

## DL\_POLY on the Manchester Cray T3E

---

### Introduction

The following is an account of the performance of the DL\_POLY program (Version 2.11) on the **CSAR**

<http://www.csar.cfs.ac.uk> Cray T3e service at Manchester Computing Centre (June-July 1999). The evaluation is based on example simulations from the standard DL\_POLY benchmark suite, which is available from the CCP5 Program Library at Daresbury Laboratory:

Preliminary remarks

Benchmark 1: Metallic aluminium

Benchmark 2: A peptide in water

Benchmark 3: Transferrin in water

Benchmark 4: Sodium chloride

Benchmark 5: Sodium potassium disilicate glass

Benchmark 6: Potassium-valinomycin complex in water

Benchmark 7: Gramicidin in water

Benchmark 8: Magnesium oxide microcrystal

Benchmark 9: Model membrane/valinomycin system

Summary

Acknowledgements

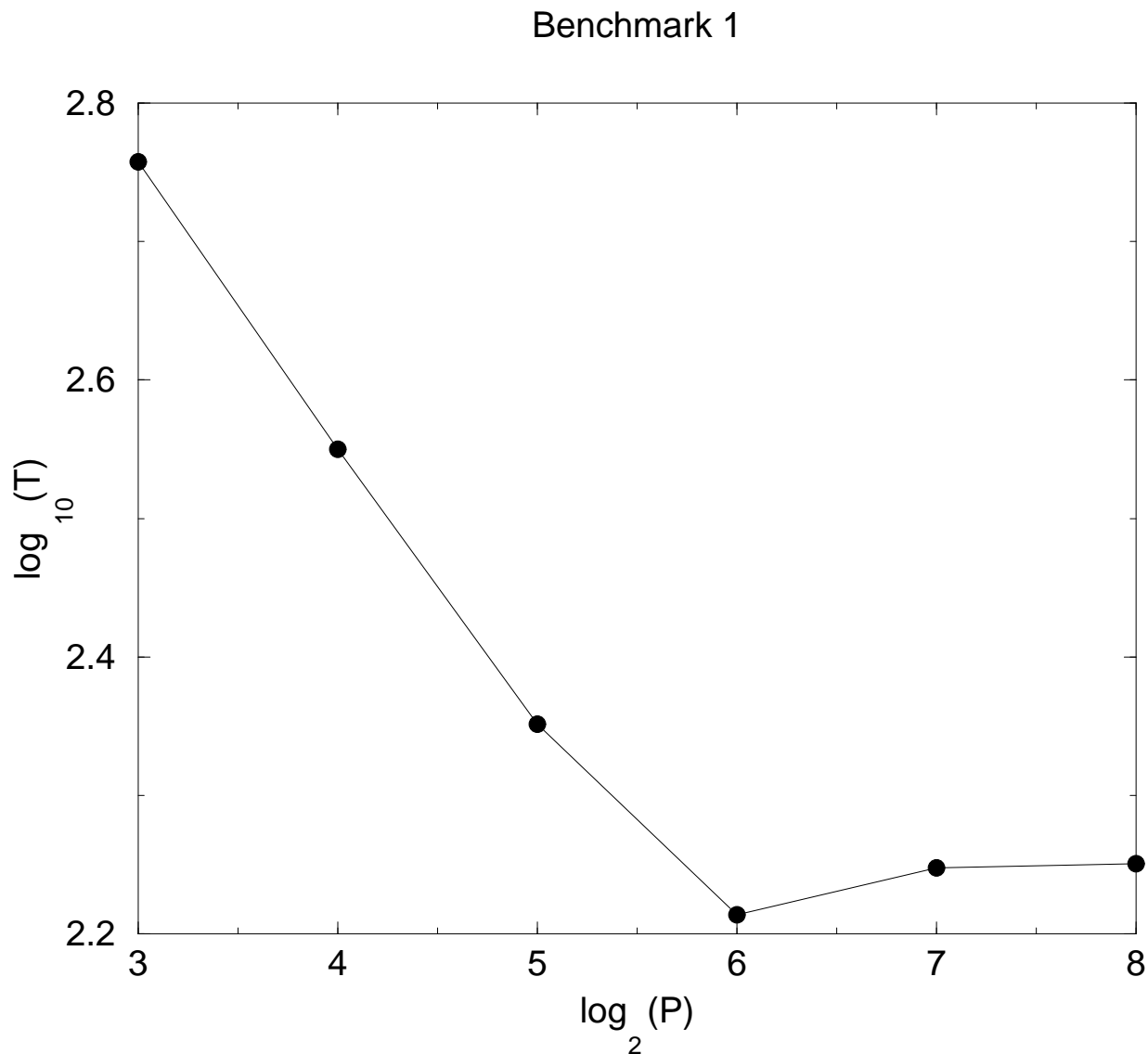
### Preliminary Remarks

The simulation program DL\_POLY is a distributed memory parallel program based on the Replicated Data (RD) strategy for parallelisation. It was designed initially for machines with up to 64 processors and systems of up to 30,000 atoms, but has since found use on much larger architectures, where memory-memory (i.e. low overhead) message passing is possible. Implicit in the RD approach is a dependence on fast global summations, which are not available on all machines. For this reason the performance may suffer markedly with increasing processor numbers. Also, the performance scaling (i.e. speed up with number of processors used) will vary according to the kind of simulation being undertaken - algorithms that require the most communication will scale less well than ones which require fewest. In practice systems possessing complex molecular topologies scale less well than ones requiring simple atomic descriptions, as they require a higher communication overhead.

The reported test give an honest indication of the capabilities of DL\_POLY in realistic applications. Each benchmark is described separately and the performance of the code on 8,16,32,64,128 and 256

processors of the T3E is given. The times quoted are wall-clock times (in sec.) to complete the described job. The plots shown are for the log (base 10) of the job times vs. log (base 2) of the number of processors.

### Benchmark 1: Metallic Aluminium



This system consists of 19,652 aluminium atoms on an FCC lattice at 300 K. The potential model is a Sutton-Chen many body potential with a cutoff at 8.6 Angstroms (A). No electrostatic forces are present in the system. The time step is 5 fs and the simulation is for 1000 time steps in the

NVE ensemble. The time quoted includes initial data input and writing restart files at the end.

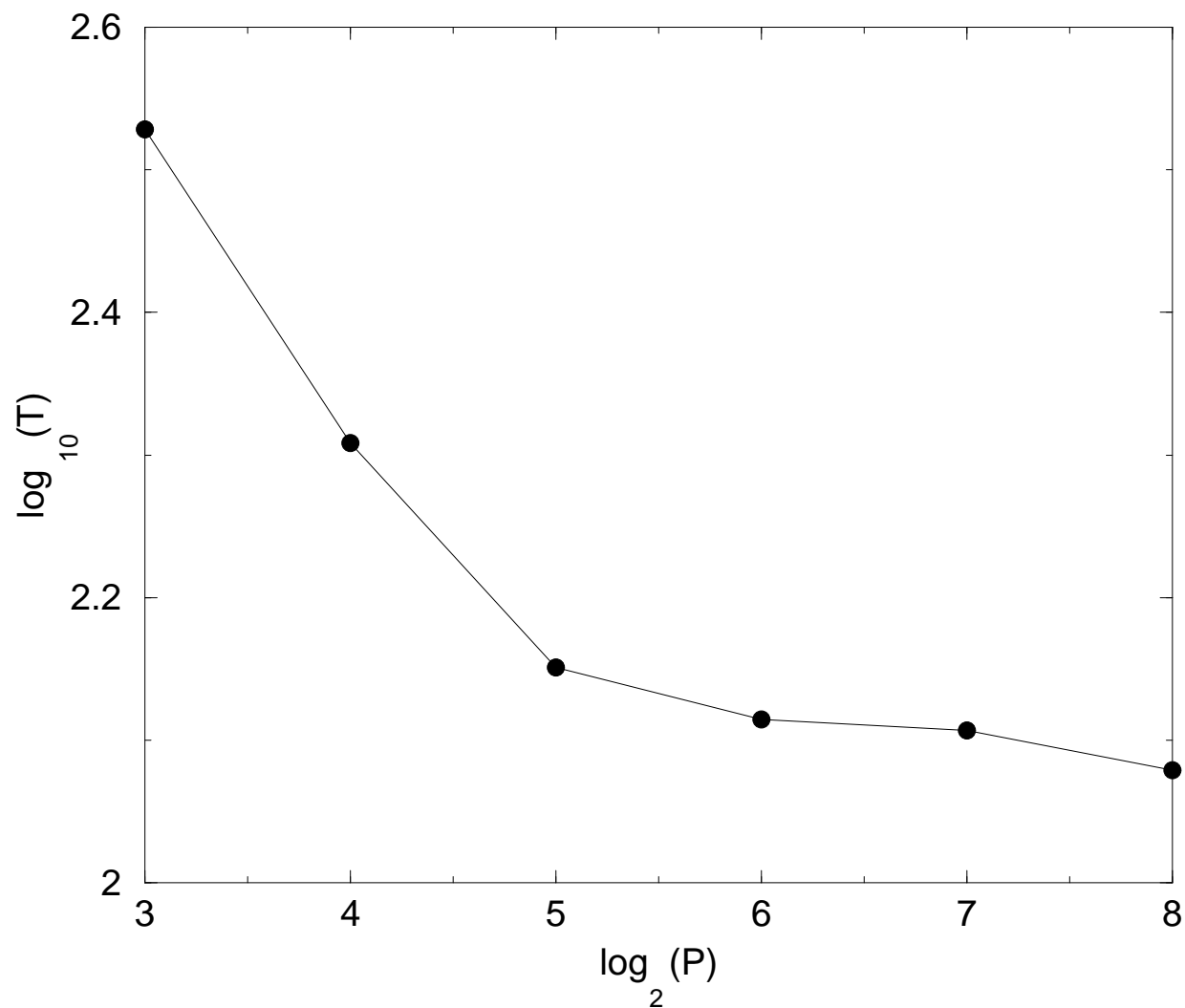
The performance scaling with processor number is very good up to 64 processors, where it achieves a maximum. There is an increase in simulation time thereafter. This is probably a reflection of the fact that the Sutton-Chen potential requires a density calculation in addition to the normal pair force terms, which demands an additional global sum during the forces calculations. Global sums generally are detrimental to performance scaling.

## **Benchmark 2: A Peptide in Water**

This simulation is of a peptide comprised of 15 amino acids in a solvent of 1247 TIP3P water molecules. The water is treated as a rigid body and the peptide bonds are handled using the SHAKE algorithm. The total number of atoms is 3,993. The electrostatics in this simulation are handled using a neutral group scheme with a reaction field. The potential cutoff, for both electrostatic and Van der Waals interactions, is set at 8 Å. The description of the peptide includes valence angle, and dihedral potentials. The simulation is for 2000 time steps with a time step of 1 fs in the NVT ensemble due to Berendsen. The time quoted includes all data input and output.

The performance plot in this case shows a gradual reduction in simulation time, without an obvious linear regime. Scaling at high processor numbers is poor and probably reflects the difficulty in apportioning the neutral group calculations across processors at this extreme. Similarly, the use of SHAKE for the bond constraints is likely to be another contributor to poor scaling, on account of its communication overheads. Gains in performance for smaller node numbers (up to 32) are much better. Nevertheless this is a rather small simulation and the result implies that better scaling is possible for larger systems.

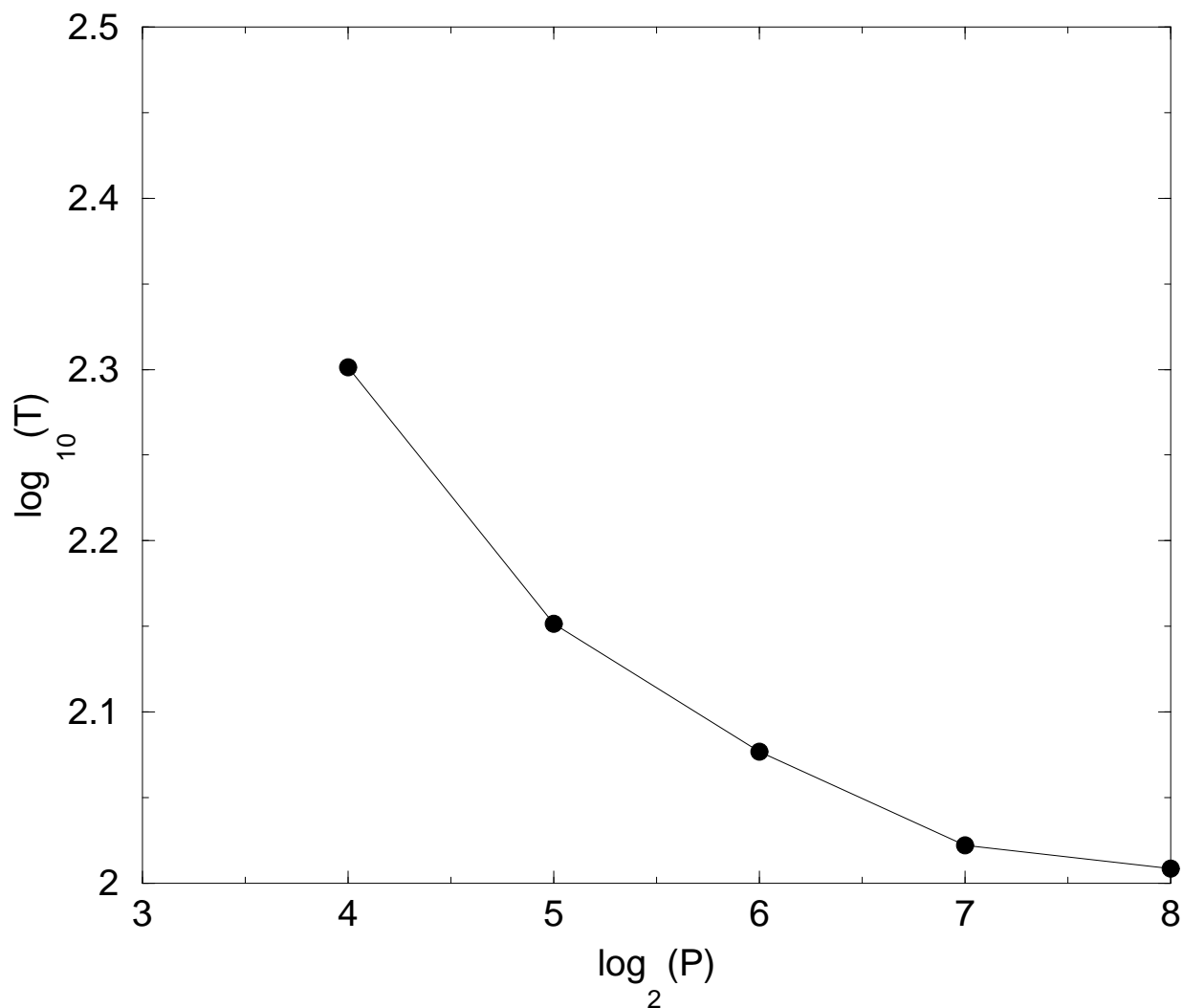
## Benchmark 2

**Benchmark 3: Transferrin in Water**

This simulation is of the enzyme transferrin in a solution comprised of 8102 TIP3P water molecules. A total of 27,593 atoms are in the system. The electrostatic forces are handled by a combination of neutral groups with the coulombic potential. All forces cutoffs are set at 8 Å. The simulation is for 250 steps with a time step of .1 fs, in the NVE ensemble. The water molecules are treated as rigid bodies and the transferrin is maintained by bond constraints using

SHAKE. Valence angles and dihedral potentials are present in the transferrin model.

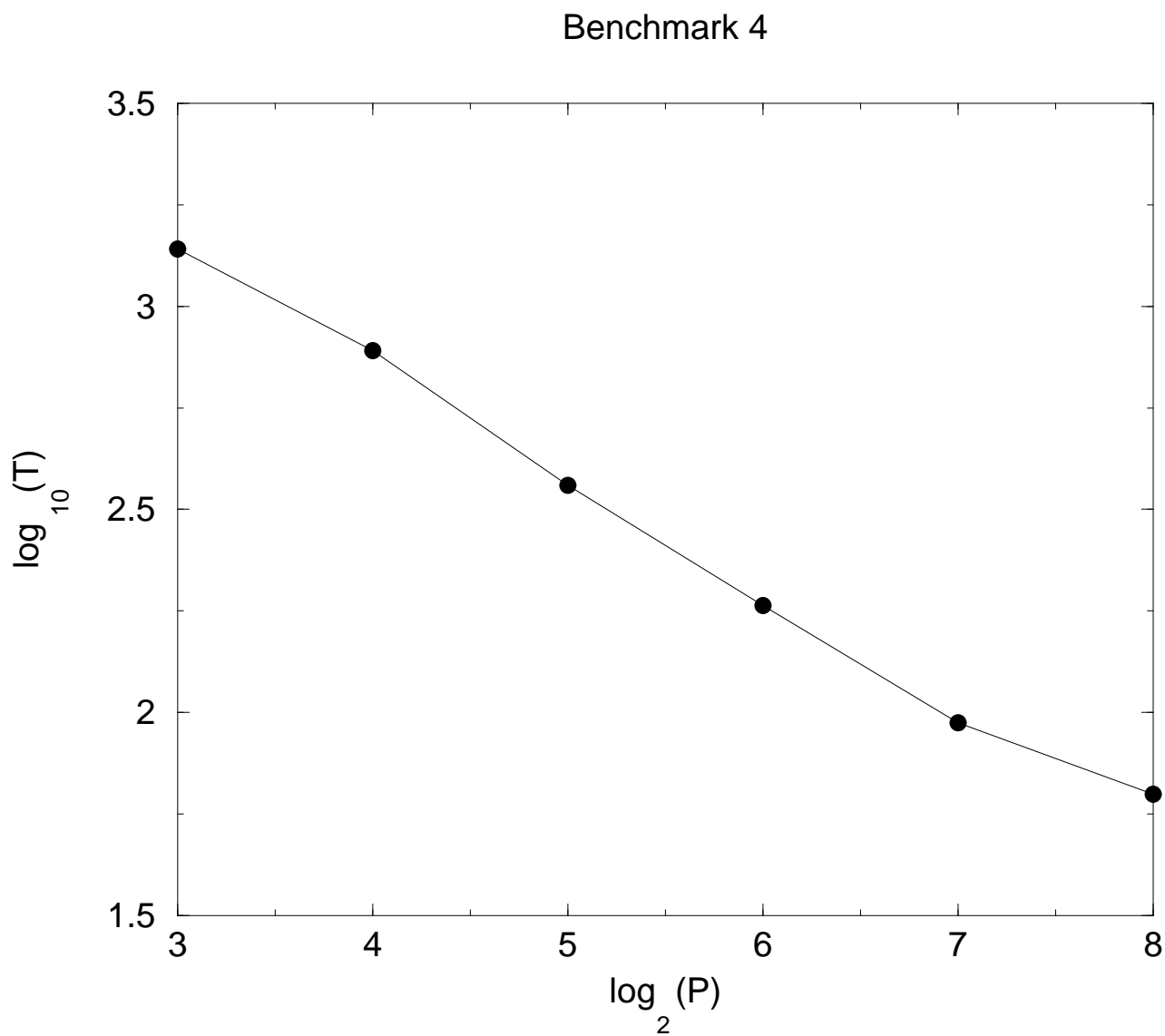
### Benchmark 3



The performance scaling resembles Benchmark 2 in that it shows no obvious linear regime, though the scaling at large processor numbers is significantly better, probably due to the better apportioning of the neutral groups in this larger system. Note that this simulation is too large to be run on less than 16 processors of the Manchester T3E.

## Benchmark 4: Sodium chloride

This represents a straightforward simulation of sodium chloride at 500K, using the standard Ewald summation method to handle the electrostatic forces. A multiple timestep algorithm is used to increase performance, which requires recalculating the reciprocal space forces only twice in every five time steps. The electrostatic cutoff is set at 24 Å in real space, with a primary cutoff of 12 Å for the multiple timestep algorithm. The Van der Waals terms are calculated with a cutoff of 12 Å. The simulation is for 200 steps with a time step of 1 fs in the Berendsen NVT ensemble. The system size is 27,000 ions. Timings include data input and output.

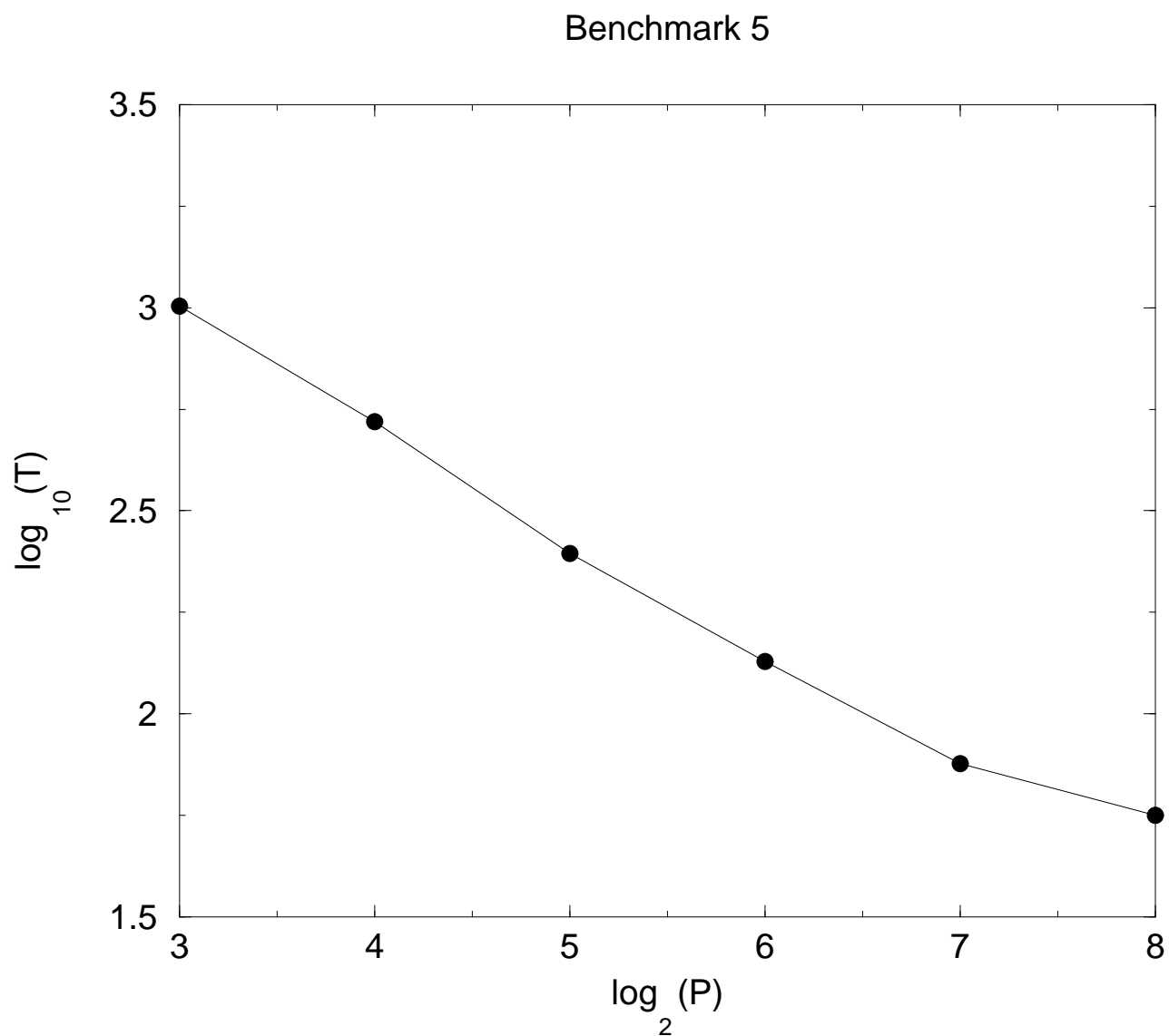


Performance scaling in this case is extremely good and is (almost) linear over the entire range of

processor numbers. This reflects the high parallel efficiency of the Ewald sum implementation.

### Benchmark 5: Sodium Potassium Disilicate Glass

This simulation is of 8,640 atoms of an alkali disilicate glass at 1000 K. The electrostatics are handled by the Ewald sum and the interaction potential includes a three-body valence angle term, which requires a link-cell scheme to locate atom triplets.



The electrostatic cutoff is 12 Å and the Van der Waals cutoff is 7.6 Å. Three body forces are cut

off at 3.45 A. The simulation is for 300 steps in the Hoover NVT ensemble, with a timestep of 1 fs. Timings include data input and output.

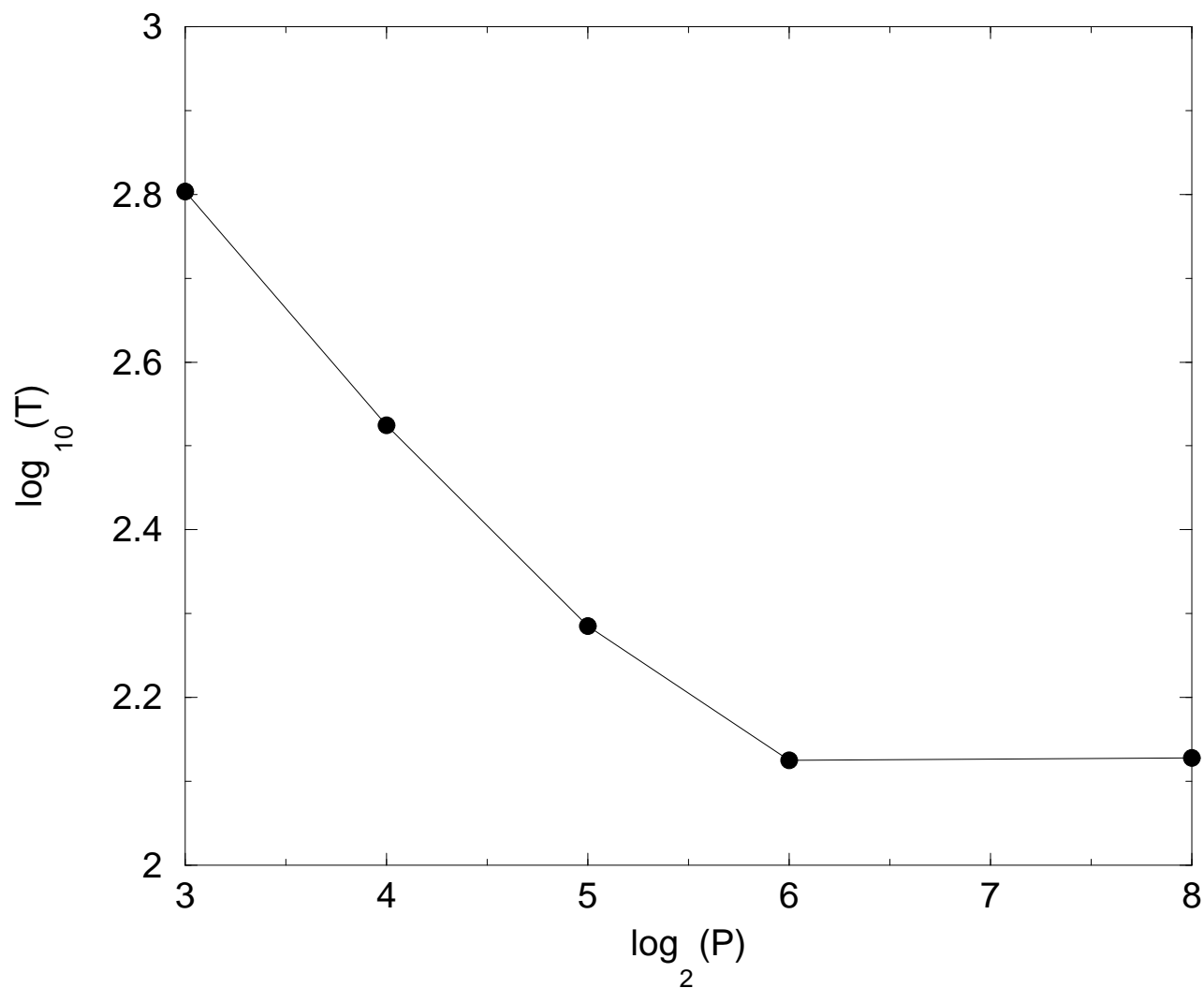
The performance scaling in this case resembles Benchmark 4, though being a smaller system, it shows a slight tendency to deviate from ideal behaviour as it approaches 256 processors. Nevertheless, performance overall is extremely good.

## **Benchmark 6: Potassium-Valinomycin Complex in Water**

Valinomycin is a naturally occurring cyclic molecule that forms a hexadentate complex with potassium. This simulation models the stability of the complex in water at 310 K. the simulation is for 500 steps with a timestep of 1 fs in the Hoover NVT ensemble. The valinomycin is modelled by a modified AMBER potential and structurally maintained by constraints with SHAKE. The water consists of 1223 SPC water molecules held rigid by bond constraints with SHAKE. The whole system is relatively small at 3838 atoms and is defined with truncated octahedral boundary conditions. The ewald sum is used to calculate the electrostatic interactions, with a real space cutoff of 16 A. A multiple timestep is used with two reciprocal space calculations every 4 time steps. The primary cutoff is 10 A. The Van der Waals interactions are truncated at 10 A. Valence angle and dihedral angle potentials are present in the valinomycin model. Timings include data input and output.

The performance scaling in this case is good up to 64 processors, but shows no improvement thereafter. The source of this difficulty lies in the use of SHAKE for the constraint bonds, which has a high communications overhead, particularly in instances where the program cannot assign complete molecules to processors and bond constraints interact across processors as is believed to be the case here. No result was obtained for 128 processors, as the program was unable to find a convenient apportioning of the constraints to each processor.

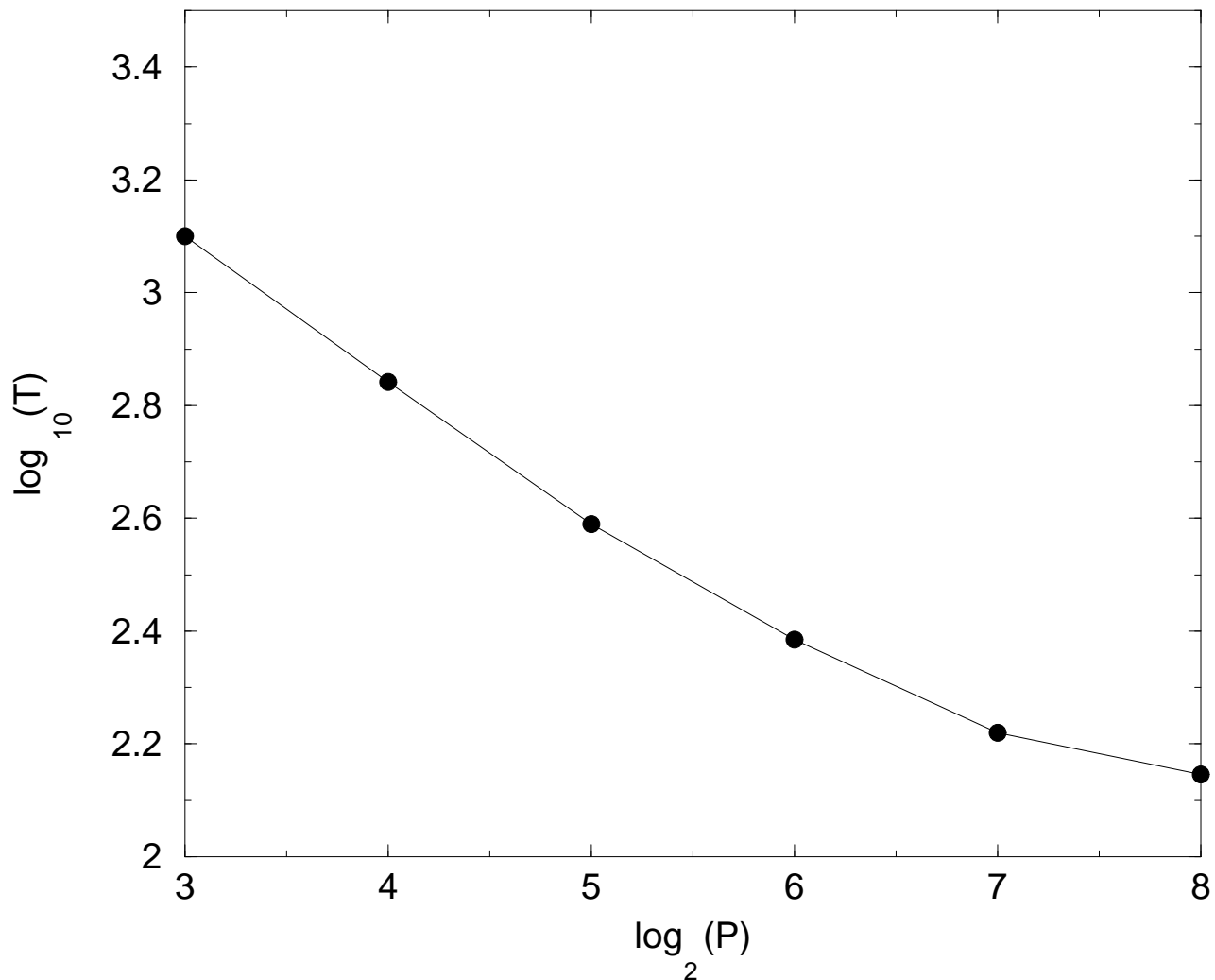
## Benchmark 6

**Benchmark 7: Gramicidin in Water**

This system is comprised of 13,390 atoms, including 4012 TIP3P water molecules solvating the gramicidin A protein molecule at 300K. Both the protein and water molecules are defined with rigid bonds and maintained by the SHAKE algorithm. The water is held completely rigid, while the protein has angular and dihedral potential terms. Electrostatic interactions are handled by the neutral group method with a coulombic potential truncated at 12 Å. The Van der Waals interactions are truncated at 8 Å. The simulation is for 500 time steps in the NVE ensemble with a 1 fs time step.

Timings include data input and output.

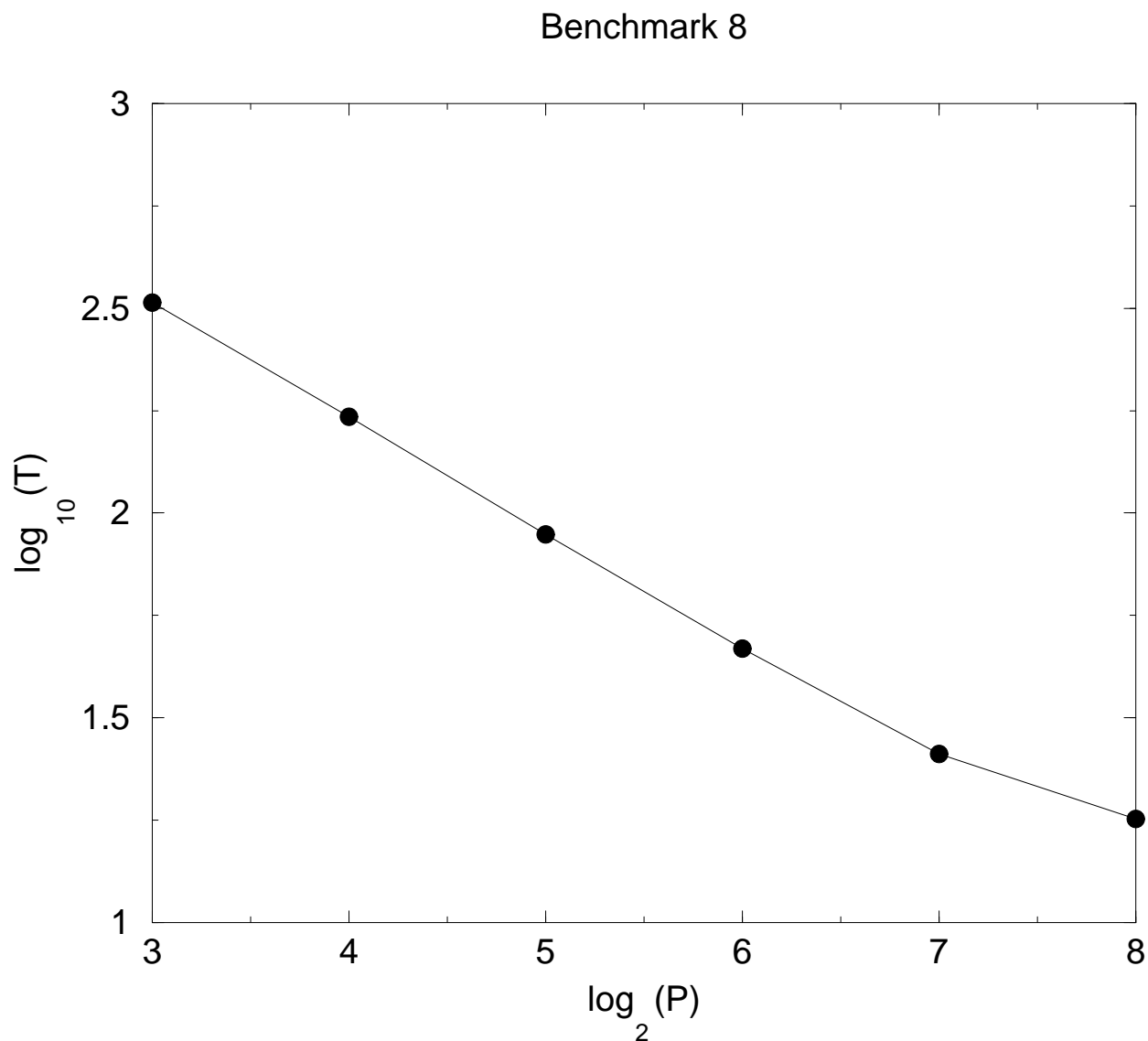
### Benchmark 7



The performance scaling resembles Benchmarks 2 and 3, in showing a reduction in job time with increasing numbers of processors, but not following an obviously linear trend. The scaling is better overall than the previous examples however. The main cause of this improvement, given that the simulations are otherwise similar, is that Benchmark 7 uses a larger cutoff in the electrostatic calculations and therefore has a lower communication/computation ratio, making for better scaling properties.

## Benchmark 8: Magnesium Oxide Microcrystal

This simulation is a roughly cubic microcrystal of 5,416 atoms of magnesium oxide in vacuo without periodic boundary conditions at 2000 K. The electrostatics are calculated directly with a cutoff of 50 Å, corresponding to an all-pairs calculation. The Van der Waals terms are truncated at 10 Å. The simulation is for 100 steps in the Hoover NVT ensemble with a timestep of 1 fs. Timings include data input and output.

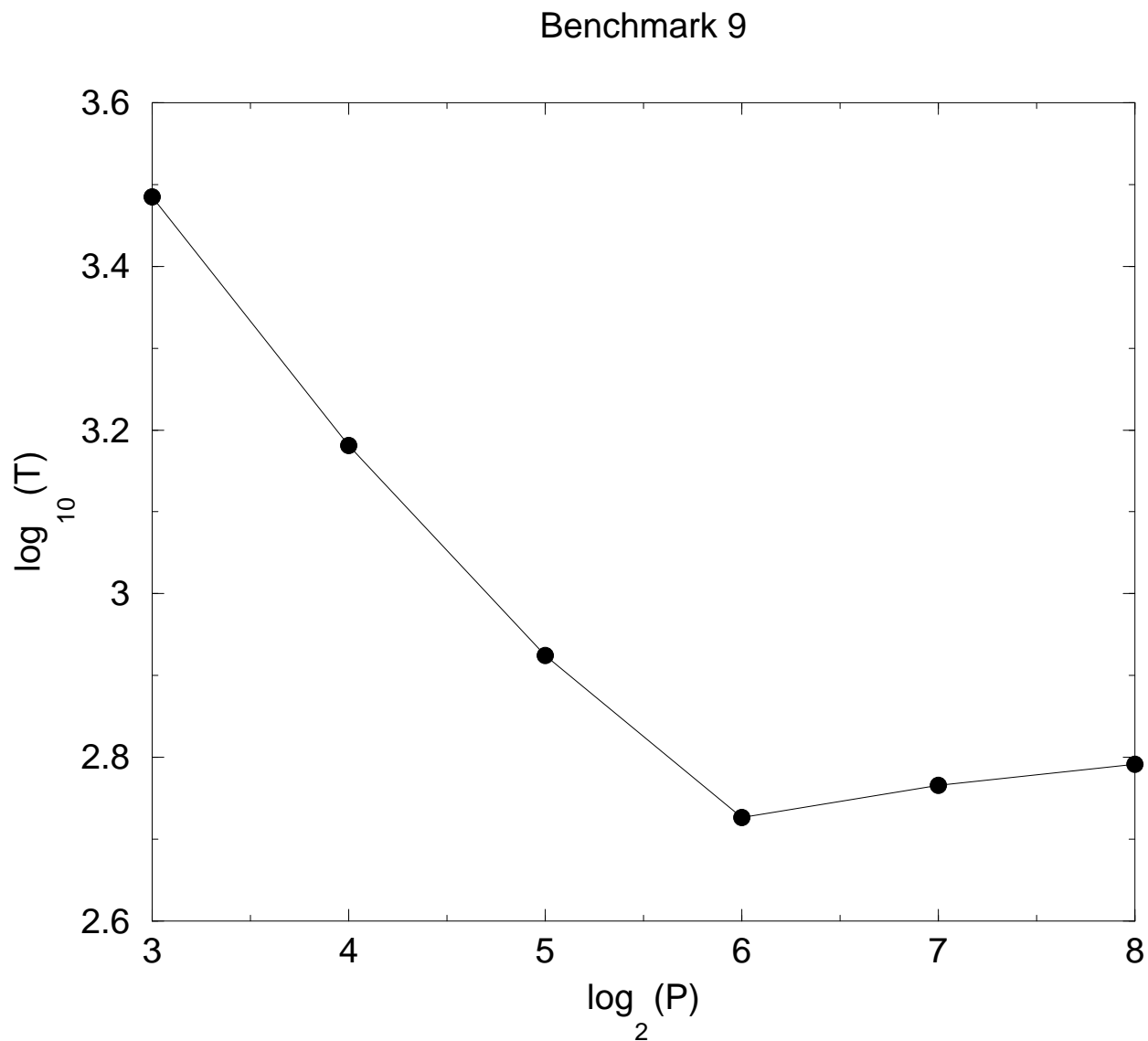


The performance scaling is almost linear for this case, except for a slight deviation at 256 processors. This simulation is heavily compute dominated and so the communication overheads have relatively little impact until large numbers of processors are used. The comparison with Benchmarks 4 and 5

is interesting, in view of the different electrostatic calculation methods.

### Benchmark 9: Model Membrane/Valinomycin System

This simulation is a model of the biological activity of valinomycin in the cell membrane and is comprised of 8 valinomycin molecules (including 4 potassium complexes), 196 hydrocarbon chains each 41 units in length, 25 molecules of potassium chloride and 3144 molecules of SPC water - making 18866 atoms in all. The electrostatics are handled by Ewald sum.



The simulation uses the multiple timestep algorithm and evaluates the reciprocal space terms twice in every 4 steps. The real space electrostatic cutoff is 14 Å, with a primary cutoff of 10.7 Å. The Van der Waals cutoff is 10 Å. The simulation is for 500 steps, with time step of 1 fs, at a temperature of 310 K in the Berendsen NPT ensemble. Timings include data input and output.

The performance scaling is similar to Benchmark 6, with good scaling up to 64 processors and no improvement afterwards. This is ascribed to the same problem, seen earlier, in being unable to assign complete molecules to individual processors in SHAKE, leading to high communication overheads.

## Benchmark Summary

The benchmarks reported here show some distinct features of running DL\_POLY on a parallel computer. Firstly it is clear that performance scaling is generally good if the simulated system does not possess constraint bonds. Secondly, if constraint bonds are present, as they usually are in bio-molecular or polymer systems, then deviations from ideal behaviour are to be expected, and the user must always be aware that using excessive numbers of nodes may be counterproductive. Of course the user is not obliged to use constraint bonds (though this is often the most sensible option) and where extensible bonds can be used, optimal scaling can be recovered. Thirdly, it is generally true that increasing the size of the problem makes for a more efficient parallel implementation, so large simulations can be expected to scale best. The corollary of this is that small systems run best on small numbers of processors.

**Table: Summary of simulations (Job Times in Sec)**

Procs	B1	B2	B3	B4	B5	B6	B7	B8	B9
8	572.0	337.7	-	1385.1	1009.3	635.9	1258.9	326.2	3053.4
16	354.9	203.5	200.2	777.9	523.9	334.9	693.7	171.7	1516.4
32	224.5	141.6	141.7	362.8	248.6	192.7	388.4	88.6	840.0
64	163.5	130.2	119.3	183.1	134.4	133.4	242.7	46.6	532.9
128	176.8	127.8	105.2	94.4	75.4	-	165.9	25.8	583.0
256	178.1	119.9	102.0	62.9	56.3	134.2	139.7	17.9	618.9

## Acknowledgements

The Manchester Computing Centre is thanked for providing access to the **CSAR**

<http://www.csar.cfs.ac.uk> Cray T3E Service. EPSRC is thanked for continuing support of DL\_POLY.