Economic Implications of Energy-Aware Pricing in Clouds

Antonis Dimakis, Alexandros Kostopoulos(✉), and Eleni Agiatzidou

Network Economics and Services Research Group, Department of Informatics, Athens University of Economics and Business, Athens, Greece
{dimakis,alexkosto,agiatzidou}@aueb.gr

Abstract. Cloud computing is a promising approach for delivering ICT services by improving the utilization of data centre resources. One candidate solution for accomplishing energy efficiency within clouds is the adoption of energy-aware pricing by the cloud service providers. In this paper, we compare the economic implications of the choice of pricing schemes under different scenarios.

Keywords: Energy-Awareness · Cloud computing · Pricing · Economics

1 Introduction

Cloud computing has received considerable attention as a promising approach for delivering ICT services. One candidate solution for accomplishing energy-efficiency is the adoption of energy-aware pricing by the cloud service providers. Charging cloud services based on energy could potentially provide the necessary incentives to the customers for achieving a more efficient resource usage.

Pricing in cloud computing has been studied extensively in the past (see [2] and references therein) and most approaches consist of a combination of a fixed or variable price per VM instance and an additional usage charge based on the actual use of computing resources such as CPU cycles, network bandwidth, memory and storage space. Recently, [3, 4] proposed pricing schemes which incorporate direct energy consumption charges. In [3] the authors do not focus on the economic implications of the proposed scheme, while [4] proposes a demand-response mechanism which the cloud employs to cope with the variability in electricity prices.

In our recent paper [1], we proposed a novel pricing scheme based on energy consumption of cloud resources. In this *two-part tariff energy-based pricing* scheme, the actual form of the price is comprised by two parts: a *fixed* one depending only on static information of a VM, and a *dynamic* one, which depends on its average power usage. For comparison, we have also considered *static pricing*, whereby the price is selected based on VM characteristics and does not vary in time.

To evaluate the effect of pricing, one needs to consider the actions taken by all the economic agents involved. For example, a price increase by an IaaS provider does not necessarily lead to an increase in its profits, as the demand of applications for VMs might drop considerably. For this reason, we consider a microeconomic model, which

C. Pham et al. (Eds.): GECON 2017, LNCS 10537, pp. 132–144, 2017.
DOI: 10.1007/978-3-319-68066-8_11

incorporates the actions of IaaS/PaaS providers, applications and their users. Since an action of any of these agents triggers a chain of subsequent responses by the others, we are interested in determining the equilibrium of such interactions.

The goal of our analysis is to compare the economic implications of the choice of pricing schemes by a service provider. In particular, our aim is to compare the static and energy-based pricing schemes proposed in [1]. To do this we consider models of cloud service providers sharing the same capabilities and the same cost structure, their only difference being the pricing scheme adopted by each. The economic quantities we consider are the level of (i) *profits for each type of provider*, (ii) *payments made by the customers of each provider type*, (iii) *overall satisfaction of the customers of each provider type*. Since the comparison depends on the market structure, we consider the actions of service providers under *monopoly* and *perfect competition*. We prove that charging VM energy in addition to a flat fee per VM, as done by the two-part tariff, is optimal for IaaS/PaaS providers in a monopoly market, as well as under competition. Similarly, we show that the profits of SaaS providers are higher when their applications are energy-aware too.

2 Model

IaaS providers: each has an infinite number of physical servers at his disposal. Each server is populated by VMs belonging to possibly different applications and the CPU speed is split equally among the VMs. Let v_i be the number of VMs used by application i. The provider is able to freely scale, i.e., the server consolidation policy is such that the number of *active* physical servers m scales in proportion to the number of VMs in the infrastructure, i.e., $\sum_i v_i/m = \rho$, where the constant ρ is the *consolidation degree*. If the CPU speed of a physical server is μ then μ/ρ is the CPU speed dedicated to each VM running in the infrastructure.

We consider a two-part tariff specified by the parameters π_0, π_1 where π_0 is the static price (in €/hour) and π_1 is the energy price (in €/watthour). Notice that a static pricing scheme has $\pi_1 = 0$. The profit per unit of time (in €/hour) for the provider is

$$\pi_0 \sum_i v_i + \pi_1 \sum_i P_i(v_i) - \pi_e \sum_i P_i(v_i) - c(m) \tag{1}$$

where $P_i(v_i)$ is the average power (in watts) consumed by the i-th application when it uses v_i VMs. π_e is the price per watthour charged by the energy provider. $c(m)$ is the per hour maintenance cost involved in operating m servers; we assume it is linear, i.e., $c(m) = cm$ for some constant $c > 0$. More specifically,

$$P_i(v_i) = p_0 m \frac{v_i}{\sum_j v_j} + p_1 \lambda_i(v_i) = \frac{p_0 v_i}{\rho} + p_1 \lambda_i(v_i)$$

where p_0 is the host's base power consumption (while no application workload is executed), p_1 is the energy in watt hours consumed in the execution of each application

request (or per CPU instruction) excluding base consumption, and $\lambda_i(v_i)$ is the throughput of application i expressed e.g., in requests (or CPU instructions) per second.

PaaS providers: we assume they are not economic agents on their own; rather they follow the strategies of IaaS providers. This is the case for example, when the PaaS layer is offered by the same economic entity, which offers the IaaS. Thus, whenever we refer to IaaS we mean the combination of IaaS/PaaS. In a more complete model, we would have considered the case where the PaaS providers are separate economic agents, which follow their own strategies.

User demand for application requests: each application i has a different throughput demand (rate of instructions or requests to be executed at the VMs of this application) λ_i^{max} which decreases to 0 if the average processing delay of each instruction/request becomes too high. In particular, we assume each request derives a benefit $R_i - \beta_i d_i(\lambda_i)$ from its execution, where R_i, β_i are constants and $d_i(\lambda_i) = 1/\left(\frac{\mu}{\rho} - \frac{\lambda_i}{v_i}\right)$ is the average processing delay based on an M/M/1 queueing model. According to this model, the benefit decreases as response delay increases. If the delay becomes too great, the benefit will become negative and requests will start balking at this point. Thus, either $R_i > \beta_i d_i\left(\lambda_i^{max}\right)$ and $\lambda_i(v_i) = \lambda_i^{max}$, or $R_i = \beta_i d_i(\lambda_i(v_i))$, i.e., $\lambda_i(v_i) = \left(\frac{\mu}{\rho} - \frac{\beta_i}{R_i}\right)v_i$. More compactly: $\lambda_i(v_i) = \min\left\{\left(\frac{\mu}{\rho} - \frac{\beta_i}{R_i}\right)v_i, \lambda_i^{max}\right\}$.

Applications: Consider application i employing v_i VMs. The profit per unit of time for the SaaS provider of this application is assumed to be given by $r_i\lambda_i(v_i) - \pi_0 v_i - \pi_1 P_i(v_i)$, where π_0, π_1 are the parameters of the two-part tariff employed by the IaaS provider, $\lambda_i(v_i)$ is the throughput of requests served by application i, and r_i is the revenue per completed request (e.g., in €/request).

The application decides how many VMs to buy from a particular IaaS provider such that its profit is maximized. Observe that it will never use more than $\lambda_i^{max}/\left(\frac{\mu}{\rho} - \frac{\beta_i}{R_i}\right)$ VMs needed to attain the maximum demand, since additional VMs only increase payments to the IaaS provider without a corresponding increase in application revenues. Thus the profit maximization problem for the SaaS provider of the i-th application is:

$$\begin{aligned} &\max\ r_i\lambda_i(v) - \pi_0 v - \pi_1 P_i(v) \\ &\text{over} \quad 0 \le v \le \lambda_i^{max}/\left(\tfrac{\mu}{\rho} - \tfrac{\beta_i}{R_i}\right) \end{aligned} \tag{2}$$

Since $\lambda_i(v), P_i(v)$ are linear functions of v in $0 \le v \le \lambda_i^{max}/\left(\frac{\mu}{\rho} - \frac{\beta_i}{R_i}\right)$, the optimal number $v_i(\pi_0, \pi_1)$ of VMs is either 0 or $\lambda_i^{max}/\left(\frac{\mu}{\rho} - \frac{\beta_i}{R_i}\right)$. It is nonzero whenever the slope of the objective function in (3) is nonnegative, i.e.,

$$r_i\left(\frac{\mu}{\rho} - \frac{\beta_i}{R_i}\right) \ge \pi_0 + \pi_1\left[\frac{p_0}{\rho} - p_1\left(\frac{\mu}{\rho} - \frac{\beta_i}{R_i}\right)\right] \tag{3}$$

In this idealized model, the number of VMs v can take any real positive value. Although this is done for simplicity, we note that a discrete model would not add anything important to our understanding, as we are mainly interested in fundamental properties of these systems. Apart from that, a continuous model is accurate for applications using a large number of VMs.

In Sect. 3, we first analyse whether energy-awareness of IaaS/PaaS providers is profitable for IaaS/PaaS and SaaS providers in the case where the latter are not energy-aware in the sense that they do not take decisions (e.g., which tasks to schedule on which VMs) on the basis of energy consumption. (Note however that they do get to decide which IaaS/PaaS provider to use on the basis of total price charged; this depends on whether the pricing scheme is energy-based or not). Section 4 considers whether energy-awareness of SaaS providers is profitable for both themselves and IaaS/PaaS providers.

3 Energy-Awareness of IaaS/PaaS Providers

3.1 Monopoly

Since the two-part tariff has two degrees of freedom while the static pricing scheme is one-dimensional (since $\pi_1 = 0$), the maximum profit achieved by an IaaS/PaaS provider acting as a monopolist is never below its profits if a static pricing scheme is used instead.

Actually, *a two-part tariff yields strictly higher profits* as the following simple example shows. Consider the case of two applications with the parameters $\pi_e, p_0, p_1, \rho, \mu, \beta_i, R_i, r_i$ satisfying

$$\pi_e\left(\frac{p_0}{\rho} - p_1\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right)\right) > r_2\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right) > r_1\left(\frac{\mu}{\rho} - \frac{\beta_1}{R_1}\right) > \pi_e\left(\frac{p_0}{\rho} - p_1\left(\frac{\mu}{\rho} - \frac{\beta_1}{R_1}\right)\right)$$

Let us compute the profits of a monopolist using the static pricing scheme (where $\pi_1 = 0$) due to application 2:

$$\begin{aligned}\pi_0 v_2 - c\frac{v_2}{\rho} - \pi_e\left[p_0\frac{v_2}{\rho} + p_1 v_2\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right)\right] &\le r_2 v_2\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right) - \pi_e\left[p_0\frac{v_2}{\rho} + p_1 v_2\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right)\right] \\ &\le r_2 v_2\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right) - \pi_e\left[p_0\frac{v_2}{\rho} - p_1 v_2\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right)\right]\end{aligned}$$

where the first inequality follows from (3) if $v_2 > 0$. (If $v_2 = 0$ then the profits due to application 2 are obviously zero). By our selection of parameter values, the profits due to 2 are strictly negative if $v_2 > 0$. Thus, a monopolist who uses static pricing clearly would not want to serve application 2 since he will suffer losses.

Now if application 1 demands a positive number of VMs under static pricing, condition (3) (with $\pi_1 = 0$) implies $r_1\left(\frac{\mu}{\rho} - \frac{\beta_1}{R_1}\right) \ge \pi_0$ holds. But then $r_2\left(\frac{\mu}{\rho} - \frac{\beta_2}{R_2}\right) > \pi_0$ must also hold, i.e., application 2 also demands a strictly positive number of VMs.

Thus under the static pricing scheme, (3) implies that it is not possible to avoid including application 2.

This does not happen under a two-part tariff with $\pi_0 = 0, \pi_1 > \pi_e$, since then $\pi_1(p_0/\rho - p_1(\mu/\rho - \beta_2/R_2)) > r_2(\mu/\rho - \beta_2/R_2)$ (i.e., application 2 is excluded) but $r_1(\mu/\rho - \beta_1/R_1) > \pi_e(p_0/\rho - p_1(\mu/\rho - \beta_1/R_1))$ (i.e., application 1 is included) and so strictly higher profits result.

As a numerical exposition, we evaluate the profits of a monopolistic provider given by (1), under two scenarios: in the first the provider employs a two-part tariff, while in the second it uses a static price. The parameter values are $R_1 = R_2 = 20, r_1 = r_2 = 1.5, \rho = 10, \pi_e = 0.285, p_0 = 10, p_1 = 5, \lambda_1^{max} = \lambda_2^{max} = 50, \mu = 50, c = 0$.

Figure 1(a) depicts the profits as a function of the maximum average request response delay tolerated by the users of application 1 (normalized by the max tolerated delay for application 2). The profits brought by the two-part tariff are always greater than those brought by the static pricing scheme. They coincide only if the quality-of-service characteristics of the two applications are the same. The greater the diversity between the applications is, the greater the difference in their profits.

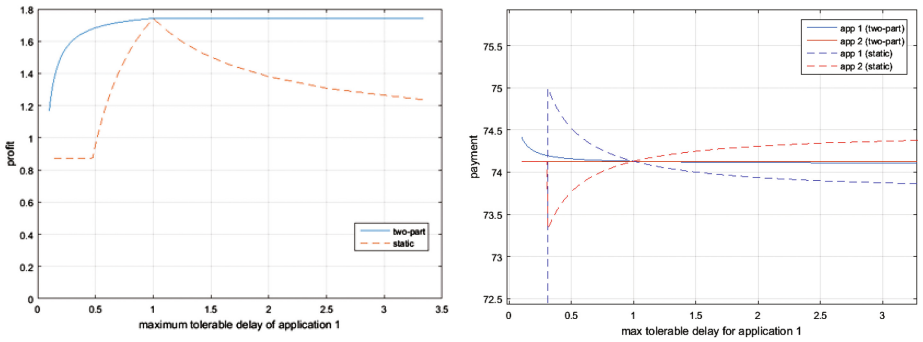

Fig. 1. **(a)** IaaS provider profits in a monopoly – Using a two-part tariff incorporating energy charges (solid curve), and a static pricing (dashed); **(b)** Comparison of payments by two applications to IaaS providers as a function of application QoS diversity.

3.2 Competition

In this section we show that for an IaaS/PaaS provider, *charging VM energy in addition to a flat fee per VM, as done by the two-part tariff, is optimal under competition*: at equilibrium prices only this type of IaaS/PaaS providers offers the maximum possible profits to SaaS providers without suffering losses (i.e., negative profits) himself.

Under ideal competition without entry costs, no IaaS provider is able to make strictly positive profits because in that case he is left without demand. This is because the demand is attracted by other providers, which choose to operate at a smaller albeit nonzero profit margin by slightly reducing their prices. Thus at market equilibrium, competitive IaaS/PaaS providers obtain zero profits and barely cover their costs. Since we are interested in comparing the effect of the pricing scheme on competition, we will compare IaaS providers under the same characterizing parameters (i.e. ρ, μ, p_0, p_1)

except those concerning their pricing scheme. Further, we assume that all IaaS providers face the same maintenance and energy costs, i.e., the c, π_e parameters are common.

We say that the IaaS provider is *competitive for applications of type i* if he makes zero profits from applications of this type, i.e., $\pi_0 v_i + \pi_1 P_i(v_i) = \frac{c}{\rho} v_i + \pi_e P_i(v_i)$ for any $v_i > 0$ with $v_i \leq \lambda_i^{max}/(\mu/\rho - \beta_i/R_i)$. (Since the profits from application i are linear in v_i, it suffices that the previous equality holds for a single v_i for it to hold over the entire range.) Observe that from application types for which an IaaS provider is competitive, the latter is able to attract a nonzero demand. This is because any application of this type pays exactly the minimum possible costs, as all IaaS providers have the same characteristics (apart of their pricing scheme). Thus, if an IaaS/PaaS provider charges prices $\pi_0 = c/\rho$, $\pi_1 = \pi_e$, i.e., charging by the true factor cost he is facing his own, he is competitive for any application irrespective of its type. This is obvious as the equation defining competitiveness is trivially satisfied for any application. In this tariff, the true energy price is passed onto the application, while the flat fee part covers the per server maintenance costs.

In contrast, IaaS providers, which use static pricing, can only be competitive for a single application type in general. This follows since $\pi_0 v_i = v_i c/\rho + \pi_e P_i(v_i)$ is possible only for $\pi_0 = c/\rho + \pi_e(p_0/\rho + p_1(\mu/\rho - \beta_i/R_i))$, using the definitions of $P_i(v_i), \lambda_i(v_i)$. Thus, the only static price which makes the IaaS provider competitive to application i depends on the application type through β_i/R_i. This means that the static price used by IaaS providers not charging energy, targets competition for a narrow set of applications. In order for these providers to attract more application types they need to offer multiple statically priced plans so that applications can select the one who find more profitable. This is essentially a pricing strategy which tries to emulate energy-based pricing using application-level information (i.e., β_i/R_i) which the IaaS provider is difficult to obtain or guess. In contrast, true energy-based pricing which uses application-independent prices is a more robust strategy by relying on industry-wide factor costs.

As an exposition of the competition between IaaS providers and the effect of the pricing scheme, we consider an example, which examines the profits of two applications as a function of their diversity. We assume the users of the applications do not tolerate average request response delays above some value, which is specific to each application. Figure 1 (b) depicts the payments per time unit incurred by each application under two different pricing schemes: i) the static price scheme, which does not take energy consumption into account, and ii) the two-part, which incorporates energy consumption. The parameter values used are $R_1 = R_2 = 20, r_1 = r_2 = 1.5, \rho = 10, \pi_e = 0.285, p_0 = 10, p_1 = 5, \lambda_1^{max} = \lambda_2^{max} = 50, \mu = 50, c = 0$. The price parameters of each scheme are chosen under the assumption of ideal competition, i.e., they are chosen as described in the previous section. The horizontal axis represents the maximum tolerable delay by users of application 1 (normalized to that of application 2).

For stringent delay requirements, when max tolerable delay is less than 0.3, application 1 does not at all use the provider with static pricing since the high costs outweigh benefits. The latter hosts application 2 only, at a competitive price. When the delay requirements of application 1 are not so stringent, the demand rises and

application 1 starts using the static provider, but at a cost which is not competitive: application 1 payments exceed the ones offered by the provider employing a two-part tariff. As applications become less diverse (i.e., max tolerable delay close to 1) the two providers are equally attractive, although the provider offering the two-part tariff is slightly more. For values of the max tolerable delay above 1, the less tolerable users belong to application 2 now, and they bare most of the costs in both providers. Nevertheless, the static provider continues not to be competitive as the payments resulting for application 2 exceed those by the provider employing the two-part tariff.

In Fig. 2 the aggregate profits over all applications is depicted for the two-part tariff and the static pricing scheme. The profits under static pricing may decrease if some applications have stringent delay requirements.

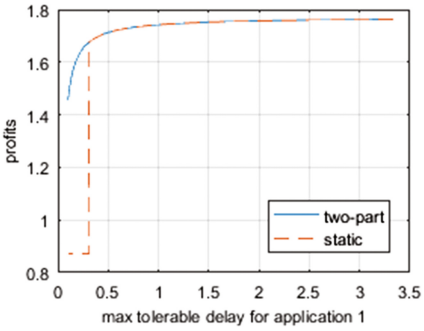

Fig. 2. Comparison of SaaS provider profits under IaaS providers competition.

4 Energy-Awareness of SaaS Providers

In this section, we analyse whether energy-awareness of SaaS providers is economically sensible. In order to make the effects of energy-awareness clearly visible, we refine the model in Sect. 2 to allow for (i) *physical hosts with different power efficiency*, (ii) *requests with different energy consumption*.

Such situations are quite common; in the sequel we consider the simplest possible case with two different host types (with the one being more power efficient) and two request types (with the one being more energy consuming).

4.1 Assumptions

Host power efficiency: The efficiency parameters of the two host types are given by (Table 1):

For simplicity, both host types consume the same power while their CPU idles. While active, type i consumes p_1^i extra power where we assume type 1 is more efficient, i.e., $p_1^1 < p_1^2$.

Table 1. Efficiency parameters of the two host types.

Host type	Idle power consumption	Active power consumption	# of hosts
1	p_0	p_1^1	H_1
2	p_0	p_1^2	$H_2 = \infty$

VM scheduling: The fact that type 1 hosts are more power efficient has an implication for the VM scheduling policy of the IaaS provider. Since the latter strives to have minimal energy costs, more power efficient hosts are preferred to less efficient ones. Thus, the VM scheduling will try to allocate type 1 hosts first to meet demand; type 2 hosts will be used only if it is not possible to meet demand only by utilizing type 1 hosts. Since the VM scheduler maintains a fixed number ρ of VMs per host, the maximum number of VMs that can be carried by type 1 hosts is ρH_1. Thus if an application employs v VMs, the number of these hosted in type 1 hosts is $\min(v, \rho H_1)$ while $\max(v - \rho H_1, 0)$ are hosted in type 2 hosts. (This is under the assumption that the VM scheduling algorithm is allowed to freely reallocate all VMs on the available hosts.) Note that if there were an infinite number of hosts for both types, the VM scheduling would never use type 2 hosts. Thus, we assume the number H_1 of type 1 hosts is finite. As in Sect. 2, to simplify the analysis the number H_2 of type 2 hosts is assumed to be infinite.

Application request types: All requests are categorized in two types described by the following parameters (Table 2):

Table 2. Request categorization.

Request type	Relative power consumption due to a unit rate of requests (normalized to type 2)	Proportion of total requests	Power consumption due to a unit rate of requests
1	$w_1 > 1$	θ_1	$p_1^i w_1$
2	$w_2 = 1$	$\theta_2 = 1 - \theta_1$	$p_1^i w_2$

We assume a unit rate of type 1 requests consumes $w_1 > 1$ times the one of type 2. The precise power consumption depends on the host type the request is executed, so the average power consumption is $p_1^i w_1$ if executed on a type i host. Let the total request rate be $\lambda(v)$ when the application employs v VMs, where we have dropped the subscript since we consider a single application. Then the power consumption (excluding idle power) due to type 1 requests is $\theta_1 \lambda(v) p_1^i w_1$ if all were executed on type i hosts. If both host types are used, the average power consumption is given by a corresponding linear combination.

Request scheduling by the application: Here we consider the implications in power consumption due to the application being energy-aware or not. First we consider the "legacy" case, where an application has no information about the power consumption

of its components. In this case, the application cannot differentiate between the more and less energy consuming request types. Moreover, it cannot have information about the energy efficiency of its VMs. Thus, the requests are scheduled on VMs independently of their type.

Now each VM receives requests of any type at rate $\lambda(v)/v$, where v is the total number of VMs. A proportion θ_j of those are type j, and so their power consumption is $p_1^i w_j$. Thus the power consumed by a VM (excluding the idle state) running on host type i is $p_1^i \frac{\lambda(v)}{v}(\theta_1 w_1 + \theta_2 w_2)$. Considering the VM scheduling algorithm outlined above, the power consumption $P(v)$ of the entire "legacy" application, including power consumption in the idle state, is:

$$P(v) = \frac{p_0 v}{\rho} + [p_1^1 \min(v, \rho H_1) + p_1^2 \max(v - \rho H_1, 0)] \frac{\lambda(v)}{v} \sum_j \theta_j w_j \tag{4}$$

Let us now consider how an energy-aware application allocates requests on its VMs. Since type 1 hosts are more power efficient and type 1 requests are more energy consuming (as $w_1 > w_2$), an energy-minimizing scheduling policy ought to place type 1 requests on type 1 hosts and use type 2 hosts only if necessary or for serving (the less consuming) type 2 requests. Under such a policy, the power consumption of the energy-aware application is given by

$$\begin{aligned} P(v) =& p_0 \frac{v}{\rho} + \frac{\lambda(v)}{v} \{\min(\theta_1 v, \rho H_1) p_1^1 w_1 + \max(\theta_1 v - \rho H_1, 0) p_1^2 w_1 \\ &+ [\min(v, \rho H_1) - \min(\theta_1 v, \rho H_1)] p_1^1 w_2 + \max(v - \max(\theta_1 v, \rho H_1), 0) p_2^2 w_2\} \end{aligned} \tag{5}$$

4.2 Monopoly

Let $v(\pi_0, \pi_1)$ be the optimal number of VMs requested by the application which is obtained by solving the optimization problem (3), where we have dropped the subscript since we have only one application. The IaaS/PaaS provider chooses prices π_0, π_1 which maximize his profits, i.e., he solves:

$$\max \pi_0 v(\pi_0, \pi_1) + \pi_1 P(v(\pi_0, \pi_1)) - c \frac{v(\pi_0, \pi_1)}{\rho} - \pi_e P(v(\pi_0, \pi_1))$$
$$\text{over } \pi_0, \pi_1 \geq 0$$

In Fig. 3 (a), we numerically solve the above problem and depict the maximum profits for the monopolist as a function of the number of power efficient hosts H_1. All curves in Fig. 3 (a) were produced under the parameters: $p_0 = 1$, $p_1^1 = 1$, $p_1^2 = 3$, $w_1 = 3$, $w_2 = 1$, $\theta_1 = 0.5$, $\rho = 0.5$, $\mu = 1$, $\lambda^{max} = 50$, $\beta = 1$, $R = 2$, $r = 1$, $c = 0.1$.

The solid curve corresponds to the case where the application is energy aware, and the dashed curve is for a "legacy" application. Energy awareness at the application level increases profits for any choice of parameters. The relative increase is at most 10%, when the energy price of the energy provider is $\pi_e = 0.05$. For cheaper energy, energy-awareness brings a smaller profit increase to the IaaS/PaaS provider.

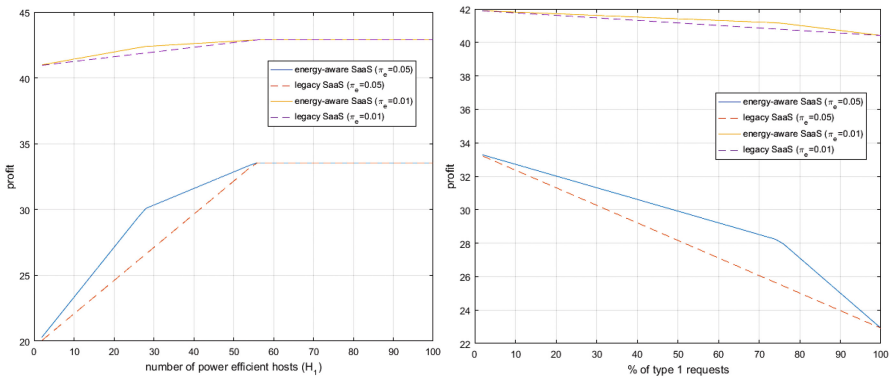

Fig. 3. **(a)** IaaS/PaaS provider profits in the case of monopoly as a function of the number of power efficient hosts; **(b)** IaaS/PaaS provider profits in a monopoly as a function of the workload mix.

For low numbers of type 1 hosts, the profits are almost identical as the majority of VMs are hosted in type 2 hosts. As the number of type 1 hosts increases, the energy-saving effect of the scheduling of requests performed by the application becomes more significant. Beyond $H_1 = 55$ there is no profit difference as all requests are served by type 1 hosts and request scheduling does not have any effect, since VM scheduling makes sure only the power efficient hosts are utilized.

It is interesting to see where the profit increase is coming from: is it because applications need to pay more or it is mostly due to a decrease in energy costs? For all parameters in Fig. 3 (a) the applications' payments are constant (and equal to 50), so the difference in profits is due to energy savings. The magnitude of the savings seems to be greater for higher energy costs ($\pi_e = 0.05$ case).

In Fig. 3 (b), we again compare profits but now as a function of the percentage of energy consuming requests, i.e., the parameter θ_1 as it ranges from 0 to 1, for $H_1 = 50$. Type 1 requests can be thought as being more CPU intensive (since they consume more energy), while type 2 as more RAM intensive. Therefore, Fig. 3 (b) shows the effect of the workload mix in profits.

All profits are decreasing in θ_1 as type 1 requests are more energy consuming. Again, *the profits with energy-aware applications are higher*. The relative profit increase due to energy-awareness is observed at approximately $\theta_1 \approx 73\%$, which involves a mix of both request types.

Figure 4 (a) depicts IaaS/PaaS provider profits in a monopoly for energy-aware (solid curves) and "legacy" (dashed) SaaS providers, as a function of how much more

energy consuming type 1 requests are relative to type 2, i.e., the w_1 parameter. The profit difference increases as the energy difference between the request types increases. At $w_1 = 10$ the profit gain due to energy awareness is 20%.

In Fig. 4 (b), we show the profits as the function of the power consumption of type-2 hosts, i.e., the parameter p_1^2. As p_1^2 increases, so type-2 hosts become less efficient, the profit gain (for IaaS providers when hosting energy-aware with respect to legacy SaaS applications) increases until a deflection point around $p_1^2 = 5.6$ where the gain start to decrease. At this point, the type-2 hosts become too expensive so the "legacy" application (which is hit most by energy costs) drops its demanded VMs such that it ceases to use type-2 hosts. (This is why the profit of the "legacy" application remains constant after $p_1^2 = 5.6$: the p_1^1 parameter is constant and no type-2 hosts are used.) On the other hand, the energy-aware application always satisfies its maximum demand λ^{max}. Of course, if p_1^2 increases beyond the range shown in the figure, the energy-aware application will lower its demand as well and meet the profit line of the "legacy" application.

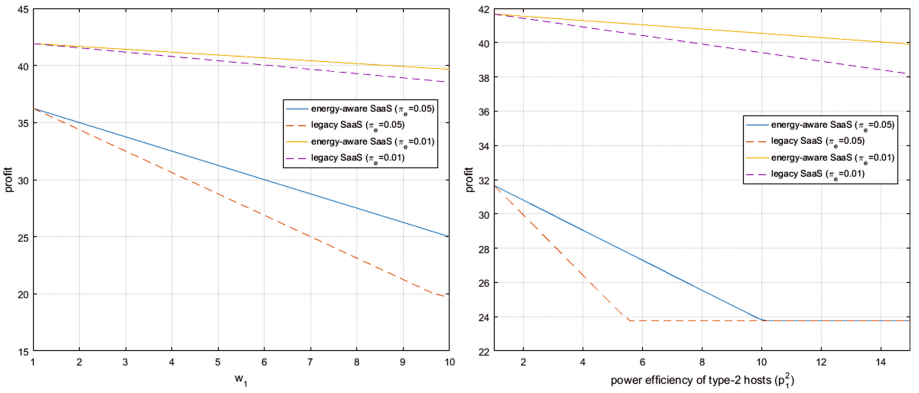

Fig. 4. **(a)**. IaaS/PaaS provider profits in a monopoly as a function of w_1; **(b)** IaaS/PaaS provider profits as a function of the power efficiency of type 2 hosts.

Given the attractive properties for the IaaS/PaaS provider that application level energy awareness has, we conclude that he has the incentive of sharing some of the cost involved for the applications to adopt energy-aware technologies. In particular, the introduction of energy awareness at the application level can have important (up to 20%) gains in IaaS/PaaS provider profitability, and does not increase the payments made by applications to IaaS compared to the "legacy" case. Additionally, the profit gains are due to energy savings resulting from scheduling diverse requests to diverse hosts, executed by the application. The more diverse the requests and hosts are the more significant the effect of application-level scheduling becomes. Finally, when either the requests consume similar energies, or the hosts have similar power efficiencies, the additional optimization performed by application does not have a significant effect.

4.3 Competition

When IaaS/PaaS providers compete with each other with no entry costs, they have zero profit margins as explained in Sect. 3. Applications however have strictly positive profits and we will see that their profits increase by being energy-aware. As the analysis of the competitive case in Sect. 3 does not depend on the precise form of power consumption function $P(v)$, the same results regarding equilibrium prices carry over to the present case, i.e., the market prices are $\pi_0 = c/\rho$, $\pi_1 = \pi_e$.

Given the market prices, an application solves problem (3) for $\pi_0 = c/\rho$, $\pi_1 = \pi_e$ to obtain the maximum profits. We show that (4) is greater than (5) for any v, and so energy-awareness increases application profits. Notice that one can move from the legacy allocation of type 1 requests, where these are distributed equally among all VMs (irrespective of the host they are running on), to the allocation produced by energy-awareness, by shifting small loads of type 1 requests that reside on any VMs on type 2 hosts to VMs on type 1 hosts. If we move a small load ϵ then the change in the total power is $-p_1^2\epsilon w_1 + p_1^1\epsilon w_1$. To keep the load of each VM balanced, the previous shift is complemented by another shift of size ϵ in the reverse direction, of type 2 requests from the VM running on the type 1 host to the VM on the type 2 host. The change in the power due to the reverse move is $-p_1^1\epsilon w_2 + p_1^2\epsilon w_2$. The total power difference is $-p_1^2\epsilon w_1 + p_1^1\epsilon w_1 - p_1^1\epsilon w_2 + p_1^2\epsilon w_2 = \epsilon(w_2 - w_1)\left(p_1^2 - p_1^1\right) < 0$, since $w_2\langle w_1, p_1^2\rangle p_1^1$. Thus, the total move yields a decreased power and so (4) is greater than (5). We conclude that *application level energy-awareness increases applications' profits*. To get a sense of the magnitude of the profit increase we numerically evaluate profits.

In Fig. 5 (a), we show the application profits for energy-aware and "legacy" applications. As expected by the previous argument, the profits of energy-aware applications surpass those of "legacy". (The parameters values were the same as those in the previous section). The maximum gain (of about 11%) is obtained for high energy costs, $\pi_e = 0.05$. The gain is marginal for low costs such as $\pi_e = 0.01$.

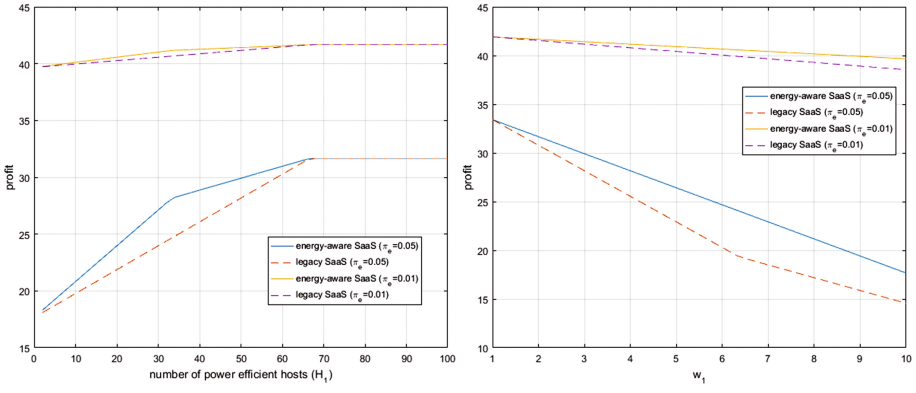

Fig. 5. **(a)** Profits of energy-aware (solid curve) and "legacy" applications (dashed) in competitive markets for IaaS/PaaS; **(b)** Application profits as a function of energy consumption of type-1 requests.

In Fig. 5 (b), we show application's profits as a function of energy consumption of type 1 requests. The profit gain becomes marginal for homogenous requests. Notice though that there is a saturation effect for $\pi_e = 0.05$ around $w_1 = 6.3$: for too large values of w_1 the energy savings due to utilization of type 1 hosts are dominated by the high energy consumption of type 1 requests on type 2 hosts. In this case, the number of type 1 hosts is too small to completely avoid the high energy consumed by type 1 requests.

We observe that in competitive markets for IaaS/PaaS, energy aware applications extract higher profits from energy-based scheduling of requests, and the profit gain is higher if the request energy characteristics are more diverse. Thus, applications themselves would want to adopt energy-based technologies because they become more profitable if IaaS/PaaS charge according to energy consumption.

5 Conclusions

In this paper, we considered a mathematical model of applications and IaaS/PaaS providers and showed that applications which adapt to energy-based information and the proposed energy-based pricing schemes by appropriately scheduling requests to VMs, extract higher profits compared to being non-adaptive. Although the model is a gross simplification of reality, it is valuable in that it clearly shows the potential economic benefits for applications to respond to appropriate pricing signals. Thus, it is not only that applications become more power efficient once they utilize an energy-aware framework (e.g., ASCETiC [5]), but they have an economic incentive to utilize it. We saw that IaaS/PaaS providers are the likely first adopters of energy-aware layers as it increases their profits even when the application providers are not energy-aware. Even if the aforementioned analysis shows that if SaaS providers adopt the energy-aware SaaS layer they will also see their profits increase, this does not mean that they will adopt an energy-aware framework as they have no means of evaluating the benefit of doing so. Our future work focuses on defining a more complete model, considering the case where the PaaS providers are separate economic agents.

References

1. Kostopoulos, A., Agiatzidou, E., Dimakis, A.: Energy-aware pricing within cloud environments. In: Bañares, J., Tserpes, K., Altmann, J. (eds.) GECON 2016. LNCS, vol. 10382. Springer, Cham (2016)
2. Al-Roomi, M., Al-Ebrahim, S., Buqrais, S., Ahman, I.: Cloud computing pricing models: a survey. Int. J. Grid Distrib. Comput. **6**(5) (2013)
3. Aldossary, M., Djemame, K.: Consumption-based pricing model for cloud computing. In: 32nd Performance Engineering Workshop, Bradford, UK (2016)
4. Wang, C., Nasiriani, N., Kesidis, G., Urgaonkar, B., Wang, Q., Chen, L., Gupta, A., Birke, R.: Recouping energy costs from cloud tenants: tenant demand response aware pricing design. In: Proceedings ACM 6th International Conference on Future Energy Systems. ACM (2015)
5. ASCETiC, EU FP-7 project. http://ascetic-project.eu/