Task Assignment in a Distributed System: Improving Performance by Unbalancing Load

Mark E. Crovella*
Department of Computer Science
Boston University
Boston, MA 02215

Mor Harchol-Balter†
Laboratory for Computer Science
MIT, NE43-340
Cambridge, MA 02139

Cristina D. Murta‡
Department of Computer Science
Boston University
Boston, MA 02215

Abstract

We consider the problem of task assignment in a distributed system (such as a distributed Web server) in which task sizes are drawn from a heavy-tailed distribution. Many task assignment algorithms are based on the heuristic that balancing the load at the server hosts will result in optimal performance. We show this conventional wisdom is less true when the task size distribution is heavy-tailed (as is the case for Web file sizes). We introduce a new task assignment policy, called Size Interval Task Assignment with Variable Load (SITA-V). SITA-V purposely operates the server hosts at different loads, and directs smaller tasks to the lighter-loaded hosts. The result is that SITA-V provably decreases the mean task slowdown by significant factors (up to 1000 or more) where the more heavy-tailed the workload, the greater the improvement factor. We evaluate the tradeoff between improvement in slowdown and increase in waiting time in a system using SITA-V, and show conditions under which SITA-V represents a particularly appealing policy.

1 Introduction

Increasingly, high performance servers are being implemented as distributed systems. We are concerned with the problem of task assignment in a such systems. While load balancing is generally considered a desirable feature of task assignment policies, in this work we explore benefits that can be obtained by unbalancing load — that is, operating different hosts in the distributed system at different utilization levels.

*Supported in part by NSF Grants CCR-9501822 and CCR-9706685. Email: crovella@cs.bu.edu.

†Supported by the NSF Postdoctoral Fellowship in the Mathematical Sciences. Email: harchol@theory.lcs.mit.edu.

‡Supported by a grant from CAPES, Brazil. Permanent address: Depto. de Informática, Universidade Federal do Paraná, Curitiba, PR 81531, Brazil. Email: cristina@dcc.ufmg.br.

The model we consider is motivated by the problem of HTML page retrieval from a cluster-based Web server. We assume a server consisting of a collection of hosts all employing the processor-sharing scheduling discipline. Task service times are drawn from a known distribution, and a task's size is known to the system at the time of task arrival. These assumptions follow from the fact that the large majority of HTML requests are for static files, whose sizes are known.

Our analysis focuses on task sizes that are drawn from a *heavy-tailed* distribution. A heavy-tailed distribution is one whose tail declines like a power-law, that is, $P[X > x] \sim x^{-\alpha}$ for $0 < \alpha \leq 2$. Heavy-tailed distributions are increasingly being observed in a wide range of computer workloads, including Web file sizes.

The metrics by which we judge system performance are user-oriented metrics: mean waiting time and mean slowdown, where slowdown is the ratio of a task's waiting time to its service demand. System designers have often focused on developing policies that minimize mean waiting time. However, in this paper we focus on minimizing slowdown, because slowdown translates more directly to user-perceived performance.

We show that when task sizes are heavy-tailed, policies that balance load can in fact do a poor job of minimizing slowdown. We demonstrate this fact by introducing a new task assignment policy for distributed servers called **SITA-V** (Size Interval Task Assignment with Variable Load) which operates different hosts at different loads. SITA-V exploits the heavy-tailed task size distribution by running the overwhelming majority of tasks (all small-sized tasks) on hosts which are loaded below the average system load and running the tiny minority (all large-sized tasks) on hosts which are loaded above the average system load. We show that using this strategy, SITA-V can reduce mean task slowdown to levels far below that of straightforward load-balancing policies.

2 Summary of Results

We assume task sizes are drawn from a heavy-tailed distribution, but have some minimum size k and some maximum size p, as would be expected in practice. Thus the probability mass function is defined as

$$f(x) = \frac{\alpha k^\alpha}{1-(k/p)^\alpha} x^{-\alpha-1} \quad k \leq x \leq p. \qquad (1)$$

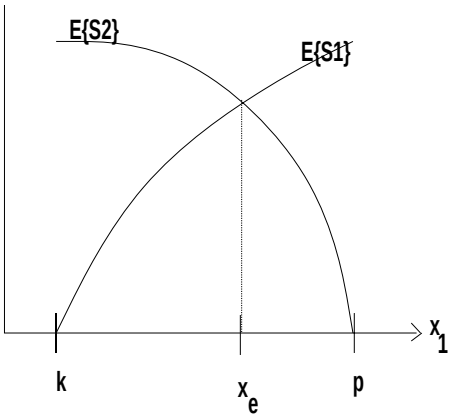

Figure 1: $\mathbf{E}\{S_1\}$ and $\mathbf{E}\{S_2\}$ shown as a function of the cutoff x_1. The point $x_1 = x_e$ depicts balanced load.

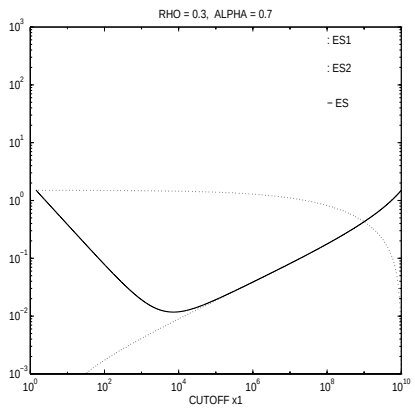

Figure 2: $\mathbf{E}\{S_1\}$, $\mathbf{E}\{S_2\}$, and $\mathbf{E}\{S\}$ under SITA-V_2 as a function of the cutoff x_1 for $\alpha = 0.7, \rho = 0.3$.

Heavy-tailed distributions have the property that while the majority of tasks are very small, more than half of the total work is made up by a tiny fraction ($< 1\%$) of the largest tasks. We will refer to this as the *heavy-tailed property*, and will exploit this property in our assignment of tasks. The lower the α parameter, the more pronounced is the heavy-tailed property.

We evaluate the effectiveness of the SITA-V task assignment scheme in terms of mean waiting time, $\mathbf{E}\{W\}$, and mean slowdown, $\mathbf{E}\{S\}$. Waiting time (W) is the time from when the task arrives at the system until the task leaves the system, minus its service time. Slowdown (S) is a task's waiting time divided by its size (execution time).

The basic idea behind SITA-V can be shown using a 2 host system (SITA-V_2). Consider a 2-host system in which tasks of size less than x_1 are sent to host 1, and tasks of size greater than x_1 are sent to host 2. Let p_i $(i = 1, 2)$ be the fraction of tasks that are assigned to host i, and let S_i be the slowdown for tasks assigned to host i. Then:

$$\mathbf{E}\{S\} = p_1\mathbf{E}\{S_1\} + p_2\mathbf{E}\{S_2\} \qquad (2)$$

where the values of p_i and $\mathbf{E}\{S_i\}$ depend on the cutoff point x_1. There is a point x_e such that when $x_1 = x_e$, we have $\mathbf{E}\{S_1\} = \mathbf{E}\{S_2\} = \mathbf{E}\{S\}$ (load is balanced). Figure 1 depicts $\mathbf{E}\{S_1\}$ and $\mathbf{E}\{S_2\}$ as a function of the cutoff, x_1. $\mathbf{E}\{S_1\}$ increases as the cutoff x_1 is increased, and $\mathbf{E}\{S_2\}$ decreases as x_1 is increased.

The key to the success of SITA-V lies in the heavy-tailed property of the task size distribution. The heavy-tailed property implies that at the equilibrium point $x_1 = x_e$ almost all tasks are being served by host 1. If the distribution is sufficiently heavy-tailed, then moving the cutoff point x_1 a little to the left (and thus lessening the load on host 1) will not change the fact that most tasks are still being served by host 1. However, since the utilization for host 1 decreases as x_1 is moved left, the mean (per-task) slowdown decreases as well. This effect is shown graphically in Figure 2. The figure shows how $\mathbf{E}\{S\}$ varies as a function of the cutoff x_1. The possible improvement in slowdown of the optimal x_1 over the $x_1 = x_e$ case is a factor of approximately 100.

In the full paper [1] we describe how this idea can be extended to multiple hosts, and we evaluate the performance of the resulting algorithm (SITA-V) over a range of system sizes, α values, and system utilizations (ρ).

In improving mean slowdown, observe that SITA-V does not sacrifice throughput. The overall system utilization (ρ) is not changed; load is merely shifted among hosts. However, SITA-V achieves its result at some cost: an increase in mean task waiting time. In the full paper we evaluate the tradeoff between reducing slowdown and increasing waiting time in a system employing SITA-V. We show that the two factors that determine the nature of the tradeoff are the variability of tasks (as measured by the exponent α in the task size distribution) and the overall system utilization ρ. In general, we find that when task sizes are highly variable ($\alpha < 1$) SITA-V can result in remarkable improvements in slowdown — by factors as great as 1000 or more. However we also show that under some conditions, the improvements under SITA-V can come at a high cost in increased waiting time. We conclude that SITA-V is a particularly appealing policy when α is small; in that case SITA-V results in significant improvements in slowdown while imposing relatively smaller costs in terms of additional waiting time.

3 Conclusion

In this paper we've shown that for distributed systems with heavy-tailed workloads, mean slowdown can be improved by large factors by adopting a policy that does not balance load. We have introduced a new policy, SITA-V, that reduces mean slowdown to levels far below those that a balanced-load policy achieves. The key idea of SITA-V is to direct the small but numerous tasks to lightly-loaded hosts, while sending the large but rare tasks to heavily-loaded hosts.

References

[1] M. E. Crovella, M. Harchol-Balter, and C. D. Murta. Task assignment in a distributed system: Improving performance by unbalancing load. Technical Report TR-97-018, Boston University Department of Computer Science, October 31 1997.