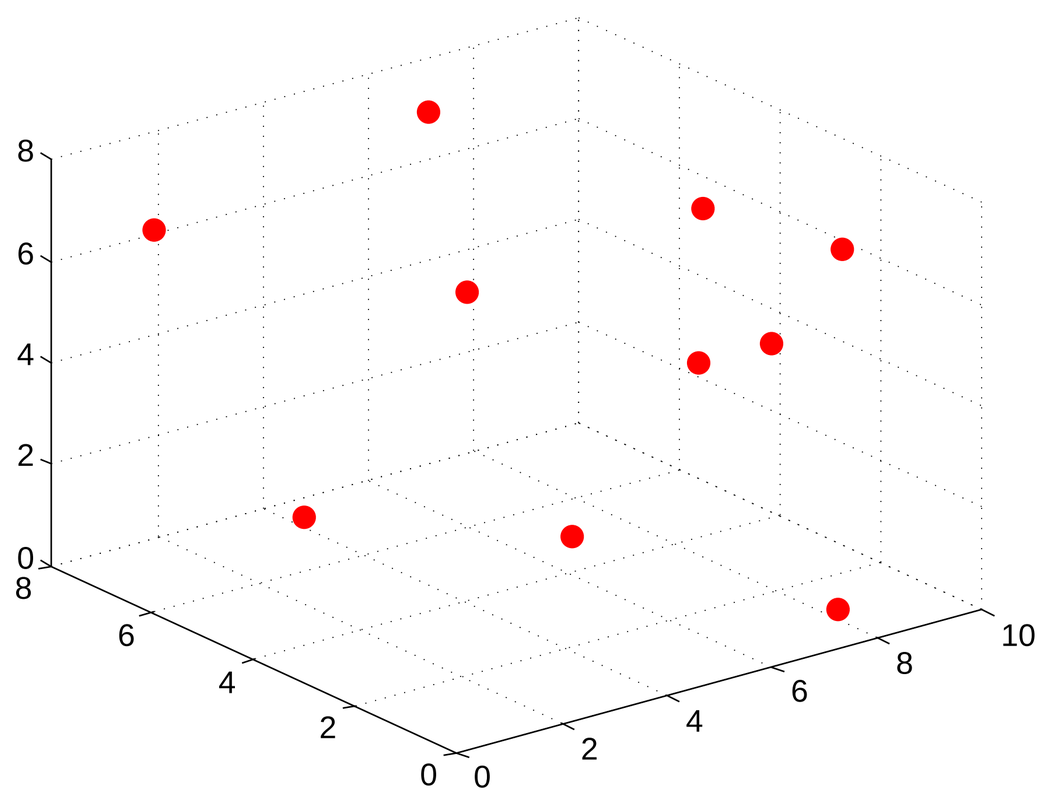
Parallel Matlab: `parfor`

Molecular Dynamics Example

VirginiaTech Invent the Future John V. Burkardt & Eugene M. Cliff ICAM

Configuration

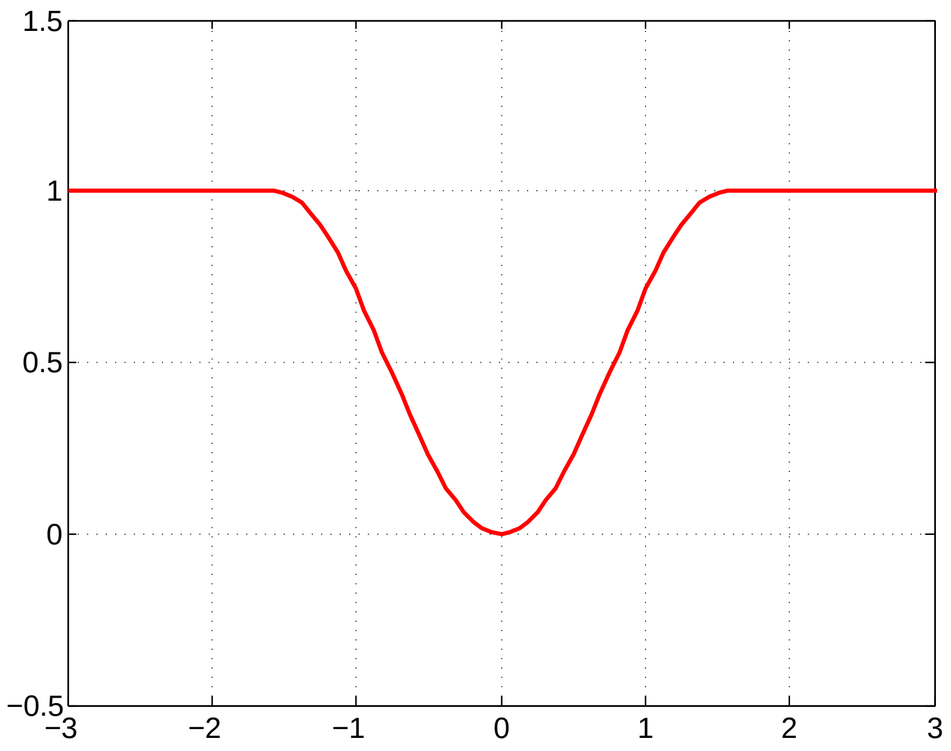


A set of NP particles move in 3D space, subject to an interparticle Lennart-Jones force.

Model Problem

Particle equations of motion

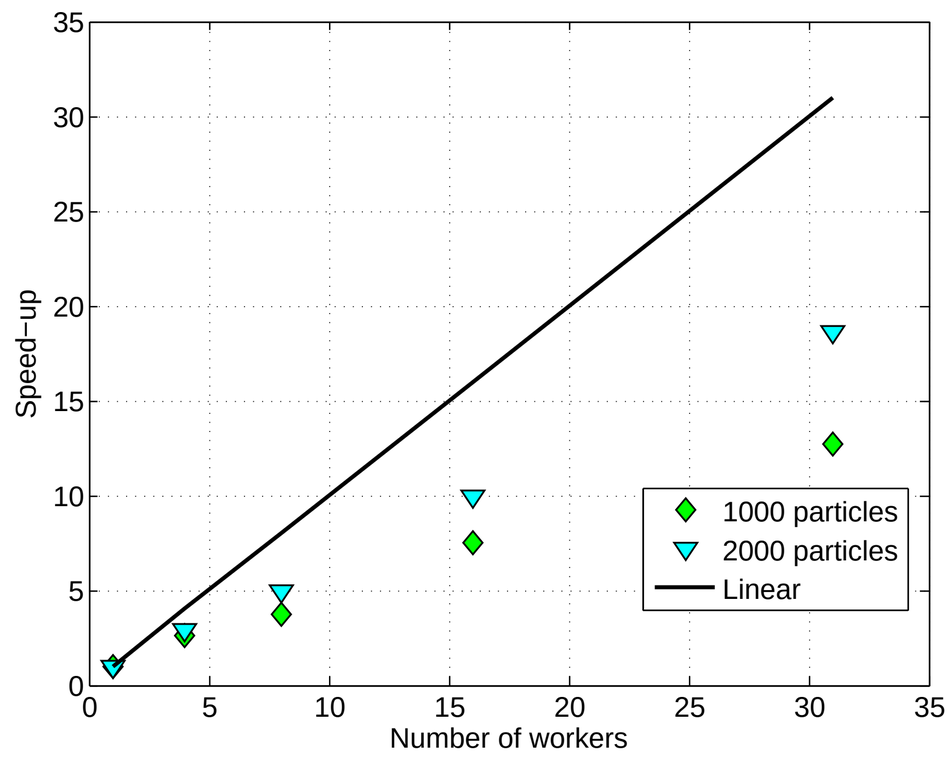
$$m\,\ddot{\vec{r}}_i(t) = \vec{f}_i = \sum_{j \neq i} \vec{f}_{j \to i}\,, = \sum_{j \neq i} \nabla V(\vec{r}_j - \vec{r}_i)\ .$$



Code Fragment

```
parfor i = 1 : np
  Ri = pos-repmat(pos(:,i),1,np);
  D = sqrt(diag(Ri'*Ri));
  Ri = Ri(:,(D>0.0));
  D = D(D>0.0);
  D2 = D.*(D<=pi2)+pi2*(D>pi2);
  pot = pot+0.5*sum(sin(D2).^2);
  f(:,i) = Ri*(sin(2*D2)./D);
end
```

Timing



Discussion

For a relatively small problem using NP=1000 particles, the program's performance improved across the range of processors; however, the speedup was only about half the ideal improvement. When the problem size was increased to 2000 particles, the program got a much greater benefit from parallelism. This suggests that, at least over this range of processors, parallel MATLAB will be increasingly efficient as the problem size increases.

Reference

http://people.sc.fsu.edu/~burkardt/...

- Burkardt, Cliff, Snow, *MATLAB Parallel Programming: Timing Results on an Intel Nehalem Cluster*, ...pdf/nehalem_matlab.pdf.
- ...m_src/md_parallel/md_parallel.html.
- .../m_src/satisfiability_parallel /satisfiability_parallel.html.