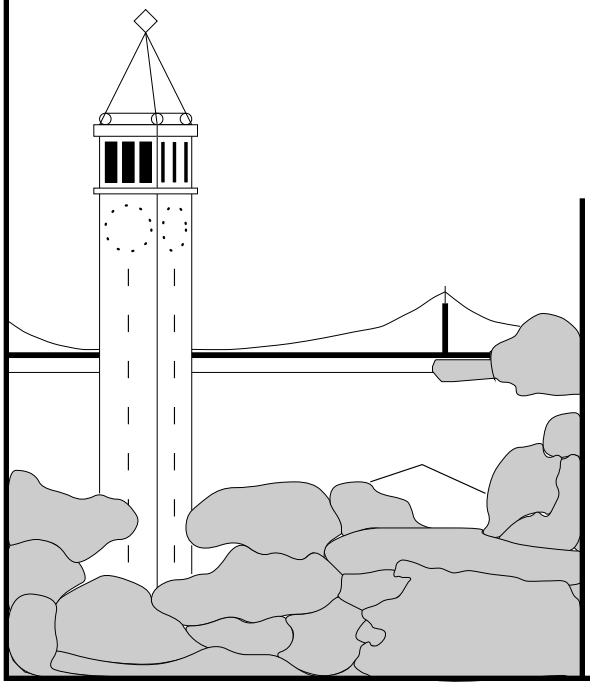
A note on "The Limited Performance Benefits of Migrating Active Processes for Load Sharing"

Allen B. Downey and Mor Harchol-Balter



Report No. UCB/CSD-95-888

November 1995

Computer Science Division (EECS)
University of California
Berkeley, California 94720

A note on "The Limited Performance Benefits of Migrating Active Processes for Load Sharing"

Allen B. Downey* Mor Harchol-Balter†

November 1995

Abstract

The 1988 paper, "The Limited Performance Benefits of Migrating Active Processes for Load Sharing," by Eager, Lazowska and Zahorjan concludes that migrating active processes for load balancing offers little additional performance benefit beyond that obtained using only remote execution (placement). This result is based on analysis and simulation of a system model that is intended to overestimate the performance benefit of migrating active processes.

This report examines the system model used by Eager, Lazowska and Zahorjan and concludes (1) that it does not describe many systems, like networks of workstations, to which its results have been applied, and (2) that it underestimates the potential performance benefit of migrating active processes.

1 Introduction

Based on analysis and simulation with synthetic workloads, Eager, Lazowska and Zahorjan ([ELZ88]) claim that "there are likely no conditions under which migration could yield major performance improvements beyond those offered by non-migratory load sharing..."

This result has been widely cited, and in several cases used to justify the decision not to implement migration or not to use migration for load balancing. For example, [ZWZD93] explain, "Our second design decision is to support remote execution only at task initiation time; no checkpointing or task migration is supported. ... For improving performance, initial task transfer may be sufficient; a modeling study by Eager, Lazowska and Zahorjan suggests that

*Partially supported by NSF Cooperative Agreement No. ASC 8902825. downey@cs.berkeley.edu

†Supported by National Physical Science Consortium (NPSC) Fellowship. Also supported by NSF grant number CCR-9201092. harchol@cs.berkeley.edu

dynamic task migration does not yield much further performance benefit except in some extreme cases."

ELZ's system model is intended to be conservative in the sense that it overestimates the benefits of migration of active processes and underestimates the benefits of non-migratory load-sharing. In this report we point out that there are, in fact, several ways in which ELZ's analysis and workload model understate the benefits of migrating active processes. We also discuss their system model and its applicability to current systems.

We conclude that the general result of ELZ does not apply to current systems. Elsewhere ([HBD96]) we use a trace-driven simulation to show a wide range of conditions in which migrating active processes provides significant performance benefit. Based on these results, and similar results from simulations ([KL88]) and implemented systems ([BSW93]), we feel that the benefits of preemptive migration in current systems should be reexamined.

In this report we refer to migration of active processes as *preemptive migration* and to implicit placement of newborn processes as remote execution or *non-preemptive migration*.

2 System model

The system ELZ model differs from some of the systems that have attempted to apply their results. We feel that it is dangerous to extrapolate these results to dissimilar systems.

In ELZ's model, a batch of jobs arrives simultaneously at an unloaded system, and is distributed, at no cost, evenly among the processors in the system. The choice of batch arrivals is intended to model the most extreme case of a bursty arrival process. The decision to distribute the processes among the hosts at no cost is based on an implicit model of a server farm in which incoming jobs have no affinity for particular hosts.

But many current systems are based on a model of a network of workstations in which users generate intermittent jobs with a natural affinity for the host to which they are submitted, *i.e.* the cost of migrating them to another host, even by remote execution, is non-trivial.

In order to model this type of system, ELZ's model should be expanded to include this cost. If this cost is not included, the model will greatly overstate the benefits of non-preemptive migration. In fact, one of the primary reasons ELZ find the marginal benefit of preemptive migration so small is that their non-preemptive strategy is so successful. They write, "The benefits of [preemptive] migration are not limited by its cost, but rather by the inherent effectiveness of [non-preemptive migration]."

In a network of workstations model, non-preemptive migration is not inherently effective, but often difficult to do well, as demonstrated on implemented systems ([ZWZD93]). Thus, the relative benefit of preemptive migration might

be much greater.

3 Distribution of processes' CPU lifetimes

ELZ consider three distributions of process CPU lifetimes, and argue that the properties of these three distributions cover the range of distributions they expect to see in practice. In fact, the distributions we have observed ([HBD96]) are significantly different from these distributions and (as we explain below) it is not possible to extend ELZ's distributions to model our observed distributions.

The first distribution ELZ propose (and the one on which most of their results are based) is a hyperexponential distribution with two branches; one branch has probability p and mean lifetime S; the other has probability $1-p$ and constant lifetime 0. By setting $p = 1/S$, they construct a distribution with mean lifetime 1. By varying S, they vary the coefficient of variation of the distribution (CV).

$$1/p = S = \frac{1 + CV^2}{2}$$

Figure 1 shows the shape of ELZ's distribution of CPU lifetimes and compares it with the observed distribution from [HBD96]. The two distributions have the same mean (1 second) and coefficient of variation (CV) of 7. For this value of CV, $p = .04$ and $S = 25$. Thus, 96% of jobs in ELZ's distribution have lifetime zero; the other 4% are chosen from an exponential distribution with mean 25.

The following are properties of ELZ's distribution model:

mostly zero-length jobs The fraction of jobs in the distribution with zero lifetime is $1 - p = 1 - 1/S$. In the case where $CV = 7$ for example, this means 96% of all jobs have zero lifetime. These jobs do not affect the performance of the system or contribute to the performance metrics.

To see the effect of the prevalence of zero-length jobs, consider a network with 10 hosts and an initial load of 10 jobs per host. When $CV = 7$, only four jobs in the system (4%) will have non-zero lifetimes. It is no wonder that migration offers little benefit in this situation, since it is unlikely that any host will have more than one process with non-zero lifetime.

few short jobs By a "short job" we mean a job which consumes between zero and one second of CPU time. In the ELZ distribution, only $(1 - e^{-1/S})/S$ fraction of all jobs are short. For example when $CV = 7$, this means $(1 - e^{-1/25})/25 \approx .0016$ fraction of all jobs are short.

ELZ claim that the prevalence of jobs with zero lifetimes overestimates the benefits of preemptive migration, because "the faster small jobs exit

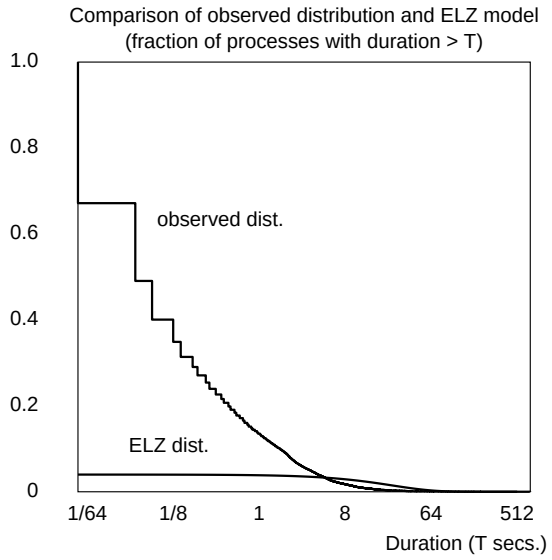


Figure 1: Comparison of a typical observed distribution with the ELZ model with the same mean and variance. The ELZ model includes many (96%) jobs with zero lifetime and few short jobs (lifetime between 0 and 1 seconds). The x axis is log scale.

the system and imbalancing occurs, the larger will be the benefits of migration."

The problem with this approach is that it eliminates from the performance metrics any benefit that preemptive migration might bestow on short jobs. But in [HBD96], we show that the most significant impact of preemptive migration is on short jobs.

The following observations explain the impact of preemptive migration on short jobs:

- If a long job stays at a busy host, it imposes slowdowns on many small jobs that arrive during its residence time (the severity of this slowdown depends on the local scheduling discipline and the particular behavior of the processes, *e.g.* I/O and interaction).
- Migrating that long job away helps not just the migrant job (by letting it run on a less-loaded host); it also helps the many short jobs that would have been slowed at the source host.[1]
- Thus, identifying long jobs is critical to the performance of short jobs. But no matter what information a non-preemptive scheme has about incoming processes, there will always be surprises — long jobs that were not migrated because they were expected to be short.
- Preemptive migration strategies can do a better job of predicting lifetimes (using the ages of processes and other collected information), and furthermore can always correct for poor predictions by migrating long jobs later.

Thus, contrary to ELZ's claim, preemptive migration provides great benefit to short jobs. By nearly eliminating small jobs from their distribution, ELZ eliminate this benefit of preemptive migration.

ELZ consider two other distributions: (1) a distribution in which jobs have lifetime either zero or S, with probabilities and values of S as above, and (2) a similar distribution that includes some processes with lifetime 1. These distributions also have the property that the majority of jobs have zero lifetime; thus neither addresses the problems we have raised.

In their paper ELZ warn, "different job service demand distributions that match with respect to both mean and variance may yield quite different results concerning the benefits of migration, thus caution is needed when developing workload models for use in migration studies." We agree wholeheartedly.

[1] Of course, it slows jobs at the target host, but there are expected to be far fewer arrivals at the target host, due to the serial correlation (burstiness) of arrivals.

4 Local scheduling

Different forms of local scheduling have a large effect on the benefits of migration. In our simulation studies we found that the extreme case of optimal local scheduling (executing the job with the shortest remaining lifetime) all but eliminates the need for migration, either preemptive or non-preemptive. [LO86] make a similar observation: "... the relative improvement from [preemptive migration] becomes more pronounced as the local scheduling policies become worse."

The reason for this effect is that under processor-sharing or round-robin scheduling, long-running jobs can impose slowdowns on many shorter jobs. Feedback scheduling mitigates this effect by lowering the priority of long-running jobs. When a new job arrives, it is given a burst of CPU time immediately; thus, many short-running jobs execute without delay, regardless of the number of longer-running jobs in queue.

Since the effect of local scheduling is large, a model of a migration system must include a model of local scheduling. There are, however, several aspects of ELZ's model that make this issue difficult to address:

- Since all jobs arrive at the same time in ELZ's model, there is no difference between processor-sharing and feedback scheduling. This aspect of the model conflicts with the observation that with continual arrivals, local scheduling has a large impact.
- Once the zero-length jobs complete, the remaining jobs have lifetimes from an exponential distribution. Thus, after time zero, all jobs have the same expected remaining lifetime regardless of how long they have run. But the effectiveness of feedback scheduling depends on the commonly-reported result that the expected remaining lifetime of a process increases monotonically with age.

The fact that ELZ's model is not affected by local scheduling, unlike real systems, makes us question the accuracy of their model.

5 Model artifacts

Some of the phenomena ELZ observe in their model are not descriptive of real systems; rather, they are artifacts of the model. For example, ELZ observe that the performance of preemptive migration (relative to non-preemptive alone) first increases as CV increases, peaks, and then decreases for higher values of CV.

The causes of this result, within ELZ's model, are explained below. We will argue that these causes do not appear in real systems, and therefore we suspect that the observed phenomenon is an artifact of the model.

1. **When CV is small** , there will be many processes with non-zero lifetime, but they will all have similar lifetimes. Thus, the initially-balanced allocation is likely to stay balanced.

   For example, consider a network with 10 hosts and an initial load of 5 jobs per host. If the CV is small, like 2, then S = 5/2 and p = 2/5. Thus 60% of the jobs have zero lifetime and the remaining 40% are chosen from an exponential distribution with mean 2.5. After the zero-length jobs complete, there will be (on average) 2 jobs on each host and the lifetimes of those jobs are likely to be similar. Thus, the system would be likely to stay balanced and the benefit of future migrations would be small.

2. **As the CV increases** , there are fewer jobs with non-zero lifetimes, and the variance in the lifetimes of those jobs is higher. Thus, the initial placement of jobs is likely to become unbalanced after the short jobs terminate. In this range, the benefits of preemptive migration are high.

3. **As the CV increases further** , the number of jobs with non-zero lifetimes becomes smaller than the number of hosts. In this case, the benefits of preemptive migration are small because the chance is small that any two jobs will share a host.

   For example, consider the same network as above, with a larger value of CV, like 5. In this case, $S = 13$ and $p = 1/13$. Of the original 50 processes, there are on average less than 4 with non-zero lifetimes. Half the time, each will be assigned to a different processor and preemptive migration will provide no benefit at all.

Thus the diminishing benefit of preemptive migration is a consequence of the large number of zero-length jobs in ELZ's workload description; it does not describe systems in which there are many short jobs (i.e. with non-zero lifetimes).

For large values of CV (greater than 5) the effect of jobs with lifetime zero is even more dominant, and the behavior of the model is even more remote from the behavior of real systems. Unfortunately, we have found that these are exactly the values of CV that are typical; the coefficient of variation in our workloads is consistently between 6 and 7.

6 Performance metrics

The primary metric ELZ use to evaluate performance is mean residence time. Mean residence time differs from the common alternative metric metric, mean slowdown, in that it is not normalized by the CPU lifetime of the job.

$$\text{mean slowdown} = \frac{1}{n}\sum_{i}^{n}\frac{\text{wallclock(i)}}{\text{cpulifetime(i)}}$$

$$\text{mean residence time} = \frac{1}{n}\sum_{i}^{n} \text{wallclock(i)}$$

This metric understates the benefit of preemptive migration enjoyed by short jobs. For example, if a job has a CPU lifetime of .01 seconds, and it runs on a processor with 100 other jobs, it will be slowed by a factor of 101, but it will only contribute 1 excess second (beyond its minimal contribution of .01 seconds) to the total residence time. At the same time, a job with a CPU lifetime of 100 seconds that shares a processor for 2 seconds, and thus takes 101 seconds, will also contribute one excess second. Thus, the slowdown imposed on small jobs will have almost no impact on ELZ's performance metrics; this omission is unfortunate because, as discussed above, short jobs are the main beneficiaries of preemptive migration.

7 Summary

The reason for this report is to suggest that the benefits of preemptive migration on current systems may in fact be greater than previously believed. This finding is contrary to ELZ, because:

- Under ELZ's system model, non-preemptive migration is able to achieve near-perfect load balance; thus, the additional benefit of preemptive migration is small. But this result may not apply to systems like networks of workstations that do not fit their model.

  ELZ use a system model in which jobs arrive at a server farm and have no affinity for particular hosts; thus the system can maintain balance by placing arrivals at hosts with low load. In this environment, non-preemptive migration is far more effective than it can be in an environment where jobs arrive at particular hosts and migration by remote execution has significant cost.

- ELZ use a workload description that has few short jobs (lifetimes greater than zero and less than one seconds). In [HBD96] we observed that short jobs are the primary beneficiaries of preemptive migration; thus ELZ ignore what we find to be a major benefit of preemptive migration — its effect on the short jobs.

- ELZ use a workload description that includes a majority of jobs with zero lifetime. This workload introduces artifacts that make it difficult to apply the results of their model to real systems.

In light of these observations we feel that the benefits of preemptive migration should be reexamined. Several recent systems have chosen to implement

preemptive migration for purposes other than load balancing (e.g. preserving autonomy). We would urge the developers of these systems to explore the benefits of load balancing by preemptive migration.

References

[BSW93] Amnon Barak, Guday Shai, and Richard G. Wheeler. *The MOSIX Distributed Operating System:Load Balancing for UNIX*. Springer Verlag, Berlin, 1993.

[ELZ88] Derek L. Eager, Edward D. Lazowska, and John Zahorjan. The limited performance benefits of migrating active processes for load sharing. In *SIGMETRICS*, pages 662–675, May 1988.

[HBD96] Mor Harchol-Balter and Allen B. Downey. Exploiting process lifetime distributions for dynamic load balancing. In *Proceedings ACM SIGMETRICS '96*, May 1996. Also appeared in Proceedings Fifteenth ACM Symposium on Operating System Principles Poster Session, December, 1995.

[KL88] Phillip Krueger and Miron Livny. A comparison of preemptive and non-preemptive load distributing. In *8th International Conference on Distributed Computing Systems*, pages 123–130, June 1988.

[LO86] W. E. Leland and T. J. Ott. Load-balancing heuristics and process behavior. In *Proceedings of Performance and ACM Sigmetrics*, volume 14, pages 54–69, 1986.

[ZWZD93] S. Zhou, J. Wang, X. Zheng, and P. Delisle. Utopia: a load-sharing facitlity for large heterogeneous distributed computing systems. *Software – Practice and Expeience*, 23(2):1305–1336, December 1993.