Incentivizing Rider Time-Shift in a Multi-Leg Public Transportation System

Megan Shearer and Michael P. Wellman
University of Michigan
{shearerj,wellman}@umich.edu

Abstract

We develop an incentive scheme for a hub-to-shuttle campus transit system, encouraging riders to shift travel times to improve system efficiency. Riders are modeled as agents with loss functions in order to evaluate social welfare in the incentive scheme. Incentives are calculated based on a predictive wait-time model trained via Gaussian process regression. Experiments show that the system's total loss decreases, but the average wait time increases. This leads to the conclusion that there is room for improvement in the predictive wait time model.

1 Introduction

In most public transportation systems, particularly bus systems, uneven ridership leads to an inefficient use of resources. There are geographic areas and times within a traffic system that require different resources, but in a bus system the same vehicles travel the same routes throughout the day. One way proposed to make public bus systems more efficient is to use shuttles for segments of bus routes with minimal riders, allowing buses to focus on high-volume routes [Mahéo et al., To appear].

Our work was performed as part of the RITMO project,[1] initiated by Pascal Van Hentenryck at the University of Michigan, which is studying the use of a hub-to-shuttle system on the campus bus network. As explained in a promotional video for the project,[2] the idea of the hub-to-shuttle system is to connect the higher-demand areas with buses, and to use shuttles to drive riders in the sparser areas. It would be inefficient to have a single shuttle service all sparse areas on the map, therefore the serviceable area is broken into zones. A shuttle will not leave its assigned zone, but rather pickup and deliver riders from hubs whom wish to travel in multiple zones.

Along with some physical routes getting significantly more attention, some temporal routes getting greater traffic volume causes problems with inefficiency. While it is inevitable that there will be uneven ridership throughout the day in a transit system, this paper addresses the problems created by the same amount of resources being deployed throughout the day by developing a solution to spread out traffic. Wait times are assumed to be important to riders and thus a useful tool to convince them to move to spread traffic. In this paper we define a *wait time* to be the time difference between when a rider requests to be picked up and when she actually boards her first vehicle. However, in a stochastic transit system like RITMO, wait times cannot be computed directly at the time the system would offer an incentive. In this paper, an *incentive* is defined as an arbitrary point system that can easily be expanded to a real quantity. For example, on the Michigan campus meal points can be allocated to students that alter their requests times. We specifically use incentives, rather than tolls, because the RITMO system is for a free campus bus system. Our goal is to spread riders out over time to reduce the cost of the system and lower wait times to increase efficiency and social welfare. *Social welfare* is the aggregate loss of riders in the system, plus the cost of providing incentives to riders.

The main contribution of this paper is the creation of an agent-based incentive scheme which utilizes a learned predictive wait time model. The incentive schemes aims to incentivize riders entering the system to change their departure time by small intervals. This helps alleviate the number of riders at a time, while not reducing the satisfaction of riders, because they still reach their destination on time. The incentive scheme models the riders as agents, then uses loss functions for the system and agents, and the expected wait time to calculate an incentive for each agent to change her trip starting time, or board time. The incentive model then decides if an agent should be offered an incentive based on if the system deems it socially optimal for the agent to move her travel time. The incentive model is judged by its effects on the system's cost and social welfare.

The creation of a learned predictive wait-time model partnered with an agent-based incentive model has previously not been used in transportation pricing. When we first created the incentive model and did not have a good wait time predictor, we found that simulated agents changed their board times to time intervals that potentially increased their wait times, which decreased the satisfaction of riders and increased costs to the system. As the predicted wait time became more accurate the rider satisfaction improved and cost decreased.

[1] https://ritmo.engin.umich.edu/

[2] https://www.youtube.com/watch?v=KFz0fEqaTrc

Therefore, the design of the predictive wait time model became integral to the success of the incentive model. Developing a predictive wait time model is not straightforward in this domain because the shuttles' paths are volatile. Our predictive wait time model uses Gaussian Process Regression [Pedregosa et al., 2011, Rasmussen, 2004] to find expected wait times for agents before their trips are scheduled, based on current traffic information. We evaluate the wait-time model on the accuracy of its predictions, and whether it helps reduce the average wait times in the system.

2 Related Work

The hub-and-shuttle concept has been extensively explored by Van Hentenryck and colleagues. Much of their published work on the topic focuses on vehicle routing optimization and analysis [Arda et al., 2014, Guimarans et al., 2015, Mahéo et al., To appear, Mouthuy et al., 2015, Pillac et al., 2015].

Literature on dynamic transportation pricing mostly involves finding optimal prices for lanes or roads to reach an equilibrium for drivers [Sharon et al., 2017, Yin and Lou, 2009, Zhang et al., 2014]. These attempt to change routes rather than departure times. Studies more similar to ours examine a similar problem of tolls for individuals and road traffic to achieve a socially optimal solution [de Palma et al., 2005, Joksimovic et al., 2014, Li, 2018, Bui et al., 2012].

Literature on pricing ride sharing systems strongly considers surge pricing, or pricing during peak travel times [Banerjee et al., 2015]. This work studies a two-sided system because drivers are also part of the pricing scheme, compared to the RITMO system where drivers are not compensated for individual trips. There is a study on pricing multi-leg ride sharing [Teubner and Flath, 2015], but their pricing scheme relies solely on trip distance and does not study traffic or incentivize riders to change requests.

Studies using machine learning to predict the wait time for public transportation riders mainly focus on arrival of buses on their predetermined routes [Chien et al., 2002, Yu et al., 2011, 2007]. Idé and Kato [2009] use Gaussian process regression to predict the travel time of vehicles in a traffic network.

As noted above, this work is part of the RITMO project at the University of Michigan. The work reported here builds on a simulation system developed by collaborating researchers on that project. Specifically, our main extensions are development of an accurate wait-time prediction model, a mechanism for the system to offer rider incentives, and incorporation of agent-based decision functions to allow riders to make choices among incentivized and non-incentivized routing options.

3 Incentive Model

To reduce the number of riders entering the system at high traffic times, an incentive model aims to incentivize some riders to change their requested board times by small time intervals. A rider enters the RITMO system by providing a request to the scheduler on an app interface. Without an incentive model, a scheduler immediately processes a riders requests by finding a trip for the rider, which is illustrated in Figure 1. Scheduling the requested trip is necessarily beneficial to other riders in the system, so we introduce an incentive model to attempt to increase the social welfare of the system by lowering the aggregate loss of the riders. Figure 2 illustrates the process of how a scheduler uses the incentive model to handle a rider's request. This request contains a rider's desired trip information, which inform the scheduler how much the rider values different aspects of the trip, for instance any aversion about early arrival at destination. After receiving a request, the scheduler decides whether or not to provide an incentive to convince the rider to change her board time. This decision is based on expected wait times for the rider and similar riders. If the scheduler decides not to offer an incentive, then the original request processes by scheduling the trip. If the scheduler offers an incentive, then the rider decides if she wants to take the incentive and change her board time, or reject the incentive and keep her original board time. Once the rider makes her decision, the scheduler respects the decision and continues with processing the request.

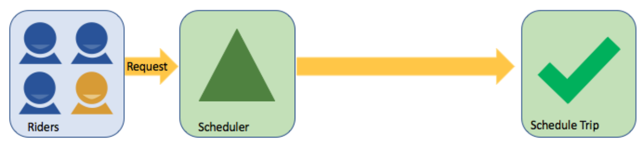

Figure 1: Without an incentive model, riders enter the system and send their desired trip to the scheduler. The scheduler immediately processes requests and schedules the route and vehicles.

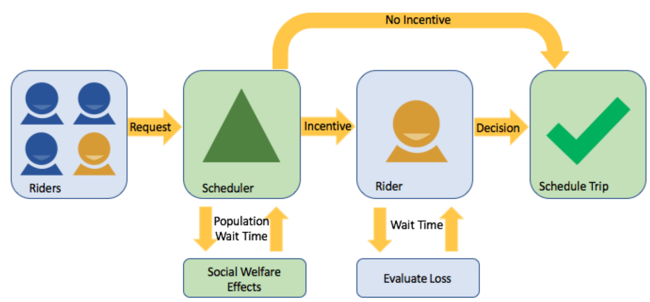

Figure 2: Riders enter the system and send their desired trip to the scheduler. It considers a request and uses the predictive wait time model to decide if offering an incentive is socially optimal. If it will not benefit the system for a rider's board time to change, then the it schedules her ride. Otherwise, it offers her an incentive, which she uses the predictive wait time model to decide if her personal loss lowers. She then returns her decision, and her trip is scheduled.

Given the information in a request, we model the riders as agents with loss functions. An agent's loss increases as wait time increases, and decreases with the addition of an incentive. Though every time an agent takes an incentive, the system endures the cost of that incentive. The entire system is better off when the agents' losses in aggregate are lower. Therefore, the total loss of this system is calculated by the sum of agent loss plus the incentive costs. This total loss can be thought of as the negative of social welfare, because the agents on average are better off with a lower system loss. An individual agent's assumed goal is to minimize her own loss. The goal of the scheduler is to maximize social welfare by minimizing the total loss of the system.

3.1 Communication Between Scheduler and Agent

An agent wishes to minimize her own loss and to reach her destination. The scheduler wants to minimize the net-loss of the system and get all agents to their desired destinations. The scheduler breaks a day into t-minute time intervals as a way to measure congestion at different points. Both the agents and scheduler are aware of traffic in the market.

An agent, x_i, initiates an interaction by sending a request with public and private parameters. We assume the scheduler has full information from agent x_i's request, that is, the request the scheduler receives reflects all private information. Given the request, the scheduler decides if the agent should be offered an incentive ψ_i to move the agent's board time by t. We calculate the incentive with agent x_i's private loss function, using values in x_i's request. Once the scheduler finds ψ_i, it looks at congestion and a threshold for incentives. Congestion is determined by the effects of an agent on the average wait time at different time intervals.

If the scheduler determines that ψ_i should be given to x_i then it returns her proposal consisting of ψ_i and a new board time. If the scheduler does not think that an incentive should be offered to x_i, then it continues on with scheduling x_i's ride. If the agent is offered an incentive, then she considers her potential loss with the proposed board time and her original requested board time. The agent chooses the board time that minimizes her loss. Since the scheduler knows the private parameters of the agent, it can choose an incentive that it knows will be sufficient to convince an agent to agree to a time-shift. The scheduler updates agent x_i's request and continues with scheduling her ride.

3.2 Details of Interaction

Request

Agents' requests consist of a set of public and private parameters that the scheduler uses to assign routes and incentives. The public parameters within an agent's request are a trip ID (i.e., an identifier for agent $x_i \in X$ where X is the set of all agents), the requested board time lower bound r_i, the pickup stop, and the delivery stop. The private parameters are her willingness to leave early α_i, willingness to leave late β_i, and malleability to incentives γ_i, which determines how much x_i appraises an incentive in proportion to how the system values it. Her wait time is the difference between her requested board time lower bound and her actual pickup time.

Agents' Loss Function

Agents have a loss function to evaluate a ride. The elements of this loss function consist of the travel time, and the effects of changing the request time if the agent chooses to change her board time lower bound. The travel time is broken down into the wait time and the time an agent spends on the actual trip. We assume agents want lower wait times. Therefore the wait time is beneficial in calculating an incentive. We assume that loss increases if the agent does not leave at her intended time, but an incentive counteracts this by decreasing loss.

Agent x_i's loss function is formally defined as:

$$\mathcal{L}_a(x_i) = \tau(p_i) + \alpha_i(r_i - p_i)I_e + \beta_i(p_i - r_i)I_\ell - \gamma_i\psi_i. \quad (1)$$

The parameters are revealed to the scheduler in x_i's request. p_i is a board time lower bound proposed to agent x_i by the scheduler, where $p_i \in \{r_i - t, r_i, r_i + t\}$. $\tau(p_i)$ is the trip time at p_i, where trip time is calculated by the time from p_i to when the agent completes her trip. I_e and I_ℓ are indicator functions where $I_e = 1$ when $r_i > p_i$ and $I_e = 0$ otherwise, i.e. when the proposal time is earlier than the request time, and $I_\ell = 1$ when $r_i < p_i$ and $I_\ell = 0$ otherwise.

The scheduler and agents cannot know exact travel times until all rides are assigned trips, so the estimated agent loss is found by:

$$\hat{\mathcal{L}}_a(x_i) = E[\tau(p_i)] + \alpha_i(r_i - p_i)I_e + \beta_i(p_i - r_i)I_\ell - \gamma_i\psi_i. \quad (2)$$

The expected trip time, $E[\tau(p_i)] = E[w(p_i)] + E[v_i]$. $E[w(p_i)]$ is the predictive wait time model's estimate at p_i, and $E[v_i]$ is an optimization of the minimum trip distance. The calculation of the estimated wait time is discussed in section 4.

System's Loss Function

The system also endures a loss from each agent and the incentive it pays her. Therefore, the system's loss is defined as the aggregate agents loss, plus the cost of the incentives. Since the system appraises the incentive at the its true value, the system's loss from a rider is defined as:

$$\mathcal{L}_s(x_i) = \mathcal{L}_a(x_i) + \psi_i. \quad (3)$$

The scheduler uses an estimate of (3) to assess the effect an agent will have on the system, because the agent's trip is not scheduled when it uses the loss to calculate an incentive. Thus, it must use the predictive wait time model to calculate the expected travel time. The estimation the scheduler uses:

$$\hat{\mathcal{L}}_s(x_i) = \hat{\mathcal{L}}_a(x_i) + \psi_i, \quad (4)$$

After the termination of the simulation, we evaluate the system by summing agent loss (3),

$$L = \sum_{i=1} \mathcal{L}_s(x_i). \quad (5)$$

Incentive

Assuming the scheduler knows each agent's private parameters, it can find $\hat{\mathcal{L}}_a(x_i), \forall i$, and use this to minimize the system's loss. We find the minimum incentive for x_i by solving:

$$E[\tau(r_i)] - \hat{\mathcal{L}}_a(x_i) = 0. \quad (6)$$

$E[\tau(r_i)]$ is the estimate loss of at r_i. We solve for ψ_i in (6):

$$\psi_i = \frac{1}{\gamma_i}\Big(E[\tau(p_i)] + \alpha_i(r_i - p_i)I_e + \beta_i(p_i - r_i)I_\ell - E[\tau(r_i)]\Big). \quad (7)$$

If the scheduler offers ψ_i found in (7), then agent x_i is guaranteed to do as well as her original request time. However, it is most likely suboptimal to move all agents, so the scheduler decides which agents to incentivize the impact on other agents with similar requests. If the scheduler determines moving x_i to have a positive affect on other agents, then ψ_i is offered to x_i, in exchange for x_i to change her request time to p_i.

Scheduler Decision
Once the scheduler calculates an agent's required incentive, it must decide whether to offer this incentive to the agent. The scheduler's goal is to maximize social welfare, so it offers x_i an incentive if moving x_i positively affects the other agents in aggregate. We determine this using the wait-time estimator to find the wait time with and without the agent at the requested time and time intervals around it. We assume that an agent affects only those requesting trips at the same stop and time interval. Therefore, we calculate the average estimated wait time of the original time interval and the intervals around it. To find the socially optimal option, we find the average estimated wait when x_i is in each time interval. Whichever provides the lowest average wait time is the interval the scheduler incentivizes agent x_i to move her request time. This decision is illustrated in Figure 3. We assume lower average wait time leads to a socially better outcome, because more agents with requests similar to x_i will be better off since a lower average wait time yields a lower average loss.

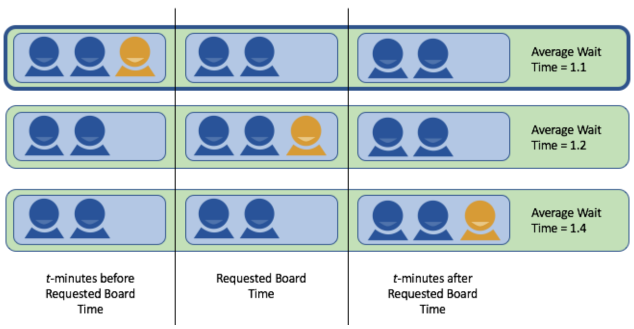

Figure 3: The scheduler decides to offer an agent an incentive if it will positively affect social welfare. To make this decision, it finds the average wait time of the desired time interval and those preceding and following it; this average wait time is defined as $E[W]$. Here it offers an incentive to change the board time to $r_i - t$, because $\min E[W] = 1.1$.

Agent Decision
An agent's goal is to minimize her own personal loss. Therefore, she will take an incentive if her overall loss lowers when she moves her request time to the proposed time. How she makes this decision is illustrated in Figure 4. An agent does not particularly care about if she enters at rush hour or not, but rather if she has a low wait time and reaches her destination on time. Choosing the board time lower bound with the lowest loss assures that she achieves these goals.

3.3 Agent Manipulation

We also explore the impacts on the incentive model when agents try to manipulate the system. *Agent manipulation* defines an agent attempting to lower her own loss by giving the scheduler an untruthful board time lower bound. Before an agent submits her request, she is assumed to have full knowledge of the system and can determine if she would be incentivized to change an untruthful request time to her true desired time. This decision happens before a request is sent, and how it fits into the incentive model is shown in Figure 5,

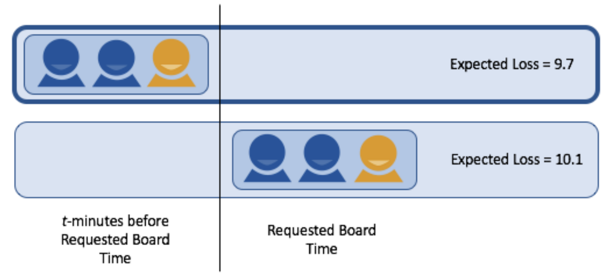

Figure 4: An agent determines whether or not to take an incentive by assessing her personal loss. She uses the predictive wait time model in order to calculate her expected loss, $\hat{\mathcal{L}}$. Here she will choose to take the incentive because $\min \hat{\mathcal{L}} = 9.7$.

and details of her decision are illustrated in Figure 6. A manipulative agent is only untruthful about her requested board time, and the other parameters in her request are truthful. The scheduler is unaware that an agent is untruthful, and assumes that the distribution of traffic throughout the system remains the same as if all agents were truthful.

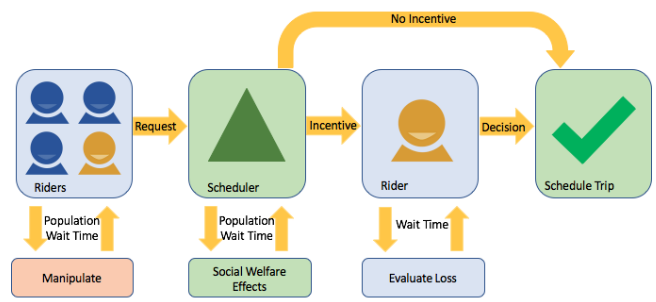

Figure 5: This depicts the same incentive model as Figure 2, except agents are now trying to manipulate the system. Agents now use the predictive wait model to determine if they should submit an untruthful requested board time to the scheduler. The incentive scheme proceeds in the same manner as Figure 2.

3.4 Evaluation

The incentive model is evaluated on its impacts on social welfare. Our goal is to minimize the system loss (5). This equation is the sum of the agents' loss and the incentives provided to them. Therefore, in minimizing this equation, we also minimize the agents' loss, while not providing unaffordable incentives. Minimizing the agents' loss should result in maximizing the social welfare. The baseline for total system cost is the value of (5) when no incentive scheme is used with the model.

4 Predictive Wait Time Model

The goal of the incentive scheme is to convince an agent to alter her request before the actual trip takes place. To determine an incentive to provide the agent, the scheduler uses (7). Thus at the creation time of the request, the scheduler needs to know the wait time at the requested board time.

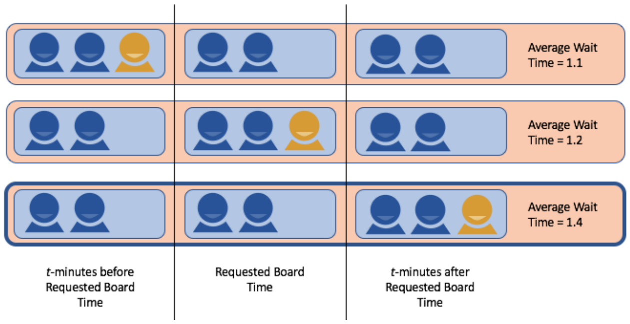

Figure 6: To decide if she should lie about her desired requested board time, an agent wants to know if she will be incentivized to change her board time to her desired time. She uses the predictive wait time model to determine her effects on social welfare, which are calculated the same way as Figure 3, with average wait time $E[W]$. If she finds that the system will be worse off if she has a different board time, then she changes her request. Here she changes her request to $r_i + t$ because max $E[W] = 1.4$.

Finding the wait time at a future time step is nontrivial because the state of the system can change. In the interim, other agents may enter and alter a shuttle's path, potentially affecting the predicted route of an agent at her creation time. Shuttles may also leave and enter the system in a stochastic manner.

4.1 Gaussian Process Regression

We estimate wait times using a model trained with Gaussian process regression (GPR). GPR generates a nonlinear and nonparametric model to predict output values. It works by measuring the distance between points using a kernel function, and constructing predictions based on a distance-based weighting of those points. The training is governed by a set of hyper-parameters [Rasmussen, 2004].

We chose GPR for its superior performance using the features available in the stochastic RITMO system. GPR training requires time cubic in samples, which limited the size of training sets we could employ in experiments. We use a radial basis kernel function, $k(x, x') = c\exp(-\frac{1}{2\ell}(x - x')^2)$, with white noise. We assume that small changes in features causes only small changes in a wait time, which makes the smoothing quality of the radial basis kernel ideal. All of the features are standardized during training, and data is split between testing and training sets.

4.2 Feature Selection

All of the selected features are independent of an agent's private variables and pertain to the current state of riders, vehicles, and stop network: number of agents expected to request board times in the same zone and time interval; distance of the predicted closest vehicle; whether requested board stop is a hub; whether the agent's pickup and drop-off stops are in different zones; whether the agent is at a hub and changing zones (i.e., waiting for a bus rather than shuttle).

4.3 Training Loop

The training loop begins by running the incentive model once to generate features and wait times used to train the predictive wait time model. The training loop and final model selection is illustrated in Figure 7. Once data is collected from this initial simulation run, it is split into testing and training data. The training data is used to train the Gaussian process regression model. The trained model's generated parameters are plugged into the simulation, and used to calculate the expected wait time on future runs. Next, the simulation runs again with a smaller proportion of agents offered incentives. This proportion is determined by offering incentives only to agents that the scheduler is sure with benefit the system. The scheduler determines which agents will benefit other agents more by altering their requests in Section 3.2. Then the same data extraction and training process are repeated. The data is extracted and trained until are desired proportions are tested. After this training loop completes, all wait time models are used on the incentive model when either the corresponding proportion of agents are offered incentives, or no agents are offered incentives. The chosen final model is the combination of the predictive wait time and incentive models that yields the lowest total system loss, L, found by (5).

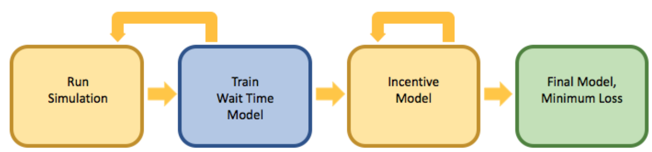

Figure 7: The model begins by generating data by running the incentive scheme once. Then the wait time model trains using this data, and this trained model is saved for later. Next the model is trained with a smaller proportion of agents offered incentives. Next these wait time models are used in the incentive model. The final model is chosen by the minimum system loss, L.

4.4 Evaluation

The predictive wait time model is evaluated on the accuracy of its expected wait times and its effects on the actual wait time. The model is therefore evaluated by its success in lowering aggregate wait times. The baseline for this comparison is the system with no incentive scheme.

5 Results

5.1 Data

We use a RITMO dataset provided by the RITMO project in our experiments. This dataset represents bus traffic on the University of Michigan campus. University bus drivers manually recorded the number of riders that boarded a bus at each stop three different months, which then was turned into a representative distribution of agent requests. There are also sets of shuttles, buses, and stops to create the University of Michigan RITMO system.

To model loss, we added personal preference data randomly to each agent. These preferences are not meant to be a realistic model, but rather reasonable values for evaluation. Using uniform distributions, we generated preferences, $\alpha_i, \beta_i, \sim U[0, 0.1]$ and $\gamma_i \sim U[0.5, 1.5]$, which specify, respectively, willingness to travel early and late, and how the rider values an incentive proportionally to the system. We also perturbed the baseline request set to test the incentive and predictive wait time models on different traffic spreads on the Michigan campus. Specifically, we added noise $\sim U[-20, 20]$ minutes to each rider's request-creation time and requested board time.

5.2 Environment Settings

We consider the RITMO and perturbed datasets, which contain 37,724 rider requests in the University of Michigan bus system. This dataset takes place over a 22 hour period. The system is split into three zones, with between 17 and 31 shuttles in each zone. There are four bus lines with a total of 17 buses.

For both the incentive and predictive wait time models, we vary the proportion of agents offered incentives to determine if this affects the total loss and average wait time. We do this by only offering agents whom the scheduler is more convinced will benefit the system by moving. This is determined by setting a limit on the change in predicted average wait time, and the scheduler will only offer incentives to agents that it is more convinced will benefit the other agents around them. We then evaluate the system by the total loss and average wait time at the different proportion of agents offered incentives.

In the incentive model we test all non-manipulative agents, and all manipulative agents. We train the predictive wait time model on a set of 1,000 requests. Each experiment is run with 10 different random seeds to produce 95% confidence intervals on the following graphs. Lastly, the baseline is defined as the total loss or wait time of the system when no incentives are offered.

5.3 Incentive Model Results

The incentive model is evaluated by the total system loss for non-manipulative and manipulative agents. The goal is to lower these from their initial baseline values. The results of the total system loss with the predictive wait time model on the RITMO and perturbed data are presented in Figures 8 and 9, respectively.

Figure 8 shows for non-manipulative and manipulative agents the total system loss initially decreases by an insignificant amount from the baseline value, depicted by the black line, then dramatically increases. The difference in proportion of agents offered incentives is determined by a threshold placed on the certainty of the scheduler that moving an agent will benefit the system. Therefore, when more agents are offered incentives, there is a lower threshold on the certainty of the scheduler, and the maximum proportion in Figure 8 occurs when any agent may potentially benefit the system by moving. The result that the loss increases as more agents are offered incentives demonstrates an inaccuracy in the system, because ideally the scheduler should not be offering that many agents, if any at all, incentives that would cause the total loss to increase. This is mainly an issue because the goal of the system is to lower the loss of the agents in aggregate, a goal which is clearly not being achieved. The loss slightly increases when agents are manipulative, but not by a significant amount compared to non-manipulative agents.

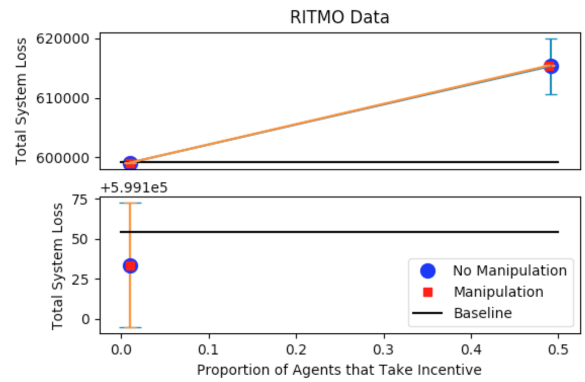

Figure 8: The total loss of the system increases if we offer more agents incentives. The bottom figure is a closer look at the total loss when a smaller proportion is offered incentives. The total loss lowers, but not by a significant amount. The loss slightly rises when agents are manipulative.

Figure 9 shows the total loss of the perturbed data set. The perturbed data yields a similar result to RITMO data, but when a lower proportion of agents are offered incentives the total loss lowers by a significant amount. One more thing to note is that the confidence intervals are tighter on the perturbed data compared to the RITMO data, and there is less variation between the cases with and without manipulation. This is most likely caused by the method of perturbation making the traffic spread more stable and have slightly lower peaks at high traffic times.

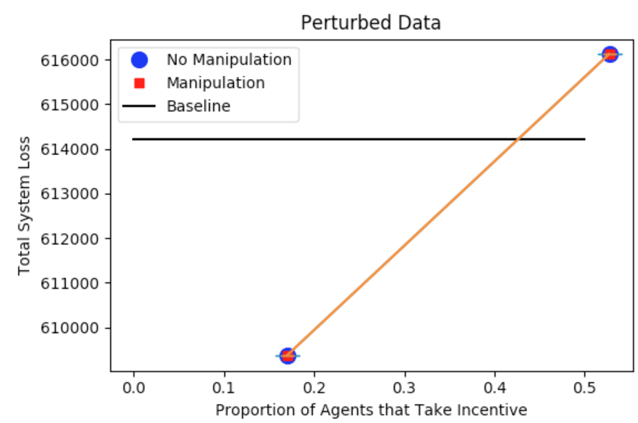

Figure 9: The total loss lowers by a significant amount when incentives are offered to a smaller proportion of agents, and the loss is higher than the baseline when more agents are offered incentives. The loss remains the same when agents are manipulative.

5.4 Predictive Wait Time Model Results

The predictive wait time model is evaluated by the average wait time at the end of the simulation. The results of the average wait time for the RITMO and perturbed data are shown in Figures 10 and 11, respectively.

Figure 10 shows the average wait for the RITMO data and training sets of non-manipulative and manipulative agents.

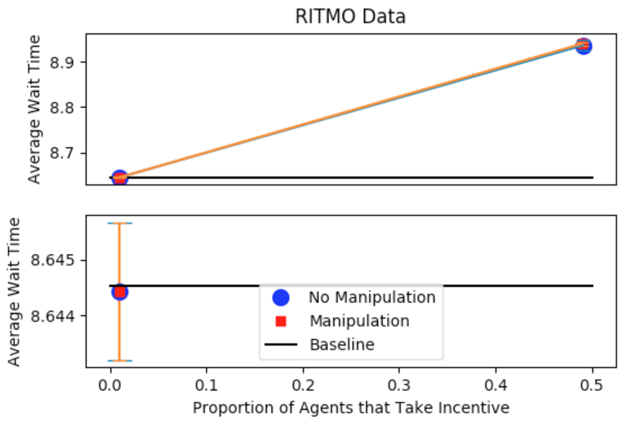

Figure 10: On the RITMO dataset, the average wait time shows the same patterns as the total loss in Figure 8. The average wait time is lower when a smaller proportion of agents are offered incentives, and the average wait time increases by a small amount when agents are manipulative.

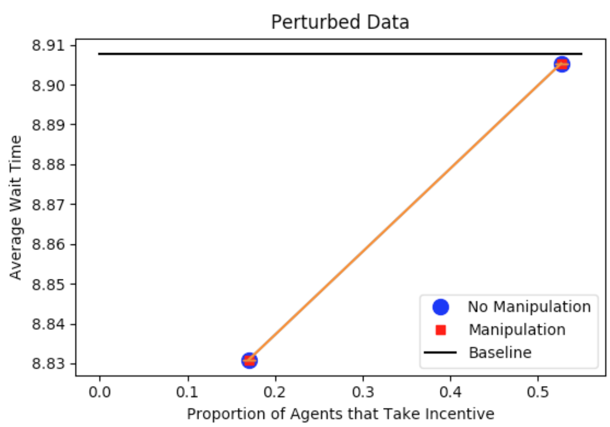

Figure 11: The average wait time is higher for all tested proportions of agents offered incentives, even though the total loss lowers in Figure 9. The average wait time remains the same when agents are manipulative.

This shows the exact same pattern as Figure 8, demonstrating that the loss and average wait times are directly related. This allows us to conclude that the inaccuracies with the scheduler's decision that an agent will benefit the system are from problems with the predictive wait time model. If the predictive wait time model accurately calculated expected wait times for agents, then the average wait time would lower when agents took incentives, and as a result the total loss would lower as well. It also shows that the incentive model is behaving how we would expect because there is a direct relationship between the average wait and total loss. Therefore, we can come to the conclusion that to improve the entire model, we need to improve the predictive wait time model.

The average wait time of the perturbed data is shown in Figure 11. Like the RITMO data, the average wait time shows a similar pattern as the total loss. While the increase in wait time as the proportion of agents that take an incentive increases is similar, the average wait time is always lower when agents take incentives. This shows that the predictive wait time model works better on the perturbed data, which is most likely a result of the stabilization of the perturbed data discussed in the previous section. Therefore there is potential in the incentive scheme and predictive wait time model, but we should proceed in attempting to improve the predicted wait time model for all datasets.

6 Conclusion

We developed an incentive scheme for the RITMO transit system with a predictive wait-time model. The incentive model aims to minimize system loss and improve social welfare, which is defined as the aggregate loss of agents in the system. In experiments with non-manipulative agents, the total system loss is successfully lowered compared to the loss in an unincentivized model. However, the average wait time is consistently higher for all incentive schemes. Therefore, the predictive wait time model can be improved.

The main contributions of this work are the creation of the agent-based incentive model and learned predictive wait model. Using learned wait times to calculate incentives for agents in a stochastic system is an innovative idea because riders in a transportation system value their time, and thus want lower wait times. This idea could expanded to a useful tool for dynamic pricing in any stochastic road network.

One potential way to improve the predictive wait time model is to look at traffic at an individual stop, rather than the entire zone. We could also switch from Gaussian process regression to a deep neural network, which could better estimate the unknown, underlying structure of the RITMO system. Given the relationship in experiments between the mean squared error and average wait times, improving the predictive wait time model will greatly improve the success of the incentive model. Other extensions are to not assume the scheduler knows private information, and to assume an estimated distribution of riders, rather than the exact number of riders. These extensions would make this model more deployable.

References

T. Arda, Y. Crama, D. Kronus, T. Pironet, and P. Van Hentenryck. Multi-period vehicle loading with stochastic release dates. *EURO Journal on Transportation and Logistics*, 3: 93–119, 2014.

S. Banerjee, R. Johari, and C. Riquelme. Pricing in ridesharing platforms: A queueing-theoretic approach. In *16th ACM Conference on Economics and Computation*, page 639, 2015.

K. T. Bui, V. A. Huynh, and E. Frazzoli. Dynamic traffic congestion pricing mechanism with user-centric considerations. In *15th International IEEE Conference on Intelligent Transportation Systems*, pages 147–154, 2012.

S. I.-J. Chien, Y. Ding, and C. Wei. Dynamic bus arrival time prediction with artificial neural networks. *Journal of Transportation Engineering*, 128:429–438, 2002.

A. de Palma, M. Kilani, and R. Lindsey. Congestion pricing on a road network: A study using the dynamic equilibrium simulator METROPOLIS. *Transportation Research Part A: Policy and Practice*, 39:588–611, 2005.

D. Guimarans, D. Harabor, and P. Van Hentenryck. Simulation and analysis of container freight train operations at Port Botany. *Eprint arXiv:1512.03476*, 2015.

T. Idé and S. Kato. Travel-time prediction using gaussian process regression: A trajectory-based approach. In *9th SIAM International Conference on Data Mining*, pages 1183–1194, 2009.

D. Joksimovic, M. Bliemer, and P. Bovy. Optimal toll design problem in dynamic traffic networks with joint route and departure time choice. *Transportation Research Record: Journal of the Transportation Research Board*, 1923:588–611, 2014.

W. Li. Accounting for travel time reliability, trip purpose and departure time choice in an agent-based dynamic feedback-control toll pricing approach. 2018.

A. Mahéo, P. Kilby, and P. Van Hentenryck. Benders decomposition for the design of a hub and shuttle public transit system. *Transportation Science*, To appear.

S. Mouthuy, F. Massen, Y. Deville, and P. Van Hentenryck. A multi-stage very large-scale neighborhood search for the vehicle routing problem with soft time-windows. *Transportation Science*, pages 223–238, 2015.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

V. Pillac, M. Cebrian, and P. Van Hentenryck. A column-generation approach for joint mobilization and evacuation planning. *CPAIOR*, pages 285–303, 2015.

C. E. Rasmussen. Gaussian processes in machine learning. *Advanced Lectures on Machine Learning*, pages 63–71, 2004.

G. Sharon, J. Hanna, T. Rambha, M. Levin, M. Albert, S. Boyles, and P. Stone. Real-time adaptive tolling scheme for optimized social welfare in traffic networks. *AAMAS*, 2017.

T. Teubner and C. M. Flath. The economics of multi-hop ride sharing. *Business & Information Systems Engineering*, 57: 311–324, 2015.

Y. Yin and Y. Lou. Dynamic tolling strategies for managed lanes. *Journal of Transportation Engineering*, 135, 2009.

B. Yu, Z. Yang, and B. Yao. Bus arrival time prediction using support vector machines. *Journal of Intelligent Transportation Systems*, 10:151–158, 2007.

B. Yu, W. H. Lam, and M. L. Tam. Bus arrival time prediction at bus stop with multiple routes. *Transportation Research Part C: Emerging Technologies*, 19:1157–1170, 2011.

G. Zhang, Y. Wang, H. Wei, and P. Yi. A feedback-based dynamic tolling algorithm for high-occupancy toll lane operations. *Transportation Research Record: Journal of the Transportation Research Board*, 2065, 2014.