A Disaggregate Quasi-Dynamic Park-and-Ride Lot Choice Model Application with Parking Capacities

John Gibb
Senior Transportation Engineer
DKS Associates
8950 Cal Center Dr, Suite 340, Sacramento, CA, USA, 95826
jag@dkssacramento.com

Submitted May 29, 2009 to the proceedings of the 12th National Transportation Planning Applications Conference of the Transportation Research Board, Houston, Texas, May 2009

Abstract

This report describes a model of park-and-ride lot choice for morning commuters driving to transit boarding stops with limited parking. It is applied to individual disaggregate trips created by an activity-based travel demand model, such that each chooses the parking location of least generalized cost. Its dynamic aspect is that it simulates the filling up of parking lots as time progresses; once a lot fills up, it is unavailable to later commuters. An efficient application with a popular transportation planning software system is described for the lot choice, auto and transit assignments of the trips, period-representative “skim” time measures, and its handling within the solution of system equilibrium.

Introduction

It is a common necessity for travel demand models to include some methodology representing the ability of travelers, especially commuters to work, to complete a trip by driving from home to a public “park-and-ride” lot located near a transit stop or station, and continuing the trip on a public transit service to someplace near the destination, to walk the rest of the way, and to return in the reverse sequence of modes. The network modeling of these methodologies are concerned with these tasks:

1) Determining the choice of which park-and-ride lot(s) used by each trip or grouping of trips. This determines how the remaining tasks are accomplished.
2) Measuring travel times and other level of service “skim” values, to furnish utilities in mode choice models,
3) Assignment of the driving portion of such trips to the highway network, and
4) Assignment of the transit portion of such trips to the transit network.

In customary four-step models, the mode choice model, and possibly the trip distribution model, would use the skim values to determine how many trips, between each park of network zones

(TAZs), will travel in the park-and-ride mode (also called drive-to-transit, transit-drive-access, etc.)

In the development of an activity-based travel demand model for the Sacramento Area Council of Governments (SACOG), came a decision whether to retain the park-and-ride methodology in their trip-based four-step model, or develop a new one.

SACOG's four-step model "SACMET" uses the standard methodology provided in TP+ and Cube Voyager for generating drive-to-transit level-of-service matrices, and transit assignment. This methodology is an old and customary approach in which special drive-access links are created from user-coded pairings of TAZs to transit station nodes. TP+/Voyager does not have a standard means to assign the auto-access vehicle trips to the highway network (since one end is a node, not a zone). Instead, a system of unusual tricks and coding conventions was developed to connect the parking-end of such trips to a nearby zone, enabling inclusion in highway assignment.

The practical means by which the user prepares TAZ to park-and-ride station pairings is to draw and list a "catchment area" of all TAZs to be given access to any given park-and-ride lot. The catchment area approach is supported by survey observations showing most users of park-and-ride lots live fairly close to them. In the former MINUTP version of SACMET, memory limitations practically required that any zone be a member of only one station's catchment area. Consequently, the choice of where anyone would park is made entirely by the user (as is whether the residents of a zone can drive to transit at all). TP+ and Cube Voyager do not have this memory limitation, so many SACMET datasets are made with some overlapping between catchment areas.

In practice, catchment-area zone lists can be difficult to code and maintain, especially when forecasting alternative scenarios. If the model overloads a park-and-ride lot beyond its capacity (actual or foreseeable), the only recourse is to remove zones from association with the lot, associate them to other lots, and run the model again, requiring time-consuming judgment-making and trial-and-error. Mistakes easily deprive some zones of all drive access to transit. Much depends on the judgment and technique of the analyst.

A different methodology has been available since at least 1994. Users of the travel model software EMME/2 (now EMME) have been applying matrix-based models of park-and-ride choice with a special zone at each park-and-ride lot. These were made practical with a "matrix convolutions" module (now called "triple-index operations") – explicit loops through each possible intermediate zone between each origin and destination zone (Blain, 1994). These are entirely matrix processes, distinct from the separate network path-building processes of auto and transit modes. This matrix methodology includes both a skimming stage, and a trip-splitting stage which converts the transit-drive trips into separate drive trips (for inclusion in auto assignment) and transit trips (for inclusion in transit assignment).

Many of these models, including Blain's, are multinomial logit choice among all accessible park-and-ride zones. Consequently, drive-to-transit trips from any origin to any destination are split in some amount to all accessible park-and-ride lots. Estimated or calibrated coefficients of these models commonly weight the drive access time between three and six times compared to transit in-vehicle time, favoring but not forcing proximity between home and parking for such trips.

Soon afterwards, parking lot capacity restraint methodologies were added to these models. One widely-used methodology (Spiess, 1996) iteratively solves for a "shadow cost" imposed on each full parking lot, so that every park-and-ride lot satisfies the rule that either its demand matches capacity, or it has no shadow cost and demand is less than capacity.

While developing the activity-based model system SACSIM for the Sacramento Area Council of Governments using DaySim (Bowman and Bradley, 2006), a methodology was sought that avoids user-coded catchment areas or similar judgmental inputs, avoids unusual tricks, and automatically satisfies parking capacity limits. While either of these two basic methodologies could have been incorporated into the SACSIM system (for pragmatic if not ideal reasons), it was realized that some new opportunities could give rise to new approaches:

1) TP+ and Voyager permits explicit and versatile user-coded loop control and matrix cell addressing capabilities in its matrix processing program, unlike those in MINUTP and most other modeling software, which basically process matrices sequentially cell-by-cell. These capabilities are more general than EMME's triple-index operations.
2) TP+ and Voyager can process general database-record data, not limited to TAZ-indexed data, and can interface such data with matrix and zonal data.
3) The activity-based demand model per se, "DaySim", creates individual disaggregate trips. Just as the individual disaggregate approach within DaySim makes complex activity-based demand models practical, individual disaggregate trips can make practical some models of park-and-ride choice that are difficult with matrix aggregations.

Some park-and-ride lot choice models were proposed for use with SACSIM, that take advantage of TP+'s capabilities. These models include:

(1) Multinomial logit with shadow cost solution,
(2) All-or-nothing choice of the least generalized cost,
(3) All-or-nothing least generalized cost choice, but with maximum drive times solved for each full lot so that demand does not exceed capacity. (A maximum drive time can be considered a catchment area radius, but with catchment areas of different lots freely overlapping.)
(4) Simulate filling of parking lots over time, making each lot that fills up unavailable to later trips.

Model (1) was not developed or tested in TP+ for this effort. Model (2) is the basis of the other models, and remains the method of park-and-ride lot choice in the first iteration of the model system, when all times are free-flow. Model (3) has appeal as an analytical non-judgment-based method of catchment areas, although its behavioral basis is unclear.

Model (4) has realism not apparent in the other models: park-and-ride spaces are available to any of the public who arrive before they fill up, and closed to any who come too late; there are no policies that a user must live in a designated area. It is straightforward and efficient to apply upon individual trips generated by DaySim. This is the park-and-ride lot choice model implemented in the SACSIM system, and described further in this report.

The remaining sections of this report lay out a specification of the park-and-ride lot choice problem, describe the solution that was employed in the SACSIM system, and then identify a slightly more complex general solution.

A Specification of the Park-and-Ride Lot Choice Problem with Capacities

A conceptual specification of park-and-ride lot choice with capacities is here laid out.

Definitions:

D_n is amount of parking demand of the *n*'th unit or bundle of park-and-ride travel.
(This demand could be a matrix cell, single trip, or infinitesimal unit depending on the application. Indices *i* and *j* each could be a zone, home, jobsite, and/or building, or separate occurrence thereof, depending on the application.)
i_n and j_n origin and destination zones of demand unit *n*.
k a park-and-ride lot alternative within choice set *K*.
U_{ik}^{drive} utility (or negative generalized cost) of driving from origin *i* to alternative *k*.
$U_{kj}^{transit}$ utility (or negative generalized cost) of transit from alternative *k* to destination *j*.

C_k capacity of alternative *k*.
P_{nk} priority of demand unit *n* to determine its eligibility to choose alternative *k*, i.e. trip *n*'s order of arrival or acceptance to lot *k* if choosing it.
Q_k minimum priority cutoff for each alternative that fills to capacity.

Specifications:

1) The utility of an alternative *k* for trip *n* is
$V_{nk} = U_{i_n k}^{drive} + U_{k j_n}^{transit}$
2) Each individual demand unit *n* chooses the best alternative k_n^* such that V_{ijk^*} is the highest among all *k* that also satisfy $P_{ijk} \geq Q_k$.
3) A solution to the problem is a solution to all Q_k so that
$$\begin{Bmatrix} \sum_{n:k=k_n^*} D_n < C_k \text{ at all } k \text{ where } P_{nk} \geq Q_k \text{ for all } n \\ \sum_{n:k=k_n^*} D_n = C_k \text{ at all } k \text{ where } P_{nk} < Q_k \text{ for some } n \end{Bmatrix}$$

Specification 1 simply identifies the total utility, or negative generalized cost, of the drive plus the transit components of each alternative, for each unit of demand. Specification 2 is choice of maximum utility, but only among those alternatives to which the demand unit has a high enough priority level or ranking. An individual demand unit's priority to one alternative may be unrelated to its utility for that alternative, and to its priority to another. Specification 3 identifies the unknown to be solved: the priority cutoff levels for each alternative. If an alternative doesn't fill, then all demand units are eligible, but if it would otherwise overfill, then the alternative's cutoff level is solved to match demand to capacity.

In park-and-ride lot choice models based on chronological filling of lots (Model 4 above), the priority cutoff level of a lot is simply the time when it fills up. Meanwhile the priority of a trip or demand unit for that lot is (or should be) the time it would arrive to park there if it could and chose to. (In Model 3 in the above section, priority is shortness of the drive trip, instead.)

In the models discussed in this paper, all drivers are assumed to make informed choices with knowledge of when park-and-ride lots of interest fill up. None are modeled as driving to a lot, finding it full, then driving from there to another, etc. until successfully parking. This is deemed reasonable for transit park-and-ride lots that mostly serve regular commuters to work in the morning; negligibly few at a time are testing the opportunities. Also, consistent with regular commuter usage, the models formulated here assume most vehicles stay parked through the day until the evening commute period; parking space turnover is not here considered.

An Efficient One-Pass Application

If we assume drive-access times are fairly short (compared to the analysis period and the transit travel times), then the departure time from home may serve as a proxy for arrival time to a parking lot, in the determination of an individual's priority. The problem to solve is then much simpler, since an individual's departure time is constant across all k; departure time for P_{nk} may be simply called P_n. The solution procedure is simply to sort the demand trips on departure time, then in this chronological order of departure time, calculate utilities, choose the best alternative among parking locations that have not yet filled, and accumulate demand to that choice.

In the SACSIM implementation of this procedure, demand is individual trips in the AM commute period generated by DaySim, each having a specific time of departure, and origin and destination TAZs, among other data. For each trip, this model selects one zone for this trip to park. Only zones with remaining parking capacity are allowed. With this selection, the trip is split into an auto trip from the origin to the parking zone, and a transit trip from the parking zone to the destination. Processing is entirely by the TP+/Voyager matrix programs, using its functions to read, sort, and write database tables, look up matrix cell values, and aggregate data records to matrices.

Each trip is linked to the same person's return trip and the return trip is split into a transit and an auto trip through the same parking zone. (The return trip may have a different origin than the original drive-to-transit trip's destination, and/or a different destination than the original trip's origin.)

The resulting auto and transit trips are then aggregated into trip matrices by time period for inclusion in the auto and transit assignments. This trip processing model is applied after DaySim (since DaySim trip predictions are input), and before auto assignment (since the auto portions of trips are included in the assignments).

The parking lot choice model makes a single choice for each drive-to-transit trip of the parking zone, among those available for parking and not filled up, having the least generalized cost combined from the auto and transit portions of travel parking at that zone. The generalized costs are as follows, for origin zone *i* and parking zone *k*:

$GC(auto)_{ik}$ = 3*Auto $Time_{ik}$ (minutes)
+ 2*(Terminal $Time_i$ + Terminal $Time_k$)
+ 2*(Auto $Distance_{ik}$ * 5 cents/mile + Parking $Cost_k$ /2) * 0.0558 minutes equivalent/cent
/ 1.28 persons per vehicle

$GC(transit)_{kj}$= In-Vehicle Time (minutes)
+ 2*Walk Time
+ 1.5*First Wait Time
+ 2*Transfer Time
+ 2*Fare * 0.0558 minutes equivalent/cent

Consequently, $V_{nk} = -\mathrm{GC(auto)}_{i_n k} - \mathrm{GC(transit)}_{k j_n}$

Costs are in 1990 cents, consistent with Sacmet data. (The U.S. Bureau of Labor Statistics reports that $1.00 in 1990 is equivalent to $1.49 in 2005 and $1.55 in 2006.) The factors on costs are taken from the Sacmet model's middle stratum of cost factors for work trips, and imply a value of time of $5.38/hour. Parking cost is a placeholder for potential policies not in place, and is specific to park-and-ride activity, rather than destination parking.

This model processes AM trips in chronological order, according to the predicted time-of-day of each trip. The trip start-times from DaySim are specified in whole minutes, and many minutes have several trips departing. To break ties, a random number is added and saved with each trip, to settle the order in which trips are processed and given priority at parking lots. One parking zone is chosen for each DaySim drive-transit trip, which has the least total generalized cost from its auto and transit legs. The remaining capacity of the chosen zone is decreased by one vehicle; if that was the zone's last available parking space, then the zone is unavailable to all later trips.

In addition to the trip pairs labeled with the parking lot choice, the AM drive-transit trip processor also outputs the time at which each parking zone fills up.

For the midday period, all lots that fill up in the AM period are modeled as unavailable. For PM and evening, all lots are assumed available for arriving drive-to-transit trips.

It is normally assumed that a commuter can reach the destination through the transit system from any available park-and-ride lot. In practice, sometimes this assumption fails, where the transit system isn't interconnected enough to reach a transit-accessible destinations from any park-and-ride lots. For these cases, exception-handling may place them into already-full parking locations, with a steep saturation penalty.

Implementation on Level of Service Measurements

DaySim uses zone-to-zone measurements of times and costs for auto-access transit among its inputs. These must be provided for all zone-to-zone movements having access to this mode, not just those having trips.

The AM transit-drive skimming module, which is run separately after auto assignment, uses the loading schedule to compute weighted-average skims for the next model system iteration. (In the first iteration of SacSim, when times are free-flow, no lots are filled, so this schedule is empty.) The loading schedule, in effect, partitions the AM period into a number of time increments, each of which has a certain set of parking zones available. The share of trips loaded during each increment serves as the weight for averaging.

For each origin-destination pair and each time increment, level-of-service measures are computed by choosing the yet-unfilled parking zone yielding the least combined generalized-cost, according to the same criteria described above for trips. This implementation takes advantage of the fact that each time increment has one less particular parking zone available than the previous, which is the zone that filled up at the start of the increment.

In the special case of the first iteration through the SacSim system, level of service measures are needed before there are any trips to fill parking lots. All parking zones are considered available for this case.

For the mid-day period, two level-of-service matrix sets are made. One has all parking zones that filled in the AM period unavailable, for use by DaySim for the off-peak periods. The other has all parking zones available, for use by the airport mode choice model (which includes an explicit transit drop-off mode). PM and evening transit skim matrices are not made. Instead, DaySim uses the opposite direction of AM skims for the PM period, and the opposite direction of mid-day for the evening.

Other Implementation Features

For each drive-to-transit trip, DaySim (usually) creates a returning transit-to-drive trip by the same person later in the itinerary. SACSIM identifies the pairings of these trips. The parking zone chosen in the drive-to-transit direction is also used for the return trip.

The table of completed drive-transit and transit-drive trip pairs includes these descriptive data fields:

- Drive-transit trip origin zone
- Drive-transit trip destination zone
- Drive-transit trip departure time
- Drive-transit trip assignment-period
- Drive-transit trip random tie-breaker
- Parking zone

Transit-drive trip origin
Transit-drive trip destination
Transit-drive trip assignment-period

Usually the drive-transit trip origin is the same as the transit-drive trip destination, but not always. Likewise the drive-transit trip destination usually but not always matches the returning origin.

This table completely specifies the auto and transit trips for assignment due to the park-and-ride transit mode. The two auto trips are drive-transit origin to parking zones, and parking to transit-drive destination, each selected to the appropriate period. The two transit trips are drive-transit parking to destination, and transit-drive origin to parking, each in the respective periods.

The DaySim activity-based model does not explicitly represent auto drop-off to transit, or auto pick-up when returning. These "kiss-and-ride" trips are implicitly present, classified as auto-access transit. Furthermore, DaySim does not distinguish multiple persons carpooling to park-and-ride. To compensate, each individual trip is assumed to accumulate 0.71 parking spaces, this rate based on past surveys.

A General Solution

For the more general parking problem where an individual would arrive at one park-and-ride lot at a different time than another, there is no certain priority order in which an individual's choice may be determined. No sort key may be identified as in the above chronological method.

The Gale-Shapley algorithm (Gale and Shapley, 1962) is applicable to the park-and-ride lot choice problem with capacities, individual trips, and alternative-dependent priority. (Some on-line resources on the Gale-Shapley algorithm are listed in the References). It is a one-to-one or many-to-one matching algorithm, with historic applications in placement of graduating medical students to hospital residencies, and other applications in college admissions and entry-level labor markets. It is iterative, and converges exactly in a finite number of iterations.

Gale-Shapely is commonly explained in terms of the "stable marriage problem." Let there be a collection of men, and an equal number of women, who gather to be matched for heterosexual monogamy. Each man knows enough about the women to form his own preference ranking of the women, and each woman likewise knows about the men. Rounds of proposals begin. Each man first proposes to his favorite woman. Each woman who receives a proposal tentatively engages her favorite proposer. In successive rounds of proposals, each man who is not tentatively engaged proposes to the next woman down his preference list (even if she is engaged); each proposed woman tentatively engages her favorite proposer, even if it means releasing a previous engagement. When every man is engaged, the process is complete.

As the process progresses, men work their way downward in their preference list, while women trade upward. It would be futile for any man to propose again to a woman higher in his list.

This process results in “stable marriages”, in that while there are men who prefer other women to their final mates, and women who prefer other men to their final mates, no man and woman prefer each other to their respective mates. There may be multiple stable pairings possible. The Gale-Shapley process yields pairings that are male-optimal (i.e. proposer-optimal; if women propose then the results are female-optimal). Convergence is assured, in finite time, to an exact solution. (If ties are present, then a tie-breaking mechanism may need to be introduced.)

In this park-and-ride problem, the “men” are units of demand (trips), and the “women” are parking spaces. The “men’s” preference for “women” is utility. The “women” are inanimate, but have preference nonetheless by order of arrival of the “men”. Actual time of arrival determines an individual’s priority, which may vary by alternative for an individual demand trip. The parking spaces in a lot are identical and interchangeable, so we need not be concerned with when each particular space fills, only when each whole lot fills.

The Gale-Shapley algorithm readily accommodates problems with more “women” than “men”. (Some “women” would be left engaged to nobody.) It is readily adapts to cardinal instead of ordinal preference data, such as utilities and times of filling.

As the algorithm progresses for the park-and-ride problem, the time at which any parking lot fills never moves later, only earlier. The utility of an individual’s best available alternative never gets better, only worse. Convergence is reached when no parking lot is oversubscribed. The converged result is user-optimal, by analogy to the male-optimal stable-marriage solution.

Further Study and Development

The disaggregate time-based placement of drive-access transit trips to park-and-ride lots begins to raise various questions of personal travel and time-use behavior, such as:

1) Do commuters, avoiding the risk of being turned away from full park-and-ride lots, prevent them from routinely filling? Although the base-year model depends on parking capacity constraint to prevent overloading certain locations, none have been recently observed suffering routine complete filling and overflowing. Risk-averse commuters may avoid the likelihood of being turned away, and in so doing, prevent lots from routinely filling.
2) Do individuals leave home earlier than necessary to get a “competitive” space? That is, do some people shift their departure time, for a higher-utility alternative that may otherwise be unavailable? Or, more broadly…
3) How can an activity-based demand model respond to the time-dependence of travel time and utility of park-and-ride travel?
4) What are the turnovers of park-and-ride spaces, being later vacated later in the day after being filled? And to what extent do the relatively smaller numbers of mid-day travelers take advantage of them?

References

Bowman, J. L., Bradley, M. A. (2006) Activity-Based Travel Forecasting Model for SACOG: Technical Memos

Number 1 - Model System Design
Number 2 - Population Synthesis
Number 3 - Design of Model System Application Software
Number 4 - Mode Choice Models
Number 5 - Intermediate Stop Location Models
Number 6 - Day Pattern Activity Generation Models
Number 7 - Time of Day / Activity Scheduling Models
Number 8 - Usual Location and Tour Destination Models
Number 9 - Household Auto Availability Model
Number 10 - DaySim05 Documentation
Number 11 - Impedance and Accessibility Effects

DKS Associates (2002), *Model Update Report, Sacramento Regional Travel Demand Model Version 2001 (SACMET 01)*, Sacramento Area Council of Governments 02-003.

DKS Associates (1994) *SACMET Regional Travel Demand Model Version 94.0 Model Development and User Reference Report*, for Sacramento Area Council of Governments.

Cube, *Voyager*, and *TP+*, Version 4.0 (2006), Citilabs, Oakland.

Blain L. (1994). *Park-and-Ride Choice using Matrix Convolutions.* Presented at the 9th Annual International EMME/2 Users Group Conference, Montréal, Québec.

Spiess, H. (1996), *A Logit Parking Choice Model with Explicit Capacities,* EMME/2 Support Center, Aegerten, Switzerland.

EMME/2 Release 9 (1998), INRO Consultants, Inc., Montréal.

D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9-15, 1962.

Some on-line resources on the Gale-Shapley algorithm:
`http://en.wikipedia.org/wiki/Stable_marriage_problem`
`http://eprints.gla.ac.uk/4495/`
`http://www1.cs.columbia.edu/~evs/intro/stable/writeup.html`
`http://www.csee.wvu.edu/~ksmani/courses/fa01/random/lecnotes/lecture5.pdf`