ONLINE APPENDIX:

<u>Derivation of ADL Equilibrium Conditions</u> <u>***with Finite Late Cost***</u> (for $\overline{C}_1 < \overline{C}_2$ and $s_2 \geq s_d$).

The following two lemmas will be used in the derivations to follow:

<u>Lemma 1:</u> Once the first user in group 2 departs, there is always a queue at the upstream bottleneck until all group 1 users have departed.

Proof: Once the users from group 1 begin to depart (at t_1^o) a queue must exist at the upstream bottleneck; otherwise, a group 2 user could depart at the same time as someone from group 1 and incur an identical cost violating $\overline{C}_1 < \overline{C}_2$. Before group 1 users begin departing, if the upstream queue vanishes then a group 2 user could depart and incur lower costs than the previous user to depart (no delay at upstream queue, no delay at downstream queue (because $s_2 \geq s_d$), no late cost, and no greater cost of being early).

<u>Corollary</u>: The downstream bottleneck has a queue over the entire period from the first departure until all group 1 users are have departed.

Proof: Because $s_2 \geq s_d$ and because, from Lemma A, the upstream queue is not empty, there always will be a queue at the downstream bottleneck.

<u>Lemma 2:</u> The downstream bottleneck has a queue from the time of the first departure until the time of the last departure, t_{max}.

Proof: The corollary above covers the period up to the last departure from group 1. Between t_1^e and t^* if the downstream queue becomes empty then a user from group 1 could depart and incur a lower cost (arrive with no delay, no late cost, and a lower cost of being early than anyone in group 1 who departed earlier). This violates the equilibrium condition (3). Between the times t^* and t_{max} the downstream queue must not be empty or a member from group 2 could depart and arrive with no delay (the upstream queue must also be empty because $s_2 \geq s_d$) and hence incur a lower cost than the user who arrives at t_{max} again violating (3).

Derivation of each equation:

Equation (5) $t_2^o = t_{\max} - (n_1 + n_2)/s_d$: From Lemma 2, the downstream queue is never empty over this period and $n_1 + n_2$ users flow through the bottleneck.

Equation (4) $t_{\max} = t_2^e = t^* + \left(\frac{\beta}{(\beta+\gamma)}\right)\left(\frac{(n_1+n_2)}{s_d}\right)$: The first user from group 2 departs at t_2^o and arrives with no delay incurring a cost of $\beta(t^* - t_2^o)$. The last user from group 2 departs at t_{max} (and arrives then) incurring a cost of $\gamma(t_{\max} - t^*)$. Equilibrium requires that these costs be equal hence $t_{\max} - t^* = \beta(t^* - t_2^o)/\gamma$. Combining this condition with (5) above results in (4).

Equation (6) $d_2(t) = \frac{\alpha}{\alpha - \beta} s_d$ for $t_2^o \le t \le t^b$: For the entire initial departure period until group 2 users, upon arriving at the downstream bottleneck encounter the first group 1 user (label this time t^b), it is easy to show that group 2 users must depart at a constant rate, d_2^o. The first of these group 2 commuters incurs a cost of $\beta(t^* - t_2^o)$ and one who leaves at a time t, $t \le t^b$, spends $t_q = \frac{(d_2^o - s_d)(t - t_2^o)}{s_d}$ in queues (by Lemma 2) and arrives $t^* - t - t_q$ early. Equating the costs of these users yields equation (6).

Equation (7) : $d_2(t) = s_2$ for $t^b \le t \le t^f$, where t^b is the time at which group 2 commuters leaving the upstream bottleneck encounter the first group 1 commuter at the downstream bottleneck and t^f is the departure time of the first group 1 commuter who will arrive late at the destination. The departure rate for group 2 falls to a rate that is equal to the capacity of the upstream queue, s_2, at t^b and the upstream queue length stays constant for the entire period that these users will encounter group 1 users at the downstream bottleneck. This is clearly a necessary condition for an equilibrium because the only difference between the costs of the two groups of users over this period is the cost that group 2 users encounter in the upstream bottleneck hence this upstream delay cost must be a constant.

Equation (8) $d_2(t) = \frac{\alpha}{\alpha + \gamma} s_d$ for $t^f \le t \le t_{\max}$: [Derivation is similar (6) above.]

Equation (9) $t^f = \frac{\alpha + \gamma}{\alpha} t^* - \frac{\gamma}{\alpha} t_{\max}$: Some group 2 user must arrive exactly at t^*, otherwise another user could depart fractionally later and incur a lower cost. If this user departed at t^f then his cost is $\alpha(t^* - t^f)$. This cost must, by the equilibrium condition, be equal to the cost of the last arrival from group 2 (who arrives at t_{max} with no delay) which is $\gamma(t_{\max} - t^*)$. Equating these costs and rearranging terms leads to (9).

Equation (10) $t_1^e = t_2^o + \frac{n_2 - s_d(t_{\max} - t^*)}{s_2}$: The number of group 2 users who arrive late is $s_d(t_{\max} - t^*)$ because the downstream queue is never empty hence the last group 2 member to arrive on time departs the upstream queue at $t_2^o + \frac{n_2 - s_d(t_{\max} - t^*)}{s_2}$. This must be the departure time of the last user from group 1 to arrive on time.

Equation (11) $t_1^o = t_1^e - \frac{n_1}{d_1^o}$: For all group 1 users to have the same cost (as required for equilibrium), the marginal value of delaying departure by an incremental amount Δt must be 0. Such a delay increases queue delay costs by $\alpha \frac{(d_1(t) + s_2 - s_d)}{s_d} \Delta t$ where $d_1(t)$ is the departure rate of group 1 users at t. Likewise, the delay reduces early arrival costs by $\beta(\Delta t + \frac{(d_1(t) + s_2 - s_d)}{s_d} \Delta t)$. Summing these two expressions and equating to 0 yields $d_1(t) = \frac{\alpha}{(\alpha - \beta)} s_d - s_2$, which is a constant. As a result, group 1 members depart at the constant rate $d_1^o = \frac{\alpha}{(\alpha - \beta)} s_d - s_2$ and $t_1^o = t_1^e - \frac{n_1}{d_1^o}$.

Equation (12) $\overline{C}_1 = \alpha\left(\frac{\gamma}{(\beta+\gamma)}\left(\frac{(n_1+n_2)}{s_d}\right) - \frac{n_2}{s_2} + \frac{\beta}{(\beta+\gamma)}\left(\frac{(n_1+n_2)}{s_2}\right)\right)$: The last member of group 1 arrives at t*. Hence, $\overline{C}_1 = \alpha(t^* - t_1^e)$. Substituting (9) for t_1^e and rearranging $t_{\max} - t^* = \beta(t^* - t_2^o)/\gamma$ from the proof of (5) and substituting for t^*, gives (12).

Equation (13) $\overline{C}_2 = \beta\left(\frac{\gamma}{(\beta+\gamma)}\left(\frac{(n_1+n_2)}{s_d}\right)\right)$: Rearranging equation (4), we have

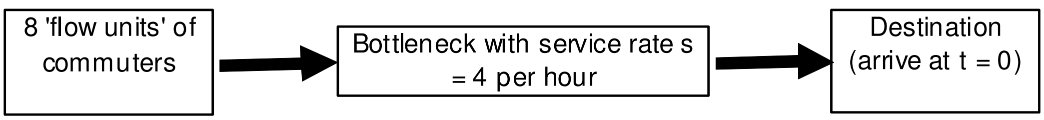
$$t_{\max} - t^* = \beta(n_1 + n_2)/s_d(\beta+\gamma).$$

Multiplying both sides by γ gives, on the LHS of the equation, the cost of the last group 2 user to arrive and, with the equilibrium condition, implies (13).

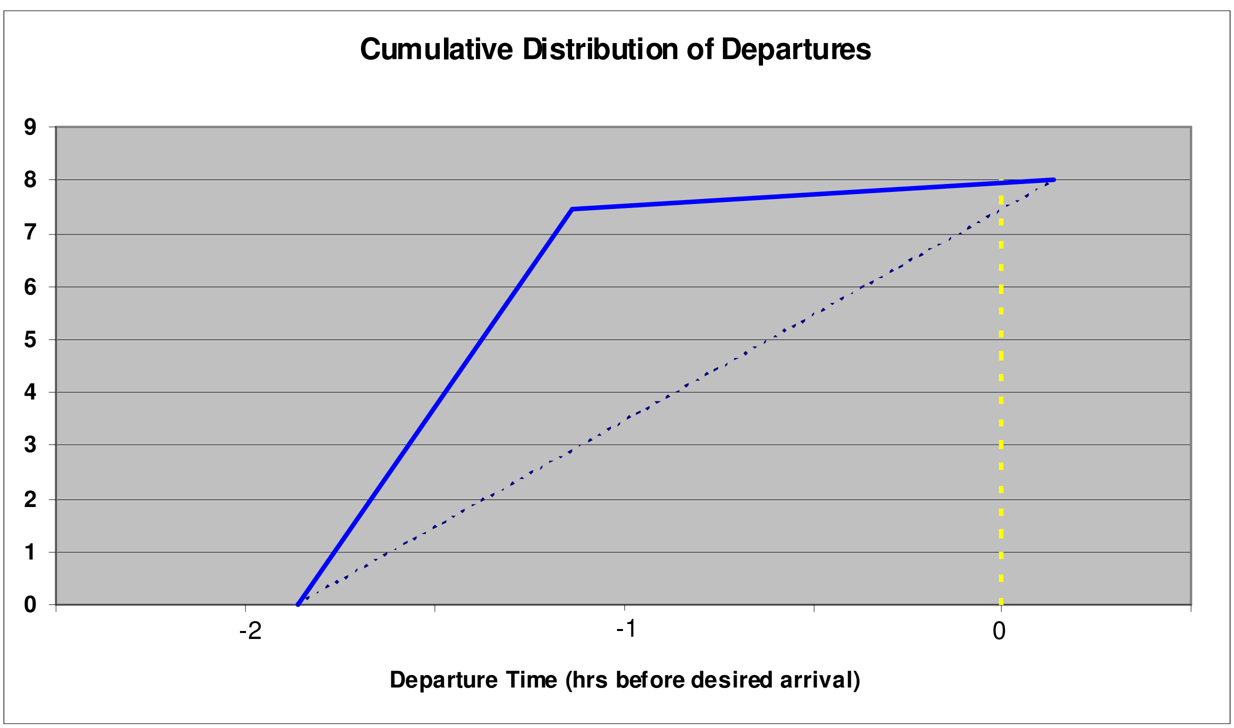
Discussion of the Equilibrium Conditions for an N- person Departure Time Model

For modeling most real traffic networks, the infinite-agent ‘flow’ model (as used in ADL and Kuwahara and others) is a good (efficient and accurate) approximation of the real network with its large numbers of commuters. However for the purposes of laboratory experimentation where practical considerations dictate that the numbers of agents be small (24 in our case), it is necessary to consider how the equilibrium changes with a finite set of agents.

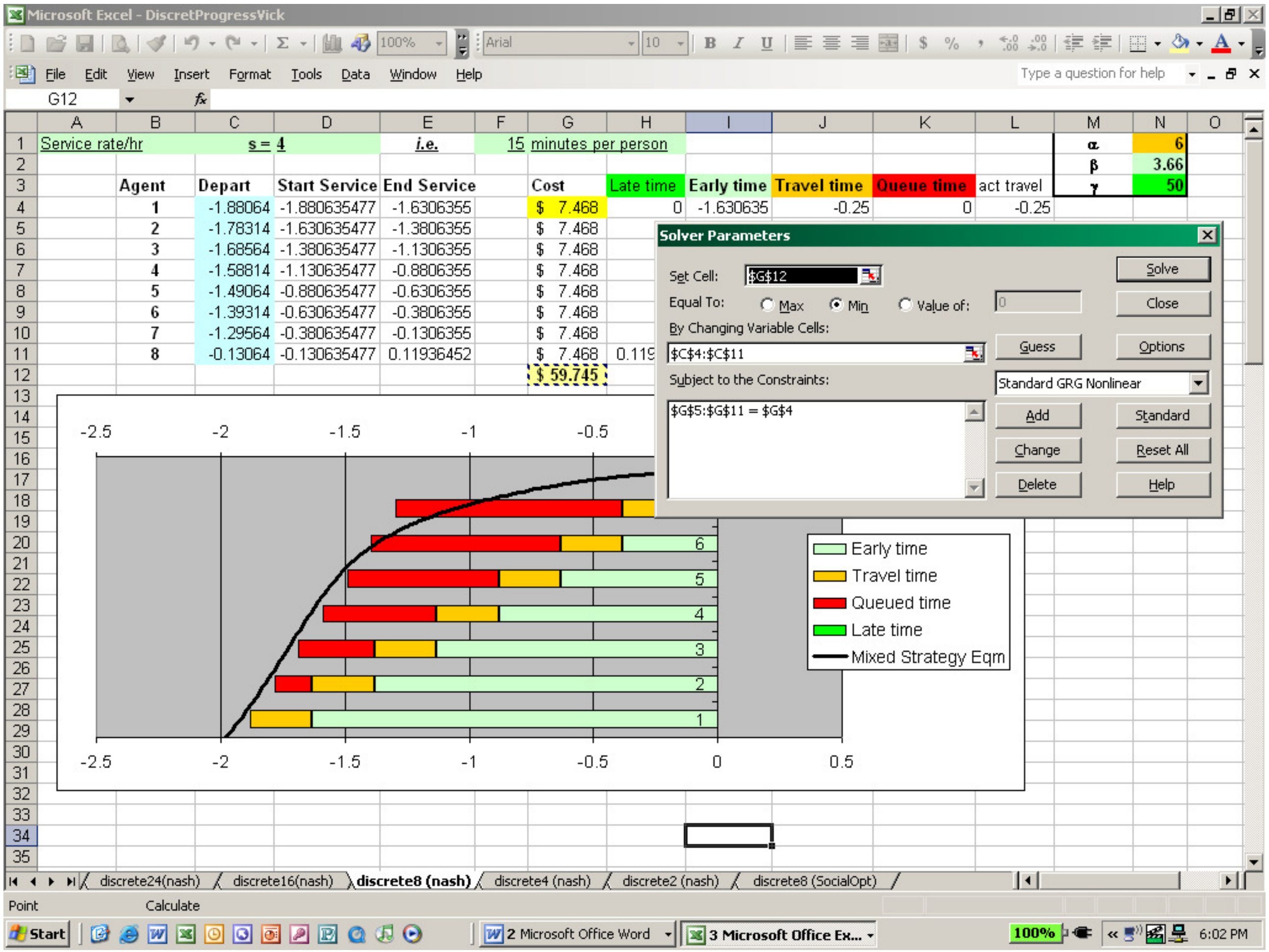
The finite agent versions of both the ADL model and even Vickrey’s original single bottleneck model become very difficult to solve for Nash Equilibrium conditions when the decision space (set of possible departure times) is large. In such cases, it becomes readily apparent that no pure strategy Nash equilibria exist and that the symmetric Nash equilibrium must be composed of mixed strategies. To illustrate this more clearly, consider the simplest case, the basic Vickrey model with a single bottleneck. Assume that we have a ‘dense’ set of 8 units of commuters (infinite number with ‘volume’ of 8) and that the bottleneck processing rate is 4 units per hour. As a result, it will take 2 hours to process the entire population.



The graph below shows the Nash equilibrium set of departure times for this model – the familiar Vickrey curve illustrating that departures occur in non-Pareto fashion with significant queuing costs.



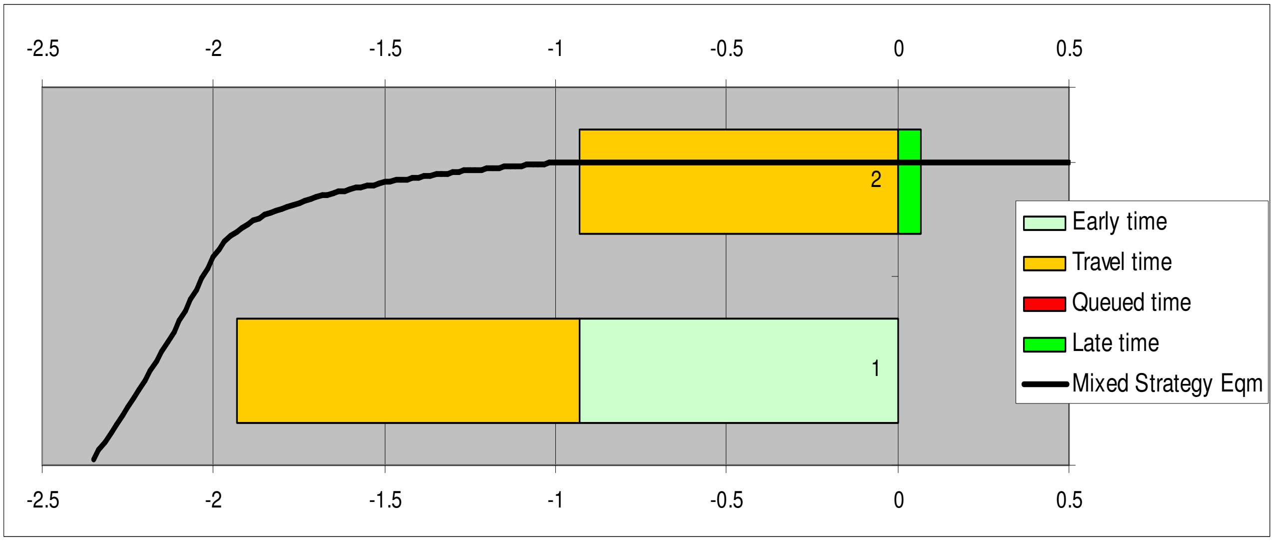
Now if we replace this infinite set of agents with a set of 8 finite commuters, then what does the equilibrium set of departure times look like? To explore this, consider the problem of finding a set of 8 departure times such that the total commuting costs of each agent were mimimized subject to the constraint that all commuters have the same total cost. The figure below shows a mathematical programming model (in a spreadsheet format) to find such a solution.



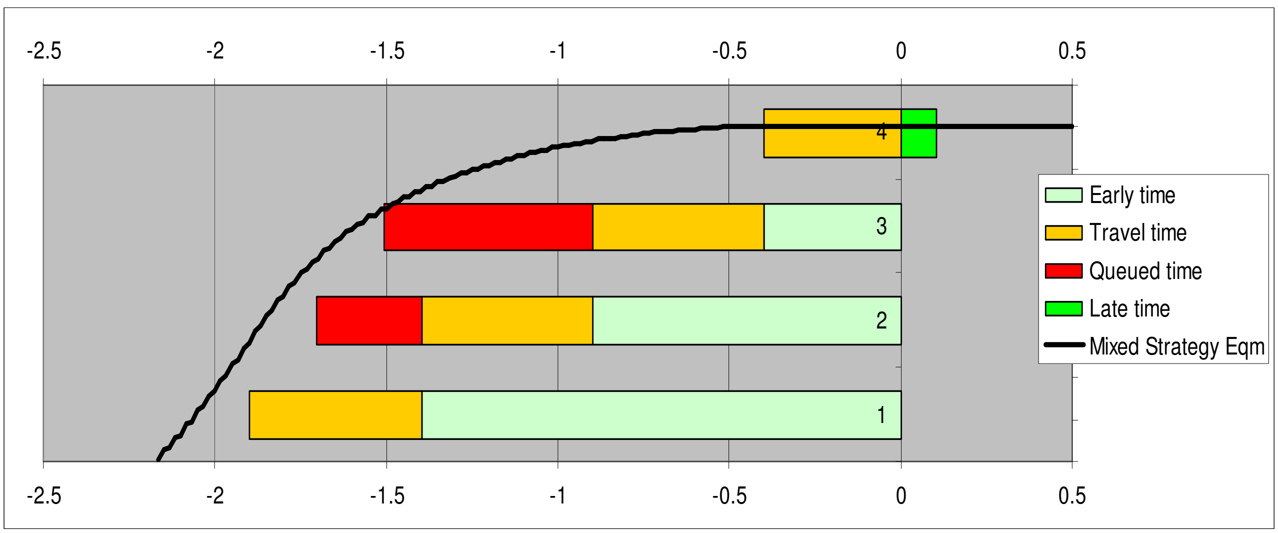
The times in the blue cells in column C in the spreadsheet are the resulting departure times that give equal but minimum costs to all agents and the bar graph shows the resulting departures and times spent in queue, being served and waiting (early) at the destination for each agent. Such a solution is referred to, in the paper, as the ‘discrete’ equilibrium although it clearly is not a Nash equilibrium. Any agent, given the departures of the others, could depart a few seconds later (not so much as to depart later than the next agent) and reduce his or her waiting and incur lower costs.

For the simple Vickrey network shown however we have been able to develop an iterative solution method based on modeling the process as a non-stationary Markov network and thereby numerically solve (in a sequence of steps) for the true mixed strategy equilibrium. The algorithm is computationally burdensome and for more than a few agents, the solution times are very large (work is proceeding on this algorithm). The black solid curve in the above figure shows this solution: the cumulative distribution of the symmetric mixed strategy Nash equilibrium for the above problem. Note the reasonably close correspondence to the ‘discrete’ solution. In the sequence of graphs on the next page we see discrete versions of the model in which the number of agents, N, ranges from 2 to 24 (the latter being the number in our experiments). As would be expected, the correspondence between the discrete solution and the mixed strategy solution rapidly improves as N increases and, as can be seen in the N = 24 case, both appear to converge to the Vickrey ‘flow’ solution.

N = 2 agents

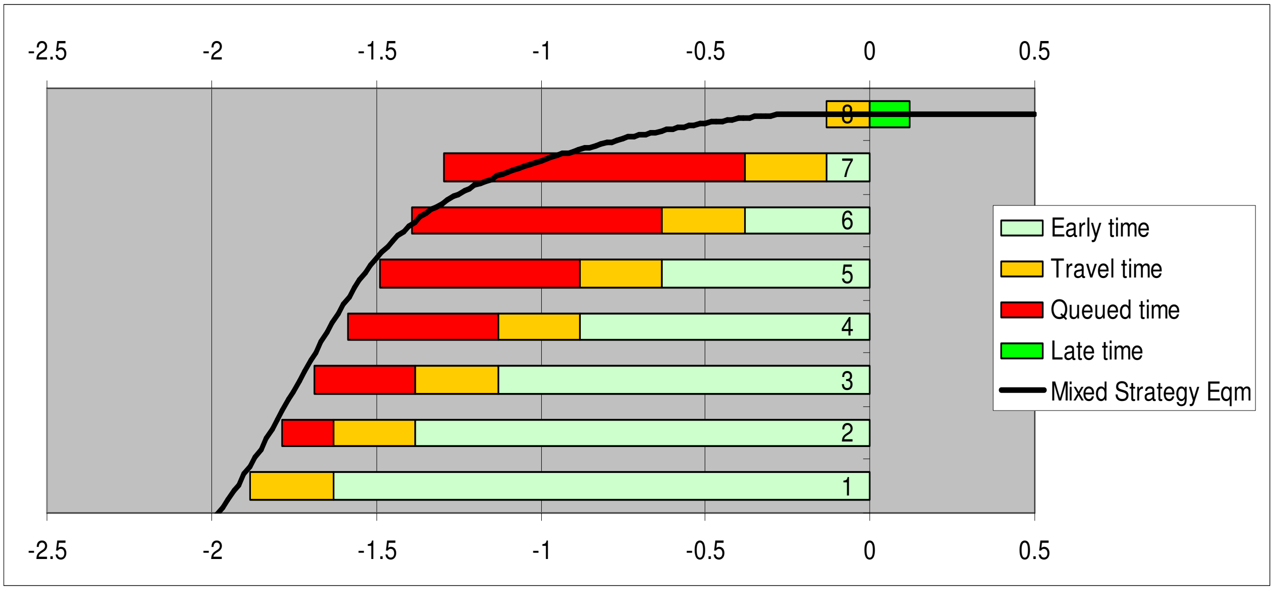


N = 4 agents`

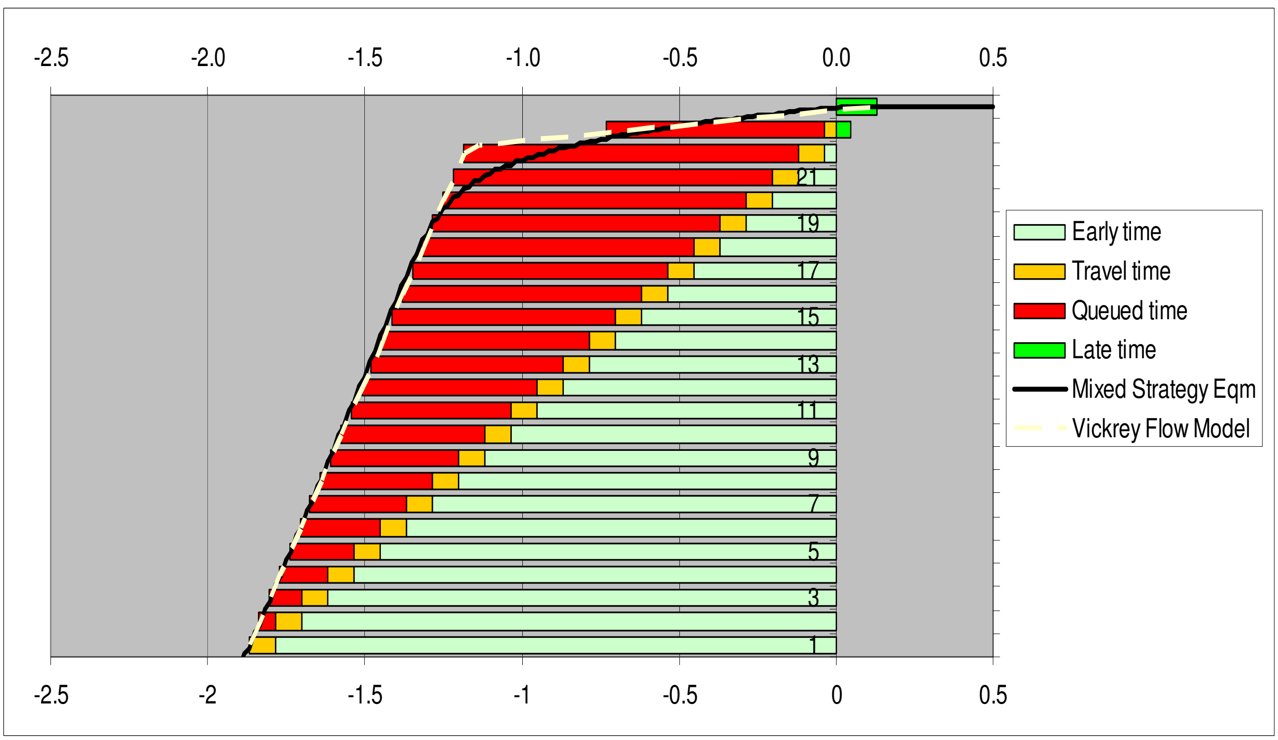


Note that with a small number of agents (2 and 4 above) the early departing agents must occasionally (in the mixed strategy Nash solution) depart quite early (more than 2 hours prior to the desired arrival time) in order to 'keep the other agents honest' (i.e. from departing just a few seconds before the expected departure time of the first-departing agent and thereby eliminating all waiting costs). As the number of agents grows, this need decreases. As can be seen below, by N = 8 the 'discrete' solution is a good approximation of the full mixed strategy set of the departure times. By N = 24 the correspondence is almost perfect and we note as well, it is virtually the same as the Vickrey flow (infinite agent) model.

N = 8 agents

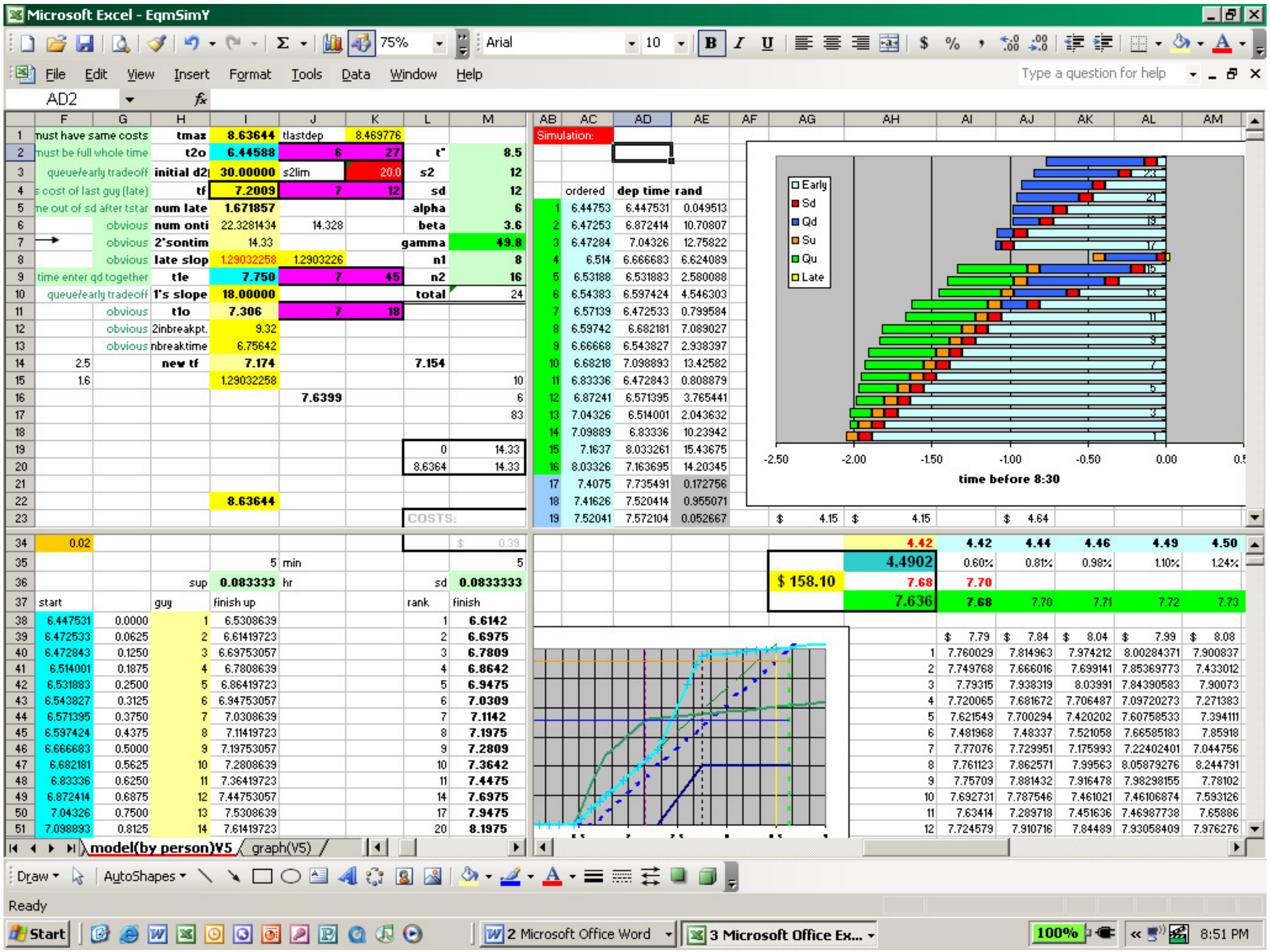


N = 24 agents



While the correspondence is very close for the N = 24 case, it is also quite close at N = 8.

In our Y-shaped network, we have two interacting populations, one of size 8 and one of size 16 (for a total of 24 commuters). While we do not have a version of the Markov algorithm to solve for the mixed strategy equilibrium for the Y-network, we can find both the ‘flow model’ solution (as derived at the beginning of this appendix) and a ‘discrete’ solution (analogous to the above discrete solution) using mathematical programming. These solutions were then used in a large simulation model as the basis for a search for a pair of departure distributions, one for each of the two populations of commuters (of sizes 8 and 16 respectively) that would constitute a symmetric mixed-strategy Nash equilibrium. For each test solution in this search, commuters randomly drew departure times from the respective distributions and commuting costs were commuted for those times. This Monte Carlo method was repeated for 10,000 'days' (trials) for each prospective solution and the results then checked for consistency with the equilibrium requirements that all agents have the same (approximate) average commuting costs and that no other departure times outside the solution set give lower commuting costs (subject to the other agents playing this solution). This simulation model for the Y-network is available on request; below is a screenshot of one iteration of the model.



As is stated in the paper, the precise (mixed strategy equilibrium) solution of the discrete agent model for the Y network is not likely to be found easily, we can be reasonably confident that that solution will not differ considerably from the three solutions (flow, discrete, simulated) that we use in the paper as benchmarks for experimental behavior. Regardless of which model is used (and they all give similar results), there is a pronounced Braess effect in the Y network. This provides behavioral confirmation to the theoretical findings of ADL.