**Distributed Computing Reduces Computation Time**

The code calculates the positions of 30 satellites over a 24-hour period with a 1-minute time interval, resulting in a large amount of computation. To reduce the computation time, the code utilizes distributed computing using the joblib library.

**Before Distributed Computing**

Without distributed computing, the calculate\_satellite\_positions function would iterate through each satellite sequentially, calculating its position at each time step. This would result in a computation time of:

30 satellites x 1440 time steps x computation time per satellite

**With Distributed Computing**

By using joblib's Parallel and delayed functions, the computation is distributed across 4 cores (adjustable to the number of available cores). Each core calculates the positions of a subset of satellites, reducing the computation time to:

30 satellites / 4 cores = 7.5 satellites per core x 1440 time steps x computation time per satellite

**Speedup**

The distributed computing approach reduces the computation time by a factor of approximately 4. This significant speedup enables the code to complete the computation in a fraction of the original time, making it more efficient and scalable.

**Benefits**

Reduced computation time

Improved scalability for larger datasets

Efficient use of multi-core processors

By leveraging distributed computing, the code can handle larger datasets and more complex computations, making it an essential technique for scientific computing and data analysis.