General framework for Dynamic Demand Simulation

Michel Bierlaire* Rabi Mishalani† Moshe Ben-Akiva‡

February 23, 2000

Report # RO-000223
ROSO-DMA-EPFL

Abstract

The development and evaluation of Dynamic Traffic Management Systems (DTMS) for Intelligent Transportation Systems (ITS) applications require sophisticated simulation tools. Many traffic simulators representing traffic at various levels of aggregation have been developed and used. Despite the importance of demand in this context, these tools focus mainly on the supply aspects of the transportation systems. Demand is usually just an input to the simulator.

The lack of dynamic demand simulators seems to be due to the difficulty of combining different models (like discrete choice models and OD matrix estimation) within a common environment. Therefore, a unifying framework, where different models can cooperate, will provide the necessary incentives for the development and implementation of a new class of dynamic demand simulators.

In this paper, we propose a general framework for the design and development of dynamic demand simulators. It is sufficiently general to encapsulate a wide variety of applications, models, data and algorithms. The paper not only describes the conceptual framework,

*Ecole Polytechnique Fédérale de Lausanne, Dpt. of Mathematics, CH-1015 Lausanne, Switzerland. Email: michel.bierlaire@epfl.ch

†The Ohio State University, Columbus, Oh. Email: mishalani.1@osu.edu

‡Massachusetts Institute of Technology, Dpt of Civil and Environmental Engineering, Cambridge, Ma. Email: mba@mit.edu

but also provides several practical considerations. The framework is illustrated using a dynamic demand simulator implemented within DynaMIT, a real-time dynamic traffic assignment system.

1 Introduction

With the emergence of Intelligent Transportation Systems (ITS), research has been conducted for several years on the design, development and evaluation of Dynamic Traffic Management Systems (DTMS). An important part of that effort focuses on traffic simulation which can be (i) macroscopic such as METANET (Messmer and Papageorgiou, 1990) and the cell-transmission model (Daganzo, 1994); (ii) mesoscopic such as in DynaMIT (Ben-Akiva et al., 1998), DYNASMART (Mahmassani et al., 1993) and INTEGRATION (van Aerde and Yagar, 1988); (iii) microscopic such as MITSIM (Yang and Koutsopoulos, 1997) and AIMSUN2 (Barceló and Ferrer, 1997). All of these traffic simulators effectively capture the performance of the network. This literature illustrates the strong emphasis placed on modeling network supply in the context of DTMS.

The research on demand aspects has also been investigated in the literature. However, very few demand-based integrated operational tools that can be used within DTMS have been developed. Such tools are necessary to capture both the demand levels and their distribution over time and space in conjunction with network performance and information provided to travelers.

As already recognized by Ben-Akiva et al. (1994) in the context of DTMS applications, demand models can roughly be grouped into two categories: aggregate (or statistical) models, and disaggregate (or behavioral) models. The development of integrated demand simulators requires an unified framework where these two categories of models can interrelate in a cooperative and complementary fashion. The first integrated real-time demand simulator has been proposed by Antoniou et al. (1997) and has been implemented within the DynaMIT system (Ben-Akiva et al., 1998). This simulator captures the effect of information provided to travelers on their departure time and route choices prior to the commencement of their trips using historical OD data and real-time link flow data.

In this paper, a conceptual framework for the development of a Dynamic Demand Simulator (DDS) is presented. This framework is based on a formal representation of demand referred to as the Disaggregate Demand Representation (DDR). The originality of the DDR is in its ability to combine different levels of aggregation in a consistent fashion. The methodology through which the DDR is used to combine different sources of demand information such as origin-destination matrices, socio-economic data and behavioral models is presented first. Moreover, a comprehensive mathematical description of demand processing is discussed together with the underlying assumptions. The general concept of a DDS, based on the DDR, is then introduced. Finally, the DynaMIT demand simulator is described as an illustration of the concepts.

2 The Disaggregate Demand Representation

A *Disaggregate Demand Representation* (DDR) $\mathcal{C}$ is characterized by a set of attributes $\{C_1, C_2, \ldots, C_n\}$ that are relevant in a transportation demand analysis context. These attributes include socio-economic characteristics (such as age, gender, level of income, address, access to real-time information, etc.) and trip characteristics (such as origin, destination, departure time, mode, path, average travel time, etc.) It may also include external characteristics describing the context when a trip occurs (such as weather condition, special events, holidays, etc.) This list is not exhaustive and may be extended to meet the requirements of any particular application. We assume, without loss of generality, that each attribute C_i may take only a finite and discrete number of distinct values $\{C_i(1), \ldots, C_i(s_i)\}$, called *states* of the attributes.

Definition 1 Given a set of n attributes $\{C_1, C_2, \ldots, C_n\}$, with a finite number of states $s_i, i = 1, \ldots, n$, a *Disaggregate Demand Representation* $\mathcal{C}$ is a table with n columns and $m = \prod_{i=1}^{n} s_i$ rows. Each row corresponds to a unique combination of the attributes states.

For example, we consider the set of attributes {Origin, Destination, Mode, Departure Time Interval}, and we assume that we have two origins O_1 and O_2, two destinations, D_1 and D_2, one mode M and two departure time intervals T_1 and T_2. The corresponding DDR is represented in Table 1, where $n = 4$, and $m = 2 \times 2 \times 1 \times 2 = 8$.

From a practical viewpoint, the general definition of the DDR may lead to intractable representations due to the combinatorial size of the DDR. Therefore, for practical purposes, and without loss of generality, several combi-

i	Origin	Destination	Mode	Departure Time
1	O_1	D_1	M	T_1
2	O_1	D_1	M	T_2
3	O_1	D_2	M	T_1
4	O_1	D_2	M	T_2
5	O_2	D_1	M	T_1
6	O_2	D_1	M	T_2
7	O_2	D_2	M	T_1
8	O_2	D_2	M	T_2

Table 1: Example of a Disaggregate Demand Representation

nations of states may be arbitrarily omitted from the representation, based on the specific characteristics of the application. For example, this typically occurs when the list of attributes contains Origin, Destination and Path. It does not make sense to include in the DDR a combination where a path does not link the associated origin and destination. Therefore, in that case, the topology of the network can be exploited to significantly reduce the size of the DDR.

The definition of a DDR is sufficiently general to capture a very wide range of applications related to transportation demand. For example, origin-destination matrices are a specific DDR, with two attributes. However, when several representations are used in a specific context, they must be compatible with one another.

Definition 2 If $\mathcal{C}_a$ and $\mathcal{C}_b$ are two DDR, we say that $\mathcal{C}_a$ is *compatible* with $\mathcal{C}_b$ and denoted by $\mathcal{C}_a \subseteq \mathcal{C}_b$, if the set of attributes in $\mathcal{C}_a$ is a subset of the set of attributes in $\mathcal{C}_b$. Moreover, the attributes common to both DDR must have the same states.

If $\mathcal{C}_a \subseteq \mathcal{C}_b$, we say that the Disaggregate Demand Representation $\mathcal{C}_a$ is *more aggregate* than $\mathcal{C}_b$. Note that $\mathcal{C}_a = \mathcal{C}_b$ if and only if $\mathcal{C}_a \subseteq \mathcal{C}_b$ and $\mathcal{C}_b \subseteq \mathcal{C}_a$. If $\mathcal{C}_a \subseteq \mathcal{C}_b$ and $\mathcal{C}_a \neq \mathcal{C}_b$, we note $\mathcal{C}_a \subset \mathcal{C}_b$.

We denote by $f_{\mathcal{C}}(i)$ the set of the attribute states corresponding to row i of a DDR $\mathcal{C}$. The subscript is dropped when no confusion is possible and we write $f(i)$. Referring to the example of Table 1, we have $f(6) = \{O_2, D_1, M, T_2\}$. f is a bijective relation between the set of indices $(1, \ldots, m)$

and the set of all combinations of attributes states. Therefore, it is meaningful to write $f^{-1}(\{O_2, D_1, M, T_2\}) = 6$.

Definition 3 The *characterization function* $f_{\mathcal{C}}(i)$ of a DDR $\mathcal{C}$ is a bijective function mapping the set of indices $(1, \ldots, m)$ into the set of all combinations of states.

The compatibility between two DDRs can be captured by a linear operator, referred to as the *compatibility matrix*, based on the characterization functions.

Definition 4 If $\mathcal{C}_a$ and $\mathcal{C}_b$ are two compatible DDRs such that $\mathcal{C}_a \subseteq \mathcal{C}_b$ with m_a and m_b rows, respectively, their *compatibility matrix* $P_b^a \in \mathbb{R}^{m_a \times m_b}$ is defined as

$$P_b^a(i,j) = \begin{cases} 1 & \text{if} f_{\mathcal{C}_a}(i) \subset f_{\mathcal{C}_b}(j) \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Note that each set of states $f_{\mathcal{C}_b}(j)$ contains exactly one set of states $f_{\mathcal{C}_a}(i)$. Therefore, each column of P_b^a contains exactly one nonzero entry. Hence, defining $J_i = \{j | P_b^a(i,j) = 1\}$ as the set of indices associated with nonzero entries in row i, then

$$J_i \cap J_k = \emptyset \quad \text{if } i \neq k, \tag{2}$$

and

$$\bigcup_{i=1}^{m_a} J_i = \{1, \ldots, m_b\}. \tag{3}$$

This concept is illustrated with the following example. Denote by $\mathcal{C}_b$ the DDR represented in Table 1 based on the set of attributes {Origin, Destination, Mode, Departure Time Interval}, and by $\mathcal{C}_a$ the DDR based on the set {Origin, Destination} represented in Table 2. We have $n_a = 2$, $m_a = 4$, $n_b = 4$ and $m_b = 8$.

The compatibility matrix P_b^a is given by

$$P_b^a = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \tag{4}$$

For example, the entry $P_b^a(2,3)$ is 1 because $\{O_1, D_2\} = f_{\mathcal{C}_a}(2) \subseteq f_{\mathcal{C}_b}(3) = \{O_1, D_2, M, T_1\}$.

i	Origin	Destination
1	O_1	D_1
2	O_1	D_2
3	O_2	D_1
4	O_2	D_2

Table 2: A more disagreggate DDR

After introducing the representation itself, an instance of a Dynamic Demand Representation is now defined.

Definition 5 An *instance* of a DDR $\mathcal{C}$ is characterized by a vector $\mathcal{C}(\alpha) \in \mathbb{R}^m$. The ith component of $\mathcal{C}(\alpha)$ represents the amount of demand, in a given unit, associated with the set of attributes $f(i)$.

The chosen unit depends on the application. Typical units are *number of travelers*, *number of vehicles* or *number of packets of vehicles*. In the example above, α_6 could be the number of travelers from origin O_2 to destination D_1 departing during time interval T_2 using mode M.

In summary, the level of aggregation of a DDR is determined by the number n of considered attributes. The more attributes, the more disaggregate the representation. The definition of a DDR is sufficiently general to capture a wide range of demand representations, from simple static origin-destination matrix, to a complete list of trip-makers with all their characteristics.

In the subsequent sections 3 through 6, the processes used to transform an instance of a given representation into a instance of another representation are described. These processes are the building blocks of Dynamic Demand Simulators.

3 The Aggregation process

Let $\mathcal{C}_b$ be a DDR based on the set of attributes $\{C_1, \ldots, C_{n_b}\}$ and $\mathcal{C}_a \subseteq \mathcal{C}_b$ a DDR based on $\{C_1, \ldots, C_{n_a}\}$, where $n_a \leq n_b$. The Aggregation process transforms an instance $\mathcal{C}_b(\beta)$ of $\mathcal{C}_b$ into an instance $\mathcal{C}_a(\alpha)$ of $\mathcal{C}_a$. Denoting by m_a and m_b the number of rows of $\mathcal{C}_a$ and $\mathcal{C}_b$, respectively, the following holds:

$$\begin{array}{rlcl} \text{Agg}: & \mathbb{R}^{m_b} & \longrightarrow & \mathbb{R}^{m_a} \\ & \mathcal{C}_b(\beta) & \rightsquigarrow & \mathcal{C}_a(\alpha) = P_b^a \mathcal{C}_b(\beta), \end{array} \tag{5}$$

where P_b^a is the compatibility matrix defined by (1).

Using the example described in Section 2, with P_b^a defined by (4), and considering

$$\mathcal{C}_b(\beta) = (\ 3.4 \quad 6.8 \quad 2.3 \quad 5.7 \quad 1.0 \quad 4.5 \quad 0.0 \quad 3.0\ )^T, \tag{6}$$

the aggregated instance is given by

$$\mathcal{C}_a(\alpha) = P_b^a \mathcal{C}_b(\beta) = (\ 10.2 \quad 8.0 \quad 5.5 \quad 3.0\ )^T. \tag{7}$$

4 The Disaggregation process

Let $\mathcal{C}_b$ be a DDR based on the set of attributes $\{C_1, \ldots, C_{n_b}\}$ and $\mathcal{C}_a \subseteq \mathcal{C}_b$ a DDR based on $\{C_1, \ldots, C_{n_a}\}$, where $n_a \leq n_b$. Consider also a matrix $Q_a^b \in \mathbb{R}^{m_b \times m_a}$ such that

$$P_b^a Q_a^b = I_{m_a}, \tag{8}$$

where P_b^a is the compatibility matrix defined by (1).

The Disaggregation process transforms an instance $\mathcal{C}_a(\alpha)$ of $\mathcal{C}_a$ into an instance $\mathcal{C}_b(\beta)$ of $\mathcal{C}_b$ as follows:

$$\begin{array}{rlcl} \text{Disagg}(Q_a^b): & \mathbb{R}^{m_a} & \longrightarrow & \mathbb{R}^{m_b} \\ & \mathcal{C}_a(\alpha) & \rightsquigarrow & \mathcal{C}_b(\beta) = Q_a^b \mathcal{C}_a(\alpha). \end{array} \tag{9}$$

Contrary to the aggregation process, where the matrix P_b^a is completely characterized by $\mathcal{C}_a$ and $\mathcal{C}_b$, the disaggregation process requires an externally defined matrix Q_a^b. Thus, there are many possible ways to disaggregate a DDR, and the matrix Q_a^b is necessary to identify one of them.

To illustrate the disaggregation process the same example is used with

$$\mathcal{C}_a(\alpha) = (\ 10.2 \quad 8.0 \quad 5.5 \quad 3.0\ )^T. \tag{10}$$

Using the matrix

$$Q_a^b = \begin{pmatrix} 0.5 & 0 & 0 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 0.5 \end{pmatrix} \tag{11}$$

to disaggregate $\mathcal{C}_a(\alpha)$ results in

$$\mathcal{C}_b(\beta') = \begin{pmatrix} 5.1 & 5.1 & 4 & 4 & 2.75 & 2.75 & 1.5 & 1.5 \end{pmatrix}^T. \tag{12}$$

4.1 Properties

Some interesting properties of the aggregation and the disaggregation processes are identified in this section. Consider $\mathcal{C}_b$ a DDR based on the set of attributes $\{C_1, \ldots, C_{n_b}\}$ and $\mathcal{C}_a \subseteq \mathcal{C}_b$ a DDR based on $\{C_1, \ldots, C_{n_a}\}$, where $n_a \leq n_b$. Consider also the matrix $P_b^a \in \mathbb{R}^{m_a \times m_b}$ defined by (1). Condition (8) immediately induces the following property.

Property 1 For any matrix $Q_a^b \in \mathbb{R}^{m_a \times m_b}$, the following holds

$$P_b^a Q_a^b \mathcal{C}_a(\alpha) = \mathcal{C}_a(\alpha) \quad \forall \alpha \in \mathbb{R}^{m_a}. \tag{13}$$

It guarantees that applying any disaggregation followed by an aggregation on an instance of a DDR does not modify that instance.

Note, however, that in general applying an aggregation followed by a disaggregation on an instance of a DDR does not result in that same instance. That is, $Q_a^b P_b^a \mathcal{C}_b(\beta) \neq \mathcal{C}_b(\beta)$. This is illustrated in the above example where the instance $\mathcal{C}_b(\beta')$ given by (12) is different from the instance $\mathcal{C}_b(\beta)$ given by (6) although $\mathcal{C}_a(\alpha)$ is an aggregation of $\mathcal{C}_b(\beta)$ as given by (7).

Property 2 For any matrix $Q_a^b \in \mathbb{R}^{m_b \times m_a}$ satisfying (8), the following holds:

$$\sum_{i=1}^{m_b} Q_a^b(i,j) = 1. \tag{14}$$

Proof. Let j be any index between 1 and m_a. From (8),

$$\sum_{i=1}^{m_b} P_b^a(j,i) Q_a^b(i,j) = 1 \quad \forall j = 1, \ldots, m_a. \tag{15}$$

By definition of J_j, $P_b^a(j,i) = 1$ if $i \in J_j$, and 0 otherwise. Therefore,

$$\sum_{i \in J_j} Q_a^b(i,j) = 1. \tag{16}$$

Let ℓ be any index between 1 and m_a, $\ell \neq j$. From (8),

$$\sum_{i=1}^{m_b} P_b^a(\ell,i) Q_a^b(i,j) = 0. \tag{17}$$

By definition of J_ℓ, $P_b^a(\ell,i) = 1$ if $i \in J_\ell$, and 0 otherwise. Therefore,

$$\sum_{i \in J_\ell} Q_a^b(i,j) = 0. \tag{18}$$

Finally, (3) allows for writing the following:

$$\sum_{i=1}^{m_b} Q_a^b(i,j) = \sum_{k=1}^{m_a} \sum_{i \in J_k} Q_a^b(i,j). \tag{19}$$

The result (14) follows directly from (16), (18) and (19). □

4.2 Specific disaggregation matrices

The general description of the disaggregation process is insufficient for an operational system. Some specific processes corresponding to realistic situations are therefore presented. As before, it is assumed that $\mathcal{C}_a \subseteq \mathcal{C}_b$ and an instance $\mathcal{C}_a(\alpha)$ is disagreggated into an instance $\mathcal{C}_b(\beta)$. Also, P_b^a is the compatibility matrix between $\mathcal{C}_a$ and $\mathcal{C}_b$.

Homogeneous disaggregation The first situation considered is when no external information is available to determine the disaggregation, and an arbitrary decision must be made. In this case, the disaggregation matrix

$$Q_a^b = (P_b^a)^T (P_b^a (P_b^a)^T)^{-1}, \tag{20}$$

the Moore-Penrose generalized inverse of P_b^a, is defined. This matrix reflects a homogeneous distribution of the total demand across the states of new attributes. The disaggregation matrix (11) used to produce $\mathcal{C}_b(\beta')$ of (12) is such a matrix.

Previous aggregation If a disaggregated instance $\mathcal{C}_b(\beta)$ is already available, it is desirable that an aggregation of this instance, followed by a disaggregation would lead back to the same instance. More formally, if $\mathcal{C}_b(\beta) \in \mathbb{R}^{m_b}$ is known, then $\mathcal{C}_a(\alpha) = P_b^a \mathcal{C}_b(\beta)$, and $Q_a^b \in \mathbb{R}^{m_b \times m_a}$ is given by

$$Q_a^b(i,j) = \begin{cases} \mathcal{C}_b(\beta)_j / \mathcal{C}_a(\alpha)_i & \text{if} f_{\mathcal{C}_a}(i) \subset f_{\mathcal{C}_b}(j) \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

so that

$$Q_a^b P_b^a \mathcal{C}_b(\beta) = \mathcal{C}_b(\beta). \tag{22}$$

Considering our example, we have

$$Q_a^b = \begin{pmatrix} 0.33 & 0 & 0 & 0 \\ 0.67 & 0 & 0 & 0 \\ 0 & 0.29 & 0 & 0 \\ 0 & 0.71 & 0 & 0 \\ 0 & 0 & 0.18 & 0 \\ 0 & 0 & 0.82 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{23}$$

In general, this matrix Q_a^b is determined when an aggregation is performed, and is used in a subsequent disaggregation. This is illustrated in section 7.

External data In order to maximize the quality of the disaggregated instance, it is desirable to make use of all available data, namely socio-economic information. Assume that the external data is available as an instance $\mathcal{C}_{\text{ref}}(\alpha_{\text{ref}})$ of a DDR $\mathcal{C}_{\text{ref}} \subseteq \mathcal{C}_b$ (so that both $\mathcal{C}_a$ and $\mathcal{C}_{\text{ref}}$ are

j	Origin	Destination	$\mathcal{C}_a(\alpha)_j$
1	O_1	D_1	100
2	O_1	D_2	50
3	O_2	D_1	250
4	O_2	D_2	20

Table 3: A static OD matrix

compatible with $\mathcal{C}_b$), and that all attributes in $\mathcal{C}_b$ are either in $\mathcal{C}_a$ or in $\mathcal{C}_{\text{ref}}$ (or both). In this case, the disaggregation matrix Q_a^b is defined such that

$$Q_a^b(i,j) = \begin{cases} g(i)/G(j) & \text{if } f_{\mathcal{C}_a}(j) \subset f_{\mathcal{C}_b}(i) \\ 0 & \text{otherwise,} \end{cases} \tag{24}$$

where $g = Q_{\text{ref}}^b \alpha_{\text{ref}}$, $G = P_b^a g$, and Q_{ref}^b is any appropriate disaggregation matrix.

In most practical applications, the choice of Q_{ref}^b is simply the homogeneous disaggregation, but any other matrix verifying (8) is valid. The vector g represents the disaggregation of the available data into the structure of the desirable DDR while the vector G represents the aggregation of this disaggregate instance to the DDR structure of the instance which needs to be disaggregated. The corresponding disaggregation matrix Q_a^b defined by the elements of g and G, therefore, allows for a disaggregation which is consistent with the information inherent to the available data.

As an example, a static OD matrix is disaggregated, knowing the gender breakdown by origin. The instance of $\mathcal{C}_a$ represented in Table 3 is disaggregated into an instance of the DDR $\mathcal{C}_b$ represented in Table 4, knowing the socio-economic data represented by the instance of $\mathcal{C}_{\text{ref}}$ represented in Table 5.

Choosing Q_{ref}^b as the homogeneous disaggragation matrix, the following are determined:

$$g = Q_{\text{ref}}^b \alpha_{\text{ref}} = \begin{pmatrix} 32.5 & 17.5 & 32.5 & 17.5 & 10 & 40 & 10 & 40 \end{pmatrix}^T, \tag{25}$$

$$G = P_b^a g = \begin{pmatrix} 50 & 50 & 50 & 50 \end{pmatrix}^T \tag{26}$$

	Origin	Destination	Gender
1	O_1	D_1	Male
2	O_1	D_1	Female
3	O_1	D_2	Male
4	O_1	D_2	Female
5	O_2	D_1	Male
6	O_2	D_1	Female
7	O_2	D_2	Male
8	O_2	D_2	Female

Table 4: A disaggregated static OD matrix

	Origin	Gender	$\mathcal{C}_{\text{ref}(\alpha)}$
1	O_1	Male	65
2	O_1	Female	35
3	O_2	Male	20
4	O_2	Female	80

Table 5: Socio-economic data

and

$$Q_a^b = \begin{pmatrix} 0.65 & 0 & 0 & 0 \\ 0.35 & 0 & 0 & 0 \\ 0 & 0.65 & 0 & 0 \\ 0 & 0.35 & 0 & 0 \\ 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0.8 & 0 \\ 0 & 0 & 0 & 0.2 \\ 0 & 0 & 0 & 0.8 \end{pmatrix}. \tag{27}$$

Therefore, the disaggregated instance of $\mathcal{C}_b$ is

	Origin	Destination	Gender	$\mathcal{C}_b(\beta)$
1	O_1	D_1	Male	65
2	O_1	D_1	Female	35
3	O_1	D_2	Male	32.5
4	O_1	D_2	Female	17.5
5	O_2	D_1	Male	50
6	O_2	D_1	Female	200
7	O_2	D_2	Male	4
8	O_2	D_2	Female	16

Probabilistic model The disaggregation using external data is designed to maintain some known proportions in the demand representation. In some cases, however, the demand must be represented by integer numbers and, therefore, cannot be split using that technique. The probabilistic model uses the same information, but creates a disaggregation matrix with only 0 or 1 entries. That guarantees that the integrality of the demand will be preserved by the disaggregation process.

Considering any disagreggation matrix Q_a^b, a new matrix $\hat{Q}_a^b$ can be built using the following procedure. For each column j, property 2 of section 4.1 is used and the entries of the column are considered to define a probability mass function (pmf). A Monte-Carlo simulation is performed based on this pmf to randomly identify one row i. Define

$$\hat{Q}_a^b(i,j) = 1 \tag{28}$$

and

$$\hat{Q}_a^b(k,j) = 0 \quad \forall k \neq i. \tag{29}$$

Column	Uniform[0,1)	Selected row
1	0.353	1
2	0.345	3
3	0.509	6
4	0.105	7

Table 6: Illustration of the probabilistic model

Note that this process is more expensive from a computational point of view. Therefore, it is preferable to use it only for off-line computation except when it is necessary to preserve integrality in a consistent way.

As an example, consider Q_a^b defined by (27). Each column contains two nonzero entries. A random number r is generated, based on a uniform distribution between 0 and 1. If r is smaller or equal to the first non zero element of a column, the corresponding entry in $\hat{Q}_a^b$ is 1. If it is large, the entry in $\hat{Q}_a^b$ corresponding to the other nonzero element is one. Table 6 illustrates this process. The first column contains the column index of Q_a^b, the second contains uniformly distributed random numbers, and the third mention the row in $\hat{Q}_a^b$ which is set to 1.

5 The splitting process

The combinatorial definition of a DDR may lead to very large instances. For practical reasons, it is desirable to split a DDR into smaller representations. The idea is to select one attribute and split the DDR creating a new DDR for each state of this attribute. The selected attribute will not appear anymore in the new DDRs.

The splitting process is defined by first assuming (without loss of generality) that the last attribute is selected. Let $\mathcal{C}$ be a DDR based on the set $\{C_1, \ldots, C_n\}$ of attributes. The splitting process creates s_n DDRs $(\mathcal{C}_1, \ldots, \mathcal{C}_{s_n})$, all based on the set $\{C_1, \ldots, C_{n-1}\}$ of attributes.

Given an instance $\mathcal{C}(\beta)$ of the original DDR, an instance of a new DDR $\mathcal{C}_k(\kappa)$ is such that $\mathcal{C}_k(\kappa)_j = \mathcal{C}(\beta)_i$ if

$$f_{\mathcal{C}}(i) = f_{\mathcal{C}_k}(j) \cup \{C_n(i)\} \tag{30}$$

Departure Time = T_1

i	$\mathcal{C}_1(\kappa)_i$	Orig.	Dest.	Mode
1	3.4	O_1	D_1	M
2	2.3	O_1	D_2	M
3	1.0	O_2	D_1	M
4	0.0	O_2	D_2	M

Departure Time = T_2

i	$\mathcal{C}_2(\kappa)_i$	Orig.	Dest.	Mode
1	6.8	O_1	D_1	M
2	5.7	O_1	D_2	M
3	4.5	O_2	D_1	M
4	3.0	O_2	D_2	M

Table 7: Split Disaggregate Demand Representation

Considering the DDR of table 1 and the instance (6)

$$\mathcal{C}_b(\beta) = (\ 3.4 \quad 6.8 \quad 2.3 \quad 5.7 \quad 1.0 \quad 4.5 \quad 0.0 \quad 3.0\ )^T, \tag{31}$$

two new DDRs and their corresponding instances are created, one for each departure time, as shown in Table 7.

6 Model application

A model application transforms an instance of a given DDR into another instance of the same DDR. It is important to distinguish between two types of models: aggregate and disaggregate.

A typical example of an aggregate model application is an OD estimation algorithm (see for instance Bell, 1983, Cascetta, 1984 or Ashok and Ben-Akiva, 1993). Such an algorithm updates an a priori OD matrix using external data such as link flow observations. If $\mathcal{C}$ is the DDR with m rows capturing the OD matrix, and $\mathcal{C}(\alpha)$ is the instance corresponding to the a priori matrix, the algorithm computes $\Delta \in \mathbb{R}^m$ such that the new instance is

$$\mathcal{C}(\alpha^*) = \mathcal{C}(\alpha) + \Delta. \tag{32}$$

Typical examples of disaggregate models are discrete choice models (see for instance Ben-Akiva and Bierlaire, 1999). These models consider one unit of demand (an individual, or a packet) and update some attribute states (such as the path choice). This produces shifts of demand across the rows of the DDR. If $\mathcal{C}$ is the considered DDR with m rows, and $\mathcal{C}(\alpha)$ is the instance corresponding to the initial demand, the disaggregate model transforms it into

$$\mathcal{C}(\alpha^*) = A\mathcal{C}(\alpha), \tag{33}$$

where $A \in \mathbb{R}^{m\times m}$ is such that A_{ij} is the proportion of demand moving from state combination $f(i)$ to state combination $f(j)$, and where f is the characterization function of the considered DDR.

For example, considering a DDR composed of Origin, Destination and Path, with $f(i) = \{O_1, D_1, P_1\}$ and $f(j) = \{O_1, D_1, P_2\}$, the entry A_{ij} is the proportion of travelers that used path P_1 between O_1 and D_1, and are now using path P_2. Note that the matrix A can be obtained either by considering the probability distribution provided by the model as proportions, or by performing a Monte-Carlo simulation in the exact same way as described for the probabilistic model discussed in Section 4.2. A compromise between accuracy and computation time has to be made here, but it does not affect the general framework.

As can be seen from the above discussion, model applications can be categorized into additive and multiplicative operations on DDR instances. More general operations could be integrated in the framework, but no operational requirement justifies complex generalizations.

7 Dynamic Demand Simulation

We now combine the concepts introduced before to define a Dynamic Demand Simulation.

Definition 6 Let $\mathcal{C}_1, \ldots, \mathcal{C}_N$ be N DDRs such that, either $\mathcal{C}_i \subseteq \mathcal{C}_{i+1}$, or $\mathcal{C}_{i+1} \subseteq \mathcal{C}_i$, for $i = 1, \ldots, N-1$. A *Dynamic Demand Simulation* is a function transforming an instance of $\mathcal{C}_1$ into an instance of $\mathcal{C}_N$.

Given an instance $\mathcal{C}_1(\alpha_1)$, the Dynamic Demand Simulation creates a sequence $(\mathcal{C}_2(\alpha_2), \ldots \mathcal{C}_N(\alpha_N))$ of instances of each DDR in the following way.

If $\mathcal{C}_i \subset \mathcal{C}_{i+1}$ a disaggregation process is applied, and

$$\mathcal{C}_{i+1}(\alpha_{i+1}) = Q_i^{i+1}\mathcal{C}_i(\alpha_i)$$

where Q_i^{i+1} satisfies (8).

If $\mathcal{C}_{i+1} \subset \mathcal{C}_i$, there are two possibilities

1. An aggregation process is applied, and

$$\mathcal{C}_{i+1}(\alpha_{i+1}) = P_i^{i+1}\mathcal{C}_i(\alpha_i)$$

where P_i^{i+1} is defined by (1); or

2. A splitting process is applied, as described is Section 5.

If $\mathcal{C}_i = \mathcal{C}_{i+1}$ a model is applied to transform $\mathcal{C}_i(\alpha_i)$ into $\mathcal{C}_{i+1}(\alpha_{i+1})$, as described in Section 6.

Therefore, to design an operational Dynamic Demand Simulation, the following steps must be performed:

1. Identification of relevant DDRs: In a typical application, the number of different DDR is small, but may be repeated several times in the sequence.

2. Selection of disaggregation matrices Q_i^{i+1} for all i such that $\mathcal{C}_i \subset \mathcal{C}_{i+1}$.

3. Choice between splitting and aggregating when $\mathcal{C}_{i+1} \subset \mathcal{C}_i$.

4. Selection of models transforming an instance of a DDR into another instance of the same DDR when $\mathcal{C}_{i+1} = \mathcal{C}_i$.

Section 8 is devoted to the description of the DDS proposed by Antoniou et al. (1997) and implemented into the DynaMIT system (Ben-Akiva et al., 1998).

8 DynaMIT implementation

DynaMIT (Dynamic Network Assignment for the Management of Information to Travelers) is a real time dynamic traffic assignment system that provides traffic predictions and travel information and guidance. DynaMIT estimates current traffic conditions using historical information and real-time data collected from a surveillance system. DynaMIT also generates prediction-based pre-trip information for departure time, path and mode choice, and en-route information for route choice.

DynaMIT combines both historical and real-time data to perform the best possible estimation and prediction. In order to achieve real-time efficiency, DynaMIT processes several data off-line, before the online operations start. The purpose of DynaMIT's DDS is to generate a list of travelers or group of travelers formed into packets each with a specific departure time and path choice reflecting the available pre-trip travel time information. The inputs to this process are historical OD flows, a habitual path choice model, a departure time and path choice model based on pre-trip traveler information, and real-time data on link flows.

The steps that make DynaMIT's demand simulator an operational Dynamic Demand Simulation, as described in Section 7, are presented here. For more details about DynaMIT's DDS (including models and algorithms), see Antoniou et al. (1997).

8.1 Identification of relevant DDRs

DynaMIT's DDS is based on four DDRs:

1. a historical database, noted $\mathcal{C}_D$,
2. a time-dependent OD matrix, noted $\mathcal{C}_{OD}$,
3. a list of unrouted packets, noted $\mathcal{C}_{UP}$, and
4. a list of packets, noted $\mathcal{C}_P$.

DynaMIT uses a historical database to exploit any relevant information collected from day to day about the considered environment. The database may contain an arbitrarily long list of information. In the context of DynaMIT's Dynamic Demand Simulation, four relevant attributes are considered, namely $\mathcal{C}_D = \{$Origin, Destination, Departure Time Interval, Day Category$\}$. Each attribute is described in what follows.

Origin Each state of that attribute corresponds to a specific centroid where trips may originate within the considered area.

Destination Each state of that attribute corresponds to a specific centroid where trips may end within the considered area.

Departure Time Interval Each state corresponds to a time interval within which a constant OD flow is realized.

Day Category Each state of that attribute corresponds to a specific category of day, such as rainy days, holidays, or special event days.

The time-dependent OD matrix DDR is $\mathcal{C}_{OD} = \{$Origin, Destination, Departure Time Interval$\}$. The DDR associated with the list of packets is the most disaggregate and is given by $\mathcal{C}_P = \{$Origin, Destination, Departure Time Interval, Mode, Path, Value of Time, Trip Purpose, Information Availability$\}$. Each of the attributes not described already are described in what follows:

Mode In DynaMIT, this attribute currently has two states: *private automobile* and *other*.

Path DynaMIT considers a restricted number of paths between each OD pair. These paths are selected and stored off-line, based on the network topology and on historical information.

Value of Time Three states are considered in the current version of DynaMIT: *high*, *medium* and *low*.

Trip purpose Three states are considered in the current version of the system: *work*, *leisure* and *other*.

Information availability Two states are considered in the current version of the system: *Equipped* and *Not equipped*.

Finally, the DDR $\mathcal{C}_{UP}$ associated with the list of unrouted packets is that of the list of packets without the *path* attribute.

The chain of mutually compatible DDRs, according to definition of a Dynamic Demand Simulator (see definition 6), can than be described as

$$\begin{array}{ccccccccccccccc} \mathcal{C}_1 & \supset & \mathcal{C}_2 & \subset & \mathcal{C}_3 & \subset & \mathcal{C}_4 & = & \mathcal{C}_5 & \supset & \mathcal{C}_6 & = & \mathcal{C}_7 & \subset & \mathcal{C}_8 \\ \Updownarrow & & \Updownarrow & & \Updownarrow & & \Updownarrow & & \Updownarrow & & \Updownarrow & & \Updownarrow & & \Updownarrow \\ \mathcal{C}_D & \supset & \mathcal{C}_{OD} & \subset & \mathcal{C}_{UP} & \subset & \mathcal{C}_P & = & \mathcal{C}_P & \supset & \mathcal{C}_{OD} & = & \mathcal{C}_{OD} & \subset & \mathcal{C}_P \end{array} \tag{34}$$

The overall process is transforming an instance of the historical data $\mathcal{C}_D(\gamma)$ into an instance $\mathcal{C}_P(\beta_E)$ corresponding to the best estimation of the current demand at the packet level. This process is illustrated in Figure 1 and is subsequently described step by step.

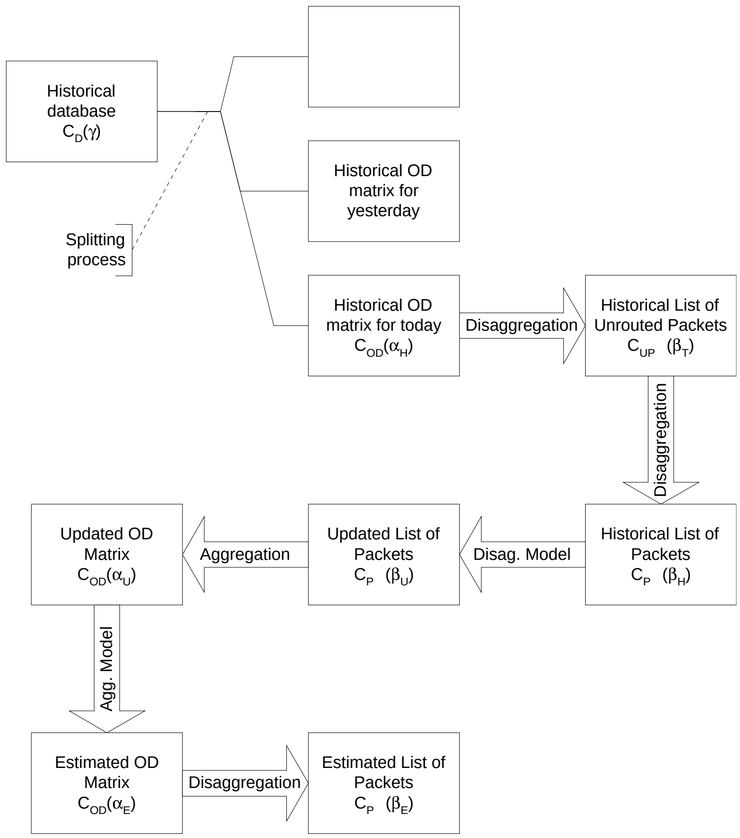

Figure 1: DynaMIT's Dynamic Demand Simulation

8.2 Splitting the historical database

A splitting process based on the attribute *Day Category* is applied to the historical database $\mathcal{C}_D$. Among the DDRs obtained from the process described in Section 5, the one corresponding to the day of interest is selected and an instance $\mathcal{C}_{OD}(\alpha_H)$ of the historical OD matrix is now available. We note that this splitting process may be performed off-line before the online system is in operation.

8.3 Disaggregation of the historical OD matrix

This disaggregation is based on external socio-economic data. As described in Section 4.2, the availability of an instance $\mathcal{C}_{\text{ref}}(\alpha_{\text{ref}})$ is assumed. For this disaggregation process to be valid, the DDR $\mathcal{C}_{\text{ref}}$ must contain at least the following attributes: {Mode, Value of Time, Trip Purpose, Information Availability}. Note that $\mathcal{C}_{\text{ref}}(\alpha_{\text{ref}})$ must reflect the composition of the population captured by the historical database. In some cases the database considers only travelers using private automobile and, consequently, the mode proportion is 1.0 for *private*.

The disaggregation process described in Section 4.2 is sufficiently flexible to exploit more detailed external data. For instance, if the composition of the population is known for each origin zone, the reference DDR to consider is then based on the attributes {Origin, Mode, Value of Time, Trip Purpose, Information Availability}. Clearly, destination-based data or even origin-destination based data can be exploited in a similar way, if available.

As discussed in Section 4.2, by default the disaggregation matrix Q^{UP}_{ref} corresponds to the homogeneous disaggregation. The disaggregation process, based on Q^{UP}_{OD} defined by (24), provides an instance $\mathcal{C}_{UP}(\beta_T)$.

8.4 Determination of habitual paths

The determination of habitual paths for each of the packets relies, again, on external data. In DynaMIT, a discrete choice model based on historical travel times is used (see Antoniou et al., 1997). The reference DDR $\mathcal{C}_{\text{ref}}$ is base on the attributes {Origin, Destination, Departure Time Interval, Path}. In this case, the disaggregation based on a probabilistic model as described in Section 4.2 is applied, so that each packet is associated with exactly one path. An instance $\mathcal{C}_P(\beta_H)$ is now available. As a direct result of using historical travel times, the path selected for each packet reflects a habitual choice. Subsequent steps will update this habitual choice to reflect the effect of information on predicted traffic conditions specific to the day of interest.

8.5 Behavioral models

Behavioral models capture the response of drivers to real-time information and guidance. The behavioral models within DynaMIT are discrete choice models capturing route choice, departure time choice and mode choice of each individual (or the group of individuals formed into packets) in the system. For details, see Antoniou et al. (1997) and Ben-Akiva and Bierlaire (1999). A matrix A captures the effects of these models on the DDR instance. As discussed in Section 6, A can be generated by interpreting the model probabilities as proportions, or by performing a Monte-Carlo simulation. The current version of DynaMIT is based on the latter. We compute

$$\mathcal{C}_P(\beta_U) = A\mathcal{C}_P(\beta_H) \tag{35}$$

to obtain an instance $\mathcal{C}_P(\beta_U)$.

8.6 Aggregation of the list of packets

As described in Section 3, the aggregation matrix P_P^{OD} is fully determined from $\mathcal{C}_P$ and $\mathcal{C}_{OD}$. Based on P_P^{OD} an instance $\mathcal{C}_{OD}(\alpha_U)$ is directly obtained through $\mathcal{C}_{OD}(\alpha_U) = P_P^{OD}\mathcal{C}_P(\beta_U)$.

8.7 OD matrix estimation

DynaMIT implements a Kalman Filter algorithm (Ashok and Ben-Akiva, 1993) where the state variables are perturbations from a reference OD matrix, in this case the Updated OD Matrix $\mathcal{C}_{OD}(\alpha_U)$. The direct output of this model is the Δ vector, introduced in Section 6, reflecting observed real-time data on link flows. Therefore, an instance $\mathcal{C}_{OD}(\alpha_E)$ is obtained through $\mathcal{C}_{OD}(\alpha_E) = \mathcal{C}_{OD}(\alpha_U) + \Delta$.

8.8 Final disaggregation

To obtain the estimated list of packets, a final disaggregation must be performed. As emphasized in Section 4, a disaggregation process is determined by a matrix Q^P_{OD}. In this case, the result of the previous aggregation that transforms $\mathcal{C}_P(\beta_U)$ into $\mathcal{C}_{OD}(\alpha_U)$ is used. The matrix Q^P_{OD} is, therefore, defined by (21) of Section 4.2 where $\mathcal{C}_a(\alpha) = \mathcal{C}_{OD}(\alpha_U)$ and $\mathcal{C}_b(\beta) = \mathcal{C}_P(\beta_U)$. The instance $\mathcal{C}_P(\beta_E)$ is finally obtained through $\mathcal{C}_P(\beta_E) = Q^P_{OD}\mathcal{C}_{OD}(\alpha_E)$.

9 Conclusion

A general framework for Dynamic Demand Simulation (DDS) is proposed. The framework is based on a formal representation of demand: the Disaggregate Demand Representation (DDR). Processes to aggregate and disaggregate that demand have been described in detail. Noting that there are infinitely many ways to disaggregate a DDR, several realistic options have been proposed. Finally, an actual DDS, DynaMIT's demand simulation, has been described in the context of the general framework.

The framework has been designed to capture most of the difficulties encountered during the design, development and implementation of DynaMIT's dynamic demand simulator. Therefore, this framework should allow for the development of more sophisticated dynamic demand simulators for evolving or new real-time applications.

References

Antoniou, C., Ben-Akiva, M., Bierlaire, M. and Mishalani, R. (1997). Demand simulation for dynamic traffic assignment, *Proceedings of the 8th IFAC Symposium on Transportation Systems*, Chania, Greece.

Ashok, K. and Ben-Akiva, M. (1993). Dynamic origin-destination matrix estimation and prediction for real-time traffic management systems, *in* C. Daganzo (ed.), *Transportation and Traffic Theory*, Elsevier Science Publishing Company Inc. Proceedings of the 12th ISTTT.

Barceló, J. and Ferrer, J. L. (1997). *AIMSUN2: Advanced Interactive Microscopic Simulator for Urban Network*, Departement d'Estadística i Investigació Operativa, Universitat Politèctina de Catalunya. User's Manual.

Bell, M. G. H. (1983). The estimation of an origin-destination matrix from traffic counts, *Transportation Science* **17**(2): 198–217.

Ben-Akiva, M. and Bierlaire, M. (1999). Discrete choice methods and their applications to short-term travel decisions, *in* R. Hall (ed.), *Handbook of Transportation Science*, Kluwer, pp. 5–34.

Ben-Akiva, M., Cascetta, E., Gunn, H., Smulders, S. and Whittaker, J. (1994). DYNA: A real-time monitoring and prediction system for inter-urban motorways, *Proceedings of the First World Congress on Applications of Transport Telematics and Intelligent Vehicle-Highway Systems*, Paris, France.

Ben-Akiva, M. E., Bierlaire, M., Koutsopoulos, H. N. and Mishalani, R. G. (1998). DynaMIT: a simulation-based system for traffic prediction, *Proceedings of the DACCORD Short-Term forecasting workshop*.

Cascetta, E. (1984). Estimation of trip matrices from traffic counts and survey data: a generalised least squares approach estimator, *Transportation Research Part B* **18**(4/5): 289–299.

Daganzo, C. (1994). The cell-transmission model. Part II: Network traffic, *PATH Working paper UCB-ITS-PWP-94-12*, University of California, Berkeley. Short version in Transp. Res., 29B(2), 79–94.

Mahmassani, H., Hu, T., Peeta, S. and Ziliaskopoulos, A. (1993). Development and testing of dynamic traffic assignment and simulation procedures for ATIS/ATMS applications, *Technical Report DTFH61-90-R-00074-FG*, Center for Transportation Research, University of Texas at Austin.

Messmer, A. and Papageorgiou, M. (1990). METANET: a macroscopic simulation program for motorway networks, *Traffic Engineering & Control* **31**: 466–470.

van Aerde, M. and Yagar, S. (1988). Dynamic integrated freeway/traffic signal networks: A routing-based modeling approach, *Transportation Research Part A* **22**(6).

Yang, Q. and Koutsopoulos, H. N. (1997). A microscopic traffic simulator for evaluation of dynamic traffic management systems, *Transportation Research C* **4**(3): 113–129.