# A Fair Comparison of Pull and Push Strategies in Large Distributed Networks

Wouter Minnebo and Benny Van Houdt, *Member, IEEE*

***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 $\lambda$, 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.**

***Index Terms*—Performance analysis, Distributed computing, Processor scheduling**

## 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], [3]) or via the processing nodes themselves (e.g., [4], [5], [6], [7]). 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. However, 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 based on 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 [4], [8]. The approach was based on a decoupling assumption and relied on numerical methods to solve some nonlinear 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 [9] 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 [10]. 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 [9], [5]. 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 [5], where it has been shown to outperform both the push and pull strategy for all loads $\lambda$. However, as indicated in [5], 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, under which nodes do not transmit probes 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 $\lambda$ 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

This is an extended version of the paper entitled "Pull versus Push Mechanism in Large Distributed Networks: Closed Form Results" published in the Proceedings of ITC-24.

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 $R$. In fact, we show that if rate based pull/push strategy matches the overall probe rate of the traditional pull/push strategy, it also matches the queue length distribution and therefore the mean response time (under the infinite system model).

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 $\lambda < (-1+\sqrt{5})/2 \approx 0.6108$.

As a side result we show that the queue length distribution of the pull and push strategy is in fact identical when they use the same rate $r$ (instead of the same $R$). 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 $\lambda$ 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 and prove the technical issues related to showing that the infinite system model is indeed the proper limit process of the sequence of finite system models, while in Section VI we show that rate-based strategies matching the probe rate of the traditional strategies also match the queue length distribution. Conclusions are drawn and future work is discussed in Section VII.

## II. Pull and Push Strategies

As in [4], [11], [9], [6] the model considered in this paper relies on the following assumptions:

1) The system consists of $N$ nodes, where each node consist of a single server and an infinite buffer to store jobs.
2) Each node is subject to its own local Poisson arrival process with rate $\lambda$. Jobs require an exponential processing time with mean 1 and are served in a first-come-first-served (FCFS) order.
3) 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).

The above assumptions also imply that all jobs can be successfully executed by any node in the system and that there are no communication failures on the network that interconnects the nodes.

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 [11, 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 $\lambda$ 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 [4], [9], [5] and discussed in Section VI (for $T = 1$) 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 repeatedly send probes until either one gets a positive reply or a predefined maximum of $L_p$ probes is reached. The overall probe rate $R$ clearly depends on $T, L_p$ and the load $\lambda$, 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 [4]), 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 [9], [5] 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 [6], [7] are more aggressive pull and push strategies. In [6] a successful probe message results in exchanging half of the jobs that are waiting, while in [7] the number of probes send under the push strategy depends on the current queue length. Although the push strategy of [7] 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 defined by a set of ordinary differential equations (ODEs) and is thus deterministic. The evolution of the finite system models (for which $N < \infty$) on the other hand will be captured by an $N$-dimensional continuous time Markov chain (CTMC) and is therefore stochastic. To define the infinite system model 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 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 $x_0 = 1$ (as all the queues contain zero or more jobs) 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 $\lambda < 1$ (see Appendix C). Let $\pi^{(N)} = (\pi_1^{(N)}, \pi_2^{(N)}, \ldots)$ be the unique stationary measure of the Markov chain $\{X^{(N)}(t)\}_{t\geq 0}$, then the overall probe rate $R^{(N)}$ for the pull, push and hybrid strategy equal $r(1-\pi_1^{(N)})$, $r\pi_2^{(N)}$ and $r_1(1 - \pi_1^{(N)}) + r_2\pi_2^{(N)}$, respectively.

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. Let $L = \{e_i, i \geq 1\} \cup \{-e_i, i \geq 1\} \cup \{e_1 - e_i, i \geq 2\}$ be the set of possible transitions (arrivals, job completions and job transfers). Next, we define

$$F(x) = \sum_{\ell \in L} \ell \beta_\ell(x),$$

where $\beta_\ell(x)$ is defined such that $\beta_\ell(x/N) = q^{(N)}(x, x+\ell)/N$. Given the above expressions for $q^{(N)}(x, x+\ell)/N$, this implies

$$\begin{aligned} \beta_{e_i}(x) &= \lambda(x_{i-1} - x_i), \\ \beta_{-e_i}(x) &= (x_i - x_{i+1}), \end{aligned}$$

for all $i \geq 1$ (with $x_0 = 1$ for $i = 1$) and

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

[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 other words,

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

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

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

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] $\phi_t(x(0))$ [12, 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 $\pi$ 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 write $\sum_{i\geq 1} F_i(\pi)$ using (2) as

$$\begin{aligned}\sum_{i\geq 1} F_i(\pi) &= F_1(\pi) + \sum_{i\geq 2}\lambda(\pi_{i-1} - \pi_i) \\ &\quad - \sum_{i\geq 2}(1 + r(1 - \pi_1))(\pi_i - \pi_{i+1}) \\ &= F_1(\pi) + \lambda\pi_1 - (1 - r(1 - \pi_1))\pi_2 = \lambda - \pi_1,\end{aligned}$$

where the last equality follows from (1). In other words, $\sum_{i\geq 1} F_i(\pi) = 0$ implies that $\pi_1 = \lambda$. 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 $\pi_i$, for $i \geq 2$. ∎

If we take the set of ODEs in (1) and (2) and replace the first $x_2(t)$ by $\pi_2$ in (1) and $x_1(t)$ by $\lambda$ in (2), then we end up with the Kolmogorov equations 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 $\lambda_i$ and the service is exponential with rate $\mu_i$ whenever the queue length equals $i$. Hence, the fixed point $\pi$ also corresponds to the steady state of a state dependent M/M/1 queue (where $\pi_i$ is the probability that the queue contains at least $i$ jobs).

The set of ODEs in (1) and (2) describes the transient behavior 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. In Appendix A we prove the following theorem:

**Theorem 3.** *All the trajectories of the set of ODEs given by* (1) *and* (2)*, starting from $x \in E$ converge towards the unique fixed point $\pi$.*

Due to the above theorem, we can now express the main performance measures of the pull, push and hybrid strategy via Theorem 2, where the overall probe rate $R$ for the pull, push and hybrid strategy are defined as $r(1 - \pi_1)$, $r\pi_2$ and $r_1(1 - \pi_1) + r_2\pi_2$, respectively:

**Corollary 1.** *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 $\lambda$ and the overall allowed probe rate

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

$R$. 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 $\lambda$ approaches one.

**Theorem 4.** *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 $\lambda$ 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 $\lambda$. 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 [13].

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 $\lambda$ approaches 1 under the pull strategy (for any $R > 0$).

**Theorem 5.** *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.*

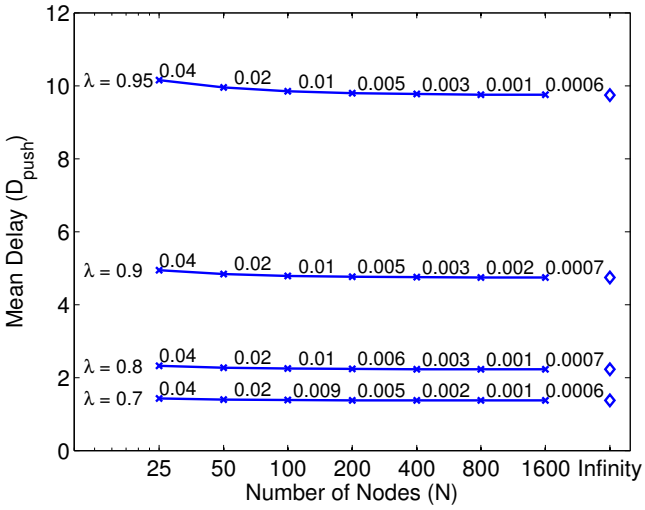


Figure 1. Mean delay and relative error of the push strategy in a finite vs. infinite system for $R = 1$

*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 $D_{push} = 1$. 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 4 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 $\lambda$ 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 $10^6$ time units (where the service time is exponentially distributed with a mean of 1 time unit) and a warm-up period of length $10^6/3$ time units.

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

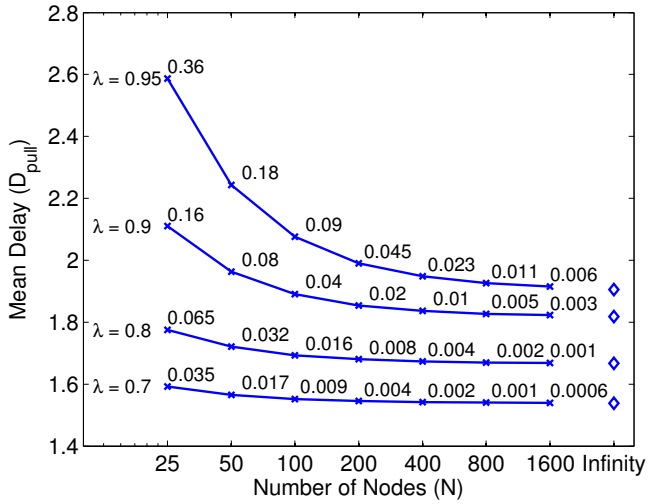


Figure 2. Mean delay and relative error of the pull strategy in a finite vs. infinite system for $R = 1$.

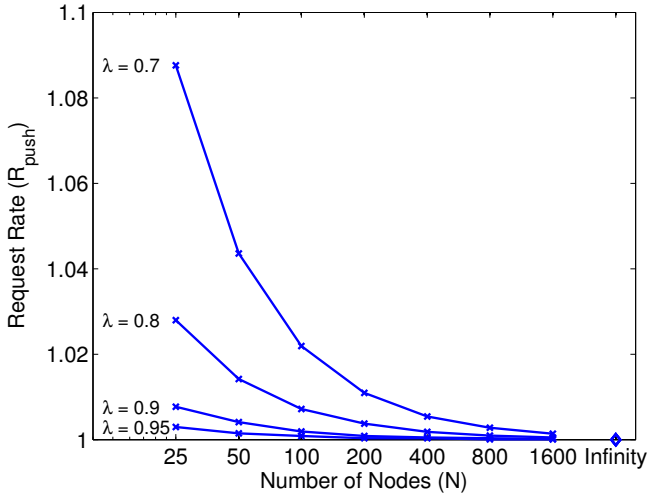


Figure 3. Overall request rate of the push strategy in a finite vs. infinite system for $R = 1$.

$\lambda = 0.7, 0.8, 0.9$ and $0.95$. For each combination of $N$ and $\lambda$ 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 $\lambda$ 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 $\lambda$ (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 $\lambda$ remains fixed). Thus, in Figure 1 we see more or less the 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 $\lambda$ 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 $\lambda$ 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 $\lambda$. 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 $\lambda \geq$

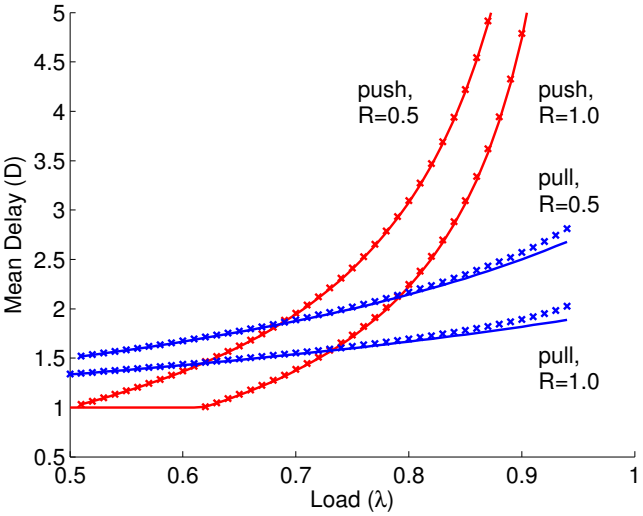


Figure 4. Mean delay of the push and pull strategy in a finite system with $N = 100$ nodes (crosses) vs. infinite system model (full lines) for $R = 0.5$ and $R = 1$.

$0$ and $R = 0.5$ and $R = 1$. The results indicate that the infinite system model provides accurate results under any load $\lambda$, 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 4). 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

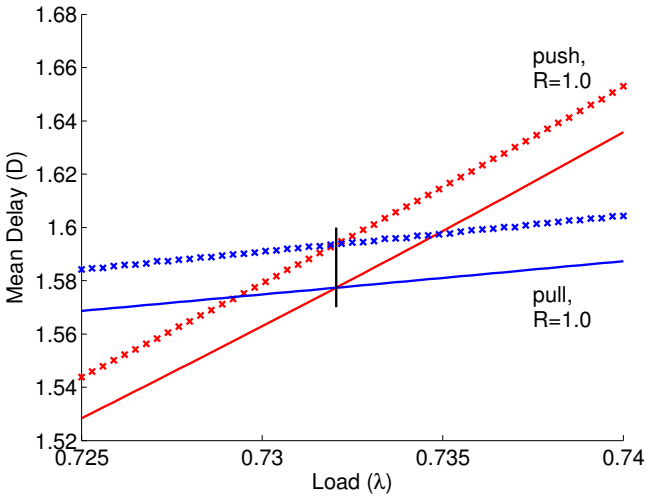


Figure 5. Mean delay of the push and pull strategy in a finite system with $N = 100$ nodes (crosses) vs. infinite system model (full lines) for $R = 1$.

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 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 and prove the technical issues related to formally showing that the steady state measures $\pi^{(N)}$ of $\{X^{(N)}(t)\}_{t\geq 0}$ converge to the unique fixed point $\pi$. 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 [6], [7]).

We start by recalling the definition of a density dependent family of Markov chains [14]. 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 $\beta_\ell$ 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 [15], where the following generalization of Kurtz's theorem was proven [15, Theorem 3.13]:

**Theorem 6** (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*

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

*then*

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

In the finite dimensional case, the set $L$ is finite and therefore (3) is automatically met. For our system, condition (3) corresponds to showing that there exists an environment $K$ such that $\sum_{i\geq 2} \sup_{x\in K}(x_i - x_{i+1}) < \infty$. The following theorem is proven in Appendix B:

**Theorem 7.** *Given $\tilde{x} \in E$ and $T \geq 0$, there exists an environment $K$ of $\{\phi_t(\tilde{x}), t \leq T\}$ such that $\sum_{i\geq 2} \sup_{x\in K}(x_i - x_{i+1}) < \infty$.*

Hence, 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 [16]. We will instead rely on a more general result in [17], 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, Corollary 1 of [17] can be rephrased as:

**Theorem 8** (Benaïm, Le Boudec)**.** *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 $K_\epsilon$ 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. In our case $E$ is not compact and the following theorem is proven in Appendix C:

**Theorem 9.** *The sequence of measures $(\pi^{(N)})_{N\geq 1}$ is tight.*

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 [18]. 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 $\pi$ occurs.

## VI. Rate-based versus traditional strategies

The aim of this section is to show that the performance of the rate-based pull/push strategies coincides with the traditional pull/push strategies when the former match the overall probe rate of the latter. To this end, we introduce an infinite system model for the following traditional pull and push strategy:

1) *Traditional Push:* A server starts sending probes whenever a job arrives in a queue with $i \geq 1$ jobs, meaning $i-1$ jobs are waiting to be served. The nodes that are probed are selected at random and a node is only allowed to accept a job if it is idle. The server starts by probing a single node. If the probe fails (because the selected node is not idle), the server sends another probe. This procedure is repeated until a probe is either successful or $L_p$ unsuccessful probes were sent.
2) *Traditional Pull:* A server starts sending probes whenever the server becomes idle. The nodes that are probed are selected at random and a node is only allowed to transfer a job if its queue length exceeds one. Probes are sent one at a time until one is successful or $L_p$ unsuccessful probes were sent.

Analytical models to assess the performance of a class of pull and push strategies that include the above two strategies were presented in [4], [8]. These models relied on a decoupling assumption and the mean response time was expressed as the solution to a nonlinear equation that was solved numerically.

In this section we present ODE models for the traditional strategies similar to the ODE model in Section III for the rate-based strategies and show that its unique fixed point can be expressed in closed form. It is not hard to verify that the nonlinear equation for the unique fixed point of the ODE corresponds to the nonlinear equation in [4] for the pull strategy with $T = 1$ and the one in [8] for the push strategy with $T = 1$ and $C = 0$.

### A. Traditional push

Let $s_i(t)$ denote the fraction of queues containing at least $i$ jobs at time $t$ and set $s(t) = (s_1(t), s_2(t), \ldots)$. The dynamics of the infinite system model for the traditional push strategy is captured by the following set of ODEs:

$$\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) + \lambda s_1(t)(1 - s_1(t)^{L_p}) - (s_1(t) - s_2(t)) \quad (4)$$

$$\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t))s_1(t)^{L_p} - (s_i(t) - s_{i+1}(t)) \quad (5)$$

for $i \geq 2$. The terms $s_i(t) - s_{i+1}(t)$, for $i \geq 1$, correspond to the service completion events (as the job durations are exponential with mean 1). The rate at which arrivals occur in a node with exactly $i-1$ jobs is $\lambda(s_{i-1}(t) - s_i(t))$ and $s_1(t)^{L_p}$ is the probability that $L_p$ probes are unsuccessful; hence queues of length $i$ are created at rate $\lambda(s_{i-1}(t) - s_i(t))s_1(t)^{L_p}$, for $i \geq 2$. Finally, queues of length 1 are created by new arrivals (at rate $\lambda(1 - s_1(t))$) or job transfers. The latter occur at rate $\lambda s_1(t)(1 - s_1(t)^{L_p})$, as $\lambda s_1(t)$ is the rate at which probes are sent to the idle nodes and $(1 - s_1(t)^{L_p})$ is the probability that one of the probes is successful.

The set of ODEs given by (4) and (5) has a unique fixed point $\hat{\pi} = (\hat{\pi}_1, \hat{\pi}_2, \ldots)$ with $\sum_{i \geq 1} \hat{\pi}_i < \infty$ given by

$$\begin{aligned} \hat{\pi}_1 &= \lambda, \\ \hat{\pi}_2 &= \lambda^{L_p+2}, \\ \hat{\pi}_{i+1} &= \hat{\pi}_i - \lambda^{L_p+1}(\hat{\pi}_{i-1} - \hat{\pi}_i), \end{aligned}$$

for $i > 2$, where the first equality follows from taking the sum of (4) and (5) for $i \geq 1$.

For the traditional push strategy, every busy node will send on average

$$1 + \sum_{i=1}^{L_p-1} \hat{\pi}_1^i = \frac{1 - \lambda^{L_p}}{1 - \lambda}$$

probes at the task arrival rate $\lambda$, meaning that the overall probe rate $\hat{R}$ equals

$$\hat{R} = \hat{\pi}_1 \lambda \frac{1 - \lambda^{L_p}}{1 - \lambda}.$$

From the relationship $R = r_{push}\pi_2$, we observe that a rate-based push strategy with

$$r_{push} = \frac{\lambda \hat{\pi}_1}{\pi_2} \frac{1 - \lambda^{L_p}}{1 - \lambda}$$

matches $\hat{R}$. By substituting the rate $r_{push}$ in Theorem 2 we find that under the infinite system model, the traditional and rate-based push strategy have the same fixed point. This indicates that the rate-based strategy matches the queue length distribution of the traditional variant, provided that we match the overall probe rate $R$.

### B. Traditional pull

A similar set of ODEs describes the evolution of the traditional pull strategy (for $L_p = 1$ this corresponds to the model in [11]):

$$\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) - (s_1(t) - s_2(t))(1 - s_2(t))^{L_p} \quad (6)$$

$$\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t)) - (s_i(t) - s_{i+1}(t)) - \frac{(s_i(t) - s_{i+1}(t))}{s_2(t)}(s_1(t) - s_2(t))(1 - (1 - s_2(t))^{L_p}), \quad (7)$$

for $i \geq 2$, where $\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t))$ if $s_2(t) = 0$ and $i \geq 2$. The intuition is similar as for the push strategy, where we note that $(s_1(t) - s_2(t))$ is the rate at which probes are generated, $(1 - (1 - s_2(t))^{L_p})$ is the probability that one of the probes is successful and $(s_i(t) - s_{i+1}(t))/s_2(t)$ is the probability that the accepting busy queue has length $i$, for $i \geq 2$.

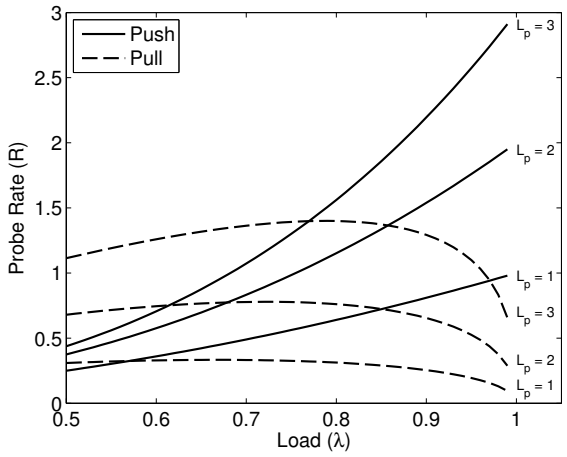


Figure 6. Mean overall probe rate for the traditional pull and push strategy as a function of $\lambda$ for $L_p = 1, 2$ and 3.

The system given by (6) and (7) also has a unique fixed point $\tilde{\pi} = (\tilde{\pi}_1, \tilde{\pi}_2, \ldots)$ with $\sum_{i\geq 1} \tilde{\pi}_i < \infty$ . As $\tilde{\pi}_1 = \lambda$, $\tilde{\pi}_2$ is found as the unique positive real root of

$$g(x) = \lambda(1-\lambda) - (\lambda - x)(1-x)^{L_p} = 0,$$

with $x \in [0, \lambda]$. The uniqueness follows by noting that $dg(x)/dx = (1-x)^{L_p-1}(1-x+L_p(\lambda-x))$ is strictly positive on $[0, \lambda]$, while $g(0) < 0$ and $g(\lambda) > 0$ (as $\lambda < 1$). Finally, $\tilde{\pi}_i$, for $i > 2$, is given by

$$\tilde{\pi}_{i+1} = \tilde{\pi}_i - \frac{\lambda(\tilde{\pi}_{i-1} - \tilde{\pi}_i)}{1 + (\lambda - \tilde{\pi}_2)(1 - (1-\tilde{\pi}_2)^{L_p})/\tilde{\pi}_2}.$$

For the traditional pull strategy, every node with exactly one job will send on average

$$1 + \sum_{i=1}^{L_p - 1} (1 - \tilde{\pi}_2)^i = \frac{1 - (1-\tilde{\pi}_2)^{L_p}}{\tilde{\pi}_2}$$

probes each time the server becomes idle (as a probe fails with probability $(1 - \tilde{\pi}_2)$), meaning that the overall probe rate $\tilde{R}$ for the traditional pull strategy equals

$$\tilde{R} = (\tilde{\pi}_1 - \tilde{\pi}_2)\frac{1 - (1-\tilde{\pi}_2)^{L_p}}{\tilde{\pi}_2}.$$

From the relationship $R = (1 - \pi_1)r_{pull}$, we observe that a rate-based pull strategy with

$$r_{pull} = \frac{\tilde{\pi}_1 - \tilde{\pi}_2}{1 - \pi_1}\frac{1 - (1-\tilde{\pi}_2)^{L_p}}{\tilde{\pi}_2}$$

amounts to the same overall probe rate $\tilde{R}$. Substituting the rate $r_{pull}$ in Theorem 2 allows us to conclude that the rate-based pull strategy matches the queue length distribution of the traditional variant, provided that we match the overall probe rate $R$.

The mean overall probe rate $\hat{R}$ and $\tilde{R}$ for the traditional push and pull strategy respectively is shown in Figure 6. As the probe rate of both strategies differs significantly, it is sometimes hard to compare these strategies in a fair manner.

## VII. 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 quite 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. Some technical issues to formally prove the convergence of the steady state measures of the finite system model to the infinite system model were identified and proven. Finally, we showed that rate-based strategies matching the probe rate of traditional strategies, also match the queue length distribution.

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 [4], [9], [5], where $T = 1$ was argued to be optimal in case the probe and job exchange times are zero. However, for the strategies studied in [4], [9], [5] smaller $T$ values result in larger probe rates $R$.

Another possible extension exists in considering strategies where nodes accept jobs whenever their queue length is small enough, instead of equal to zero as in the current paper. Models in which each node contains multiple servers instead of one may also be of interest, in this case a variety of pull strategies may be considered. One could also try to extend the current set of results to a heterogeneous model (i.e., where nodes can have different job arrival and service rates), but obtaining closed form results (or even the stability condition) may be quite challenging.

## 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] N. Vvedenskaya, R. Dobrushin, and F. Karpelevich, "Queueing system with selection of the shortest of two queues: an asymptotic approach," *Problemy Peredachi Informatsii*, vol. 32, pp. 15–27, 1996.

[3] 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.

[4] D. Eager, E. 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.

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

[6] 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.

[7] 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.

[8] D. Eager, E. Lazowska, and J. Zahorjan, "Adaptive load sharing in homogeneous distributed systems," *Software Engineering, IEEE Transactions on*, vol. SE-12, no. 5, pp. 662 –675, may 1986.

[9] 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.

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

[11] M. Mitzenmacher, "Analyses of load stealing models based on families of differential equations," *Theory of Computing Systems*, vol. 34, pp. 77–98, 2001.

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

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

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

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

[16] M. Benaïm and J. Le Boudec, "A class of mean field interaction models for computer and communication systems," *Performance Evaluation*, vol. 65, no. 11-12, pp. 823–838, 2008.

[17] ——, "On mean field convergence and stationary regime," *CoRR*, vol. abs/1111.5710, Nov 24 2011.

[18] F. 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.

[19] J. Walker, *Dynamical Systems and Evolution Equations. Theory and Applications.* Plenum Press, New York, 1980.

[20] P. Billingsley, *Probability and Measure*. John Wiley and Sons, 1979.


## APPENDIX A
## PROOF OF THEOREM 3

We start by proving the following Lemma:

**Lemma 1.** *Let $x(t)$ be the unique solution of the ODEs given by* (1) *and* (2) *with $x(0) \in E$. The $L_1$-distance to the unique fixed point $\sum_{i\geq 1} |x_i(t) - \pi_i|$ does not increase as a function of $t$.*

*Proof:* Define $\epsilon_i(t) = x_i(t) - \pi_i$, for $i \geq 1$, such that $\Phi(t) = \sum_{i\geq 1} |\epsilon_i(t)|$ represents the $L_1$-distance. As $\frac{d}{dt}x_i(t) = \frac{d}{dt}\epsilon_i(t)$ and $\pi$ is a fixed point of (1) and (2), we find

$$\begin{aligned}\frac{d}{dt}\epsilon_1(t) &= -\lambda\epsilon_1(t) + (1 + r(1-\pi_1))\epsilon_2(t)\\ &\quad -r\epsilon_1(t)(\epsilon_2(t)+\pi_2) - \epsilon_1(t),\end{aligned} \tag{8}$$

and

$$\begin{aligned}\frac{d}{dt}\epsilon_i(t) &= \lambda(\epsilon_{i-1}(t) - \epsilon_i(t))\\ &\quad -(1 + r(1-\pi_1))(\epsilon_i(t) - \epsilon_{i+1}(t))\\ &\quad +r\epsilon_1(t)(\epsilon_i(t) - \epsilon_{i+1}(t) + \pi_i - \pi_{i+1}),\end{aligned} \tag{9}$$

for $i \geq 2$. Assume for now that $\epsilon_i(t) \neq 0$ for all $i$ such that $\frac{d}{dt}\Phi(t)$ is properly defined as

$$\frac{d}{dt}\Phi(t) = \sum_{i:\epsilon_i(t)>0} \frac{d}{dt}\epsilon_i(t) - \sum_{i:\epsilon_i(t)<0} \frac{d}{dt}\epsilon_i(t).$$

If $\epsilon_i(t)$ has the same sign for all $i$, one finds that $\frac{d}{dt}\Phi(t) = -|\epsilon_1(t)|$ by summing (8) and (9). We will show that

$$\frac{d}{dt}\Phi(t) \leq -|\epsilon_1(t)|,$$

holds in general. Let $I = \{i_1, i_2, \ldots\}$, with $i_1 < i_2 < \ldots$, be the set of indices where $\epsilon_i(t)$ changes sign, that is, $\epsilon_{i-1}(t)$ and $\epsilon_i(t)$ have a different sign if and only if $i \in I$. Assume $\epsilon_1(t) < 0$ and let $I_+ = \{i_1, i_3, \ldots\}$ and $I_- = \{i_2, i_4, \ldots\}$ such that $i \in I_+$ implies that $\epsilon_{i-1}(t) < 0$ and $\epsilon_i(t) > 0$, while $i \in I_-$ implies that $\epsilon_{i-1}(t) > 0$ and $\epsilon_i(t) < 0$.

When $\epsilon_{i-1}(t)$ and $\epsilon_i(t)$ are both positive (for $i \geq 2$), the terms $\lambda\epsilon_{i-1}(t)$, $-(1 + r(1-\pi_1))\epsilon_i(t)$ and $r\epsilon_1(t)(\epsilon_i(t) + \pi_i)$ in $\frac{d}{dt}\epsilon_i(t)$ are canceled by $\frac{d}{dt}\epsilon_{i-1}(t)$ when computing $\frac{d}{dt}\Phi(t)$. However, if $\epsilon_{i-1}(t) < 0$ and $\epsilon_i(t) > 0$, $\frac{d}{dt}\epsilon_{i-1}(t)$ is replaced by $-\frac{d}{dt}\epsilon_{i-1}(t)$ in $\frac{d}{dt}\Phi(t)$ and therefore contains these three terms twice. Hence, in general

$$\begin{aligned}\frac{d}{dt}\Phi(t) &= \epsilon_1(t) + 2\sum_{i\in I_+} \underbrace{(\lambda\epsilon_{i-1}(t) - (1 + r(1-\pi_1))\epsilon_i(t))}_{<0}\\ &\quad -2\sum_{i\in I_-} \underbrace{(\lambda\epsilon_{i-1}(t) - (1 + r(1-\pi_1))\epsilon_i(t))}_{>0}\\ &\quad +2\sum_{i\in I_+} r\epsilon_1(t)(\epsilon_i(t)+\pi_i) - 2\sum_{i\in I_-} r\epsilon_1(t)(\epsilon_i(t)+\pi_i).\end{aligned} \tag{10}$$

This implies that $\frac{d}{dt}\Phi(t) \leq \epsilon_1(t)$ provided that

$$\sum_{i\in I_+}(\epsilon_i(t) + \pi_i) - \sum_{i\in I_-}(\epsilon_i(t) + \pi_i) \geq 0,$$

which clearly holds as this expression is equal to $(x_{i_1}(t) - x_{i_2}(t)) + (x_{i_3}(t) - x_{i_4}(t)) + \ldots$ and $x_i(t) \geq x_j(t)$ for $i < j$. Hence, $\frac{d}{dt}\Phi(t) \leq -|\epsilon_1(t)|$ if $\epsilon_1(t) < 0$. A similar argument can be used for $\epsilon_1(t) > 0$.

Finally, we consider the technical issue of defining $\frac{d}{dt}\Phi(t)$ in case $\epsilon_i(t) = 0$ for some $i$ and $t = t_0$. In this case the above proof remains unchanged provided that we rely on the upper right-hand derivative (as in [1, Theorem 3]), that is, if we define $\frac{d}{dt}|\epsilon_i(t_0)|$ as

$$\frac{d}{dt}|\epsilon_i(t_0)| = \lim_{t\to t_0^+} \frac{|\epsilon_i(t)|}{t - t_0}.$$

■

The above lemma shows that the $L_1$-distance to the fixed point does not increase along any trajectory $x(t)$ in $E$, and only remains the same whenever $x_1(t) = \pi_1$ (as $\epsilon_1(t) = 0$ in such a case).

**Lemma 2.** *The only trajectory $x(t)$ of the ODE given by* (1) *and* (2) *with $x(0) \in E$ for which $x_1(t) = \pi_1$ for all $t$ is given by $x(t) = \pi$ for all $t$.*

*Proof:* If $x_1(t) = \pi_1 = \lambda$ for all $t$, then (1) implies that $x_2(t) = \pi_2$. Similarly, for $i \geq 2$, if $x_j(t) = \pi_j$ for all $j \leq i$ and $t$, then (2) implies that $x_{i+1}(t) = \pi_{i+1}$. ■

We now recall La Salle's invariance principle for Banach spaces, where a (positively) invariant subset of $K \subset E$ of an ODE defined on $E$ is such that $x(t) \in K$ for all $t$ provided that $x(t)$ is the unique solution of the ODE with $x(0) \in K$.

**Theorem 10** ([19])**.** *Let $V(x)$ be a continuous real valued function from $E$ to $\mathbb{R}$ with $\frac{d}{dt}V(x) = \limsup_{t\to 0^+} \frac{1}{t}(V(x(t)) - V(x)) \leq 0$, where $x(t)$ is the unique solution of an ODE with $x(0) = x$. Let $K = \{x \in E | \frac{d}{dt}V(x) = 0\}$ and let $M$ be the largest*

*(positively) invariant subset of $K$. If $x(t)$ is precompact (i.e., remains in a compact set) for $x(0) \in E$, then*

$$\lim_{t\to\infty} dist(x(t), M) = 0,$$

*where $dist(x, M)$ represents the Banach distance between the point $x \in E$ and the set $M \subset E$.*

We are now in a position to prove Theorem 3:

*Proof of Theorem 3:* We rely on La Salle's invariance principle for Banach spaces by setting $V(x)$ equal to the $L_1$-distance to the fixed point, i.e., $V(x) = \sum_{i\geq 1} |x_i - \pi_i|$. Lemma 1 implies that $\frac{d}{dt}V(x) \leq 0$, while Lemma 2 shows that $M = \{\pi\}$ is a singleton. Hence, $\pi$ is a global attractor provided that we can show that the trajectory $x(t)$ remains in a compact set if $x(0) \in E$. Let $m = \sum_{i\geq 1} |x_i(0) - \pi_i|$, then by Lemma 1 we know that $x(t)$ remains in the set $E_m = \{x \in E | \sum_{i\geq 1} |x_i - \pi_i| \leq m\}$. This set is not compact in the Banach space $E$ equipped with the $L_1$-norm, but La Salle's invariance principle holds in any Banach space. If $E$ is equipped with the weighted $L_1$-norm $\sum_{i\geq 1} \frac{|x_i|}{2^i}$, the sets $E_m$ are compact and global attraction follows from Theorem 10. ∎

## APPENDIX B
## PROOF OF THEOREM 7

We start by proving the following lemma:

**Lemma 3.** *For any $T > 0$ and $x(0) \in E$, the unique solution $x(t) = (x_1(t), x_2(t), \ldots)$ to the initial value problem defined by* (1) *and* (2) *satisfies*

$$\sum_{i\geq 2} \sup_{0\leq t\leq T} x_i(t) \leq \exp(\lambda T) \left(1 + \sum_{i\geq 2} x_i(0)\right). \tag{11}$$

*Proof:* By (2) we have $\frac{d}{dt}x_i(t) \leq \lambda x_{i-1}(t)$ for $i \geq 2$. Hence, for $i \geq 2$

$$x_i(t) = x_i(0) + \int_{u=0}^{t} dx_i(u) \leq x_i(0) + \lambda \int_{u=0}^{t} x_{i-1}(u)du.$$

Combining the above inequality with the fact that $x_1(t) \leq 1$, readily allows us, by induction on $i$, to establish the following inequality:

$$x_i(t) \leq \sum_{j=2}^{i} x_j(0) \frac{(\lambda t)^{i-j}}{(i-j)!} + \frac{(\lambda t)^{i-1}}{(i-1)!}.$$

Interchanging the order of summation therefore yields

$$\sum_{i\geq 2} \sup_{0\leq t\leq T} x_i(t) \leq \sum_{j\geq 2} x_j(0) \sum_{i\geq j} \frac{(\lambda T)^{i-j}}{(i-j)!} + \sum_{i\geq 2} \frac{(\lambda T)^{i-1}}{(i-1)!},$$

which implies (11). ∎

Theorem 7 readily follows from (11) by defining $K$ as

$$K = \{x \in E | \exists t, \forall i \geq 0 : |x_i - \tilde{x}_i(t))| < 2^{-i}\},$$

such that

$$\sum_{i\geq 2} \sup_{x\in K} x_i < \sum_{i\geq 2} \left( \sup_{0\leq t\leq T} \tilde{x}_i(t) + 2^{-i} \right)$$
$$\leq \exp(\lambda T)(1 + \sum_{i\geq 2} \tilde{x}_i(0)) + 1 < \infty.$$

## APPENDIX C
## PROOF OF THEOREM 9

Define the set $F_m \subset E$ as $F_m = \{x \in E | \sum_{i\geq 1} x_i \leq m\}$. If we consider the metric space $(E, \rho)$, where $\rho$ is the weighted $L_1$-norm $\sum_{i\geq 1} \frac{|x_i|}{2^i}$, then $F_m$ is compact for any $m > 0$. Note, it suffices to prove tightness in this metric space as Prokhorov's theorem holds in any separable metric space [20]. To prove that the measures $(\pi^{(N)})_{N\geq 1}$ are tight, we will show that for any $\epsilon > 0$, setting $m_\epsilon = \frac{1}{(1-\lambda)\epsilon}$ implies that $\mathbb{P}\{\pi^N \in F_{m_\epsilon}\} > 1-\epsilon$ for all $N$.

We start by considering a modified system consisting of $N$ nodes in which we give preemptive priority to *local* jobs, that is, transferred jobs are interrupted whenever a local job arrives (and can be transferred to yet another node). Let $X^{(N)}_{i,mod}(t) \in \{0, \ldots, N\}$ be the number of nodes with *at least* $i$ jobs in the queue at time $t$ in the modified system. Due to the exponential job size durations we have $X^{(N)}_{i,mod}(t) = X^{(N)}_i(t)$, which implies that the modified system consisting of $N$ nodes has the same stationary measure $\pi^{(N)}$, meaning it suffices to prove tightness for the modified system.

Using the modified system consisting of $N$ nodes, we can now rely on a simple sample path argument to show that the length of queue $i$, for $i = 1, \ldots, N$, in the modified system is upper bounded by one plus the queue length of the $i$-th queue in a system consisting of $N$ independent M/M/1 queues. After all, in the modified system service to the local jobs is never prevented by a transferred job, each queue contains at most one transferred job and some local jobs may even be transferred. As the stationary queue length distribution in an M/M/1-queue is geometric with a mean equal to $\lambda/(1-\lambda)$, we may conclude that the mean queue length in the $i$-th node of the modified system is upper bounded by $1/(1-\lambda)$, for $i = 1, \ldots, N$.

Let $Y^{(N)}_{i,mod}(t)$ be the random variable representing the queue length of the $i$-th queue in the modified system at time $t$ and set $Y^{(N)}_{i,mod} = \lim_{t\to\infty} Y^{(N)}_{i,mod}(t)$, then

$$\frac{1}{N}\sum_{i=1}^{N} Y^{(N)}_{i,mod}(t) = \frac{1}{N}\sum_{i=1}^{N} X^{(N)}_{i,mod}(t).$$

Therefore, by the Markov inequality,

$$\mathbb{P}\{\pi^N \in F_{m_\epsilon}\} = 1 - \mathbb{P}\left\{\frac{1}{N}\sum_{i=1}^{N} Y^{(N)}_{i,mod} > m_\epsilon\right\}$$
$$\geq 1 - \mathbb{E}\left\{\frac{1}{N}\sum_{i=1}^{N} Y^{(N)}_{i,mod}\right\}/m_\epsilon$$
$$\geq 1 - \frac{1}{(1-\lambda)m_\epsilon} = 1 - \epsilon,$$

with $m_\epsilon = \frac{1}{(1-\lambda)\epsilon}$ for all $N$.

**Wouter Minnebo (wouter.minnebo@ua.ac.be) received his M.Sc. degree in Computer Science from the University of Antwerp (Belgium) in 2011. In October 2011, he joined the "Performance Analysis of Telecommunication Systems" research group as a PhD-student, at the Mathematics and Computer Science Department of the University of Antwerp. His main research interests include the performance analysis of load balancing and sharing strategies in large distributed networks.**

**Benny Van Houdt (benny.vanhoudt@ua.ac.be) received his M.Sc. degree in Mathematics and Computer Science, and a PhD in Science from the University of Antwerp (Belgium) in July 1997, and May 2001, respectively. From August 1997 until September 2001 he held an Assistant position at the University of Antwerp. Starting from October 2001 onwards he has been a postdoctoral fellow of the FWO-Flanders. In 2007, he became a professor at the Mathematics and Computer Science Department of the University of Antwerp, where he is a member of the PATS research group. His main research interest goes to the performance evaluation and stochastic modeling of computer systems and communication networks. He has published several papers, containing both theoretical and practical contributions, in a variety of international journals (e.g., IEEE JSAC, IEEE Trans. on Inf. Theory, IEEE Trans. on Comm., Performance Evaluation, Journal of Applied Probability, Stochastic Models, Queueing Systems, etc.) and in conference proceedings (e.g., ACM Sigmetrics, Networking, Globecom, Opticomm, ITC, etc.).**