The Report Committee for Joshua Charles Ruedin
Certifies that this is the approved version of the following or report:

Spatial and Temporal Pricing Support for Consumer Services

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

____
Christine Julien

____
K. Suzanne Barber

Spatial and Temporal Pricing Support for Consumer Services

by

Joshua Charles Ruedin, B.S.

Report

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

August, 2011

Abstract

Spatial and Temporal Pricing Support for Consumer Services

Joshua Charles Ruedin, M.S.E.

The University of Texas at Austin, 2011

Supervisor: Christine Julien

Consumer acceptance of buying goods and services online via the Internet is growing, although e-ecommerce has been mostly a mirror of traditional methods of pricing transactions – fixed price or auctions. The proliferation of personal mobile devices with pervasive Internet access and localization capability means a richer set of pricing parameters can be used. Allowing buyers and sellers to more explicitly price requests and filter offers, including information about time and place, allows for better transaction results for both parties. This paper examines the impacts of including the time and place of performance of a service as part of the price. A system for implementation is proposed, a simulation of the system is evaluated, and the results presented.

Table of Contents

List of Tables

List of Figures

Problem Statement and Background

Auction services like Ebay.com provide a mechanism for matching buyers to sellers. For a buyer purchasing a commodity item, price is the main dimension differentiating seller offers. However, there are other transactions where the price a buyer will pay depends on the time or how soon a service can be provided at their location. There are also sellers whose acceptable price is based on their current schedule and how far the place of work is from their current location.

An example scenario is buyer seeking a one-time lawn service. In this case a buyer wants his grass cut, and will pay $30 if it can be done on Friday, for example, so the yard will look nice for the weekend. This buyer may only pay $20 if it is cut any other day that week, and will not pay if it is cut on Saturday or Sunday. Lawn service providers exist throughout the geographic area and are willing to take less if they are not currently working and the job is close by. If the job is far away or if they have other better paying prospects, they will not accept the job.

With the assumption that the parties involved in these transactions each have a mobile device such as a smart phone, a solution to this problem is to provide a website or phone application that allows buyers to request the service, provide the price they are willing to pay, and capture the location where the service will be performed. The service provider accesses the website or phone application and enters times that they wish to accept work requests, and are then notified as requests from buyers are found. Service providers could then see the requests for work, and could provide a bid based on the distance to the job and the requested time of performance. The buyers would be notified as bids for the work are presented by sellers, and could accept or deny the offers.

Several methods of capturing buyers' preferences are discussed by Guttman (1992), including entering buyers' price sensitivity to a variety of factors associated with the specific item being purchased. The focus was on a basic e-commerce model or an extension of catalog or phone ordering. This so called “Kasbah” online agent system described by Guttman allowed for richness in buyer and seller pricing in the example of a market for used textbooks. Buyers could set their desired price, highest acceptable price, and a date by which the transaction should complete, and also adjust the curve of how strongly their offer increases over time. Allowing pricing linked to the timeliness of the transaction is interesting for the used textbook case, as once the semester begins the values drop. The main disadvantage of Guttman’s approach is that there is no automatic way to preferentially purchase textbooks from near one’s own location and not from someone on the other side of campus or at another school.

The goal of the MARI system described by Tewari (2000) was to specifically look at improving online marketplaces that involve the buying and selling of services or non-tangible goods. It provided a mechanism for buyers and sellers to more comprehensively specify their requirements for the service and the transaction, including reputation, expertise, and preferred task completion time. Relative weights were applied to each attribute to help allow for better matching of sellers and buyers. However, there was no mechanism to include pricing linked to timeliness of tasks, to account for the impact of distance between parties in the transaction with location information.

Tewari (2003) extended the ideas presented the MARI system in Tewari (2000) to include personalized location-based brokering in the matching of buyers and sellers. However, this extension lacks a means of variably pricing requests for services to capture the value of timeliness of service. Also, there is an assumption that participants in the system will provide their location information freely, but there is no discussion of security.

The system proposed in this report combines the managed connections of buyers and sellers with detailed price specification in the MARI system of Tewari (2000) and location as developed in Tewari (2003) but further extends those to include time-variable pricing which improves the market efficiency, with additional consideration of reputation and security concerns.

Key Concerns

As discussed, the key concerns of the proposed system are placing a value on the time services are performed via variable pricing, considering location in the matching of buyers and sellers, providing a trust mechanism for tracking reputation of the participants of the system, and addressing concerns about the security of sharing location information.

Placing Value on Time

For buyers of services, the time that a service is performed is sometimes very important. For the case of the lawn service, the buyer may be a homeowner who needs grass mowed, but also has a preference that it is mowed on Friday so it can be enjoyed over the weekend. Such a buyer might be willing to pay a premium for the service to be performed on that specific day, but demand a discount for other days. In another example, consider a ride-sharing service. Say the buyer in this case is a person who needs transportation from his current location to another point. The service provider may be another individual who is willing to take other people along if it is convenient. The buyer's window of acceptable performance of this service depends on how willing he is to wait around to be picked up. He may be willing to pay a premium to be picked up right away, instead of waiting for a traditional fixed-rate taxi-service.

In both these cases, the buyers are willing to pay a premium for the performance of a service during a certain time. If the buyers could specify this preference to the pool of available service providers, there may be a greater chance of getting the work done during their preferred period of performance. Also, the service providers would be able to preferentially perform tasks to maximize their income. In this way, the buyers' satisfaction is increased and the service providers have the opportunity to increase their revenue per service performed. The mechanism for specifying this variable pricing is to provide a vector of the buyer's asking price at each point in time, where the granularity of time depends on the type of service being requested. Figure 1 represents a possible schedule of pricing for a lawn service.

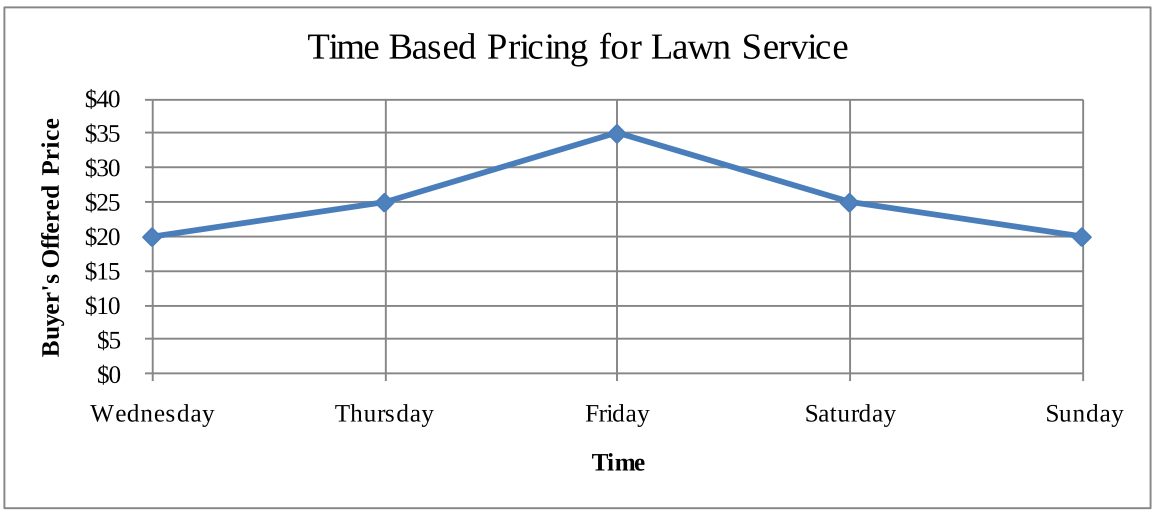

Figure 1: Time Based Pricing for Lawn Service

Here, the required granularity of time is days, and the buyer can specify a price offered for the service each day in a period that service may be performed.

For a service like car-sharing, the granularity of time might be on the order of 5-minute intervals with the desired period of performance starting immediately, as shown in Figure 2.

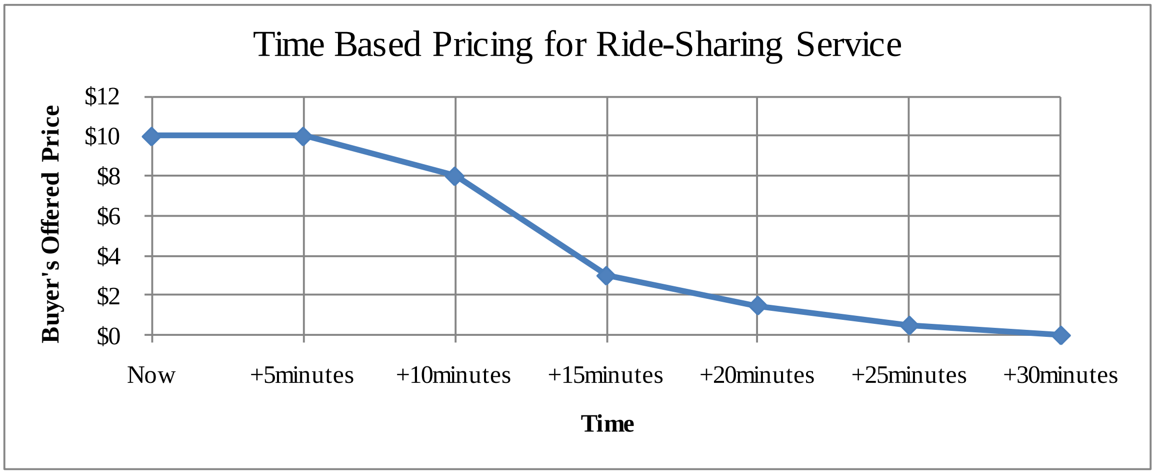

Figure 2: Time Based Pricing for Ride-Sharing

Here, the buyer can price the request for service to include a price incentive for being picked-up immediately; with the offered price dropping the longer the buyer is waiting.

Including Location

For certain types of services, the distances involved in travelling to perform the services are important. Especially for small-value transactions with narrow profit margins, the transportation costs to accept offers from far-away buyers can cut into the profitability of the service providers. Returning to the example of the lawn service, the lawn service provider has a large truck and trailer, and is wary of gasoline expenses. He prefers to do business in his local area to maintain profitability. If this lawn service provider knew the distance to the place of performance and the price the buyer is willing to pay, he could determine whether the profit was sufficient to accept the job. Also, in the face of several offers from buyers, he could prioritize the opportunities and accept the ones that make the most profit. For the scenario of the ride-share service, the service provider may be willing to accept buyer offers if the buyers are close-by and if they wish to travel in the same direction. The same logic would apply to the other types of service transactions where transportation costs are a factor.

The Issue of Trust

Some services are more interested in the reputation of the service provider. The motivating example of the lawn service requires a limited amount of trust in the provider and could be considered a commodity service, compared to hiring a pet sitter or house cleaner. However in either case, buyers may still prefer to deal with service providers with a certain reputation for successful transactions. Of systems for tracking and managing reputation, positive reputation systems, for example those that share the number of positive transactions a buyer or seller has committed, appear to be more effective for online transactions than a negative ones (Yamamoto, 2004). EBay’s reputation system provides information about buyers and sellers positive feedback rating, and leaves it to users to decide how to proceed (Houser and Wooder, 2006). Amazon's Mechanical Turk system, which is a service for automating computing tasks that require human input, provides a mechanism to request workers with a certain number of successfully completed and approved jobs (Kittur, Chi, and Suh, 2008).

To address this, at the end of the transaction, the proposed system sends a message to the buyers asking them to rate the sellers. The response is sent back to the system server where it is maintained and not editable by the service providers. Buyers are able to specify in their service requests a minimum level of successful transactions they required of potential service providers – by percent successful, or total number.

Likewise, service providers have an interest in accepting tasks from buyers who pay their bills. At the completion of the service, the service providers are also sent a message asking them to rate the buyer. The response is sent back to the system server where is it maintained, visible to other service providers when viewing requests from that buyer, and can be used to aid decisions to accept or deny a service request.

Security

Part of the value of this system is in allowing service providers to base their decision on accepting a buyer's request based on how far away the buyer is located. However, buyers may have security concerns about broadcasting their position to an unknown number of unfamiliar service providers.

Using this system to full advantage requires users to share their location so the best matches can be found. In the proposed system, it is clear to users what information is being requested, and why the information is needed. Having clear justification of the scope and intended use for the information improves users' willingness to participate in sharing (Consolvo, 2005) (Brush, 2010). Naturally there are fears of security risks posed by broadcast or publication of this location (Leonhardt, 1998). Approaches to maintaining security in location-based services include intention obfuscation of the location (Duckam, 2005), anonymization of users (Gruteser, 2003), and limiting access of user information to the smallest number of individuals possible to perform a function (Saltzer, 1975).

The proposed approach to addressing this issue is to provide a mechanism at buyers' and service providers' mobile devices to encrypt their location before sending it to the system server. This mechanism includes symmetric encryption based on a key created by the system server and shared between the system server and each user at the time of creation of the user account. Now, the users' position information is only accessible by the system server which only uses the location information to match buyer requests with service providers. Service providers looking for work are only provided the distance to buyers with open service requests, not their location. The location of the buyer is only revealed to the service provider once the service provider has accepted to perform the service for that buyer. The service provider's location is never provided to the buyer. In this way, the locations of users are provided to each other on a need-to-know basis, which follows the security principal of least privilege (Saltzer, 1975).

A System to Support Time and Location Based Pricing

This section describes an implementation of such a system with the assumption that both the buyers and service provides are individuals with mobile devices or smart phones that provide for essentially pervasive internet access and localization. Two applications resident on the participants' mobile devices are proposed: a BuyerApp, and a ServiceProviderApp. Both applications communicate with a SystemServer. The SystemServer performs the coordination of sellers and buyers, determines relative distances between the participants' locations, and also acts as intermediary in any communication between the buyers and sellers.

The BuyerApp resides on the buyer's mobile device. The buyer uses BuyerApp to enter requests for service, provide a price schedule for the request, and for requests where buyer mobility is a factor, BuyerApp sends periodic updates of the buyer's position to the SystemServer. The mechanism for specifying this variable pricing schedule is to provide a vector of the buyer's asking price versus points in time, by table entry on the mobile device.

A ServiceProviderApp resides on the service provider's mobile device and sends the provider's location updates to the SystemServer and also notifies the provider of any buyer offers. The service providers use the ServiceProviderApp to register with the SystemServer and declare what types of services are being offered, as well as the minimum price and the maximum distance he/she is willing to accept.

The sequence diagram of Figure 3 outlines the steps involved.

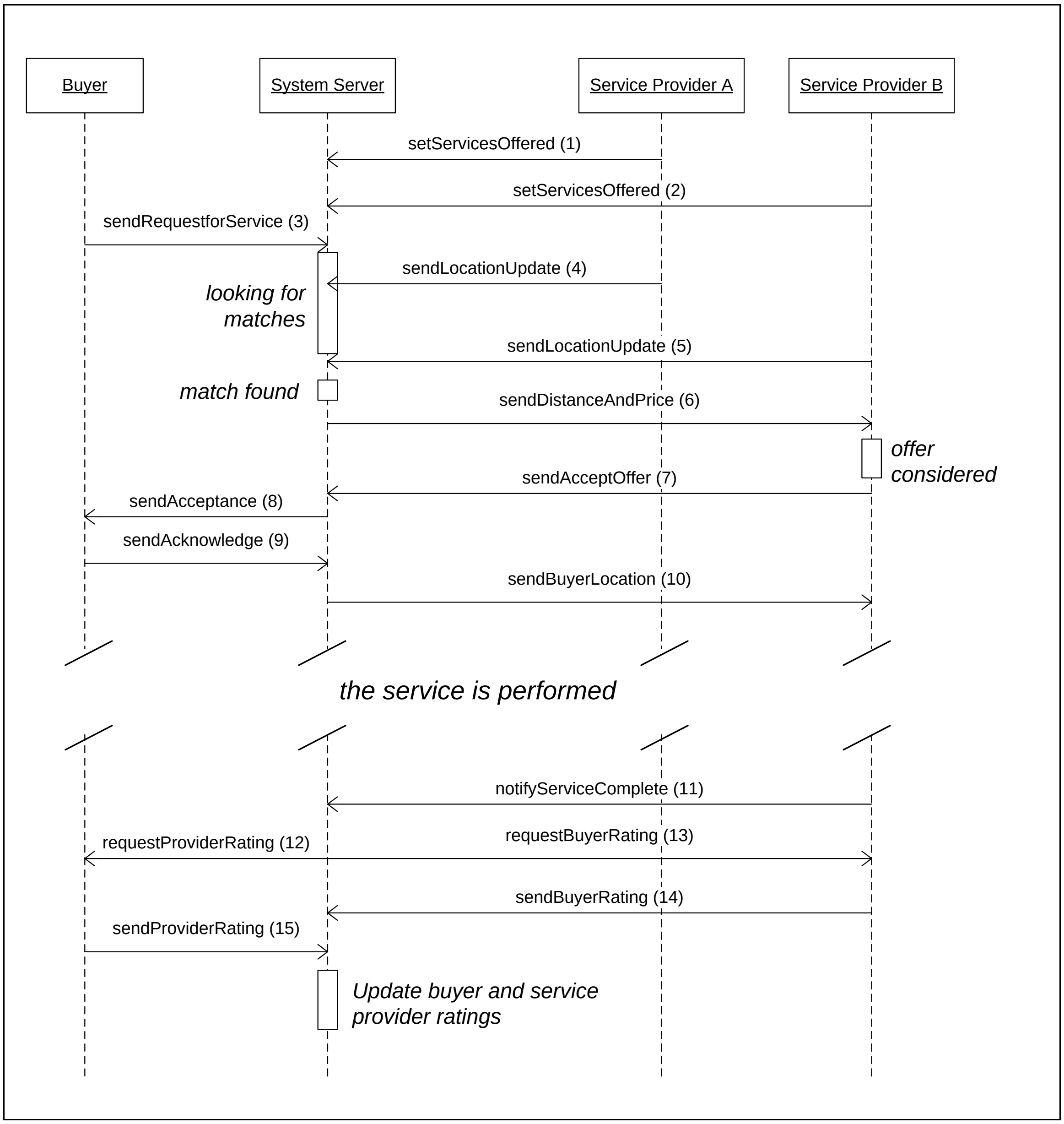

Figure 3: Sequence Diagram of Proposed System Operation

As discussed previously, the actors in this system are the Buyer who uses BuyerApp on his mobile device for making the request for service, the SystemServer that finds matches and coordinates communication between buyers and service providers, Service Provider A and B, who are two separate performers of the service being requested by the buyer and use ServiceProviderApp to communicate with the System Server.

.

The details of the steps in the sequence diagram are as follows –

1. setServicesOffered – Service Provider A uses ServiceProviderApp to register with the SystemServer by sending the type of service offered, the distance willing to travel, and the minimum price willing to accept.
2. setServicesOffered – another provider, Service Provider B, uses ServiceProviderApp to register with the SystemServer, sending the same type of information.
3. sendRequestforService – The Buyer uses BuyerApp to send a request for service to the SystemServer. This includes the type of services requested, and a price schedule that allows the buyer to specify how much he is willing to pay at each point in time in the future.
4. sendLocationUpdate – ServiceProviderApp sends a location update to the SystemServer, indicating Service Provider A's position has changed. The SystemServer checks this position against all outstanding buyer offers to see if the new positions put him in range of any open buyer offers.
5. sendLocationUpdate – the ServiceProviderApp belonging to Service Provider B sends a location update.

This time, the SystemServer determines there is an open buyer offer meeting the service provider B's criteria set in step 2.

6. SendDistanceandPrice – the SystemServer send the distance and buyer's price offer to service provider B via ServiceProviderApp. In the event that multiple service provider matches are found for a single buyer offer, all service providers are notified.
7. sendAcceptOffer - Service provider B considers the buyer's terms, and uses ServiceProviderApp to send an acceptance to the SystemServer. In the event that multiple service provider matches are found for a single buyer offer, all service providers are may respond.
8. sendAcceptance – the SystemServer sends an acceptance message to the BuyerApp to indicate that is request for server has been answered, and the specific time to anticipate the service to be completed. In this implementation, if there are multiple responses from service providers, the first service provider meeting the buyer's requirements is selected as the winner of the job – in this case service provider B.
9. sendAcknowledge – the buyer agrees and uses BuyerApp to sends an acknowledgement of acceptance of service provider B's offer.
10. sendBuyerLocation - the SystemServer now sends the full location of the buyer to the seller.

At some time in the future, the service is performed and the transaction takes place. At this point the sequence diagram continues.

11. notifyServiceComplete – service provider B notifies the SystemServer that the transaction is complete.
12. requestProviderRating - the SystemServer sends a request to the buyer to provide a rating of the service provider B.
13. requestBuyerRating - the SystemServer sends a request to the service provider B to provide rating of the buyer
14. sendBuyerRating – ServiceProviderApp B sends the buyer rating to the SystemServer where is it recorded in the buyer’s record.
15. sendProviderRating. – the buyer sends the rating of service provider B to the SystemServer where it is recorded in the buyer’s record.

Simulation

To evaluate the performance of the system with time and location based pricing, a simulation was constructed in Java. The motivating example of a lawn service was used as the scenario for the simulation. Before a simulation begins, buyers are randomly placed on a two dimensional map. These buyers want their yard serviced in the next two weeks, and would prefer it to be done on a certain day. They register with the system server and enter their pricing schedule. In the case of fixed pricing, the buyer offers $25. In the case of variable pricing, the buyer offers $35 for lawn service to be completed on the preferred day, $25 for the service to be completed on the adjacent two days, and only $20 to be completed two days away from the preferred day.

Figure 4 compares two pricing scenarios for the lawn service.

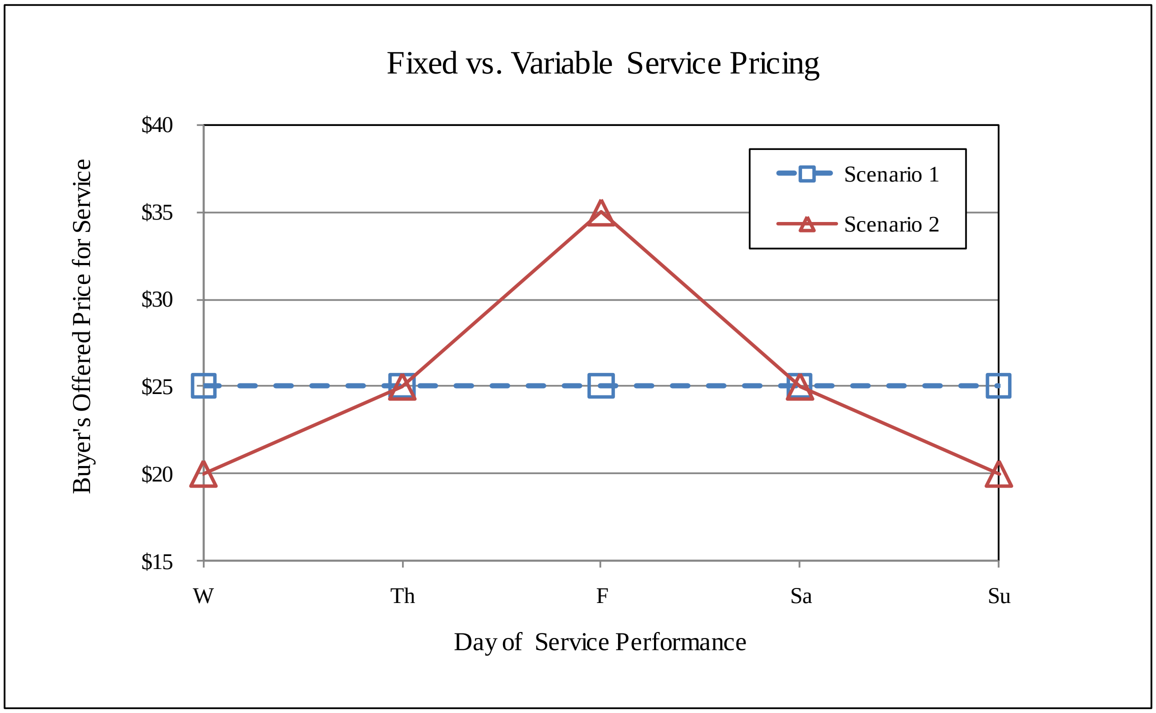

Figure 4: Fixed versus Variable Pricing of a Lawn Service

In both scenarios, buyers are requesting lawn service some time in five day period. In Scenario 1, the buyer offers a fixed price for the service to be completed. In Scenario 2, the buyer offers a price that varies by time, with the intent that there will be a better chance of having the service completed on the preferred day. Note that in both scenarios, the average price offered for performance of the work over the time period is the same. The buyers register their request for service with these pricing schedules with the system server and wait to receive matches to lawn service providers that accept their terms.

Lawn service providers are also randomly placed on the map. The each provider has a minimum amount of profit that is acceptable for a lawn service transaction. Providers also have a transportation cost in $/mile that reduces the profit on jobs depending on the prospective buyer's distance from the providers current location. Lawn service providers update the system server with their location each time step and the system server will only notify providers about buyer offers that are within a threshold distance from the providers' current location.

In this example case of lawn service, it is assumed that the buyers are homeowners who remain stationary, while the lawn service providers are mobile, although the same system could be used to simulate services with mobile buyers and mobile service providers. Figure 5 shows an example of the initial location of buyers and providers in a simulation. At each time step, the lawn service providers move to a new buyer location if they've accepted an offer, or if not, move in a random direction.

For this simulation, the mechanisms for security of location information and for reputation tracking were not considered.

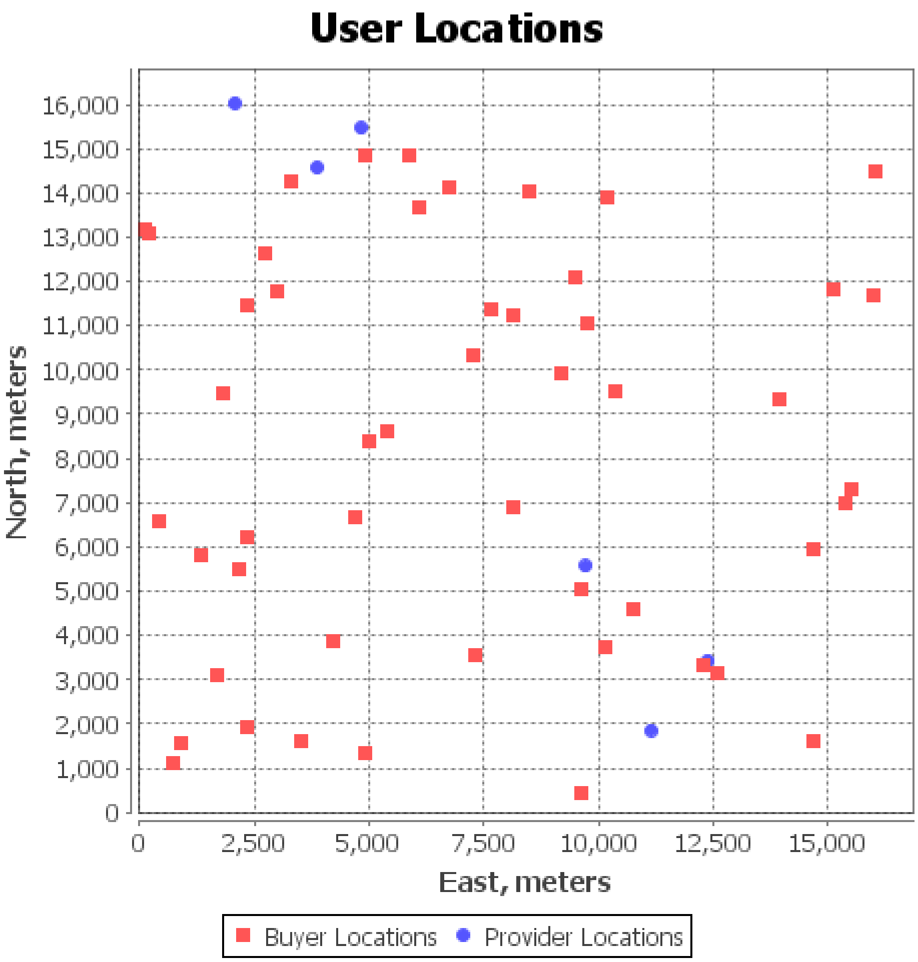

Figure 5: Initial Locations of Buyers and Service Providers in the Simulation

Simulation Approach

At the start of the simulation, each buyer submits a request of service for some time in the next two week period. Each service provider submits distance acceptable to travel to perform a task. Service providers also locally maintain a cost per distance associated with traveling, and minimum amount of money acceptable to perform the task.

Service providers use these two values to determine if they will accept a buyer's offer forwarded to them from the system server.

The simulation then steps through the two-week period with the time step of one day for simplicity. Each day, the system server looks at buyer and servicer provider locations and notifies the providers if there are any buyer offers that meet their criteria. If a provider sees multiple buyer offers, the provider selects the one that yields the best profit when considering the offer price and the transportation cost. If a provider accepts a buyer's request, the provider is revealed the buyer's location, moves there, and performs the lawn service. For the purpose of this simulation, an assumption is made that a provider can only service one buyer request per day, which is in this case one per time step. As buyers' lawns are serviced, their requests are closed, and the providers wait for notification of new offers from the system server. Providers receiving no acceptable offers for that day move in a random direction on the map

Simulation Results

The simulation was run twice with the same initial locations of buyers and service providers and the same initial conditions. For all runs 50 buyers were included, so there were 50 opportunities for service in the two-week simulation period.

In Scenario A, buyers have offered a fixed price over their period of performance – \$25 per day for 5 days. In Scenario B, buyers offer a variable price -\$35 for their most desired day, \$25 for the days adjacent, but only \$20 for 2 days before or after the target date. This is the same as described previously in Figure 1, where the average price per day is the same between the two distributions.

The measurements of interest are the total number of jobs done, the number of buyers' target days that were hit – that is to say, the lawn was serviced on the day most preferred by the buyer - and the average transaction price.

The simulation was run over a range of buyer to provider ratios. For each buyer/provider ratio setting, random initial locations for buyers and providers were selected and then simulated over a two-week period with variable pricing; the same initial conditions were then used in a simulated two-week period with fixed pricing. This procedure was repeated 3 times at each buyer/provider ratio with the results averaged. The simulation parameters are results for the fixed pricing runs are shown in Table 1, and the results for the variable pricing runs are shown in Table 2.

Pricing Type	Fixed	Fixed	Fixed	Fixed	Fixed	Fixed	Fixed
Number of Buyers	50	50	50	50	50	50	50
Number of Providers	1000	100	50	25	10	5	2
Buyer:Provider Ratio	0.05	0.5	1	2	5	10	20
Total Jobs Done	50	50	47.3	42.3	26.7	15.7	6.3
Target Days hit	5.7	5.3	5.7	7.7	4	2.3	2
Ave Transaction price	$ 25	$ 25	$ 25	$ 25	$ 25	$ 25	$ 25

Table 1: Simulation Results from Fixed Price Scenarios

Pricing Type	Var	Var	Var	Var	Var	Var	Var
Number of Buyers	50	50	50	50	50	50	50
Number of Providers	1000	100	50	25	10	5	2
Buyer:Provider Ratio	0.05	0.5	1	2	5	10	20
Total Jobs Done	50	50	50	49.3	43	27.7	14
Target Days hit	5.7	10.7	18	33.3	33.7	22.3	11.7
Ave Transaction price	$ 26	$ 27	$ 29	$ 32	$ 33	$ 33	$ 33

Table 2: Simulation Results from Variable Price Scenarios

With the variable pricing, the average increase in total jobs complete was 40% over the fixed pricing results, and the average number of buyers' target days hit was nearly 4 times greater than with fixed pricing.

Figure 6 shows the number of lawn service jobs completed for the fixed pricing scenarios, versus the variable price scenarios. For the fixed two-week time period of the simulation, the number of jobs completed dropped off as the ratio of buyers to providers increased. This is an intuitive result as there are fewer workers able to service a larger set of requests, they are only able to complete so many tasks in a given time period. The comparison of the fixed-pricing runs and the runs where the buyers were allowed to preferentially price certain days shows that the overall number of jobs completed is higher for the variable pricing case. The temporarily higher pricing offered during buyers' preferred days appears to be enough to attract providers who otherwise find it too far to travel and turn down offers.

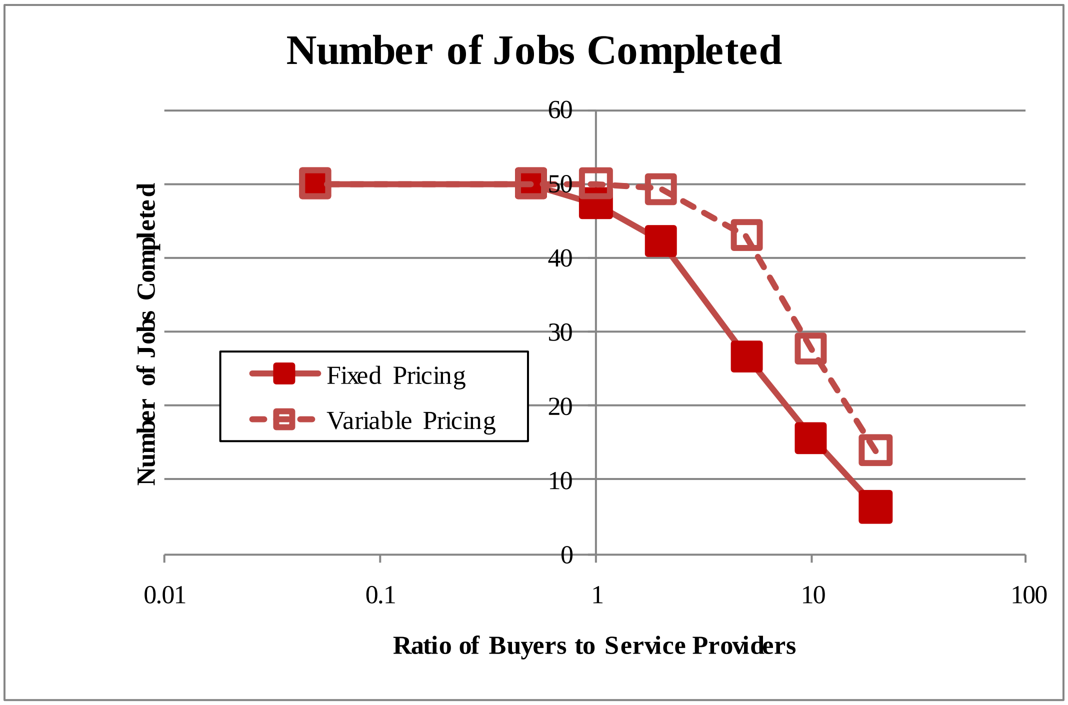

Figure 6: Number of jobs completed during the simulation

As the ratio of buyers to sellers drops below 1:1, the advantage of variable pricing to lure providers in the face of competition is reduced. In this condition, there are so many service providers it becomes highly likely that one is nearby any given buyer, will not incur much transportation cost, and is willing to accept the buyer's offers. So all the offers available in the 2-week period are accepted.

Figure 7 compares the number of jobs performed on buyers' preferred days, for the fixed pricing simulations and for the variable price simulations. For the conditions of the simulation, it appears that approximately 4:1 may be an optimal ratio between buyers and service providers that maximizes the opportunities for jobs to be performed on buyers' preferred days. When there are many more providers than buyers, the advantage of variable pricing is less. In this condition, it may be advantageous for buyers to further reduce the offered price on their non-preferred days.

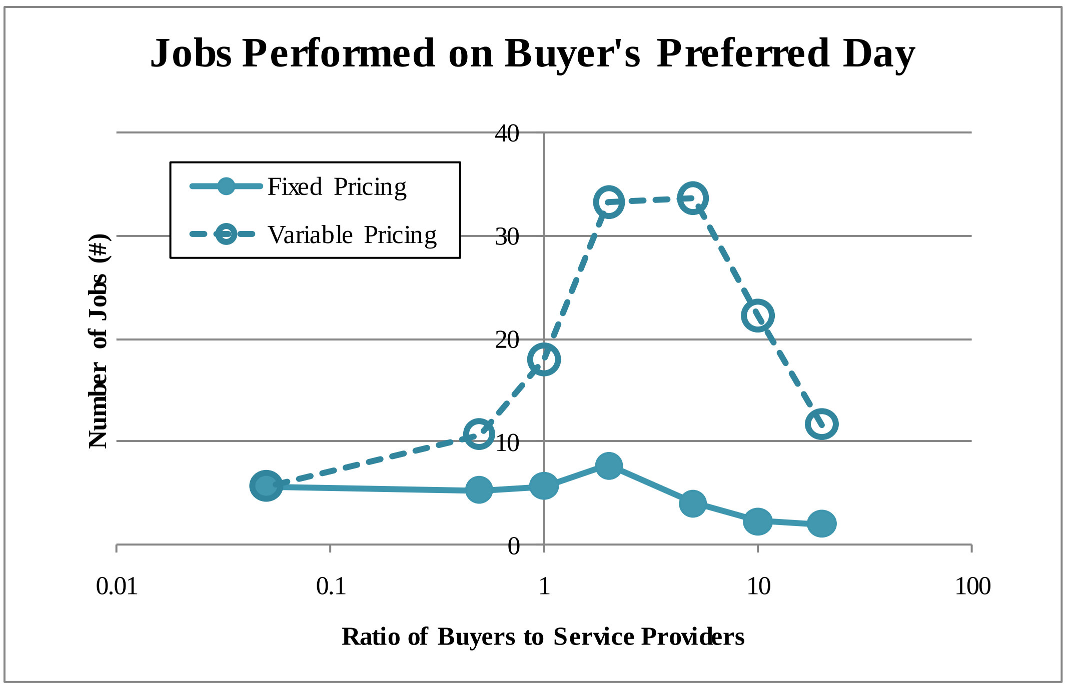

Figure 7: Number of jobs performed on buyers' preferred days

However, the more striking result is that in the conditions of buyer-provider parity or when there are more buyers than sellers (i.e. ratios of 1-10,) the ability to extend variable pricing offers clearly increases the likelihood of buyers having service performed on their preferred days. With variable pricing, over three times as many jobs were performed on the buyer's preferred days than with fixed pricing. In fact, the only condition where there was not a difference between the fixed and variable pricing was at the extreme lowest ratios corresponding to 1000 service providers for 50 buyers. In this case, it appears that there is so much competition between service providers that they will take any job available as soon as possible and forego the premium return possible by waiting for a buyer's preferred day.

Figure 8 shows the average transaction price for jobs performed during the simulation. The fixed pricing scenarios are clearly shown as the solid line at $25. The variable pricing results in premium return for service providers especially when there are more buyers than providers.

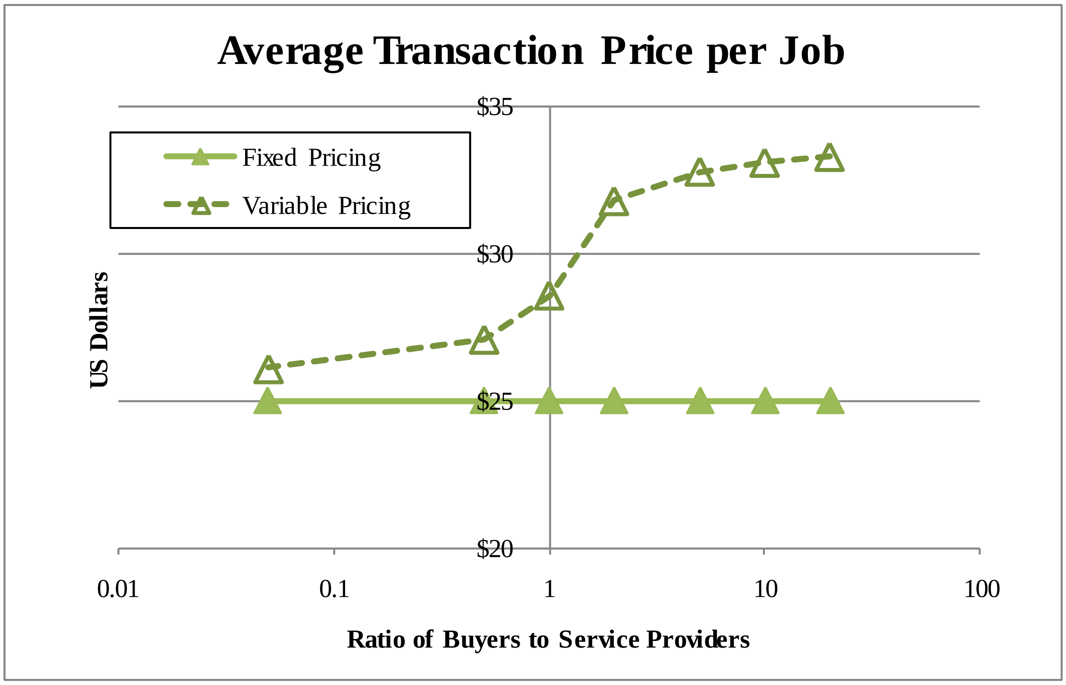

Figure 8: Average transaction price for jobs completed

This is an intuitive result. If there are many buyers and few providers, the providers will prefer to take the jobs that pay the most. In the opposite condition of many more sellers than buyers, the simulation shows that the sellers are still able to take advantage of the buyers' premium pricing for having service performed on the preferred day.

Note that there were several simplifications of the participants' behavior in the simulation. First, the assumption was made that the lawn service providers can only perform one job each day. In practice, they may be able to do multiple jobs in a day. Accounting for this in the simulation could be done by simply reducing the time step, for example, to 4-hour increments. The impact to the results would be a shift the effective ratio of buyers to providers, but the trends in the results would remain the same. Another simplification was that the service provider model was somewhat naive and only looked at the current available buyer offers. It did not use any strategy to further maximize profits, for example, by turning down an offer today for accept a more lucrative offer tomorrow. However, if this additional complexity were added to the behavior, service providers would work to maximize profits even further, so at least the same or more even jobs would end up occurring on the buyers' preferred days, and as such the average transaction price would increase as well. Also, if the focus of the simulation were changed to a investigate shorter term tasks like pizza delivery or a car-service with pricing schedule similar to that shown earlier in Figure 2, then there would be less opportunity for the service provider to use any predictive strategy, and the simplification of a naive service provider in the simulation would be quite appropriate since there would be effectively no lead-time in which to strategize.

Paths for extending the simulation in the future include modeling types of services where both the buyers and sellers are mobile, adding more variety to the types of buyer requests and seller behaviors that occur, and also investigating the impacts of cancellations or reschedule requests. However, there is a risk of skewing the simulation results if the added level of detail of the buyer and seller behavior is diverges in some way from the behavior of actual system users. To mitigate this effect, it would be advantageous to perform a small test or survey with actual human participants to determine if the added detail in the way buyer and seller behaviors are modeled is appropriate.

Extension to Other Services

This same application approach can be applied to many other types of transactions for services simply by allowing the buyers and sellers to modify their acceptable pricing versus time and distance, or providing more weight to the provider reputation. Other scenarios could include services such as a car service, medical house call, handyman, rideshare, or even a ghost-runner to stand in line for movie tickets.

Table 3 lists several examples along with what elements of the transactions make them different from the others.

Service	**Parameters Differentiating the Services**							
	Skilled 0-1	Trusted 0-1	Buyer Mobility 0-1	Service Provider Mobility 0-1	Task Periodicity 0-1	Required Completion Reliability 0-1	Task Cost 0-1	Duration of Execution window 0-1
Lawn service	0	0	0	1	0.7	0.3	0.5	0.8
House cleaning	0	0.8	0	1	0.5	0.2	0.8	0.5
Pool service	0.2	0.3	0	1	0.9	0.1	0.8	0.8
Tree trimming	0.2	0.3	0	1	0.1	0	0.9	0.7
Baby sitting	0.5	1	0	1	0.3	1	0.4	0.1
Pet sitting	0	0.8	0	1	0.1	1	0.2	0.1
Ride-share service	0	0.4	1	1	0.1	0.9	0.2	0.1
Mobile mechanic	0.5	0.8	0.7	1	0.1	0.7	0.7	0.4
Meal delivery	0	0.5	0.2	1	0.8	0.9	0.2	0.1
Massage therapy	0.2	0.2	0.2	1	0.7	0.8	0.6	0.4
Ticket queue stand-in	0	0	1	1	0.1	0.5	0.3	0.2

Table 3: Comparisons of Other Services

For example, for a task such as lawn service, there may be a large acceptable duration of execution; the buyer just wants his grass cut this week sometime. However, with a taxi service, the buyer most likely needs the service on the order of minutes and could price the need for that service to reflect the timeliness required. Tasks with a high value of trust required for the provider are shown such as baby sitting, or house cleaning, and as discussed earlier, a mechanism for accounting for service provider reputation could be weighted. For example, a buyer may specify that only service providers that have completed ten or more jobs with a buyer satisfaction rating of 95% or greater may accept their offer. Another method would be to allow buyers to preferentially select service providers they have done business with before.

The barriers to entry to become a service provider are also low – a provider just needs the mobile application, to get started. An individual with a car running errands for the day could register as a car service provider, and be notified of any buyer requests in his area from people wanting to travel in the same direction. In this case, the buyer mobility and service provider mobility are both high, however the system would operated the same way as in the lawn service example, only both buyer and service provider locations would need to be updated at the system server on a timelier basis.

The task periodicity column captures how likely this type of service would be a one-time-only task versus something that occurs on a regular pre-determined basis, such as every second Tuesday. This system is especially useful for tasks with low periodicity in which case there is no pre-determined preferred service provider for the buyer's request, as it handles the matching of service providers to buyers in an ad-hoc basis. In tasks with high-periodicity, it may be more advantageous for specific buyer and service provider to agree ahead of time on a regular schedule of work, as the service provider may offer a reduced price as compensation for the guaranteed future work.

Conclusion

Motivation for the inclusion of time and location in the pricing of services were discussed along with an example of a lawn service. Other research was evaluated and examples included systems using detailed buyer criteria, or including location, but not both as presented here. An approach to implementing this system was presented including customized buyer pricing, using location in the selection of possible matches between buyers and service providers, and providing a mechanism for service providers to predict the profitability of a job before accepting. Other important practical issues were discussed including security concerns about the disclosure and use of location information, and an approach to tracking successful outcomes as a proxy for the reputation of buyers and service providers. A simulation of the operation of proposed approach was created, and exercised over a range of conditions and numbers of buyers and service providers. The simulation results showed that variable pricing was beneficial to buyers by improving the chances that service was performed at their preferred time, and including location information in the form of buyer-to-provider distance allowed service providers to increase profits by preferentially selecting jobs with lower transportation costs. Furthermore, the general nature of the proposed system allows it to easily be applied to a wide range of other tasks.

Appendix A. Simulation Code Listing

```
public class BuyerProviderSim {

	static int distE = 16000, distN = 16000;
	//playing field size 10x10miles, 16000meters

	// sim input variables
	static double BuyerDensity =  0.50;
	// buyers per square mile (default 0.25)
	static double BuyerRatio = 5;
	// buyer to seller ratio. (default 5)

	static int nbuyers =  (int)(distE/1600*distN/1600*BuyerDensity) ;
	static int nsellers = (int)(nbuyers / BuyerRatio);

	//initialize buyers and sellers
	static MBuyer[] buyerlist=new MBuyer[nbuyers];
	//make a list of buyers
	static MSeller[] sellerlist=new MSeller[nsellers];
	//make a list of sellers

	public static void main(String[] args) {
		int i,j,k,t;

		//measures for the sim
		int TotalJobsDone = 0;
		double TotalValueofTransactions = 0.0;
		double TotalSellerProfitofTransactions = 0.0;
		double AverageTransactionPrice = 0.0;
		double TransportCost = 0.0;
		double ratiow = 0.0;
		// total transport cost / total transaction value

		Random randomGenerator = new Random();
		// Randomize the initial locations of buyers and sellers
		int rx, ry;
		for (i=0;i<nbuyers;i++){
			rx = randomGenerator.nextInt(distE)+100;
			ry = randomGenerator.nextInt(distN)+100;
			buyerlist[i] = new MBuyer(i,rx ,ry);
		}

		for (i=0;i<nsellers;i++){
			rx = randomGenerator.nextInt(distE)+100;
			ry = randomGenerator.nextInt(distN)+100;
			sellerlist[i] = new MSeller(i,rx,ry);
		}

		//set buyer pricing comparison
		// fixed price:        25,25,25,25,25
```

```java
// variable pricing:   20,25,35,25,20
int m1; //random offsets to vary buyers' preferred days
for (i=0;i<nbuyers;i++){
      m1 = randomGenerator.nextInt(13);
      buyerlist[i].pricetable[(0+m1)%14]= 0;
      buyerlist[i].pricetable[(1+m1)%14]= 0;
      buyerlist[i].pricetable[(2+m1)%14]= 0;
      buyerlist[i].pricetable[(3+m1)%14]= 0;
      buyerlist[i].pricetable[(4+m1)%14]= 0;
      buyerlist[i].pricetable[(5+m1)%14]= 20; //20
      buyerlist[i].pricetable[(6+m1)%14]= 25; //25
      buyerlist[i].pricetable[(7+m1)%14]= 35; //target day
      buyerlist[i].pricetable[(8+m1)%14]= 25; //25
      buyerlist[i].pricetable[(9+m1)%14]= 20; //20
      buyerlist[i].pricetable[(10+m1)%14]= 0;
      buyerlist[i].pricetable[(11+m1)%14]= 0;
      buyerlist[i].pricetable[(12+m1)%14]= 0;
      buyerlist[i].pricetable[(13+m1)%14]= 0;
      //note this buyer's target day
      buyerlist[i].targetday = (7+m1)% 14;
}

//begin sim
int customernum;
int TargetDaysHit = 0;
double profit, bestprofit, revenue, bestrevenue;
double dist;
for (t=0;t<14;t++) {   //do for 14 time steps (2 weeks)

      for (j=0;j<nsellers;j++) {   //for each seller

            profit = 0;
            bestprofit = 0;
            bestrevenue = 0;
            customernum = 0;

            for (i=0;i<nbuyers;i++) {
            //loop over each buyer

                  //calculate distance
                  dist = Math.sqrt( (buyerlist[i].bx
                              - sellerlist[j].sx)
                              * (buyerlist[i].bx
                              - sellerlist[j].sx)
                              + (buyerlist[i].by
                              - sellerlist[j].sy)
                              * (buyerlist[i].by
                              - sellerlist[j].sy)) ;
                  //calculate value of deal to seller
                  revenue = buyerlist[i].pricetable[t];
                  profit = buyerlist[i].pricetable[t]
                              - sellerlist[j].kd*dist ;
                  if (profit > bestprofit) {
```

```
                    //update the best value
                          bestprofit = profit;
                          bestrevenue = revenue;
                          customernum = i;
                    }
              }
              //Check if best offer is
              // above seller's minimum threshold
              if (bestprofit > sellerlist[j].Pb ) {
                    //deal is made.
                    TotalJobsDone++;
                    TotalValueofTransactions =
                                TotalValueofTransactions
                                + bestrevenue;
                          TotalSellerProfitofTransactions =
                          TotalSellerProfitofTransactions
                                + bestprofit;

                    //Check if this was the
                    // buyer's target day
                    if ( t ==
                          buyerlist[customernum].targetday )
                    {
                          TargetDaysHit++;
                    }
                    //seller goes to buyers position,
                    // buyers price curve is zeroed.

                    for (k=0;k<14;k++){
                    buyerlist[customernum].pricetable[k] = 0;
                    }
                    // update seller's position
                    sellerlist[j].sx =
                          buyerlist[customernum].bx;
                    sellerlist[j].sy =
                          buyerlist[customernum].by;
              }
              else {
              // if the best offer wasn't acceptable,
              // move to a new position, up to 3 miles away
                    sellerlist[j].sx = sellerlist[j].sx
                                + randomGenerator.nextInt
                                (3*1600)-3*800;
                    sellerlist[j].sy = sellerlist[j].sy
                                + randomGenerator.nextInt
                                (3*1600)-3*800;
              }
        }

        //Calculate evaluation measures for this time step
        TransportCost = TotalValueofTransactions
                    - TotalSellerProfitofTransactions;
```

```java
                Plot(t);  //make a plot for each time step

            } //end of time step loop

            // Calculate evaluation metrics for the sim
            TransportCost = TotalValueofTransactions
                        - TotalSellerProfitofTransactions;
            System.out.println("TotalJobsDone: "
                        + TotalJobsDone
                        +"/"+nbuyers+", TransportCost: "
                        +TransportCost);
            System.out.println("TotalValueofTransactions:"
                        +TotalValueofTransactions
                        +", TotalSellerProfitofTransactions:"
                        +TotalSellerProfitofTransactions);
            System.out.println("TargetDaysHit:" + TargetDaysHit);

            if (TotalJobsDone > 0) {
                  ratiow = TransportCost
                              / TotalValueofTransactions;
                  AverageTransactionPrice =
                              TotalValueofTransactions / TotalJobsDone;
                  System.out.println("AverageTransactionPrice: "
                              +AverageTransactionPrice
                              +", Tranport Loss Ratio: "+ratiow);
            }

            //reset performance measures
            TotalJobsDone = 0;
            TransportCost = 0;
            TotalValueofTransactions = 0;
            TotalSellerProfitofTransactions = 0;
            TargetDaysHit = 0;

      }
}


/*
 * A buyer of services
 */
public class MBuyer {

      //index in price table of preferred day
      public int targetday;
      //location
      public float bx,by;
      public float bname;
      public int bindex;
      public int[] pricetable = {0,0,0,0,0,0,0,0,0,0,0,0,0,0};

      public MBuyer(int index, int x, int y) {
            // Constructor
```

```
		bx = x;
		by = y;
		bname = index;
	}

}

/*
 * A Provider of services
 */
public class MSeller {
	//location
	public double sx,sy;
	public double sname;
	public int sindex;

	public double Pb = 20.0; //base price
	public double kd = 5.0/1600;  //distance multiplier, $/meter

	public MSeller(int index, int x, int y) {
		// Constructor
		sx = x;
		sy = y;
		sname = index;
	}
}
```

Bibliography

Brush, A.J., Krumm, J., and Scott, J. 2010. Exploring end user preferences for location obfuscation, location-based services, and the value of location. In *Proceedings of the 12th ACM international conference on Ubiquitous computing* (Ubicomp '10). ACM, New York, NY, USA, 95-104.

Consolvo, S., Smith, I., Matthews, T., LaMarca, A., Tabert, J. and Powledge, P. 2005. Location disclosure to social relations: why, when, \& what people want to share. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (CHI '05). ACM, New York, NY, USA, 81-90.

Duckham, M. and Kulik, L. A Formal Model of Obfuscation and Negotiation for Location Privacy. Pervasive Computing. Lecture Notes in Computer Science. 2005. Springer Berlin / Heidelberg. pp243-251. Volume 3468Anon y mouse

EBay. Accessed 7-20-11. http://www.ebay.com/

Gruteser, M. and Grunwald, D. 2003. Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. In *Proceedings of the 1st international conference on Mobile systems, applications and services* (MobiSys '03). ACM, New York, NY, USA, 31-42.

Guttman, R. H. 1998. *Merchant differentiation through integrative negotiation in agent-mediated electronic commerce*." Masters Thesis, MIT Media Laboratory, Cambridge, MA.

Hauser, C. and Kabatnik, M. Towards Privacy Support in a Global Location Service. In IFIP Workshop on IP and ATM Traffic Management, Paris, pages 81–89, 2001.

Houser, D. Wooders, J. 2006. Reputation in Auctions: Theory, and Evidence from eBay. Journal of Economics & Management Strategy. Volume 15, Issue 2. Blackwell Publishing Inc.

iOS Development Guide. Retrieved July 7, 2011. http://developer.apple.com/library/ios/#documentation/Xcode/Conceptual/iphone_development/000-Introduction/introduction.html

Kittur, A. Chi, E.., and Suh, B. 2008. Crowdsourcing user studies with Mechanical Turk. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems* (CHI '08). ACM, New York, NY, USA, 453-456.

Leonhardt, U. Magee, J. Security Considerations for a Distributed Location .Journal of Network and Systems Management.1998-03-01.Springer New York. Issn: 1064-7570Subject: Computer Sciencepp51-70. Volume 6, Issue 1.

Local and Push Notification Programming Guide. Retrieved July 7, 2011 from http://developer.apple.com/library/ios/#documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/CommunicatingWIthAPS/CommunicatingWIthAPS.html

Saltzer, J.H.; Schroeder, M.D.; , "The protection of information in computer systems," *Proceedings of the IEEE* , vol.63, no.9, pp. 1278- 1308, Sept. 1975
doi: 10.1109/PROC.1975.9939
URL: http://ieeexplore.ieee.org.ezproxy.lib.utexas.edu/stamp/stamp.jsp?tp=&arnumber=1451869&isnumber=31196

Tewari, G. and Maes, P. 2000. Design and implementation of an agent-based intermediary infrastructure for electronic markets. In *Proceedings of the 2nd ACM conference on Electronic commerce* (EC '00). ACM, New York, NY, USA, 86-94.

Tewari, G., Youll, J., and Maes, P., Personalized location-based brokering using an agent-based intermediary architecture, Decision Support Systems, Volume 34, Issue 2, January 2003, Pages 127-137.

Wang, D. June 1999. "Market Maker: An Agent-Mediated Marketplace Infrastructure." Master of Engineering Thesis, Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA.

Yamamoto, H., Ishida, K., Ohta, T. Modeling Reputation Management System on Online C2C Market. Computational & Mathematical Organization Theory. 2004. Springer Netherlands. 165-178. Volume 10, Issue 2.