Power Efficiency in High Performance Computing

Shoaib Kamil
LBNL/UC Berkeley
sakamil@lbl.gov

John Shalf
LBNL/NERSC
jshalf@lbl.gov

Erich Strohmaier
LBNL/CRD
estrohmaier@lbl.gov

ABSTRACT

After 15 years of exponential improvement in microprocessor clock rates, the physical principles allowing for Dennard scaling, which enabled performance improvements without a commensurate increase in power consumption, have all but ended. Until now, most HPC systems have not focused on power efficiency. However, as the cost of power reaches parity with capital costs, it is increasingly important to compare systems with metrics based on the sustained performance per watt. Therefore we need to establish practical methods to measure power consumption of such systems in-situ in order to support such metrics. Our study provides power measurements for various computational loads on the largest scale HPC systems ever involved in such an assessment. This study demonstrates clearly that, contrary to conventional wisdom, the power consumed while running the High Performance Linpack (HPL) benchmark is very close to the power consumed by any subset of a typical compute-intensive scientific workload. Therefore, HPL, which in most cases cannot serve as a suitable workload for performance measurements, can be used for the purposes of power measurement. Furthermore, we show through measurements on a large scale system that the power consumed by smaller subsets of the system can be projected straightforwardly and accurately to estimate the power consumption of the full system. This allows a less invasive approach for determining the power consumption of large-scale systems.

1. INTRODUCTION

We are entering an era where Petaflop HPC systems are anticipated to draw enormous amounts of electrical power. Concerns over total cost of ownership have moved the focus of the HPC system architecture from concern over peak performance towards concern over improving power efficiency. The increase in power consumption can be illustrated by comparing typical top HPC systems. In November of 2001, NERSC's new 3 teraflop HPC system was able to reach #3 on the Top500 list of most powerful machines using less than 400 KW of electrical power. In November 2007, NERSC's 100 teraflop successor consumes almost 1,500 KW without even being in the top 10. The first petaflop-scale systems, expected to debut in 2008, will draw 2-7 megawatts of power. Projections for exaflop-scale computing systems, expected in 2016-2018, range from 60-130 megawatts [16]. Therefore, fewer sites in the US will be able to host the largest scale computing systems due to limited availability of facilities with sufficient power and cooling capabilities. Following this trend, over time an ever increasing proportion of an HPC center's budget will be needed for supplying power to these systems.

The root cause of this impending crisis is that chip power efficiency is no longer improving at historical rates. Up until now, Moore's Law improvements in photolithography techniques resulted in proportional reductions in dynamic power consumption per transistor and consequent improvements in clock frequency at the same level of power dissipation– a property referred to as Dennard scaling. However, below 90 nm, the static power dissipation (power lost due to current leakage through the silicon substrate) has overtaken dynamic power dissipation. This leads to a stall in clock frequency improvements in order to stay within practical thermal power dissipation limits. Thus, the free ride of clock frequency and power efficiency improvements is over. Power is rapidly becoming the leading design constraint for future HPC system designs. After many years of architectural evolution driven by clock frequency improvements at any cost, architectural power efficiency matters once again.

In this paper we address how power consumption on small and large scale systems can be measured for a variety of workloads. We do not address the related but independent question of performance measurement itself. In Section 2 we discuss various approaches for defining workloads, procedures for power measurements, and different power efficiency metrics. In Section 3 we describe the experimental setup for the systems in our study. Results for single node measurements are presented in Section 4, for single cabinet measurements in Section 5, and for a full large scale system in Section 6. In Section 7 we demonstrate that full system power consumption can be approximated with high accuracy through extrapolations based on power consumption at the cabinet level. Our conclusions are presented in Section 9.

2. RELATED WORK IN POWER EFFICIENCY METRICS

While metrics for assessing performance such as SPEC-FP [5], the NAS Parallel Benchmarks [9], and the Top500 [6] list have gotten considerable attention over the past decade, similarly robust assessments of power-efficiency have received comparably less attention. In order to foster an industry-wide focus on keeping power consumption under control, it is necessary to provide appropriate power-efficiency metrics that can be used to compare and rank systems in a manner similar to how LINPACK is used for ranking peak delivered performance. Such efforts are already underway for commercial data centers. For example the EPA Energy Star program has defined a rigorous set of Server Metrics [1] and testing methodologies oriented towards transactional workloads. However, they are inappropriate for assessing the power efficiency delivered for HPC and scientific workloads. The emerging SpecPower metric is valuable for assessing technical applications on workstation-class computing platforms, but may have limited applicability to large scale HPC systems.

It is our intent to foster development of power metrics by popular HPC rankings such as the Top500 list to develop efficiency standards that are appropriate for scientific workloads. To arrive at a sound power efficiency metric we need to define a suitable workload for performance measurements as well as power measurements, a power measurement procedure, and an appropriate metric itself.

2.1 Workload Definition for Performance Measurements

For any serious evaluation of performance, it is critically important to develop a workload that correctly reflects the requirements of large-scale scientific computing applications. Contrary to first impressions this continues to be largely unsolved and has not proven straightforward.

There have been numerous alternative computer architectures proposed to address power efficiency concerns, but little information on sustained power efficiency. A number of novel architectures such as General Purpose GPUs [14], the STI Cell Broadband Engine [18], and embedded-processor-based systems like the IBM BG/L hold some promise of improving the power efficiency of HPC platforms, but the lack of a uniform basis for comparing such systems makes it difficult to determine whether any of these approaches offer genuine power efficiency benefits for relevant scientific computing problems.

2.2 Workload Definition for Power Consumption

One important question addressed in this study is whether the computational workload defined for performance measurements has to be used for power measurements as well. Similarly, it is important to know whether power measurements of a different and potentially simpler workload can be used without loss of accuracy.

It is generally well understood that power consumed under a full system load is considerably higher than the idle power. However, we do not understand how much the application mix on a fully loaded system can effect power draw. In this study we examine the extent to which the choice of application effects power consumption. We find that the power consumed under a full LINPACK workload is very close to the power consumed by a more diverse set of scientific applications. Therefore, the power consumed under LINPACK can be used as a proxy for the power consumed by a much broader variety of workloads (eg. power efficiency based on NAS benchmark performance or power efficiency based on HPCC benchmark performance). There is no need to measure the power consumption under each different benchmark workload— the power drawn while running LINPACK can suffice.

2.3 Methodologies for Measuring Power Consumption

Several different methods for measuring power usage on current architectures have been proposed. These methods differ in the tools used for measuring power consumption and in the various places where valid measurements can be collected. In this study we explore several different measurement methods, and compare their effectiveness in our experience.

We investigated a variety of measurement techniques to fit into the diverse constraints of existing facility infrastructure. For example, in many facilities more than one system shares the same PDU or metered circuit, therefore making it very difficult to isolate system power. In Warren et al [17], the authors use Transmeta processors and infrastructure allowing them to utilize an off-the-shelf UPS system to measure power because of low power consumption as well as the use of a system that uses standard 3-prong wall sockets. Their power consumption methodology is not generally applicable, since most cluster systems do not use wall socket connectors for power; in addition, current cluster designs attempt to perform a single AC to DC conversion for the entire rack.

A 2005 study due to Feng et al [11] proposed a framework for building cluster systems from commodity components and integrating a set of extension wires from the power supply, each connected to a sensor resistor and a digital multimeter. Experimentally, they correlate each wire with its associated components, and then measure the power consumption for various NAS parallel benchmark codes. Although their power measurement hardware is infeasible for many systems, they provide important results that agree with our findings.

Previous work has shown that it can be extremely challenging (and likely impractical) to measure power for a complete HPC system while running the LINPACK benchmark. Systems that have already collected performance data are loath to take a system out of service to collect the data yet again while measuring the power consumption. Our study demonstrates that measuring the power consumed by LINPACK on a small fraction of the system (even a single rack) can be used to accurately project power consumed by the overall system.

Overall the proposed Top500 power-consumption measurements should be very practical to collect and have a low impact on center operations.

2.4 Power Efficiency Metrics and Procedures for Ranking Systems

Having defined how to measure performance and power consumption of an HPC system, the question remains how to combine the measured values into a power efficiency metric to compare technologies, architectures, and systems and rank them. Extreme ranking procedures would use only performance values, such as the LINPACK numbers used in the TOP500 [6] or only total power consumption, such as occasionally shown in presentations [15].

The ratio of performance per watt drawn called “power efficiency” is a popular metric used to compare systems.

The recently published Green500 ranks the TOP500 systems based on estimates of this ratio [8].

Power efficiency is certainly a proper metric to compare competing technologies or different system architectures. However, as any metric, power efficiency does have limitations and one must be careful not to use it beyond its limits as results otherwise become misleading.

For embarrassingly parallel workloads and system interconnects with linear power scaling, performance and power draw are essentially extensive quantities, values that grow linearly with an appropriately chosen system size parameter. This in turn makes power efficiency a ratio of two extensive properties and thus an intensive property— it is independent of system size! Intensive system properties and efficiencies are not suitable to rank systems of various sizes, as they are constructed in a way to not to depend on system size.

The effect of this misuse of power efficiency for ranking systems can be seen clearly in the Green500. LINPACK values grow slightly sub-linearly with system size measured by node count, while the power estimates used grow linearly with it. As a result the Green500 power efficiency rank decreases for any given system architecture with system size. This results in, for example, all BlueGene/L systems being inversely sorted by system size, leading to the misleading statement that smaller systems are more power efficient than larger systems. An appropriate metric for ranking individual systems in a similar fashion as currently done in the TOP500 needs to utilize a metric which implicitly grows with system size.

3. EXPERIMENTAL TESTBED

We assess a variety of standalone systems similar to nodes in much larger HPC systems, and one large-scale Cray XT4 system. We also use a variety of measurement methods to collect power consumption data on the targeted systems. In the sections below, we describe the hardware platforms involved in this assessment, the measurement methods employed, and the benchmarks that we use in our study.

3.1 Individual Node Configurations

Our work examines the power consumed when running scientific kernels and microbenchmarks on individual nodes that comprise common cluster system architectures. Our testbed consists of a dual IBM PowerPC970-based system, a dual-socket AMD Opteron system, and a dual-core Intel Core Duo based system.

Our AMD system was a typical dual-socket single-core motherboard with each processor operating at 2.2 GHz. The cores provide a peak double-precision floating point performance of 4.4 GFlop/s per core. Each socket includes its own dual-channel DDR2-667 memory controller, delivering 10.66 GB/s, for an aggregate NUMA (non-uniform memory access) memory bandwidth of 21.33 GB/s for the dual-socket Tyan system examined in our study.

The Apple PowerMac G5 hardware platform contains two 2.7 GHz IBM PowerPC 970FX CPUs, have a peak performance of 10.8 double precision GFLOP/s, with2 GB PC-3200 DDR memory providing 6.4 GB/s of memory bandwidth. The overall structure of the system is generally similar to platforms that include Intel or AMD processors; while the specific details may differ, the general out-of-order processing, memory configuration, and operation closely resemble our evaluated platform.

The Intel Core Duo machine we used was an Apple MacBook Pro with a 2.0 GHz dual-core processor and 2 GB of DDR2 RAM. To eliminate the display and battery as a source of power drain, we ran all tests using AC power only (no battery installed) and an external display with a separate power supply that was not instrumented. The MacBook Pro was running Mac OS X 10.4.8 and no applications other than those running in a normal startup of the OS.

3.2 Full System Configuration

Our tests were conducted on Franklin, NERSC's Cray XT4 system, which consists of 9,660 compute nodes (plus 20 spares), each with a dual-core 2.6 GHz AMD Opteron processor, for a total of 19,320 available compute processor cores (40 spares). The theoretical peak performance is 5.2 GFlop/s per node. Each compute node contains 4 GB of memory and runs either a custom Cray OS called Catamount or a lightweight OS based on Linux called Compute Node Linux. In addition to the compute nodes, there are 16 login nodes running a full Linux system with 8 GB of memory per login node, as well as 96 filesystem service nodes that provide a 350 TB Lustre shared filesystem using a Data Direct Networks (DDN) backend. The entire system is contained in 102 cabinets of 96 nodes each. The interconnect topology of the XT4 is a 3D torus, with each node connected via HyperTransport to a dedicated SeaStar2 network router.

Each cabinet contains three blade modules. Each module contains 8 blades mounted so that air can flow through them vertically (from the bottom to the top of the cabinet). Each XT4 blade consists of 4 AMD sockets, which are independent nodes, and their associated memory and the SeaStar routers. Therefore, each rack contains a total of 96 nodes consisting of 96 AMD dual-core processors for a total of 192 cores per cabinet. The power feed to each cabinet is 208 VAC 3-phase and is capable of handling 25 KW per rack. Each cabinet has a single 92 percent efficient power supply at the bottom of the rack for the AC to DC conversion. The 48 VDC power is then distributed to each of the blade modules using substantially large copper conduits to handle the large currents.

3.3 Power Measurement Methods

Unfortunately, measuring power usage remains an inexact process, with accuracy and feasibility varying due to the many methods used to measure power and at what concurrency (from single nodes to entire HPC systems). In the course of this work, we examined several possible ways to obtain power usage readings and compare their effectiveness and practicality.

3.3.1 Line Meters

We used inline meters for our single-node tests. These meters are fairly simple to use, and can output readings at one second intervals over serial, making record keeping easy and accurate. In addition, the rated accuracy for most measurements was 0.5-1.0%, which is quite good. However, such meters generally work with only one or two voltages (i.e. 110 V or 212 V) and require disconnecting the system to be measured. For a single node, this is rarely a problem, but inline meters are infeasible for measuring, for example, the power usage of an entire rack.

We used an Electronic Product Design PLM-1-PK calibrated power meter that measures instantaneous power usage and outputs a variety of measurements every second. The meter measures power at 120 VAC, and for the instantaneous power usage (in watts), the error factor is $0.5\% + 1LSD$. These meters are calibrated yearly by the manufacturer to ensure they remain accurate and precise.

For larger-scale systems, such as the Cray XT4 (Franklin), using inline meters is not as practical. Power is supplied to the XT4 cabinets via 208 V 3-phase power conduits that are rated at 60 A. At that current rating, the power whips are bolted down on both ends (both the power supply and the panel end-points), making it impractical to use inline meters for the power feed to each cabinet. We point out that Power3 and Power5 SP systems also employ a similar power distribution method. In the case of the Power5-Federation systems (NERSC Bassi and the ASCI Purple system), the power is converted from 408 VAC to 350 VDC for distribution within the rack. The SGI ICE and Altix systems also use 48 VDC power distribution within the rack. Therefore, unlike commodity clusters that use conventional AC power plugs, typical high-end HPC systems are not amenable to power measurement using in-line meters.

3.3.2 Clamp Meters

Clamp meters provide a way to measure power without needing to disconnect a system. In addition, clamp meters work with a large range of voltages and wire types (single core, multicore). We attempted to use an AVO International Megger FlexiClamp 200 meter to investigate how useful clamp meters could be for measuring HPC power consumption. Normally, clamp meters can only be used to measure current on individual conductors of a 2-phase or 3-phase multi-conductor cable. The manufacturer of our clamp meter claimed that it was capable of measuring current in multi-conductor wires (2-phase or 3-phase). However we found the meter incapable obtaining precise measurements with a clamp meter when encountering wire with more than one wire core for our single-node tests. Even for measurements on simple 2-phase (3-conductor) wires, the measurements varied by over 50% in some cases. In fact, the specifications for our clamp meter only claim accuracy to within 12%.

We examined the possibility of using the clamp meters for measurements on our large-scale HPC systems since it was infeasible to use an in-line meter. It is theoretically possible to measure current on a 3-phase carrier by clamping each phase individually and taking into account the phase angle. However, there was not a convenient or safe location to separate out the phases at either end of the power whips (either the panel or the power supply). In theory, one could build a custom cable with the phases broken out, but this would involve taking the system down to reinstall the affected power whips (a non-starter for a production supercomputing center), and violate local standards in the electrical code.

Ultimately, we do not feel it is feasible to use clamp meters successfully except if individual phases are broken out into separate clamp-able wires. This proved to be infeasible for systems we examined.

3.3.3 Integrated Meters

Power supply manufacturers have recently begun providing mechanisms within their products to monitor and record power usage. However, such devices are fairly new and not widely deployed, leaving their accuracy unknown.

In the case of our Cray XT4 system, the power supply contains an ethernet interface that allows us to measure the current and voltage of each of the 3 DC outputs that feed the 3 compute shelves in the rack. The measurements are relatively coarse-grained because the commands must be submitted through an interactive command-line interface. It is not feasible to monitor all of the racks on a continuous basis because the current ALPS system management interface allows direct connection to only one rack at a time for manual diagnostic purposes. The utility was never intended to provide manual access to all of the racks at the same time for this particular purpose. It is possible that future implementations may provide comprehensive and continuous power monitoring.

3.3.4 Power Panels

Lastly, power panels in power distribution units (PDUs) are one way to measure the power usage of a large system. The power systems in most facilities incorporate metering capabilities as an important diagnostic function for building electrical infrastructure. The panel does not provide fine-grained measurements of individual pieces of a system, but rather allows the entire system to be characterized. Generally, the accuracy may only be to the nearest kilowatt, and may require the observer to manually record each reading. Nevertheless, PDU power panels provide the only way to monitor large pieces of an HPC system or the entire HPC system itself. We utilized power panels at NERSC to record the power consumption of the entire XT4 system made up of several thousand cores.

3.4 Benchmark Codes

We selected a number of microbenchmark codes to exercise orthogonal hardware components. In addition, we included a mix of full-fledged scientific applications from the NERSC workload and some application kernels (such as the NAS PB) to represent the requirements of a typical scientific workload.

3.4.1 Single Node Tests

For a CPU-centric benchmark code, we used the C port of the LINPACK benchmark [2] by Bonnie Toy (HPL), which performs CPU-intensive linear algebra routines to test peak CPU floating-point performance. For our test runs, we used double precision algebra with unrolled loops to exploit as much of the CPU as possible.

In order to exercise machine features somewhat orthogonal to the CPU, we use the standard STREAM [4] benchmark, which exercises the memory subsystem of a computer, measuring the maximum utilizable bandwidth for a set of simple loops. Our test runs used the C version of the benchmark and increased the appropriate memory sizes and iteration counts to ensure the benchmark ran for a longer period of time (long enough to measure the power usage).

For a third data point that explores the power usage of the I/O subsystem, we used the IOZone benchmark [7]. IOZone is a self-contained benchmark suite that stress-tests the filesystem of a machine by varying a number of parameters and measuring the I/O performance. We used the default "automatic" test suite (which uses 256 MB files) and measured how power varied throughout.

As proxies for single-node scientific applications, we used a subset of the serial implementations of the NAS Parallel Benchmarks [9]. Although they are much simpler than typical full applications, we ran FT, CG, and LU from the benchmark suite because they represent a mix of memory-intensive and CPU-intensive operations for solving a simplified version of an actual problem of interest. Therefore, their memory and CPU usage is interesting in that it exercises both subsystems, as opposed to STREAM and LINPACK.

For all of the benchmark codes, we also ran two instances of each to test multi-core or SMP power usage, because all of our single-node test machines contain at least two cores/processors. The goal was to see if using both cores increases power consumption substantially.

3.4.2 *Multiple Node Tests*

The workload in the single-cabinet power tests consists of a set of microbenchmarks and production applications. We run High Performance LINPACK (the standard benchmark for the Top500 list) and MPI-STREAM to test the memory and compute power usage, along with a subset of the NAS Parallel Benchmarks and two applications: Paratec [10] and PMEMD [3], which are both used in production at NERSC and are components of the NERSC SSP (Sustained System Performance) metric [12].

The NERSC SSP is a methodology for measuring real-world performance using a set of production applications and simplified versions of production codes. For the full system test, we again run HPL and STREAM, but also run a mix of SSP applications on the machine to simulate a full-system workload in a production situation.

4. SINGLE NODE POWER

Our test methodology consisted of running our single-node test applications in sequence, allowing the processor to return to idle operation in between each test. We ran each application for 3 minutes and allowed 2 minutes for the processor to return to idle. In addition to single process tests, we also tested running two instances of the benchmarks at the same time (one per core/processor).

4.1 Single Node Results

In this section, we show power consumption results for standalone machines while running our test benchmarks. We first show the non-I/O results, then present IOZone power usage on the test machines.

4.1.1 *Non-I/O Benchmarks*

Figure 1 shows the results of running our benchmark suite on the 2.0 GHz Core Duo MacBook Pro. Interestingly, the highest power usage occurs in the memory-intensive benchmarks (running two instances of STREAM, for example) while benchmarks that focus mainly on CPU usage, such as LINPACK, result in lower power measurements.

The absolute power usage of this system is quite low at both idle and at peak. At idle, it consumes about 20 watts of power, and at peak, we saw power usage of 32 W, or 60% more power than at idle. The relative power usage at peak is high compared to the power used under no load.

The Opteron machine, slaphappy, consumes more than 10x the power of our MacBook Pro system, as shown in Figure 2. At idle, power consumption is just under 250 W, while peak power usage was measured at about 295 W. On this machine, the benchmarks did not show much differentiation—running one iteration of any of the benchmarks yielded almost the same power usage. For two instances, the power signatures of all the benchmarks except for STREAM were relatively similar. The relative power usage at peak is only 19% more than at idle.

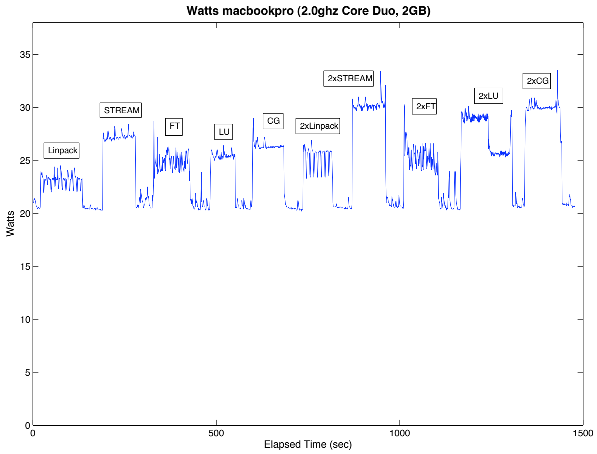

Figure 1: Results on our 2.0GHz Intel Core Duo machine.

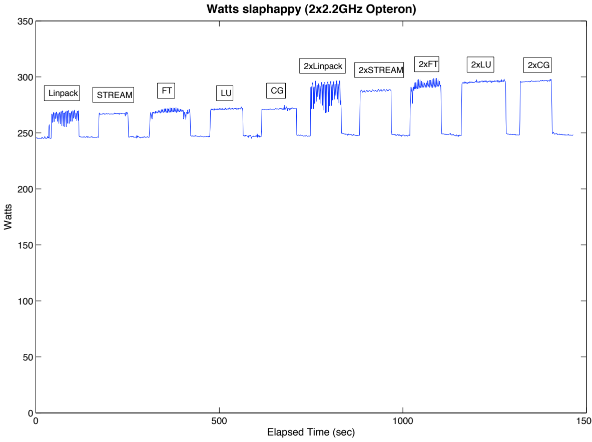

Figure 2: Results on our dual-processor Opteron test system.

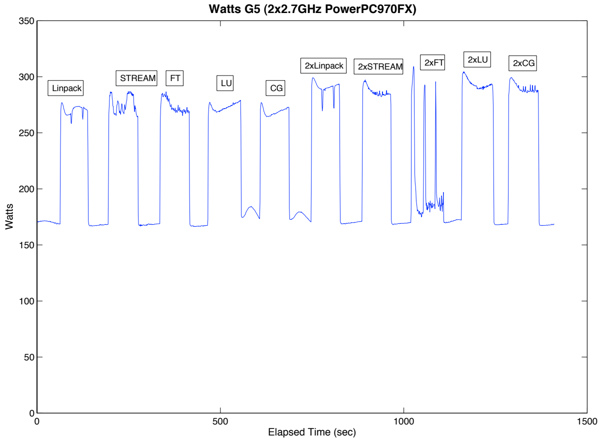

Figure 3: Results on our dual-processor G5 system.

Lastly, we present the power usage of our PowerMac G5 in Figure 3. Idle power usage is around 170 W, with peak being 80% higher at 302 W. Interestingly, the power usage does not vary much whether we run on or two processes. In addition, when beginning a benchmark run, the power usage spikes up and then comes down somewhat; this was reflected in the fan noise, which would initially spin up quite high before slowing down. This system displayed the highest difference between peak and idle power usage.

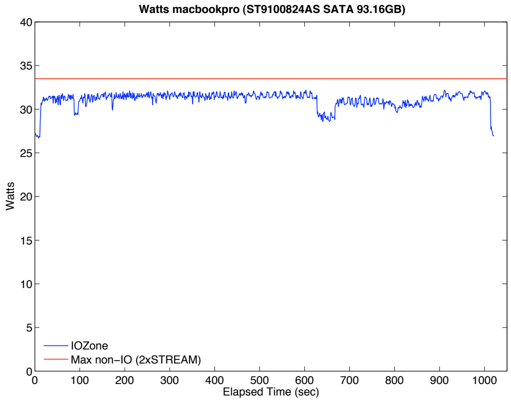

Figure 4: IOZone results on our 2.0GHz Intel Core Duo machine.

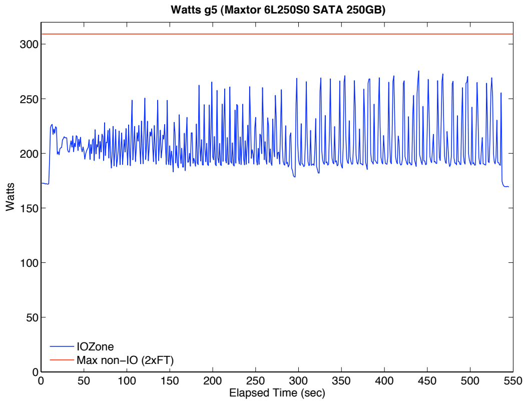

Figure 5: IOZone results on our PowerPC G5 machine.

4.1.2 *IOZone Results*

Our test Core Duo machine is equipped with a SATA 100 GB drive; the results for running IOZone are shown in Figure 4. Note that the power usage does not reach the maximum we observed in the non-I/O benchmarks. In addition, power usage throughout the test is somewhat variable—ranging from 28 to 32 watts usage over the course of the test.

Figure 5 shows the results of running IOZone on our PowerPC G5 test system. The power usage is quite variable throughout the test, ranging from a near-idle 170 watts to up to almost 270 W, for a variability of 100 watts throughout the I/O test.

4.2 Discussion

A tabular summary of the power usage while running our benchmark suite on the three test systems is given in Table 1.

	CoreDuo	Opteron	G5
Idle	21 W	245 W	170 W
LINPACK	23 W	267 W	270 W
STREAM	27 W	267 W	282 W
NAS FT	25 W	269 W	273 W
NAS LU	25 W	271 W	274 W
NAS CG	26 W	271 W	270 W
2xLINPACK	24 W	290 W	291 W
2xSTREAM	30 W	287 W	286 W
2xNAS FT	26 W	292 W	240 W
2xNAS LU	29 W	295 W	294 W
2xNAS CG	30 W	296 W	288 W

Table 1: Approximate average power usage for each of the benchmarks on our three test systems.

When examining our results across the three machines, we see that in each case, the power usage immediately increases when the processor system begins running a benchmark, then stays relatively level for the length of the benchmark run. After the run is over, we see an immediate drop in power usage to idle.

Another insight that is readily apparent is that the power usage does not vary much depending on which benchmark we run. For the G5 system, running one or two instances of each benchmark resulted in essentially the same power signature, while for the Opteron, each of the benchmarks had almost the same power usage (although running two instances resulted in different power usage than running one instance). The MacBook Pro showed the most difference between the benchmarks, with as much as 20% difference between benchmark power usage.

4.3 Results

5. MODELING A FULL MACHINE

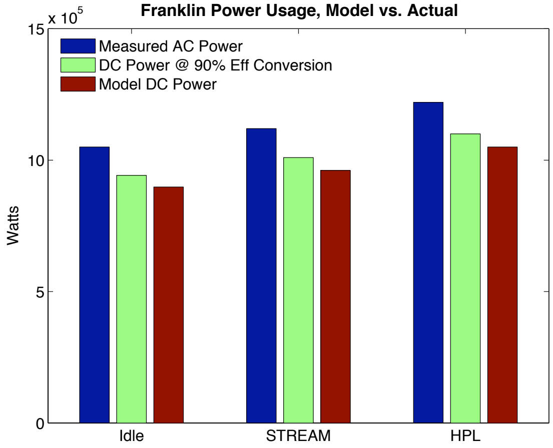

Figure 8: Model vs Actual power on the entire Franklin system.

6. SINGLE-CABINET POWER USAGE

In this section, we examine the power usage in a single cabinet of a large HPC system. Using benchmarks that are a mix of full applications, microbenchmarks, and application-like proxies, we measure their power consumption for a sin-

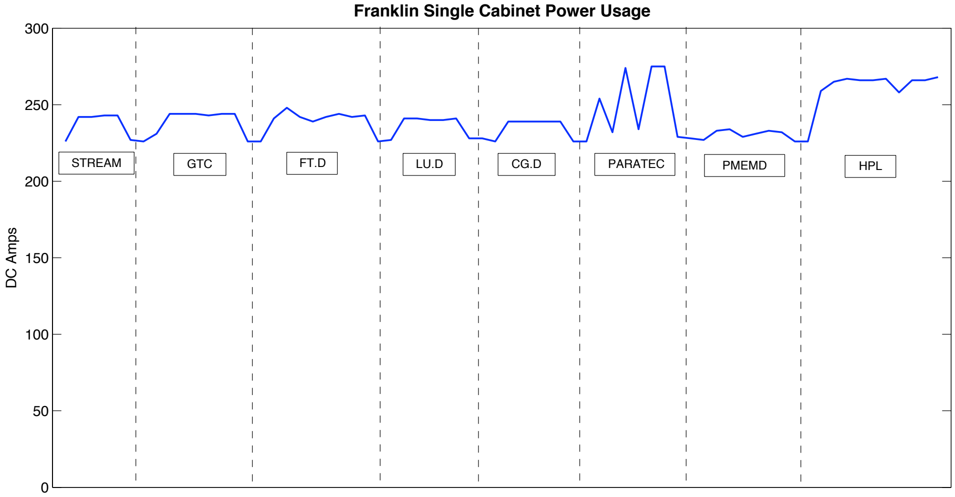

Figure 6: Power usage on a single cabinet of Franklin (192 cores) for various applications and benchmarks.

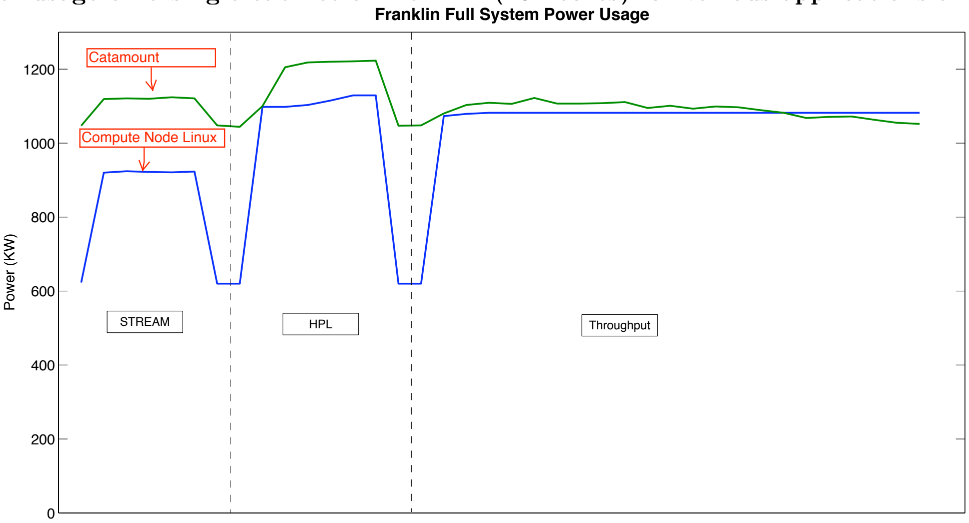

Figure 7: Power usage on the entire Franklin system. Throughput is realistic workload mix run at NERSC. The upper line is power under the Catamount OS, while the lower line is under Compute Node Linux.

gle 96-node cabinet of Franklin, the Cray XT4 system at NERSC.

Idle power was measured to be in the vicinity of 220 amps DC, and for most of the applications, the loaded power is about 245 amps. HPL and Paratec display the highest power usage at around 270 amps. The difference between idle and peak is almost 20%, while the difference between HPL and the rest of the applications is less than 9%.

In the next section, we report full-system power usage, and compare that to a modeled version of the system based on the power used for a single cabinet.

7. FULL SYSTEM POWER USAGE

Measuring power usage on an entire system is rather difficult; few mechanisms exist to measure power across a large number of cabinets. For our tests, we used the power readings from a panel that connects the machine to the main power intake into NERSC. For these tests, we ran the codes across all 19,320 compute nodes of Franklin. We ran one set of tests using the Catamount OS on the compute nodes and another set using CNL, and characterized power usage for three workloads: HPL, STREAM, and a mixed workload used by NERSC that consists of a number of production applications run simultaneously to fill the machine.

The results are shown in Figure 7. Interestingly, the power usage under load is the same for both OSs, although the idle power is quite different, due to Linux having better power management. In particular, the Linux idle() loop uses DVS to save power when the system load average is low, but DVS apparently provides very little benefit for the simulated workload (the ”throughput” test). In addition, we see that the HPL power consumption is quite similar to the mix of production codes in the throughput test.

Based on our results in the previous two sections, we attempted to model the full Franklin system power usage using single-cabinet results. In addition to the nodes of Franklin, an additional power draw in the full system test is due to our DDN-based file system backend. Using actual power consumption numbers provided by DDN (and comparing with measured data from a power sub-panel in the machine room) we estimate that the disk subsystem consumes approximately 50 KW, which is a tiny fraction of the entire system usage.

Next, we linearize the power data obtained from a single rack to the full 102 racks and attempt to correct for power loss in the AC to DC conversion. That is, our model is simply

$$DCWatts_{system} = 102 \times DCWatts_{rack} + 50KW$$

$$= 102 \times DCAmps_{rack} \times Volts_{rack} + 50KW$$

However, since the power panels where we obtain measurements for the full machine deal with AC, we must correct for power loss due to conversion. According to Cray, the AC-to-DC converters in each rack are approximately 90% efficient. Thus, in Figure 8, we compare the power usage from our model to DC power at 90% conversion efficiency of the measured AC power.

Our model manages to capture the power usage of Franklin quite well. Using our simple linearized model, we obtain less than 5% error from the measured power usage under a 90% efficient conversion from AC to DC. Although this model may seem trivial, it is important that such a simple estimation strategy can deliver a fairly accurate number for overall power consumption. For some supercomputing centers, it may be impossible to take an entire machine offline to measure power consumption; the close agreement between our simple model that uses measurements from a single cabinet to estimate power across the machine and the actual usage points to a possible methodology for measuring power consumption without requiring a full shutdown of a machine.

8. CONCLUSIONS

In this paper, we present some initial work for measuring the power consumption of large-scale supercomputers under HPC workloads. Beginning with single-node tests, we increase the scale of our power measurements up to a full-scale modern supercomputer, a Cray XT4.

We confirmed the general understanding that nameplate (or "rated") power was generally much higher than any realistic power measurement under a production scenario, and CPU power is also an inaccurate predictor of power consumption. Therefore, it is essential to measure power consumed when running a suitable workload. We found tests on individual workstations to yield results similar to tests run on a large scale system. That is, there did not seem to be a large difference in power usage due to switch fabric or other cluster-wide issues for the benchmarks we used. In addition, the maximum power usage we saw was during the runs of HPL, a compute-intensive benchmark; our memory-intensive MPI-STREAM benchmark yielded the lowest power usage. However, the difference between the two was miniscule; therefore, we believe that running HPL is a good proxy for the power consumption of a general HPC workload. Furthermore, we were able to model full system power by linearly extrapolating from a smaller piece of the system; thus, it is probably good enough to measure the power consumption of a rack or group of racks and extrapolate when direct measurement is impossible. For measurement apparatus, we recommend using an isolated PDU or vendor-included functionality.

In our future work, we will measure power on several a wider array of large-scale systems to verify our results. Motivated by these measurements and observations, we will define a procedure and metrics for fairly comparing the power usage of large-scale HPC systems across the variety of architectures present today and in the future.

9. REFERENCES

[1] EPA ServerMetrics Workshop. http://www.energystar.gov/serverconference.

[2] HPL - a portable implementation of the high-performance linpack benchmark for distributed-memory computers. http://www.netlib.org/benchmark/hpl/.

[3] PMEMD Homepage. http://amber.scripps.edu/pmemd-get.html.

[4] Stream: http://www.cs.virginia.edu/stream/.

[5] Standard Performance Evaluation Corporation CFP2000. http://www.spec.org/cpu/CFP2000/, 2000.

[6] Top500: http://www.top500.org, 2005.

[7] Iozone filesystem benchmark. http://www.iozone.org, 2006.

[8] Green 500 list. http://www.green500.org, 2007.

[9] NAS Parallel Benchmarks. http://www.nas.nasa.gov/Resources/Software/npb.html, 2007.

[10] A. Canning, L.W. Wang, A. Williamson, and A. Zunger. Parallel empirical pseudopotential electronic structure calculations for million atom systems. *J. Comput. Phys.*, 160:29, 2000.

[11] Xizhou Feng, Rong Ge, and Kirk W. Cameron. Power and energy profiling of scientific applications on distributed systems. *ipdps*, 01:34, 2005.

[12] William Kramer, John Shalf, and Erich Strohmaier. The nersc sustained system performance (ssp) metric. *LBNL Tech Report 58868*, 2005.

[13] L. Oliker, A. Canning, J. Carter, J. Shalf, and S. Ethier. Scientific computations on modern parallel vector systems. *IEEE Supercomputing 2004*.

[14] John D. Owens, Shubhabrata Sengupta, and Daniel Horn. Assessment of graphic processing units (gpus) for department of defense (dod) digital signal processing (dsp) applications. *UC Davis Tech Report*, 2006.

[15] John Shalf and David Bailey. Power efficiency metrics for the top500. *Top500 BoF, Supercomputing 2006*, 2006.

[16] H. D. Simon, T. Zacharia, and R. Stevens. Modeling and simulation at the exascale for energy and the environment. *Department of Energy Technical Report*, 2007.

[17] Michael S. Warren, Eric H. Weigle, and Wu-Chun Feng. High-density computing: A 240-processor beowulf in one cubic meter.

[18] S. Williams, L. Oliker, J. Shalf, P. Husbands, S. Kamil, and K. Yelick. The potenial of the cell processor for scientific computing. *Proceedings of the ACM Computing Frontiers Conference 2006.*