Vectorization of the Generalized Born Method in AMBER

C. P. Sosa and T. Hewitt

Cray, Inc.

and

David Case

Scripps Institute

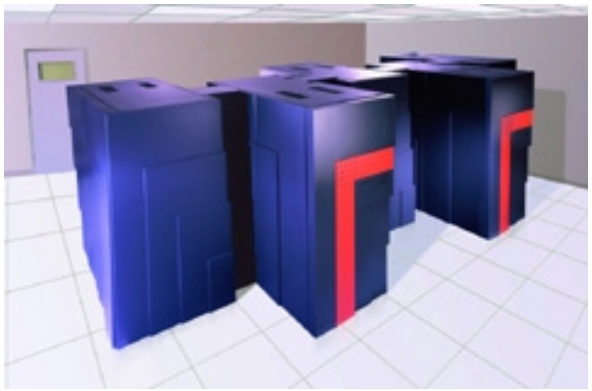
Agenda

- *Motivation*
- *The Generalized Born Method*
- *Generalized Born Routine Vectorization*
- *CRAY SV1 Architectural Features*
- *Single CPU Performance*
- *Parallel Performance*
- *Summary*
- *Future Work*

CRAY SV1 – Target Applications

- Gaussian
- AMBER
- CHARMM
- CASTEP
- AMPAC
- Gamess-US
- Gamess-UK
- Dgauss
- MND097
- CADPAC
- SHELXL

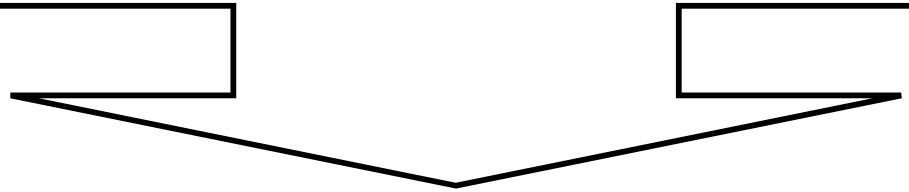
CRAY T3E – Target Applications



- Gaussian
- AMBER
- CHARMM
- CASTEP
- Gamess-US
- Gamess-UK
- DGauss

Motivation

- In biological simulations is important to have accurate representation of the system being studied
- Most chemical processes take place in a solvent



Realistic simulations require some sort of representation of the solvent or solvent environment

Solvent Effects

In simulations with waters of solvation, the water component tends to be the most CPU intensive part of the calculation

- Systems where solvent has high-degree of interaction with solute => use solvent explicitly

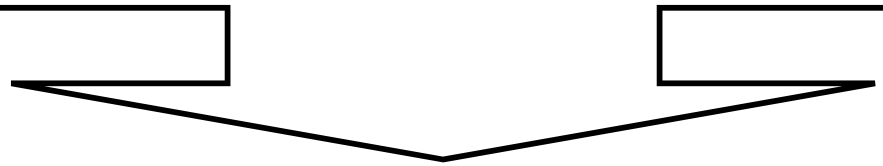
- Systems where solvent does not interact with the solute but it provides an environment that affects the behavior of the solute => use solvent environment

Simulations Limitations

Simulations of more realistic systems in presence of solvent is limited by using explicit waters of solvation

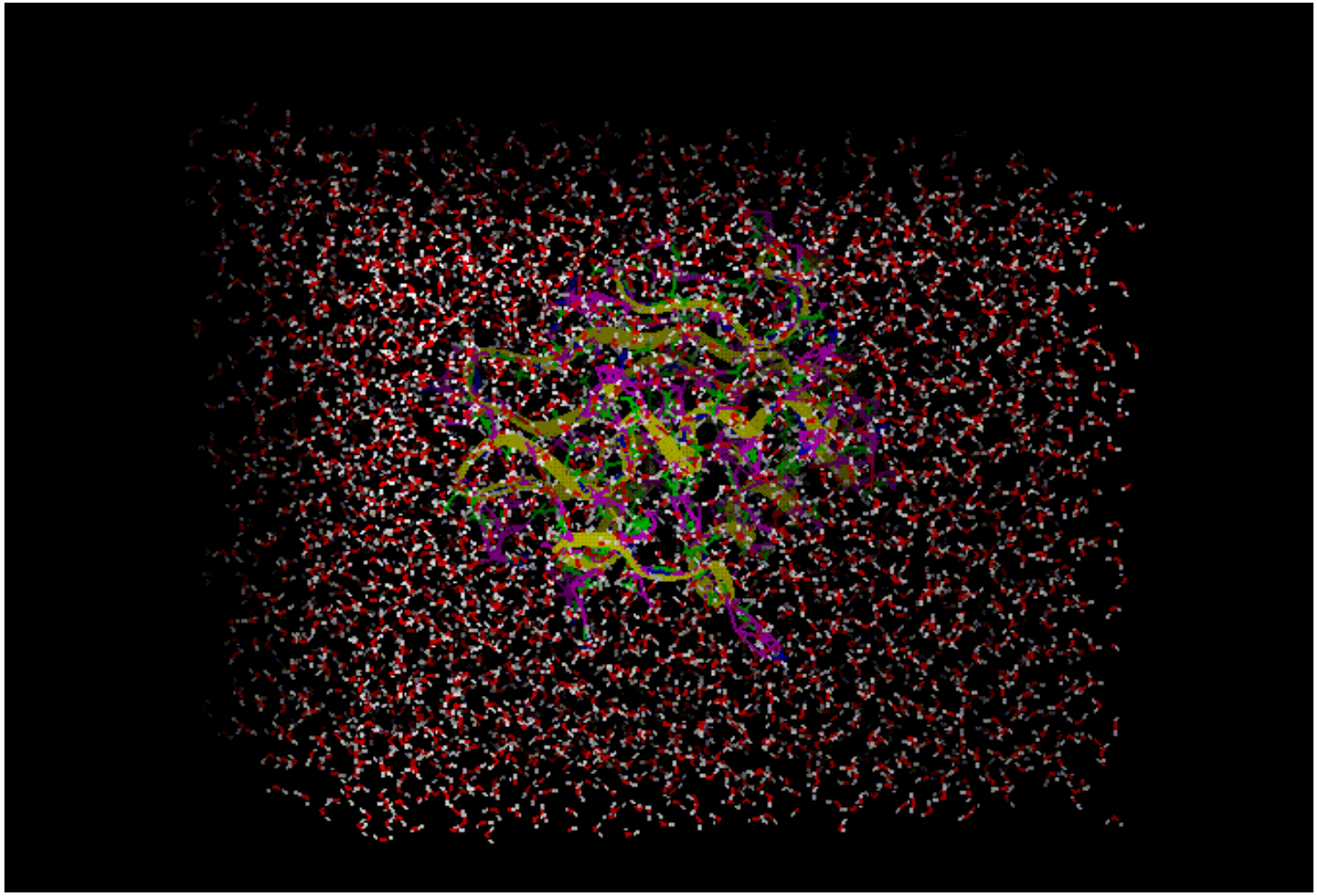
Continuum Models

The computation time can be reduced by representing solvent effects by a modified set of interactions between atoms that mimic relevant features of a particular solvent



Continuum models represent the solvent effects in a uniform medium having the average properties of the real solvent and surrounding the solute near the surface of the solute

Discrete Hydration

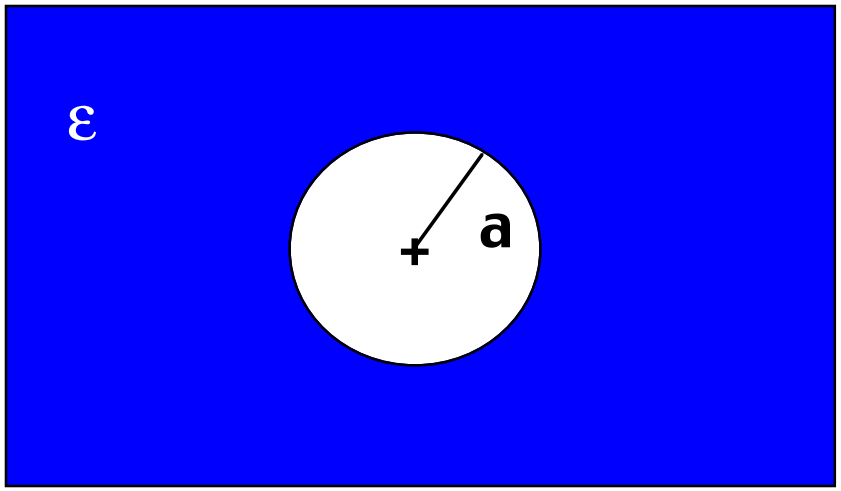


Solvation Free Energy

$$\Delta G_{sol} = \Delta G_{elec} + \Delta G_{vdw} + \Delta G_{cav}$$

"The solution free energy (ΔG_{sol}) is the free energy change to transfer a molecule from vacuum to solvent"

Electrostatic Contribution to the Free Energy of Solvation



Born Model: $\Delta G_{elec} = -q^2/2a(1 - 1/\varepsilon)$

Generalized Born (GB) Equation

$$G_{elec} = (1 - 1/\varepsilon)\Sigma\Sigma q_i q_j / r_{ij} - 0.5(1 - 1/\varepsilon)\Sigma q_i^2 / a_i$$

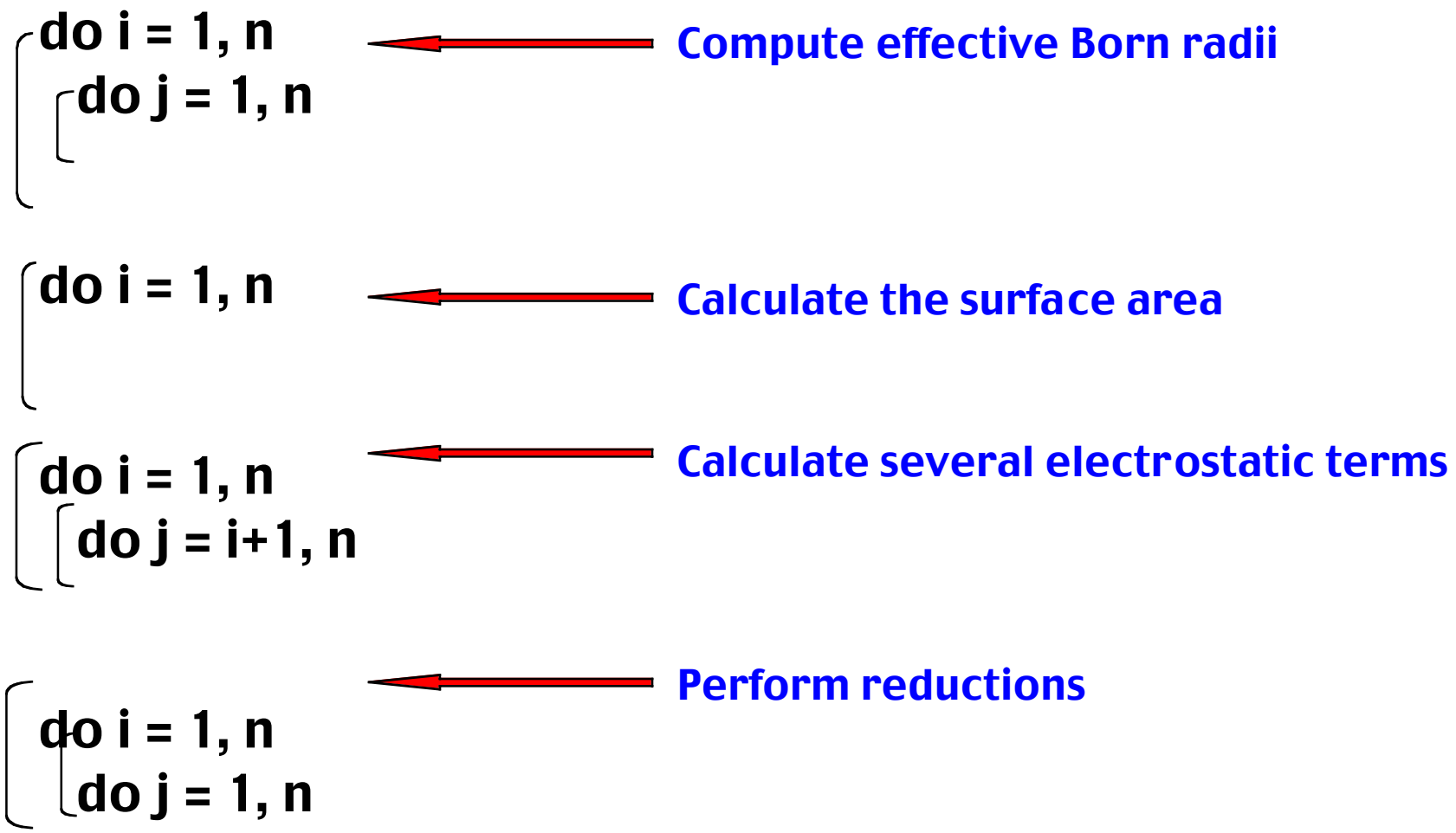
q_i: charges
ε: dielectric constant
a_i: born radii
r_{ij}: interparticle distance

Non-electrostatic Contributions to the Solvation Energy

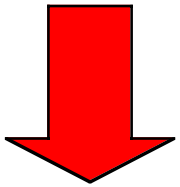
$$G_{cav} + G_{vdw} = \Sigma \sigma_i SA_i$$

$SA_i(A^2)$: total solvent-accessible area
$\sigma_i(Kcal/MolA_2)$: empirical solvation parameter

Generalized Born Routine Optimization



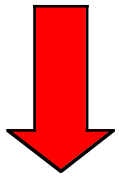
Original Loop 200



Several recurrences within the loop

```
485.  1-----<      do i=1,maxi
486.  1            xi = x(3*i-2)
487.  1            yi = x(3*i-1)
488.  1            zi = x(3*i  )
489.  1            qi = charge(i)
490.  1            if( igb.eq.1 ) then
491.  1              qid = dielfac * qi
492.  1              ri = reff(i)
493.  1            endif
494.  1            iaci = ntypes * (iac(i) - 1)
495.  1            jexcl = iexcl
496.  1            jexcl_last = iexcl + numex(i) -1
497.  1            nexcl = natex( jexcl )
498.  1            if( jexcl .gt. jexcl_last ) nexcl = 0
499.  1            dumx = 0.0d0
500.  1            dumy = 0.0d0
501.  1            dumz = 0.0d0
502.  1      c
503.  1      cdir$ ivdep
504.  1 2---<       do 200 j=i+1,natom
505.  1 2    c
506.  1 2    c       -- check the exclusion list for eel and vdw:
507.  1 2    c
508.  1 2            skip = .false.
509.  1 2            if( j .eq. nexcl ) then
510.  1 2              skip = .true.
511.  1 2              jexcl = jexcl + 1
512.  1 2              nexcl = natex( jexcl )
513.  1 2              if( jexcl .gt. jexcl_last ) nexcl = 0
514.  1 2            endif
515.  1 2    c
516.  1 2            de = 0.0
517.  1 2    c
518.  1 2            xij = xi - x(3*j-2)
```

Loop 200 – Step 1



Remove recurrence from nexcl

```
129.  1-----<      do i=1,maxi
130.  1            xi = x(3*i-2)
131.  1            yi = x(3*i-1)
132.  1            zi = x(3*i  )
133.  1            qi = charge(i)
134.  1            iaci = ntypes * (iac(i) - 1)
135.  1            jexcl = iexcl
136.  1            jexcl_last = iexcl + numex(i) -1
137.  1            nexcl = natex( jexcl )
138.  1            if( jexcl .gt. jexcl_last ) nexcl = 0
139.  1            dumx = 0.0d0
140.  1            dumy = 0.0d0
141.  1            dumz = 0.0d0
142.  1      c
143.  1            skipv(i+1:natom)=.false.
144.  1 Vr--<        do jj=jexcl,jexcl_last
145.  1 Vr           skipv(natex(jj))=.true.
146.  1 Vr-->        enddo
147.  1            nkeep=0
148.  1 2---<        do j=i+1,natom
149.  1 2            xij = xi - x(3*j-2)
150.  1 2            if( abs(xij) .gt. cutxyz ) go to 2001
151.  1 2            yij = yi - x(3*j-1)
152.  1 2            if( abs(yij) .gt. cutxyz ) go to 2001
153.  1 2            zij = zi - x(3*j  )
154.  1 2            if( abs(zij) .gt. cutxyz ) go to 2001
155.  1 2     c
156.  1 2            r2 = xij*xij + yij*yij + zij*zij
157.  1 2            if( r2.gt.cut ) go to 2001
158.  1 2            nkeep=nkeep+1;keepj(nkeep)=j
159.  1 2    2001     continue
160.  1 2--->        enddo
161.  1           if(igb.eq.1)then
162.  1      cdir$ ivdep
163.  1 V---<      do 200 jj=1,nkeep
164.  1 V          j=keepj(jj)
```

CRAY SV1 Architectural Features

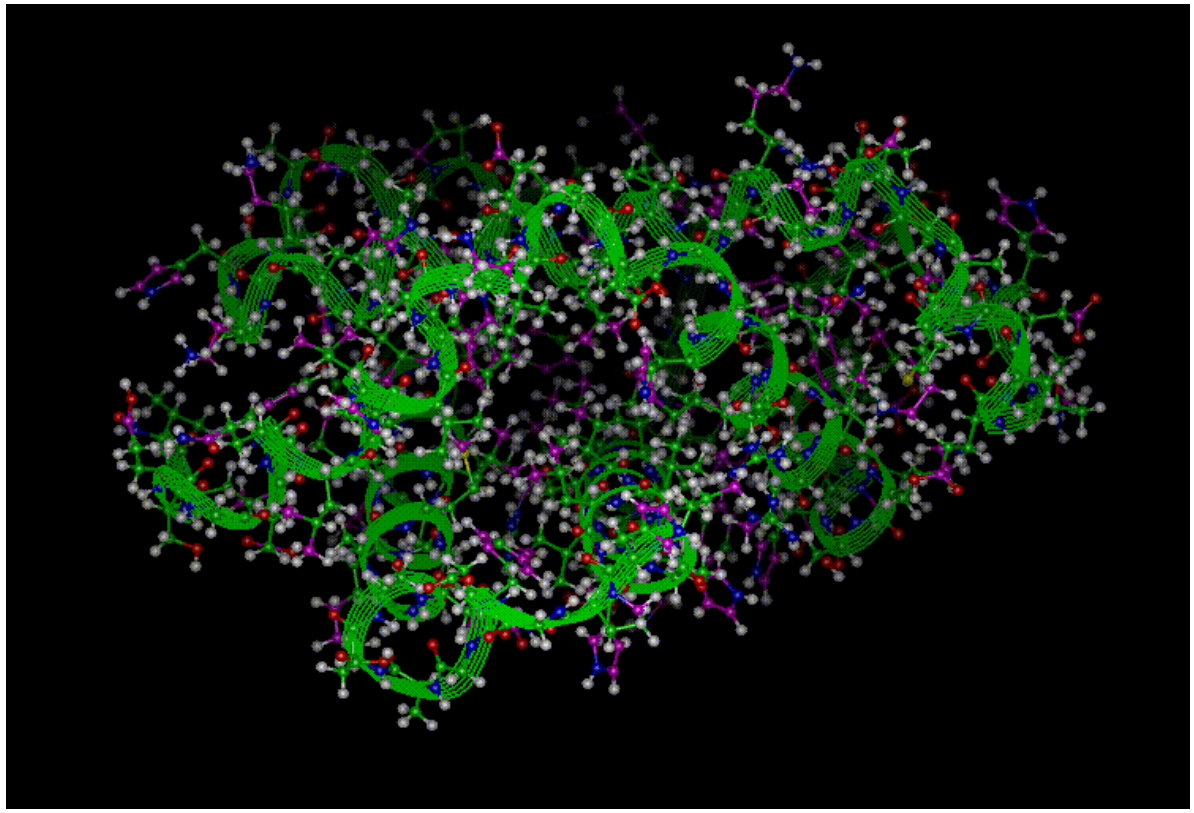
Processor:
300 MHz, 32 CPUs
Cache:
256 KB, 4-way set associative
Memory Size:
4GW
Mainframe:
SN3202, ICE

Single CPU Performance

gb_rna: RNA in a generalized Born Solvent, 10 MD Steps
640 atoms, 20 residues

	Scalar	Vector (Optimized)
Mflops	64	290
Time (sec.)	210	46

Myoglobin Single CPU Performance on a CRAY SV1

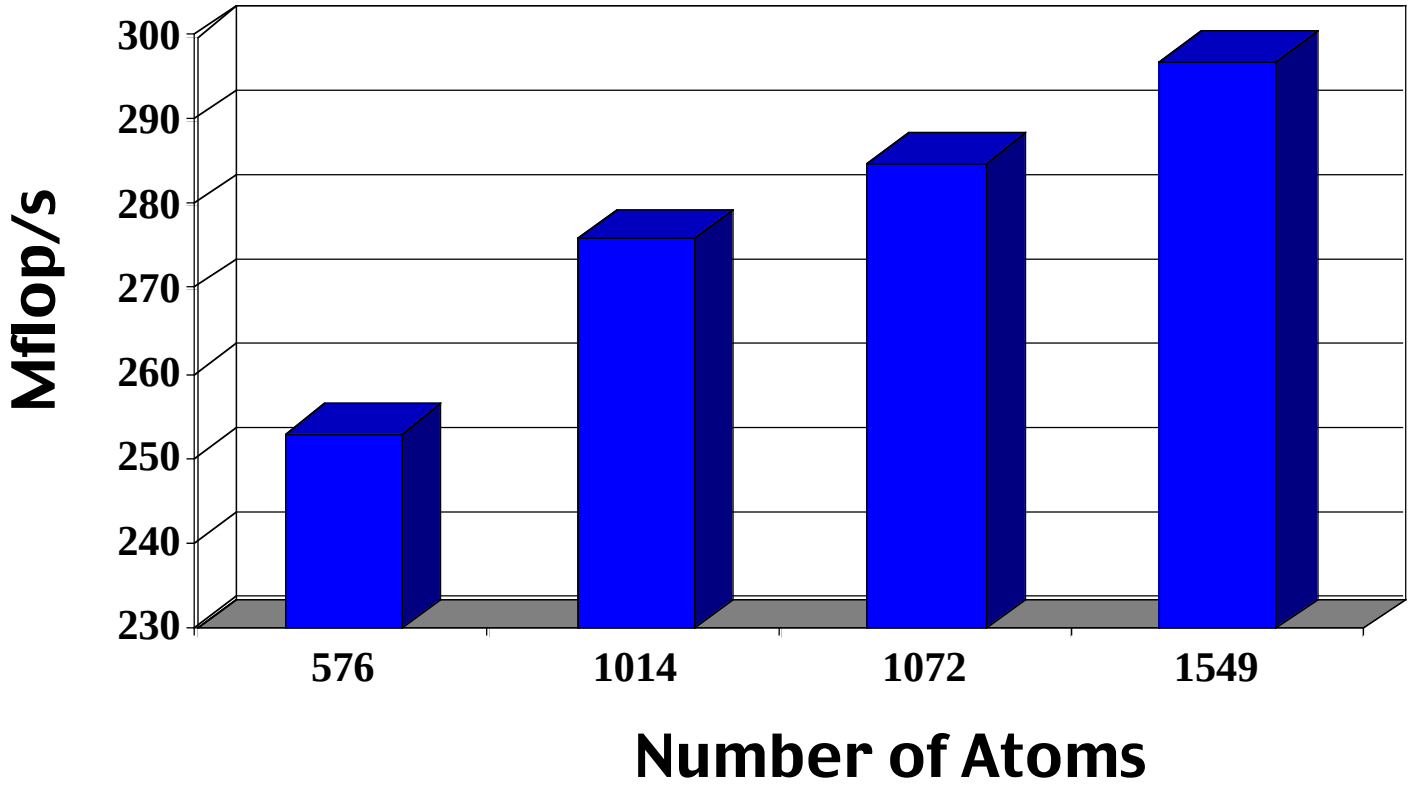


Myoglobin:
153 residues 2492 atoms
1000 MD Steps

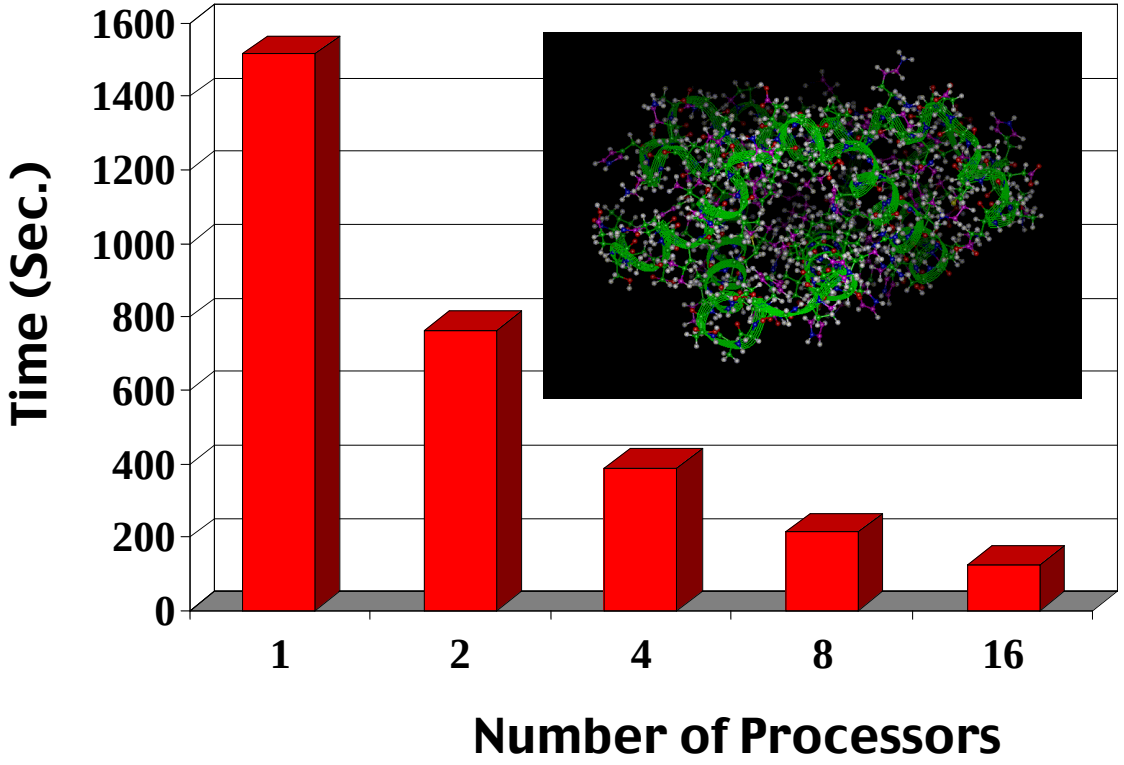
377 Mflop/s

"Mb is a relatively simple oxygen-binding protein found in almost all mammals, primarily in muscle tissue"

Mflops versus Number of Atoms for Four Different Proteins



Myoglobin Multiple CPU Performance on CRAY SV1



Summary

- Calculations based on the GB formalism spend more than 90% of the CPU in one routine (egb.f)
- All the loop indices are directly related to the number of atoms
- All the do-loops that compute all the different contributions within the GB formalism can be vectorized
- Mflops increases as the size of the system increases

Future Work

- Profile and optimized cases that use discrete or explicit solvent
- Quantify performance differences between continuum and discrete solvation models

Acknowledgements

Matthew R. Lee, University of California San Francisco
Dr. Michael Crowley, Scripps Institute
Jeff Dawson, Cray Inc