The Demand for Computational Speed

There is a continual demand for greater computational speed from a computer system than is currently possible.

Areas in need of great computational speed:

- numerical modeling and simulation of scientific and engineering problems:
  - Huge repetitive calculations on large amount of data
  - Computations must be completed within a "reasonable" time period
    - weather forecasting have a specific deadline for the computations.

A **grand challenge problem** is one that cannot be solved in a reasonable amount of time with today's computers.

Numerical Weather Prediction

- The atmosphere is modeled by dividing it into three-dimensional regions or cells.
- Rather complex mathematical equations are used to capture various effects.
  - Conditions in each cell such as (temperature, pressure, humidity, wind speed, direction, etc.) are computed at time intervals using various conditions existing in the previous time interval.
- The calculations of each cell are repeated many times to model the passage of time.

The key feature that makes this simulation significant is the number of cells necessary:

Cell size: 1 mile x 1 mile x 1 mile to a height of 10 miles (10 cells high)

Number of cells: 5×10^8

Each calculation requires 200 fp operations

To forecast weather over 10 days using 10-minute intervals: ? ? ?

A computer operating at 100 Mflops would take . . . seconds to perform the calculation.

N-Body Problem

- Predicting the motion of the astronomical bodies in space.
- There are long-range forces and the movement of each body can be predicted by calculating the total force experienced by the body.
  - ***N* bodies ->**
    - ***N-1* forces to calculate for each body,**
    - **so approximately N^2 calculations, in total.**
- After determining the new positions of the bodies, the calculations are repeated.

Even the most efficient algorithms require $N \log_2 N$ calculations.

Example: A galaxy might have **10^{11}** stars --> **10^{22} calculations.**

- **If each calculation is done in 1 msec, one iteration would take 10^9** years for N^2 algorithm, and 1 year for $N \log_2 N$ **algorithm.**

Parallel Computing

One way of increasing the computational speed is by using multiple processors operating together on a single problem.

- The overall program is split into parts
- each of which is performed by a separate processor

Writing programs for this form is known as parallel programming.

The computing platform, a parallel computer, could be

- a specially designed computer system containing multiple processors
- or several independent computers interconnected in some way.

The idea is that n computers could provide up to n times the computational speed of a single computer. This ideal situation is rarely achieved in practice!