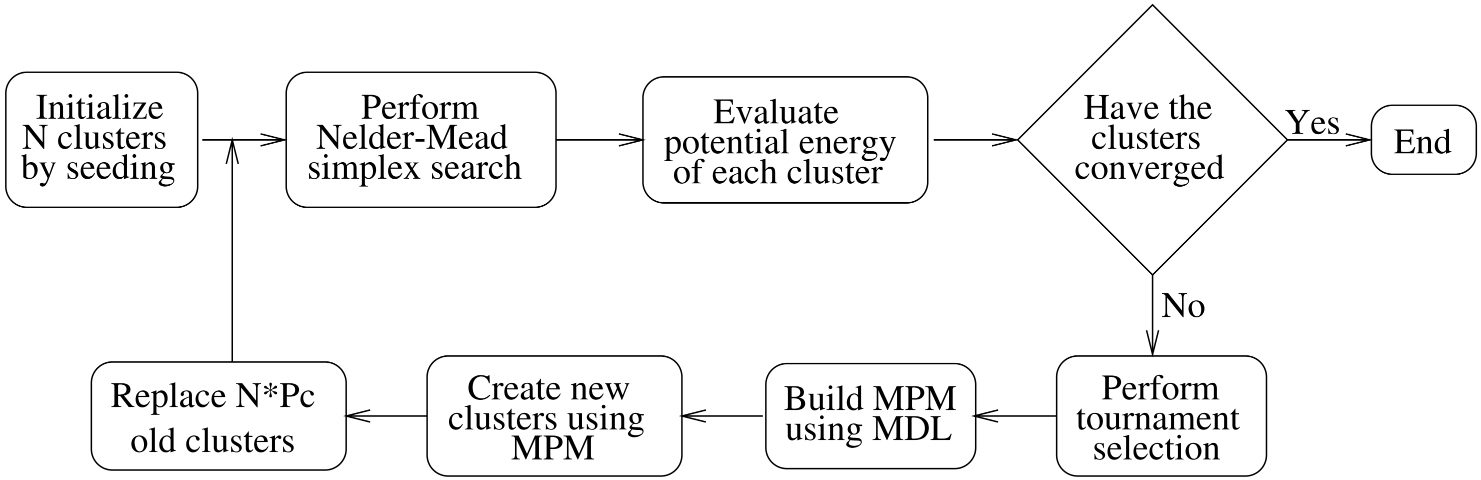
Hybridization

- Nelder-Mead simplex (Press et al, 1989)
  - Requires $3n + 1$ initial points
  - The individual accounts for one point
  - Perturb an atom in one coordinate
  - Creates $3n$ points
- Local search for every individual
- Use fully lamarckian approach
  - Local search solution replaces the individual

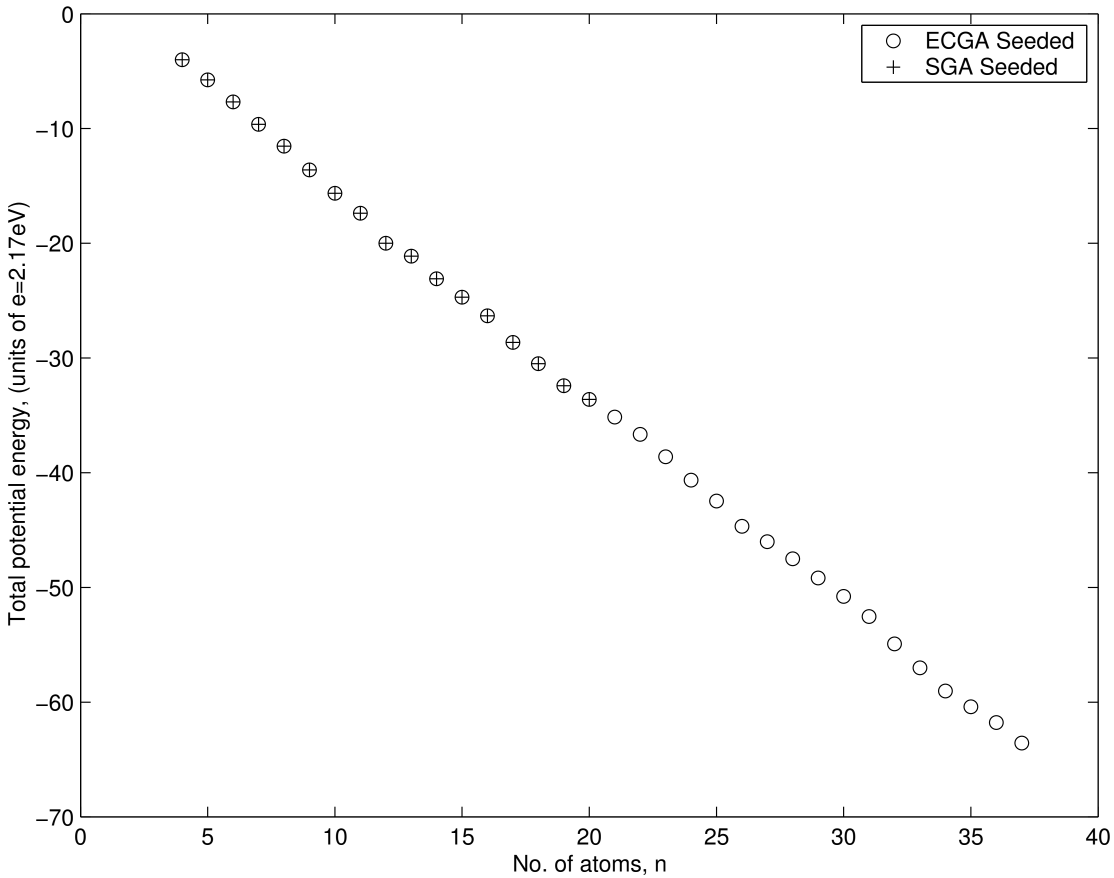
Creating New Individuals

- Create n_p individuals using MPM
  - Generate each partition independently
  - Assign values proportional to the frequency
  - m point crossover between n_p individuals.
- Elitist replacement scheme
  - Select top n_p individuals from
    - n_p new individuals, and
    - n_p old individuals

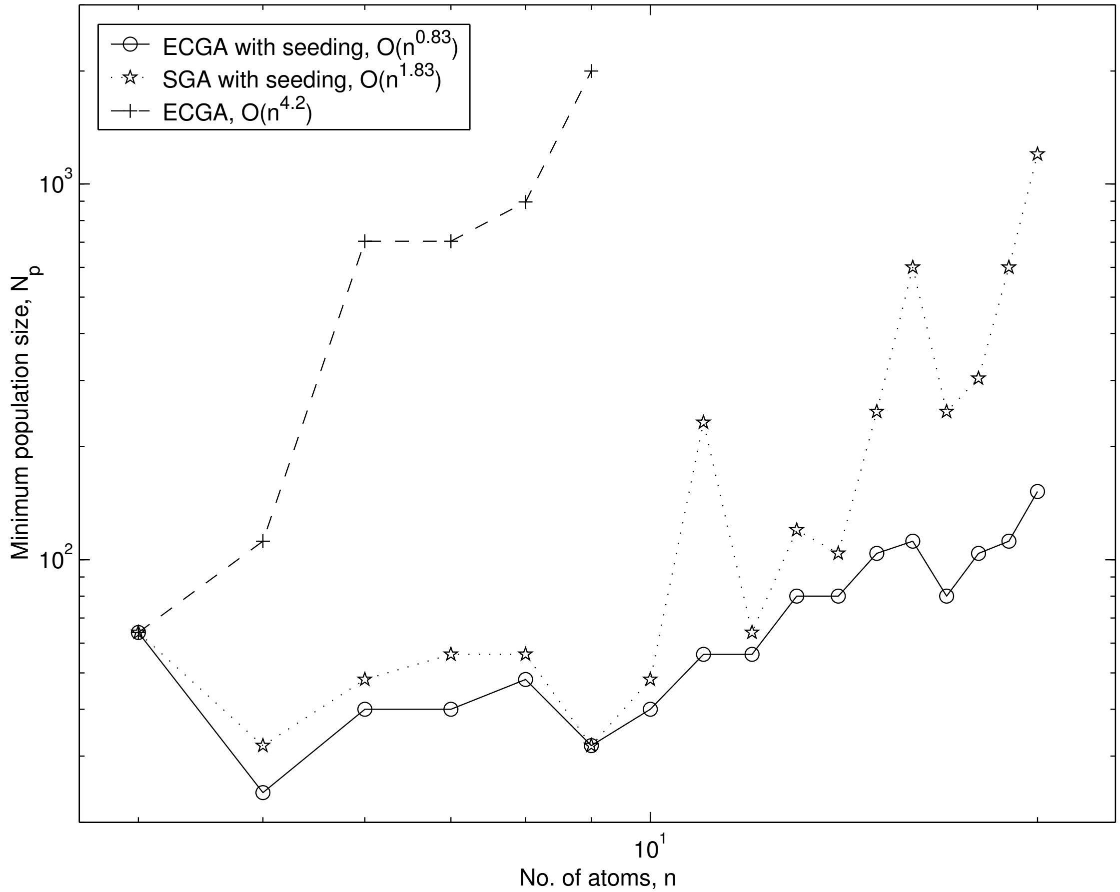
Algorithm Flowchart



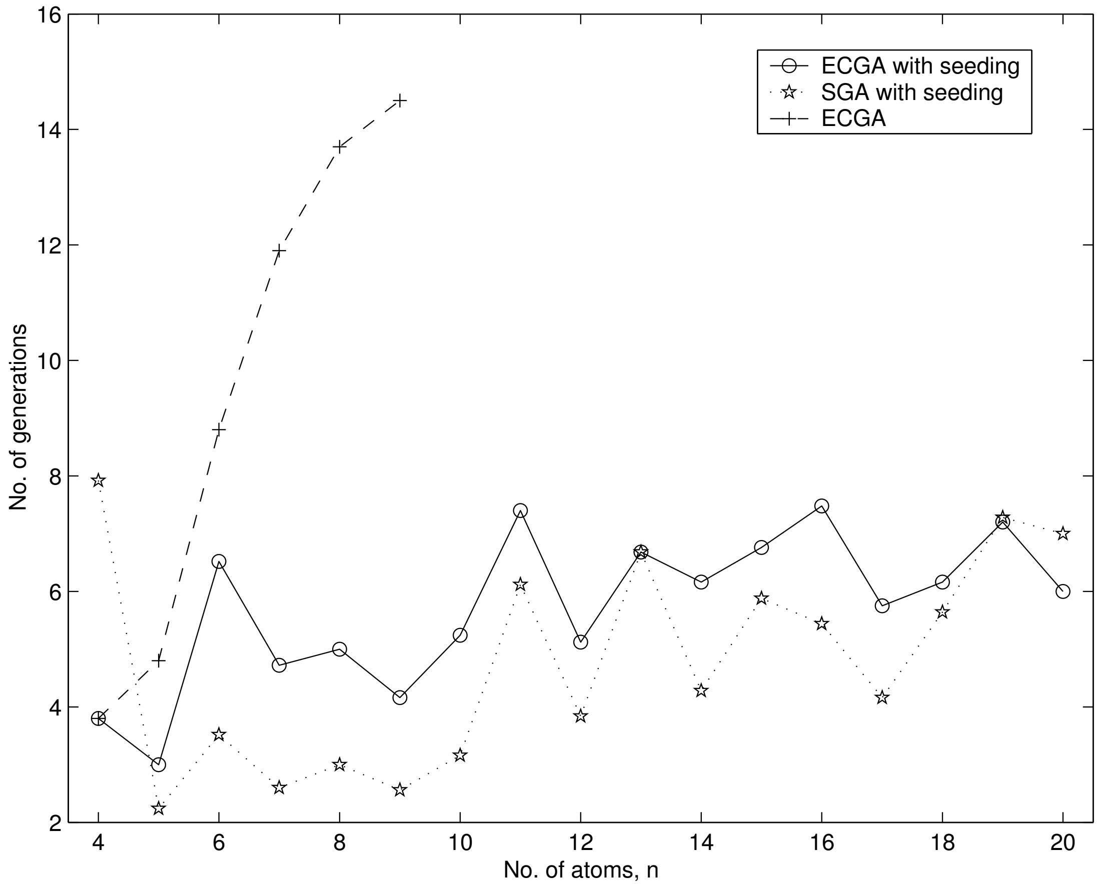
Results: Minimum Energy



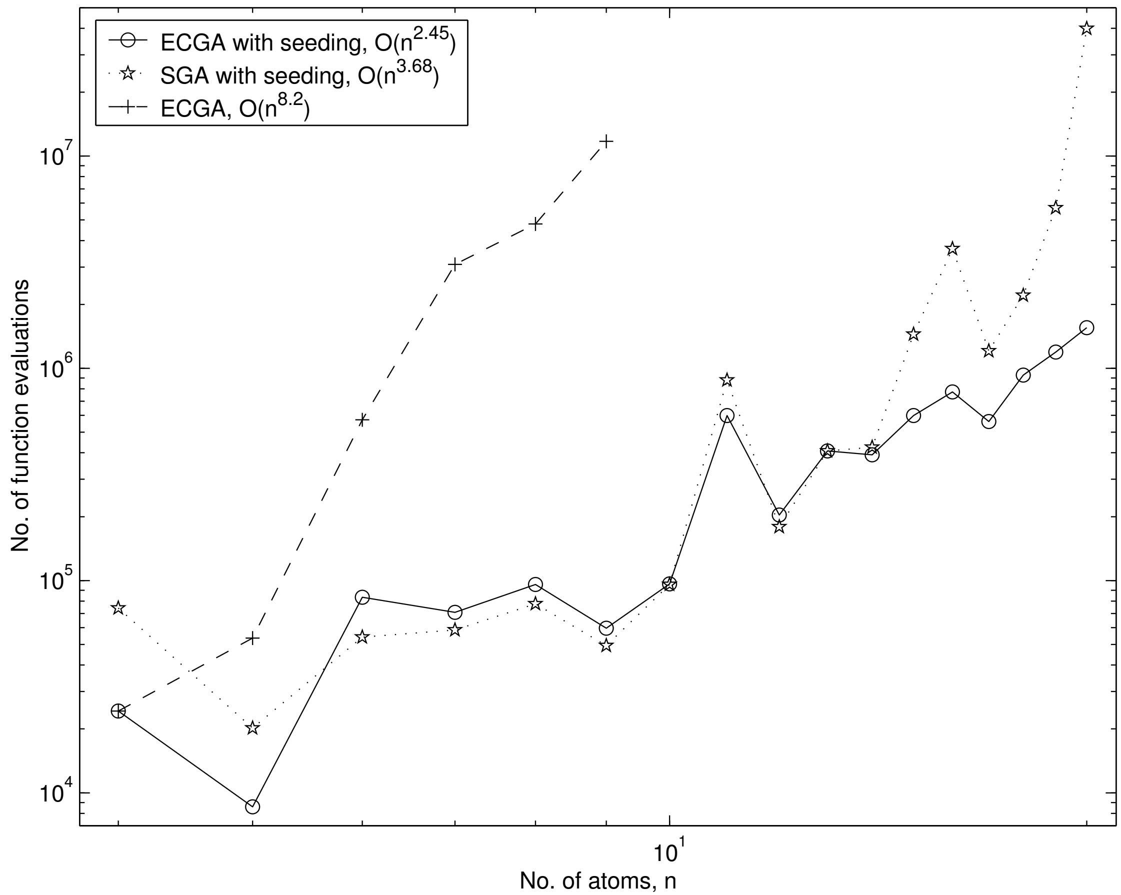
Results: Population Size



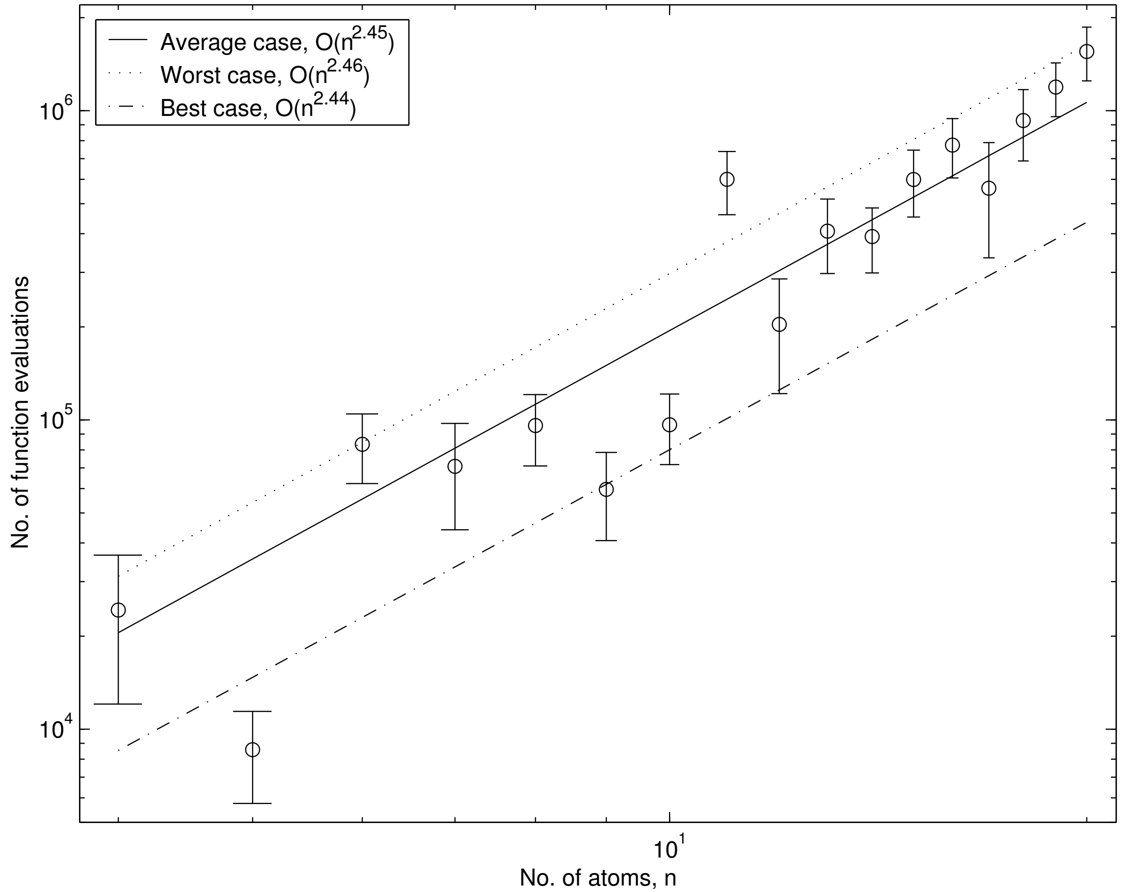
Results: Convergence Time



Results: Function Evaluations



Results: Scale Up



Summary

- An efficient hybrid cluster optimizer
  - Solves larger clusters (Up to 40 atoms)
  - High reliability: 96%
  - Minimum population size: $\mathrm{O}\left(n^{0.83}\right)$
  - Total No. of func. evals.: $\mathrm{O}\left(n^{2.45}\right)$
- Successfully predicts global optimum
- Iwamatsu (2000): 15 atoms
- Niesse & Mayne (1996): $\mathrm{O}\left(n^{3.3}\right)$

Acknowledgments

- David E. Goldberg & David Ceperley
- Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant F49620-00-1-0163.
- National Science Foundation under grant DMI-9908252.
- U. S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-0003.

Building MPM using MDL

Uses a steepest ascent search:

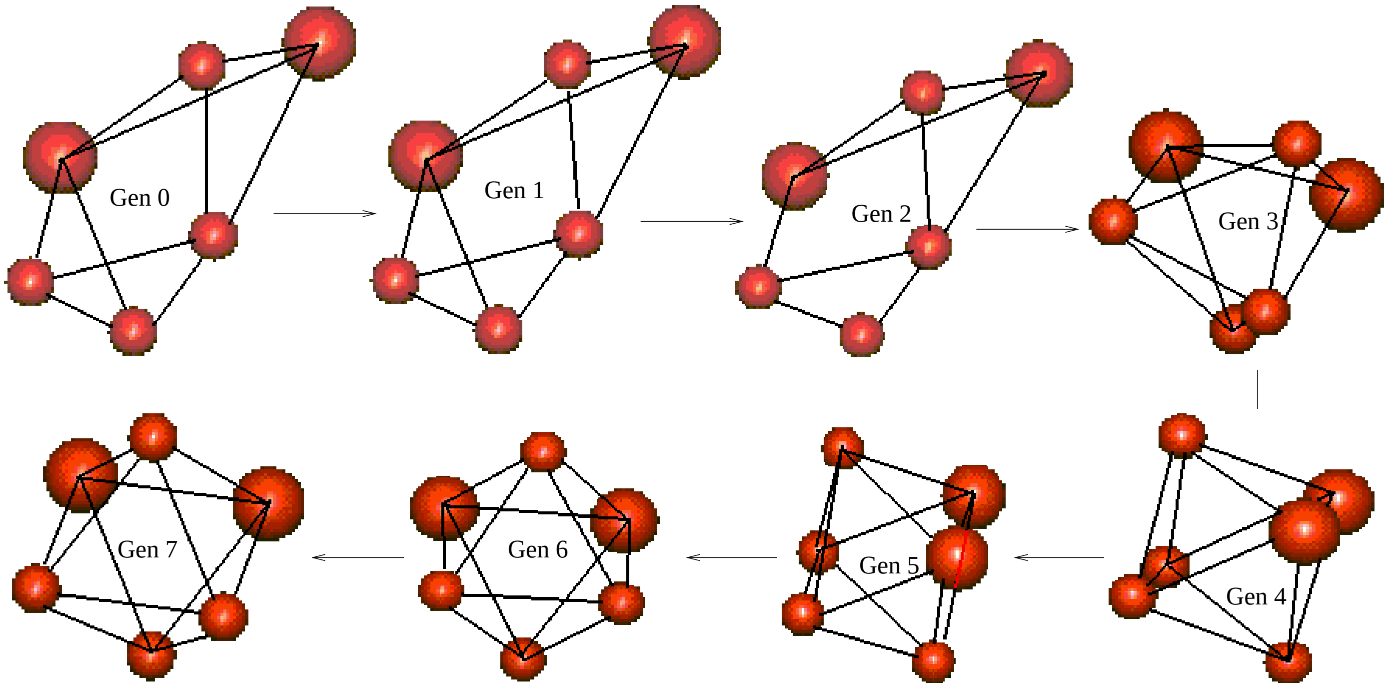
1. Compute C_c for independent genes ([1],[2],$\cdots$,[L])
2. Form all possible combinations $(m(m-1)/2)$ of merging two subsets. eg., ([1,2],[3],$\cdots$,[L]), $\cdots$, ([1],[2],[3],$\cdots$,[L-1,L]).
3. Select set with minimum combined complexity (C_c').
4. If $C_c > C_c'$ go to step 6.
5. MPM is the set with C_c'. Go to step 2.
6. Merging is not possible, exit.

Gong Potential Equations

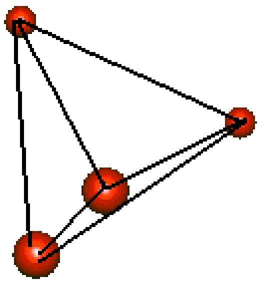
$$
\begin{aligned}
U_{\text{tot}} &= \sum_{i<j}^{n} v_2(i,j) + \sum_{i<j<k}^{n} v_3(i,j,k) \\
v_2(i,j) &= A\left(B r_{ij}^{-p} - r_{ij}^{-q}\right) \exp\left[\left(r_{ij}-a\right)^{-1}\right], \quad |r_{ij}| < a \\
v_3(i,j,k) &= h\left(r_{ji}, r_{ki}\right) + h\left(r_{kj}, r_{ij}\right) + h\left(r_{ik}, r_{jk}\right) \\
h\left(r_{ji}, r_{ki}\right) &= \begin{array}{ll} \lambda \exp\left[\gamma\left(\left(r_{ij}-a\right)^{-1} + \left(r_{ki}-a\right)^{-1}\right)\right] & |r_{ij}| < a \\ \left(\cos\theta_{jik} + \frac{1}{3}\right)^2 \left[\left(\cos\theta_{jik} + c_0\right)^2 + c_1\right], & |r_{ki}| < a \end{array}
\end{aligned}
$$

- A = 7.0496, B = 0.6022, a = 1.8, p = 4, q = 0.
- λ = 25, γ = 1.2, c_0 = -0.5, c_1 = 0.45

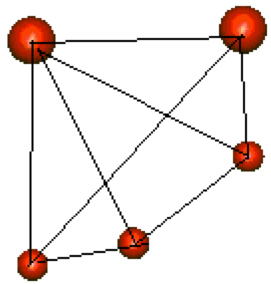
Results: Single GA Run



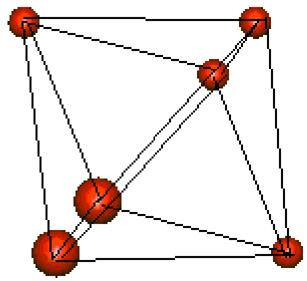
Results: Optimal Structures



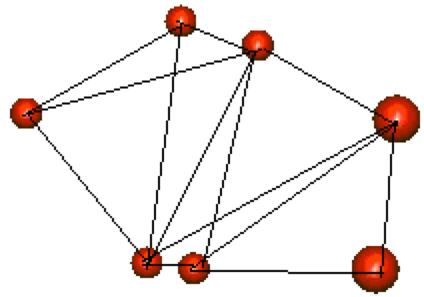
Tetrahedron
U = -4.0016



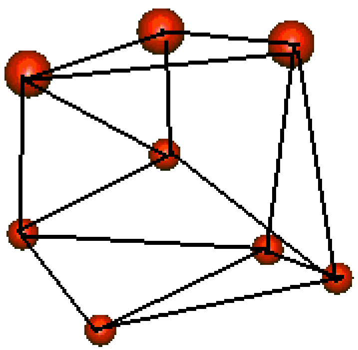
Compressed trigonal bipyramid, U = -5.7518



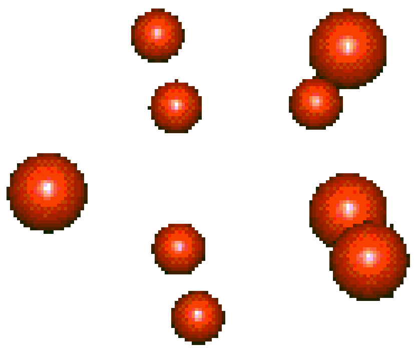
Octahedron
U = -7.6696



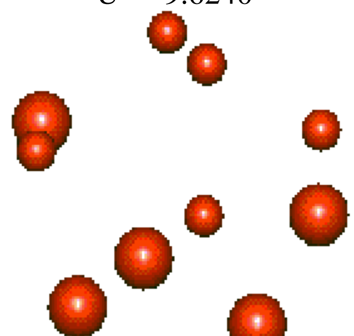
Pentagonal bipyramid
U = -9.6240



Unicapped distorted pentagonal bipyramid
U = -11.5346



Tricapped trigonal prism, U = -13.5999



Bicapped tetragonal antiprism, U = - 15.6487