Pull versus Push Mechanism in Large Distributed Networks: Closed Form Results

Wouter Minnebo and Benny Van Houdt

Department of Mathematics and Computer Science
University of Antwerp - IBBT
Middelheimlaan 1, B-2020 Antwerp, Belgium
Email:{wouter.minnebo,benny.vanhoudt}@ua.ac.be

Abstract—In this paper we compare the performance of the pull and push strategy in a large homogeneous distributed system. When a pull strategy is in use, lightly loaded nodes attempt to steal jobs from more highly loaded nodes, while under the push strategy more highly loaded nodes look for lightly loaded nodes to process some of their jobs.

Given the maximum allowed overall probe rate R and arrival rate λ, we provide closed form solutions for the mean response time of a job for the push and pull strategy under the infinite system model. More specifically, we show that the push strategy outperforms the pull strategy for any probe rate $R > 0$ when $\lambda < \phi - 1$, where $\phi = (1 + \sqrt{5})/2 \approx 1.6180$ is the golden ratio. More generally, we show that the push strategy prevails if and only if $2\lambda < \sqrt{(R+1)^2 + 4(R+1)} - (R+1)$. We also show that under the infinite system model, a hybrid pull and push strategy is always inferior to the pure pull or push strategy.

The relation between the finite and infinite system model is discussed and simulation results that validate the infinite system model are provided.

I. INTRODUCTION

Distributed networks typically consist of a set of nodes interconnected through a network, each equipped with a single server to process jobs. Jobs may enter the network via one or multiple central dispatchers (e.g., [1], [2]) or via the processing nodes themselves (e.g., [3], [4], [5], [6]). In the former case the dispatchers will distribute the jobs among the nodes using some load balancing algorithm. In the latter case, lightly loaded nodes may attempt to take/steal/pull jobs from more highly loaded nodes or highly loaded nodes may try to forward/push some of their pending jobs to lightly loaded nodes. When the initiative is taken by the lightly loaded nodes only, we say that a *pull* strategy is used. If only the highly loaded nodes initiate the exchange of jobs, we say that a *push* strategy is used. Pull strategies are often called work stealing schemes, while push strategies are sometimes called work sharing solutions.

To facilitate the exchange of jobs some central information may be stored. Though as the network size grows continuously updating this information becomes more challenging. Therefore, fully distributed networks do not rely on centralized information. Instead a node that wishes to pull/push a job will transmit a probe to one (or multiple) other nodes that are typically selected at random. If a node that receives such a probe is willing to take part in the exchange, it will send a

positive reply and the job exchange can proceed. Clearly, the overall rate at which probe messages are sent by a strategy plays a pivotal role in its effectiveness.

The performance of both push and pull strategies has been studied by various authors. A comparison for a homogeneous distributed system with Poisson arrivals and exponential job lengths was presented in [3]. The approach was based on a decoupling assumption and relied on numerical methods to solve some non-linear equation. Numerical examples showed that pull strategies achieve a lower mean response time under high loads, while push strategies are superior under low to medium loads. Similar observations were made for heterogeneous systems in [7] again by relying on a decoupling assumption. A similar approach to study the influence of task migrations in shared-memory multi-processor systems was presented in [8]. In each of these papers, nodes send out probes as soon as the number of jobs drops below some threshold $T \geq 1$ in case of the pull strategy, while for the push strategy probes are sent whenever the number of jobs in the queue is at least T upon arrival of a new job. When job transfer and signaling delays are assumed negligible setting $T = 1$ is optimal [7], [4]. Another common feature in these papers is that the number of nodes N is not a model parameter, instead they provide a numerical approach for a system with $N = \infty$, which we will call the infinite system model.

Although the insights provided by these comparisons are very valuable, the strength of the push and pull mechanism was only compared to some extent, mainly because the overall probe rate R of both strategies may be very different (and depends on the load $\lambda < 1$). This is especially true for the hybrid pull/push strategy introduced in [4], where it has been shown to outperform both the push and pull strategy for all loads λ. However, as indicated in [4], such a hybrid strategy results in a (far) higher probe rate R. Our aim in this paper is to compare the pull and push mechanism given that both generate the same overall probe rate R. Further, we also provide closed form expressions for the main performance measures under the infinite system model.

To this end we introduce a slightly different pull and push strategy, where probes are not transmitted at job completion or job arrival times. Instead idle nodes will generate probes at some rate r under the pull strategy, while under the push

strategy nodes send probes at some rate r whenever they have jobs waiting. A desirable property of both these strategies is that they can match any overall probe rate R under any load λ by setting r in the appropriate manner. More specifically, let R be the average number of probes send by a node per time unit, irrespective of its queue length. Clearly, the overall probe rate R will be less than the rate r. By establishing a simple relationship between r and R, we will determine r to match any predefined overall probe rate R.

Given some $R > 0$, we will show that under the infinite system model the push strategy outperforms the pull strategy for any

$$\lambda < \frac{\sqrt{(R+1)^2 + 4(R+1)} - (R+1)}{2},$$

in terms of the mean response time (as well as in the decay rate of the queue length distribution). As R approaches zero, the right-hand side decreases to $\phi - 1$, where $\phi = (1+\sqrt{5})/2$ is the golden ratio, which indicates that the push strategy prevails for any R when $\lambda < (1 + \sqrt{5})/2 \approx 0.6108$.

We also consider a hybrid strategy where idle nodes probe at rate r_1 and nodes with pending jobs probe at rate r_2, where $r = r_1 + r_2$ is again determined by matching R, leaving one degree of freedom. We will show that for any for λ and R the optimal policy exists in setting either r_1 or r_2 to zero. This implies that the hybrid strategy is in fact never better when the overall probe rate R is not allowed to increase.

The infinite system model corresponds to a distributed system with an infinite number of nodes N. Using simulation results, we will show that the infinite system is quite accurate for both strategies and moderate to large size systems, e.g., for $N \geq 100$ the relative error is typically below 1 percent. For smaller systems, e.g., $N = 25$, the infinite system model results in higher relative errors, especially for the pull strategy under high loads. The loads for which the push strategy outperforms the pull strategy are however still quite accurately predicted by the infinite system model, even for small systems.

The paper is structured as follows. In Section II we introduce the push, pull and hybrid strategy considered in this paper and discuss its relation with many existing strategies studied before. The infinite system model is presented in Section III and closed form results are derived for the queue length distribution and mean delay. Using these results we identify the loads at which the push strategy outperforms the pull strategy and prove that a hybrid strategy is always inferior. Simulation results that validate the infinite system model are presented in Section IV. In Section V we discuss some technical issues related to proving that the infinite system model is indeed the proper limit process of the sequence of finite system models. Conclusions are drawn and future work is discussed in Section VI.

II. Pull and Push Strategies

We consider a continuous-time system consisting of N queues, where each queue consists of a single server and an infinite buffer. As in [3], [9], [7], [5]: each queue is subject to its own local Poisson arrival process with rate λ, jobs require an exponential processing time with mean 1 and are served in a first-come-first-served (FCFS) order. We also assume that the time required to transfer probe messages and jobs between different nodes can be neglected in comparison with the processing time (i.e., the transfer times are assumed to be zero). We consider the following three basic strategies:

1) *Push:* Whenever a node has $i \geq 2$ jobs in its queue, meaning $i - 1$ jobs are waiting to be served, the node will generate probe messages at rate r. Thus, as long as the number of jobs in the queue remains above 1, probes are sent according to a Poisson process with rate r. Whenever the queue length i drops to 1, this process is interrupted and will remain interrupted as long as the queue length remains below 2. The node that is probed is selected at random and is only allowed to accept a job if it is idle.

2) *Pull:* Whenever a node has $i = 0$ jobs in its queue, meaning the server is idle, the node will generate probe messages at rate r. Thus, as long as the server remains idle, probes are sent according to a Poisson process with rate r. This process is interrupted whenever the server becomes busy. The probed node is also selected at random and the probe is successful if there are jobs waiting to be served.

3) *Hybrid:* This strategy combines the above two strategies. When queue length i equals 0 a node generates probes at rate r_1, while for $i \geq 2$ the probe rate is set equal to r_2.

We will show that under the infinite system model (i.e., $N = \infty$) the push and pull strategy result in exactly the same queue length distribution when the same rate r is used. This is even true for the hybrid strategy if we define $r = r_1 + r_2$ (i.e., the queue length only depends on the sum of r_1 and r_2). However, when the same rate r is used by these different strategies, the overall probe rate R will typically differ. Hence, we aim at comparing these strategies when the rates r are set such that the overall probe rate matches some predefined R.

The pull strategy considered in this paper is in fact identical to the pull strategy with repeated attempts considered in [9, Section 2.4], except that our nodes do not immediately generate a probe message when the server becomes idle. Generating probes in that way would automatically result in a high probe rate R when the load λ is small and would no longer allow us to match any $R > 0$ by setting r in the appropriate manner.

The traditional pull and push strategies considered in [3], [7], [4] are somewhat different. The pull strategy tries to attract a job whenever a job completes and the resulting queue length is below T, while the push strategy tries to push arriving jobs that find T or more jobs in the queue upon arrival. Further, instead of sending a single probe, both strategies send a batch of L_p probes, in the hope that one gets a positive reply. The overall probe rate R clearly depends on T, L_p and the load λ, which makes it hard to compare the pull and push strategies

in a completely fair manner.

When the time required to transfer probes and jobs between nodes is neglected (as in [3]), setting $T = 1$ ensures that exchanged jobs can immediately start (as in our setup). Assuming zero transfer time for probe messages is quite realistic as transferring jobs typically requires considerably more time than sending a probe. The models in [7], [4] do take an exponentially distributed job transfer time into account (while still assuming zero transfer time for the probes). The results show that when increasing the transfer time, the delays also increase, while the performance differences between the pull and push strategies become less significant (but remain similar). Further, the setting $T = 1$ minimizes the mean response time when the job transfer times are sufficiently small (but also results in a higher overall probe rate R).

The strategies considered in [5], [6] are more aggressive pull and push strategies. In [5] a successful probe message results in exchanging half of the jobs that are waiting, while in [6] the number of probes send under the push strategy depends on the current queue length. Although the push strategy of [6] significantly reduces the mean response time and outperforms the pull strategy, its overall probe rate is also much higher.

III. INFINITE SYSTEM MODEL

In this section we present various analytical results in closed form for the system with $N = \infty$ nodes, termed the infinite system model. The evolution of the infinite system model will be captured by a set of ordinary differential equations (ODEs); hence, the infinite system model is deterministic as opposed to the finite system models where $N < \infty$. To define the infinite system we first consider a system with a finite number of nodes N. Due to the assumptions on the arrival process, processing times and transfer times, it suffices to keep track of the N queue lengths in order to obtain a continuous time Markov chain (CTMC). Further, as the system is homogeneous, it also suffices to keep track of the number of nodes that have i jobs in their queue for all $i \geq 1$. More precisely, we define a CTMC $\{X^{(N)}(t) = (X_1^N(t), X_2^N(t), \ldots)\}_{t \geq 0}$, where $X_i^{(N)}(t) \in \{0, \ldots, N\}$ is the number of nodes with *at least* i jobs in the queue at time t (the superscript N is used to indicate that we consider a finite system consisting of N nodes). For any state $x = (x_1, x_2, \ldots)$ we clearly have that $x_i \geq x_{i+1}$ for all $i \geq 1$.

Let us first indicate that the transition rates of this CTMC are identical for the push, pull and hybrid strategy provided that they use the same rate r. Transitions take place when either one of the following three events takes place: an arrival, a job completion, or a job exchange between an idle node and a node with at least two jobs. Let $q^{(N)}(x, y)$ be the transition rate between state $x = (x_1, x_2, \ldots)$ and state $y = (y_1, y_2, \ldots)$. If an arrival occurs in a queue with $i - 1$ jobs, then x_i will increase by one. Thus, due to the arrivals we have

$$q^{(N)}(x, y) = \lambda(x_{i-1} - x_i),$$

for $y = x + e_i$ and $i \geq 1$, where $x_0 = 1$ and e_i is a vector with a 1 in position i and 0s elsewhere. Similarly, a job completion

in a queue with i jobs reduces x_i by one:

$$q^{(N)}(x, y) = (x_i - x_{i+1}),$$

for $y = x - e_i$ and $i \geq 1$. A job exchange between an idle node and a node with i jobs increases x_1 by one and decreases x_i by one; hence, $y = x + e_1 - e_i$. Under the push strategy the rate of such exchanges equals the number of nodes with exactly i jobs $x_i - x_{i+1}$ times r times the probability that a probe message is successful[1], which equals $(N - x_1)/N$. Hence, for the push strategy we have

$$q^{(N)}(x, y) = r(1 - x_1/N)(x_i - x_{i+1}),$$

for $y = x + e_1 - e_i$ and $i \geq 2$. Under the pull strategy this event takes place with a rate equal to the number of idle nodes $(N - x_1)$ times r times the probability $(x_i - x_{i+1})/N$ that we select a node with i jobs. The transition rate is therefore the same in both systems. For the hybrid strategy these events occur at rate $r_1(1 - x_1/N)(x_i - x_{i+1})$ (due to pull) plus $r_2(1 - x_1/N)(x_i - x_{i+1})$ (due to the push), which results in the same overall rate. Using a coupling argument one can prove that this CTMC is positive recurrent for all $\lambda < 1$ (as the overall mean queue length is dominated by the overall mean queue length of a set of N independent M/M/1 queues).

We will now define the infinite system model, the evolution of which is described by a set of ODEs, using the rates $q^{(N)}(x, y)$. As these rates are the same for the three strategies, they also result in the same set of ODEs. Define $\beta_\ell(x/N) = q^{(N)}(x, x + \ell)/N$, such that

$$\beta_{e_i}(x/N) = \lambda(x_{i-1}/N - x_i/N),$$
$$\beta_{-e_i}(x/N) = (x_i/N - x_{i+1}/N),$$

for all $i \geq 2$ and

$$\beta_{e_1 - e_i}(x/N) = r(1 - x_1/N)(x_i/N - x_{i+1}/N),$$

for $i \geq 2$. Denote by

$$F(x) = \sum_{i \geq 1}(e_i\beta_{e_i}(x) - e_i\beta_{-e_i}(x)) + \sum_{i \geq 2}(e_1 - e_i)\beta_{e_1-e_i}(x),$$

where $x = (x_1, x_2, \ldots)$, with $x_i \in [0, 1]$ and $x_i \geq x_{i+1}$ for $i \geq 1$. The set of ODEs describing the evolution of the infinite system model is now given by $\frac{d}{dt}x(t) = F(x(t))$, where $x_i(t)$ represents the fraction of the number of nodes with at least i jobs at time t in the infinite system. This set of ODEs can be written as

$$\frac{d}{dt}x_1(t) = (\lambda + rx_2(t))(1 - x_1(t)) - (x_1(t) - x_2(t)), \quad (1)$$

and

$$\frac{d}{dt}x_i(t) = \lambda(x_{i-1}(t) - x_i(t))$$
$$- (1 + r(1 - x_1(t)))(x_i(t) - x_{i+1}(t)), \quad (2)$$

[1]We assume that the probed node is selected at random, in fact we even allow a node to select itself with probability $1/N$. Disallowing nodes to select themselves results in the same limiting process.

for $i \geq 2$. In Section V we will discuss the relation between this dynamical system and the finite system models for large N.

Let $E = \{(x_1, x_2, \ldots)|x_i \in [0,1], x_i \geq x_{i+1}, i \geq 1, \sum_{j\geq 1} x_j < \infty\}$. The next two theorems show that this set of ODEs is Lipschitz on E and it has a unique fixed point in E.

Theorem 1. *The function F is Lipschitz on E.*

Proof: F is Lipschitz provided that for all $x, y \in E$ there exists an $L > 0$ such that $|F(x) - F(y)| \leq L|x - y|$. By definition of $F(x)$ one finds

$$|F(x) - F(y)| \leq 2(\lambda + 1 + 2r)|x - y| +$$
$$2r \sum_{i\geq 2} |x_1(x_i - x_{i+1}) - y_1(y_i - y_{i+1})|.$$

The above sum can be bounded by

$$\sum_{i\geq 2} |(x_1 - y_1)(x_i - x_{i+1}) + y_1(x_i - x_{i+1} - y_i + y_{i+1})|,$$

which is bounded by $2|x - y|$ on E. Hence, F is Lipschitz by letting $L = 2\lambda + 2 + 8r$. ∎

As E is a Banach space the Lipschitz condition of F suffices to guarantee that the set of ODEs $\frac{d}{dt}x(t) = F(x(t))$, with $x(0) \in E$, has a unique solution[2] $\phi_t(x(0))$ [10, Section 1.1].

Theorem 2. *The set of ODEs given by* (1) *and* (2) *has a unique fixed point $\pi = (\pi_1, \pi_2, \ldots)$ with $\sum_{i\geq 1} \pi_i < \infty$. Further,*

$$\pi_i = \lambda \left(\frac{\lambda}{1 + (1 - \lambda)r}\right)^{i-1}.$$

Proof: Assume π is a fixed point with $\sum_{i\geq 1} \pi_i < \infty$, meaning $F_i(\pi) = 0$ for $i \geq 1$, where $F(x) = (F_1(x), F_2(x), \ldots)$. When $\sum_{i\geq 1} \pi_i < \infty$, we can simplify $\sum_{i\geq 1} F_i(\pi) = 0$ to $\lambda - \pi_1 = 0$. Hence, π_1 must equal λ. Further, by defining $\eta_i = \pi_i - \pi_{i+1}$, the condition $F_i(\pi) = 0$, for $i \geq 2$, readily implies that $\eta_{i+1} = \lambda\eta_i/(1 + (1 - \pi_1)r)$ and therefore by induction we find the expression for π_i, for $i \geq 2$. ∎

If we take the set of ODEs in (1) and (2) and replace the first $x_2(t)$ by π_2 in (1) and $x_1(t)$ by λ in (2), then we end up with the Kolmogorov differential equation for a state dependent M/M/1 queue with $\lambda_0 = \lambda + r\pi_2$, $\lambda_i = \lambda$, for $i \geq 1$, $\mu_1 = 1$ and $\mu_i = 1 + (1 - \lambda)r$, for $i \geq 2$. The arrival process of such an M/M/1 queue is Poisson with rate λ_i and the service is exponential with rate μ_i whenever the queue length equals i. Hence, the fixed point π also corresponds to the steady state of a state dependent M/M/1 queue (where π_i is the probability that the queue contains at least i packets).

The set of ODEs in (1) and (2) describes the transient evolution of the infinite system, while we are in fact interested in its behavior as t goes to infinity. Thus, we are interested in the limit of all the trajectories of this set of ODEs. Numerical

[2]The solution $\phi_t(x)$ belongs to the class of continuously differentiable functions as in the finite dimensional case.

experimentation indicates that the trajectories starting from an arbitrary point $x \in E$ converge towards the unique fixed point π. Moreover, larger r and smaller λ values seem to result in faster convergence. To formally prove that π is a global attractor in E, one needs to define an appropriate Lyapunov function, which seems hard (except for $r = 0$). We have been able to prove a slightly weaker result that indicates that the L_1-distance on these trajectories cannot increase, meaning π is a stable fixed point. This is similar to the set of ODEs considered in [11] for which stability, but not global attraction, is shown. We will therefore rely on the following assumption throughout the paper.

Assumption A1: All the trajectories of the set of ODEs given by (1) and (2), starting from $x \in E$ converge towards the unique fixed point π.

Under the assumption A1, we can now express the main performance measures of the pull, push and hybrid strategy via Theorem 2:

Corollary 1. *Under assumption A1, the mean response time D of a job under the push, pull and hybrid strategy equals*

$$D = 1 + \frac{\lambda}{(1 - \lambda)(1 + r)}.$$

Under the hybrid strategy the overall probe rate R can be expressed as

$$R = (1 - \lambda)r_1 + \frac{\lambda^2 r_2}{1 + (1 - \lambda)r},$$

with $r = r_1 + r_2$. Setting $(r_1, r_2) = (r, 0)$ and $(0, r)$ results in the probe rate R of the pull and push strategy, respectively.

Proof: The mean response time D can be expressed as $\sum_{i\geq 1} \pi_i/\lambda = 1 + \lambda/(1 + (1 - \lambda)r - \lambda)$ by Little's law. The overall probe rate under the pull and push strategy equals $r(1 - \pi_1)$ and $r\pi_2$, respectively. Under the hybrid strategy the overall probe rate equals $r_1(1 - \pi_1) + r_2\pi_2$. ∎

Our interest lies in comparing the mean response time D of the three policies given λ and the overall allowed probe rate R. Using the above result, we can easily set r such that the overall probe rate equals some predefined R. For the hybrid policy this still leaves one degree of freedom as only the sum of $r_1 + r_2$ has been determined. The above result also indicates that R converges to $\lambda^2/(1 - \lambda)$ as r goes to infinity under the push strategy (which is in contrast to the pull strategy where R also goes to infinity). This indicates that an overall probe rate R close to $\lambda^2/(1 - \lambda)$ suffices to get a mean response time close to 1 under the push strategy. We should however also note that this rate R becomes large as λ approaches one.

Theorem 3. *Under assumption A1, the mean response time D of a job under the push strategy equals*

$$D_{push} = \frac{\lambda}{(1 - \lambda)(\lambda + R)},$$

for $R < \lambda^2/(1 - \lambda)$ and $D_{push} = 1$ for $R \geq \lambda^2/(1 - \lambda)$. Under the pull strategy we get

$$D_{pull} = \frac{1 + R}{1 - \lambda + R}.$$

Hence, given λ the push strategy outperforms the pull strategy if and only if $(1 + R) > \lambda^2/(1 - \lambda)$ and given R the push is the best strategy if and only if

$$\lambda < \frac{\sqrt{(1 + R)^2 + 4(1 + R)} - (1 + R)}{2}.$$

Further, the push strategy outperforms the pull strategy for all $\lambda < \phi - 1$, where $\phi = (1 + \sqrt{5})/2$ is the golden ratio.

Proof: The expressions for D_{push} and D_{pull} are readily obtained from Corollary 1 by plugging in the appropriate value for r in the expression for D. Requiring that $D_{push} = D_{pull}$ results in a quadratic equation for R with roots in 0 and $\lambda^2/(1 - \lambda) - 1$, which results in the condition for $(1 + R)$ and λ. The last result is obtained by noting that $\sqrt{(1 + R)^2 + 4(1 + R)}/2 - (1 + R)/2$ is an increasing function in R and its limit for R going to zero equals $\sqrt{5}/2 - 1/2$. ∎

Looking at the expression for the mean delay in Corollary 1, we note that a strategy with a lower mean response time actually has a larger r value when matching R. By Theorem 2 we also know that the queue length distribution decays geometrically with parameter $\lambda/(1 + (1 - \lambda)r)$. Hence, a smaller mean delay therefore also implies a faster decay of the queue length distribution. In fact, in this case a smaller mean delay even implies that the queue length distribution becomes smaller in the usual stochastic ordering sense [12].

We observe another fundamental difference between the push and pull strategy when the load approaches 1. In this case the mean delay of the push strategy still goes to infinity as in the M/M/1 queue (the mean response time of which is $1/(1 - \lambda)$). For the pull strategy the mean delay approaches $1 + 1/R$, hence remains finite. We should note that r does go to infinity when λ approaches 1 under the pull strategy (for any $R > 0$).

Theorem 4. *Under assumption A1, the mean delay under the hybrid strategy (r_1, r_2) with overall probe rate R is minimized by setting r_1 or r_2 equal to zero. Hence, a pure pull or push strategy is always optimal.*

Proof: Let R_1 and $R_2 = R - R_1$ be the overall probe rate generated by the pull and push operations, respectively. By Corollary 1, we have $R_1 = (1 - \lambda)r_1$ and $R_2 = \lambda^2 r_2/(1 + (1 - \lambda)r)$, while we also note that D is minimized by maximizing $r = r_1 + r_2$. Hence, by letting $R_2 = y$ and $R_1 = R - y$, we wish to maximize

$$g(y) = \frac{R - y}{1 - \lambda} + \frac{y(1 + R - y)}{\lambda^2 - (1 - \lambda)y},$$

for $y \in [0, R]$ and $R < \lambda^2/(1 - \lambda)$. For $R \geq \lambda^2/(1 - \lambda)$ the response time is minimized by setting $r_1 = 0$ as $D_{push} = 1$.
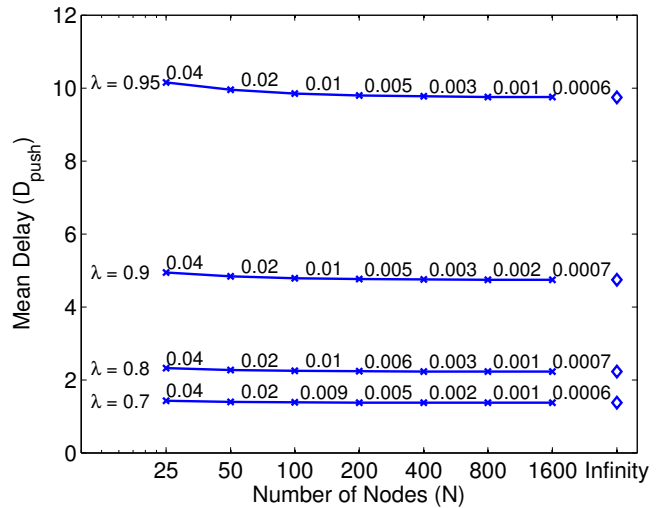

Figure 1. Mean delay and relative error of the push strategy in a finite vs. infinite system for $R = 1$

Some basic algebraic manipulations show that

$$\frac{d}{dy}g(y) = \left((1 + R) - \frac{\lambda^2}{1 - \lambda}\right)\left(\frac{\lambda}{\lambda^2 - (1 - \lambda)y}\right)^2,$$

on $y \in [0, R]$ with $R < \lambda^2/(1 - \lambda)$. Depending on the sign of $(1 + R) - \lambda^2/(1 - \lambda)$ the derivative of $g(y)$ is therefore positive or negative on the entire interval and the minimum is found in $y = 0$ (i.e., $r_2 = 0$) or $y = R$ (i.e., $r_1 = 0$). ∎

IV. MODEL VALIDATION

In this section we validate the infinite system model by comparing the closed form results of Theorem 3 with time consuming simulation results for systems with a finite number of nodes N. The infinite and finite system model only differ in the system size. Hence, the rate r in the simulation experiments is independent of N and was determined by λ and R using the expression for R in Corollary 1. Each simulated point in the figures represents the average value of 25 simulation runs. Each run has a length of 10^6 (where the service time is exponentially distributed with mean 1) and a warm-up period of length $10^6/3$.

Figure 1 compares the mean delay in a finite system with N nodes with the mean delay in the infinite system model under the push strategy with $R = 1$ for $N = 25, 50, \ldots, 1600$ and $\lambda = 0.7, 0.8, 0.9$ and 0.95. For each combination of N and λ we also show the relative error. The error clearly decreases to zero as N goes to infinity. Further, even for a system with $N = 100$ nodes we observe a relative error of 1% only. It may seem unexpected that the relative error is nearly insensitive to the load, as one might expect higher errors as λ increases. In fact, if r is kept fixed we would observe an increased error. However, we are looking at the curves for $R = 1$, meaning $r = 1/(\lambda^2 - (1 - \lambda))$ decreases with λ (see Corollary 1). As setting $r = 0$ gives exact results for any finite N, we can expect an improved accuracy for smaller r values (if λ remains fixed). Thus, in Figure 1 we see more or less the
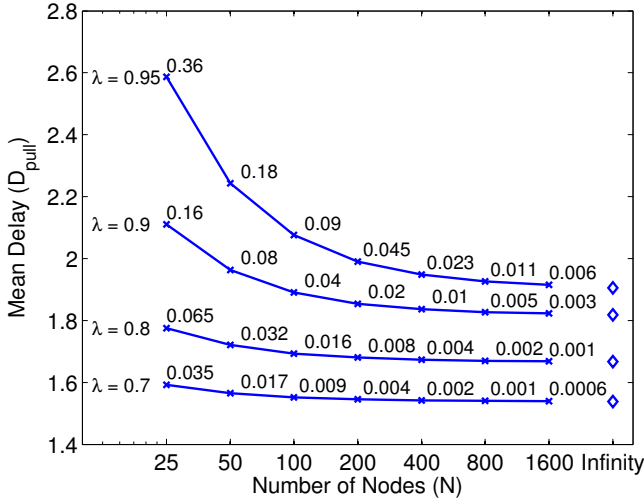
Figure 2. Mean delay and relative error of the pull strategy in a finite vs. infinite system for $R = 1$.

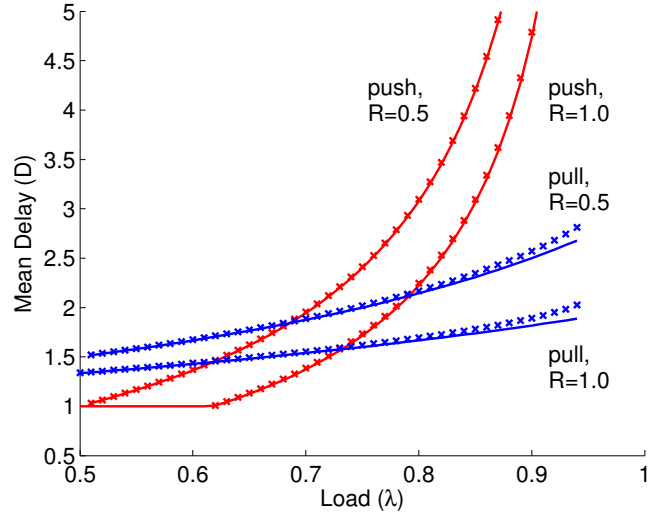Figure 3. Overall request rate of the push strategy in a finite vs. infinite system for $R = 1$.

Figure 4. Mean delay of the push and pull strategy in a finite system with $N = 100$ nodes (crosses) vs. infinite system model (full lines) for $R = 0.5$ and $R = 1$.
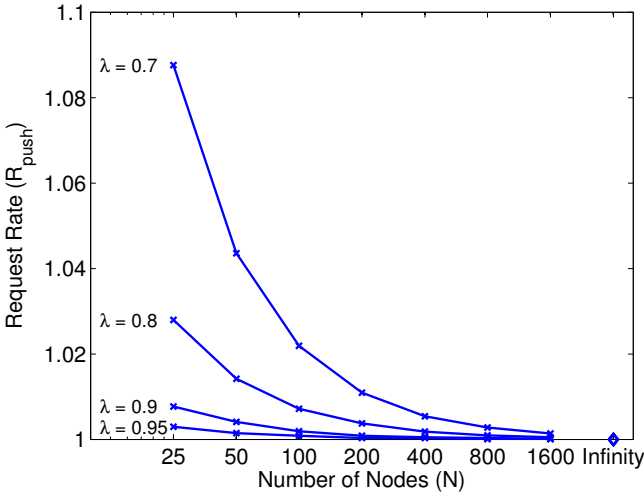
same relative errors because higher loads, which worsen the accuracy, correspond to lower r values, which improve the accuracy.

Figure 2 depicts the same results as Figure 1, but for the pull strategy. Although we still see the convergence as N goes to infinity, the relative errors grow quickly with λ and an error of 9% is observed even for a system with $N = 100$ nodes. Under the pull strategy $r = 1/(1-\lambda)$ for $R = 1$, which implies that larger λ values also correspond to larger r values. Therefore, the less accurate results for higher loads are not unexpected.

The overall request rate observed in the simulation experiments was typically within 0.1% of the targeted R value, meaning the relation $R = (1-\lambda)r$ seems highly accurate even for finite systems. This is not unexpected as the fraction of idle nodes should also match $(1-\lambda)$ in the finite system. Figure 3 shows the observed overall request rate for the

push strategy, which exceeds the targeted value of R and decreases as a function of N and λ. Hence, the relation $R = \lambda^2 r/(1+(1-\lambda)r)$ of Corollary 1 is not highly accurate for small system sizes. This can be explained by noting that the infinite system model is optimistic with respect to the queue length distribution for N finite and therefore also predicts a lower overall probe rate.

In Figure 4 we compare the mean delay of the push and pull strategy in the infinite system model (full lines) with a finite system consisting of $N = 100$ nodes (crosses) for $\lambda \geq 0$ and $R = 0.5$ and $R = 1$. The results indicate that the infinite system model provides accurate results under any load λ, while the pull strategy becomes less accurate as the load increases (which is in agreement with the results in Figures 1 and 2). Note, under the push strategy setting $\lambda < (\sqrt{5}-1)/2 \approx 0.6180$ implies that r can be chosen arbitrarily large such that the overall probe rate R remains below 1 (see Theorem 3). For $r = \infty$ the mean delay becomes 1 and there is little use in simulating the system for finite N.

In Figure 5 we have zoomed in on the intersection of the pull and push curves for $R = 1$ to indicate that the region where the push strategy outperforms the pull strategy is in perfect agreement with the infinite system model. This can be understood by noting that the r value used during the simulation is determined by the relation between R and r in Corollary 1. When $\lambda = \sqrt{(1+R)^2 + 4(1+R)}/2 - (1+R)/2$, we therefore make use of the same r value for the push and pull strategy. Hence, the evolution of the finite system model with N nodes is captured by the same Markov chain $(X^{(N)}(t))_{t \geq 0}$, meaning both strategies have the same queue length distribution and mean delay for all N. We should however keep in mind that the observed overall probe rate tends to exceed R under the push strategy, especially for small systems. It is therefore fair to say that when N is small, the region where the push strategy outperforms the pull strategy
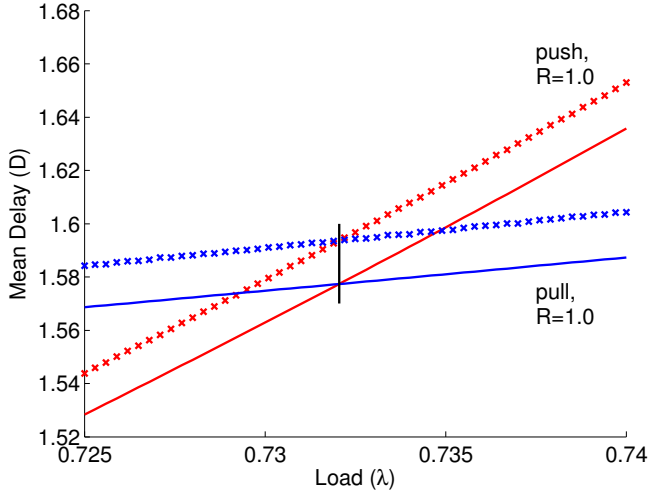
Figure 5. Mean delay of the push and pull strategy in a finite system with $N = 100$ nodes (crosses) vs. infinite system model (full lines) for $R = 1$.

is in fact overestimated by the infinite system model.

V. FINITE VERSUS INFINITE SYSTEM MODEL

In this section we discuss the relation between the set of ODEs in (1) and (2) and the sequence of Markov chains $\{X^{(N)}(t)\}_{t \geq 0}$ as N tends to infinity. More specifically, we will identify the technical issues related to formally proving that the steady state measures $\pi^{(N)}$ of $\{X^{(N)}(t)\}_{t \geq 0}$ converge to the unique fixed point π. Apart from Assumption A1, these issues arise from having an infinite dimensional state space E. Replacing the infinite size buffer in each node by a finite large buffer (such that the loss rate can be neglected) would result in a finite dimensional (compact) space E and would resolve most of the issues. This also explains why large finite buffers are often considered as opposed to infinite buffers (see [5], [6]).

We start by recalling the definition of a density dependent family of Markov chains [13]. A set of Markov chains $\{X^{(N)}(t)\}_{t \geq 0}$, with $N \geq 1$, where $E_N = E \cap \{k/N, k \in \mathbb{Z}^m\}$ is the state space of $\{X^{(N)}(t)\}_{t \geq 0}$, is a family of density dependent Markov chains provided that the transition rates $q^{(N)}(x, y)$ between state $x \in E_N$ and $y \in E_N$ can be written as

$$q^{(N)}(x, y) = N\beta_{(y-x)N}(x),$$

where β_ℓ is a function from $E \subset \mathbb{R}^m$ to $\mathbb{R}^+$. Let $F(x) = \sum_{\ell \in L} \ell \beta_\ell(x)$, where L is the set of all possible transitions. Note, the set of CTMCs considered in Section III matches this definition, with $L = \{e_i, i \geq 1\} \cup \{e_{-i}, i \geq 1\} \cup \{e_1 - e_i, i \geq 2\}$, except that E is not a part of $\mathbb{R}^m$ for some finite m. However, this definition was extended to $\mathbb{R}^\infty$ in [11], where the following generalization of Kurtz's theorem was proven [11, Theorem 3.13]:

Theorem 5 (Kurtz). *Consider a family of density dependent CTMCs, with F Lipschitz. Let $\lim_{N \to \infty} X^{(N)}(0) = \tilde{x}$ a.s. and*

let $\phi_t(\tilde{x})$ be the unique solution to the initial value problem $\frac{d}{dt}x(t) = F(x(t))$ with $x(0) = \tilde{x}$. Consider the path $\{\phi_t(\tilde{x}), t \leq T\}$ for some fixed $T \geq 0$ and assume that there exists a neighborhood K around this path satisfying

$$\sum_{\ell \in L} |\ell| \sup_{x \in K} \beta_\ell(x) < \infty, \qquad (3)$$

then

$$\lim_{N \to \infty} \sup_{t \leq T} \left| X^{(N)}(t) - \phi_t(\tilde{x}) \right| = 0 \quad a.s.$$

The condition in (3) is however not easy to verify as it is not sufficient to have an upper bound on the jump rate over the space E. In the finite dimensional case, the set L is finite and therefore (3) is automatically met. For our system, this condition corresponds to showing that there exists an environment K such that $\sum_{i \geq 1} \sup_{x \in K}(x_i - x_{i+1}) < \infty$. Simulation experiments seem to indicate that the tail of the sample paths $\{\phi_t(\tilde{x}), t \leq T\}$ decreases sufficiently fast such that this condition should be valid, but we have no formal proof at this stage. As such we will rely on the following assumption:

Assumption B1: Given $\tilde{x} \in E$ and $T \geq 0$, there exists an environment K of $\{\phi_t(\tilde{x}), t \leq T\}$ such that $\sum_{i \geq 2} \sup_{x \in K}(x_i - x_{i+1}) < \infty$.

Given that assumption B1 holds, the set of ODEs given by (1) and (2) describes the proper limit process of the finite systems over any finite time horizon $[0, T]$.

A natural question is whether this convergence extends to the stationary regime. Sufficient conditions for the finite dimensional case can be found in [14]. We will instead rely on a more general result in [15], which considers a family of stochastic processes on some Polish space E, which includes the set of infinite dimensional, separable and complete spaces. As $E = \{(x_1, x_2, \ldots) | x_i \in [0, 1], x_i \geq x_{i+1}, i \geq 1, \sum_{j \geq 1} x_j < \infty\}$ is a subspace of the space $l1 = \{(x_1, x_2, \ldots) | \sum_{j \geq 1} |x_j| < \infty\}$, it is separable. E is clearly also complete and therefore Polish. Let $\pi^{(N)} = (\pi_1^{(N)}, \pi_2^{(N)}, \ldots)$ be the unique stationary measure of the Markov chain $\{X^{(N)}(t)\}_{t \geq 0}$. Given that we have a unique solution $\phi_t(x)$ (which is continuous in t for all x) and that convergence over finite time intervals occurs under assumption B1, Corollary 1 of [15] can be rephrased as:

Theorem 6 (Benaïm, Le Boudec). *Under Assumption A1 and B1 and given that $\phi_t(x)$ is continuous in x for all t and that the sequence $(\pi^{(N)})_{N \geq 1}$ is tight, we have*

$$\lim_{N \to \infty} \lim_{t \to \infty} \left| X^{(N)}(t) - \pi \right| = 0,$$

in probability.

The sequence $(\pi^{(N)})_{N \geq 1}$ is tight if for every $\epsilon > 0$ there exists some compact set K_ϵ such that $\mathbb{P}\{\pi^N \in K_\epsilon\} > 1 - \epsilon$ for all N. Note, if E is compact (as is often the case in finite dimension), tightness is immediate.

The continuity of $\phi_t(x)$ in x for all t is guaranteed by the uniqueness of the solution in finite dimensions, but this result does not in general extend to Banach spaces of infinite dimension [16]. However, for F Lipschitz, as in our case, the classical finite dimensional results still hold and we may conclude that convergence of the steady state measures to the fixed point π occurs under assumptions A1, B1 and B2, with

Assumption B2: The sequence of measures $(\pi^{(N)})_{N \geq 1}$ is tight.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we compared the ability of the push and pull strategy to reduce the mean delay in a homogeneous distributed system given an overall probe rate R. We showed that the push strategy outperforms the pull strategy if and only if $\lambda < \sqrt{(R+1)^2 + 4(R+1)}/2 - (R+1)/2$ in the infinite system model and showed, by simulation, that this formula is also accurate for small finite systems, e.g., systems with $N = 25$ nodes. We further demonstrated that a hybrid strategy is always inferior to the pure push or pull strategy when the overall probe rate R is not allowed to increase. The technical issues (Assumptions A1, B1 and B2) to formally prove the convergence of the steady state measures of the finite system model to the infinite system model were also identified, but not proven.

For future work we intend to study push strategies where the probe rate depends on the current queue length. The push strategy considered in this paper generates probes at rate r whenever the queue length i exceeds 1. We believe we can further decrease the mean delay by making r a function of i without increasing the overall probe rate R. Note, this is very much related to the choice of the threshold T in [3], [7], [4], where $T = 1$ was argued to be optimal in case the probe and job exchange times are zero. However, for the strategies studied in [3], [7], [4] smaller T values result in larger probe rates R.

REFERENCES

[1] M. Mitzenmacher, "The power of two choices in randomized load balancing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 12, pp. 1094–1104, October 2001.

[2] Y. Lu, Q. Xie, G. Kliot, A. Geller, J. R. Larus, and A. Greenberg, "Join-idle-queue: A novel load balancing algorithm for dynamically scalable web services," *Perform. Eval.*, vol. 68, pp. 1056–1071, 2011.

[3] D. L. Eager, E. D. Lazowska, and J. Zahorjan, "A comparison of receiver-initiated and sender-initiated adaptive load sharing," *Perform. Eval.*, vol. 6, no. 1, pp. 53–68, 1986.

[4] R. Mirchandaney, D. Towsley, and J. A. Stankovic, "Analysis of the effects of delays on load sharing," *IEEE Trans. Comput.*, vol. 38, no. 11, pp. 1513–1525, 1989.

[5] N. Gast and B. Gaujal, "A mean field model of work stealing in large-scale systems," *SIGMETRICS Perform. Eval. Rev.*, vol. 38, no. 1, pp. 13–24, 2010.

[6] B. Van Houdt, "Performance comparison of aggressive push and traditional pull strategies in large distributed systems," in *Proceedings of QEST 2011, Aachen (Germany), IEEE Computer Society*, SEP 2011, pp. 265–274.

[7] R. Mirchandaney, D. Towsley, and J. A. Stankovic, "Adaptive load sharing in heterogeneous distributed systems," *J. Parallel Distrib. Comput.*, vol. 9, no. 4, pp. 331–346, 1990.

[8] M. S. Squillante and R. D. Nelson, "Analysis of task migration in shared-memory multiprocessor scheduling," *SIGMETRICS Perform. Eval. Rev.*, pp. 143–155, 1991.

[9] M. Mitzenmacher, "Analyses of load stealing models based on families of differential equations," *Theory of Computing Systems*, vol. 34, pp. 77–98, 2000. [Online]. Available: http://dx.doi.org/10.1007/s002240010010

[10] K. Deimling, *Ordinary Differential Equations in Banach spaces*. Lect. Notes in Math. 596, 1977.

[11] M. Mitzenmacher, "The power of two choices in randomized load balancing," Ph.D. dissertation, University of California, Berkeley, 1996.

[12] M. Shaked and J. G. Shanthikumar, *Stochastic Orders and their Applications*. Associated Press, 1994.

[13] T. Kurtz, *Approximation of population processes*. Society for Industrial and Applied Mathematics, 1981.

[14] M. Benaïm, "Recursive algorithms, urn processes and chaining number of chain recurrent sets," *Ergodic Theory and Dynamical Systems*, vol. 18, pp. 53–87, 1998.

[15] M. Benaïm and J. Le Boudec, "On mean field convergence and stationary regime," *CoRR*, vol. abs/1111.5710, Nov 24 2011.

[16] F. S. de Blasi and G. Pianigiani, "Uniqueness for differential equations implies continuous dependence only in finite dimension," *Bulletin of the London Mathematical Society*, vol. 18, no. 4, pp. 379–382, 1986.