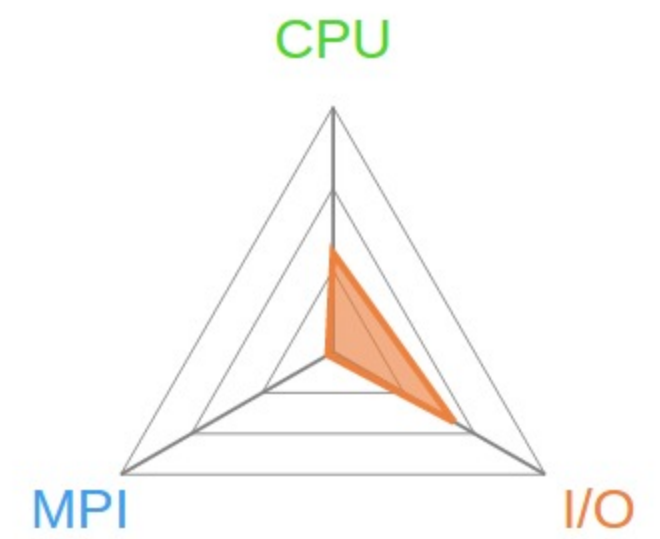


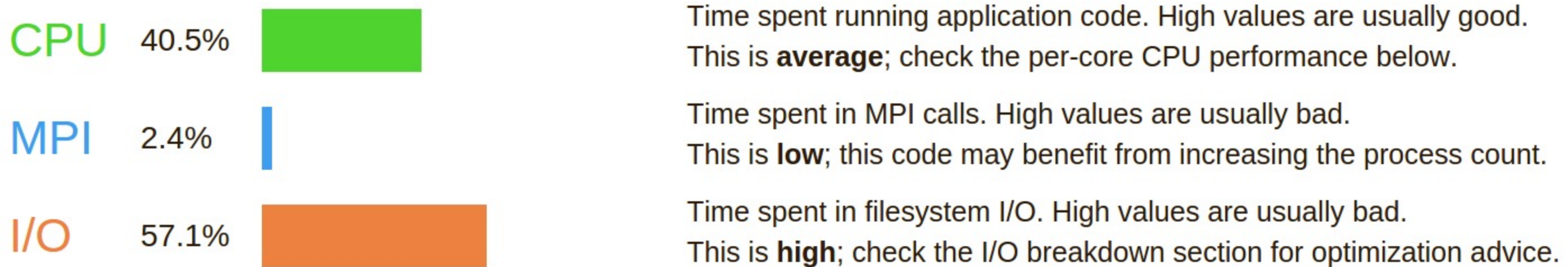


Executable: MADbench2
 Resources: 4 processes, 1 node
 Machine: sandybridge2
 Start time: Mon Nov 4 12:37:08 2013
 Total time: 3 seconds (0 minutes)
 Full path: /tmp/MADbench2
 Notes: 12-core server / HDD / 4 readers + writers



Summary: MADbench2 is I/O-bound in this configuration

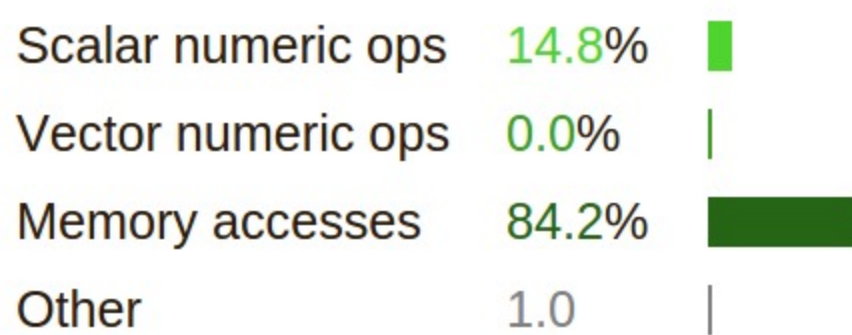
The total wallclock time was spent as follows:



This application run was **I/O-bound**. A breakdown of this time and advice for investigating further is in the **I/O** section below. As very little time is spent in **MPI calls**, this code may also benefit from running at larger scales.

CPU

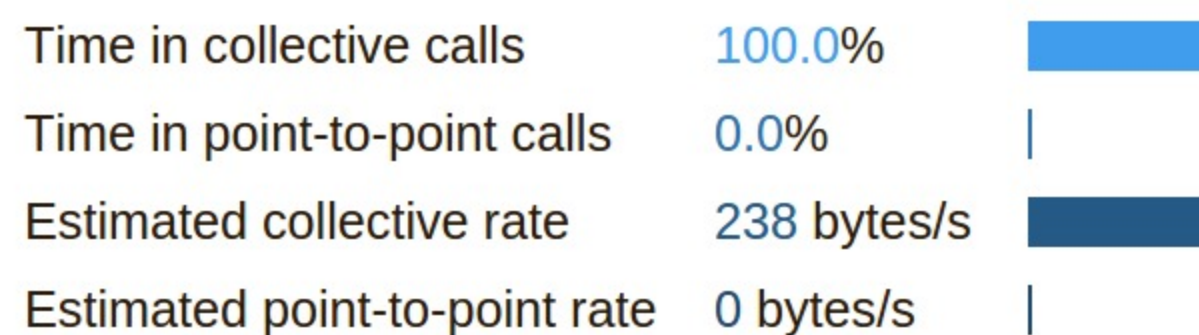
A breakdown of how the **40.5%** total CPU time was spent:



The per-core performance is **memory-bound**. Use a profiler to identify time-consuming loops and check their cache performance. No time was spent in **vectorized instructions**. Check the compiler's vectorization advice to see why key loops could not be vectorized.

MPI

Of the **2.4%** total time spent in MPI calls:



All of the time is spent in **collective calls** with a very low transfer rate. This suggests a significant load imbalance is causing synchronization overhead. You can investigate this further with an MPI profiler.

I/O

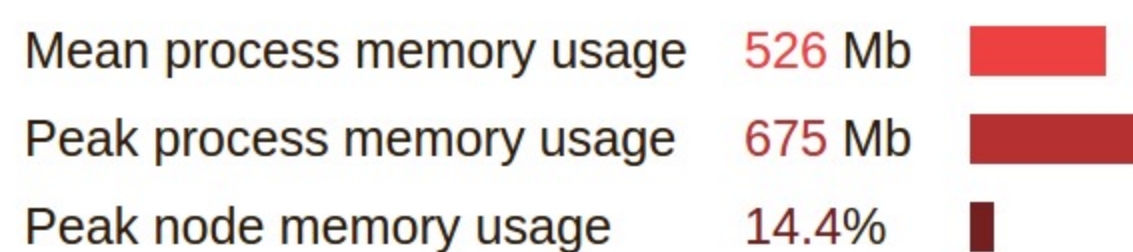
A breakdown of how the **57.1%** total I/O time was spent:



Most of the time is spent in **write operations**, which have an average transfer rate. It may be possible to achieve a higher effective transfer rate using asynchronous file operations.

Memory

Per-process memory usage may also affect scaling:



The **peak node memory usage** is low. You may be able to reduce the total number of CPU hours used by running with fewer MPI processes and more data on each process.