Mapping Between Models for Pathway Dynamics and Structural Representations of Biological Pathways

Gokhan Yavas and Z. Meral Ozsoyoglu
EECS Department, Case Western Reserve University, Cleveland, Ohio
{gxy10, mxo2}@case.edu

ABSTRACT: Mathematical modeling and simulation of biochemical reaction networks gained a lot of attention recently since it can provide valuable insights into the interrelationships and interactions of genes, proteins and metabolites in a reaction network. A number of attempts have been made for modeling and storing biochemical reaction networks without their dynamical properties but unfortunately storing and efficiently querying of the dynamic (mathematical) models are not yet studied extensively. In this paper, we present a novel nested relational data schema to store a pathway with its dynamic properties. We then show how to make the mapping between this dynamic pathway schema with the corresponding static pathway representation.

1. INTRODUCTION

Systems biology is a re-emerging field of biological research that aims at a system level understanding of genetic or metabolic pathways by investigating interrelationships (organization or structure) and interactions (dynamics or behavior) of genes, proteins and metabolites. In systems biology, mathematical modeling and simulation of biochemical reaction networks is an important research area and received a lot of attention recently. The dynamics of large and complex biochemical reaction networks can be hard to understand by intuitive approaches alone. This emerges the need for mathematical methods for modeling and simulation. Modeling can suggest novel experiments for testing hypotheses, based on the modeling experiences.

Several attempts are made to model and store the biochemical reaction networks without their dynamic properties [2, 3, 4, 5]. However, the dynamic model of a metabolic pathway, shown in Figure 2, differs from its ***static*** representation, shown in Figure 1, and storing and efficiently querying of these dynamic models are not yet studied extensively to the best of our knowledge (by ***static***, we mean the pathway representation without its additional dynamical properties such as *kinetic law* which is used to represent the flux of each reaction in the pathway). Our goal here is to integrate the static and dynamic knowledge on pathways to give a more comprehensive view to the users, as well as leveraging the dynamic models by utilizing the static data about pathway components.

In this paper, we present a novel nested relational data schema to store a pathway with its dynamic properties. We then show how to make the mapping between this dynamic pathway schema with the corresponding static pathway representation. For this purpose, we use the foreign key constraints defined in the schemas and the inter-schema correspondences which are defined between the components of two schemas. Given a dynamic (mathematical) representation of a pathway, our main goal is to store it efficiently with our dynamic pathway schema and then to match the corresponding components in the static pathway database representing the pathway structure. This functionality can be very beneficial to systems biologists designing the mathematical models for pathways and biologists who want to see the relationship between these two different representations.

2. PATHWAY REPRESENTATION AND PATHWAY DYNAMICS

2.1 Pathway Representation

In general, a metabolic pathway is represented in the form of a graph structured data, where nodes represent molecular entities; edges represent reactions relating the molecular entities involved in the reaction; and the direction of the edge is from the substrate to the product of the reaction [3]. Since a reaction may have one or more substrates, and one or more products, the edge representing a process is actually a hyper-edge, and the graph representing a pathway is a hyper-graph. A given reaction may have a gene product (e.g. catalyzing enzyme) associated with the reaction, one or more cofactors, inhibitors and activators. The basic building blocks of pathway's ***static*** representation can be summarized [3] as: (i) ***molecular entities*** which are used to denote any molecular object (Thus we say that substrates, products, co-factors, inhibitors, and activators, are all molecular entities), (ii) ***processes*** which are used to denote reactions (where each process has molecular entities as its building blocks), and (iii) ***pathways*** which can be considered as interconnected arrangements of processes, where processes are viewed as building blocks of a pathway. As an example, consider the graphical representation of Purine metabolism in human, taken from [2], which is presented in Figure 1. In this pathway graph, all the processes and metabolites that belong to the generic purine metabolism pathway are shown.

2.2 Mathematical Models for Metabolic Pathways

Understanding the dynamic properties of biochemical reaction networks can provide us valuable insights into the working and

general principles of organization of biological systems. In order to analyze the dynamic properties of these networks, mathematical models are essential.

There are several different mathematical models such as Michaelis-Menten Rate laws, S-System models or Generalized Mass Action models (GMA) [7, 8, 9] used for analyzing the dynamics of biochemical pathways. In these models, the concentrations of metabolites are considered as variables, and the change of each variable's concentration with respect to time is written as a differential equation. The rate equations are written in different ways in different mathematical modeling techniques. The dynamic model of a metabolic pathway differs from the static representation in several aspects irrespective of which mathematical model is used. In order to express these differences, we will give some example reactions from the mathematical model of the purine metabolism in man presented in [1] and their counterparts in static purine pathway representation in man which is given in Figure 1. For this, we first give the graphical representation of the dynamic pathway model of purine metabolism which is studied in [1] in Figure 2.

In mathematical model given in Figure 2, each box represents a variable where a variable can represent a single metabolite or a pool of metabolites. The straight, heavy arrows between the boxes represent a reaction and its flux. Regulatory signals are represented by curved, light arrows; inhibitions are indicated by dashed lines and activations by solid lines.

We now summarize our observations from this model.

- The molecular entities Guanosine monophosphate (GMP), Guanosine diphosphate (GDP) and Guanosine triphosphate (GTP) are aggregated into one metabolite pool which represents all of these entities as one variable as shown in Figure 2. However, in the ***static*** pathway representation, (Figure 1), these are represented as different entities.

- As shown in the mathematical purine model (Figure 2) it is possible for several reactions, which forms a linear chain, to be lumped into single processes. In this model, the linear reaction chain between Phosphoribosylpyrophosphate (PRPP) and Inosine monophosphate (IMP) in purine metabolism is represented as a single reaction. There is more than one reaction from PRPP to IMP in the ***static*** pathway (see Figure 1) where the metabolite PRPP is the substrate of the first reaction, the metabolite IMP is the product of the last reaction and each reaction in the chain has the previous reaction's product as its substrate.

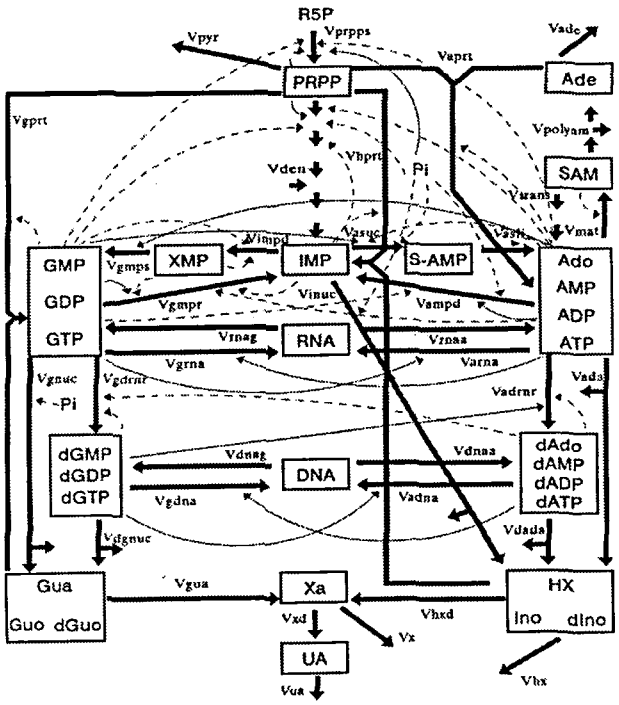

Figure 2: Mathematical model for Purine metabolism in man

- In a mathematical model of a pathway, all enzymatic reactions have a rate law which represents the flux of the reaction. Thus, the dynamic model of a pathway should store the kinetic law of each reaction which is not stored in the static pathway representation.

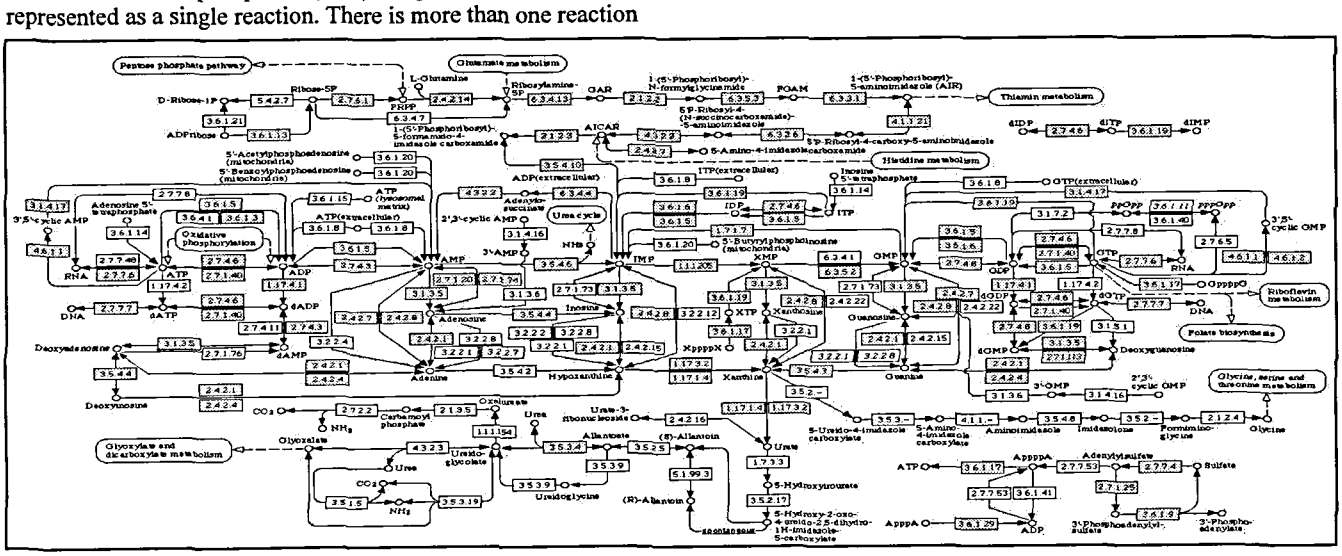

Figure 1: Purine metabolism pathway in human from KEGG DB

3. SCHEMA DEFINITION FOR DYNAMIC PATHWAY MODELS

In the light of the above observations, we now present a novel nested relational data schema to store a pathway with its dynamic properties. In section 2.1, we have seen that there are ***molecular entities*** which take part in the reactions as products or substrates or modifiers (i.e., catalyzing enzymes, activators, inhibitors, and cofactors). Note that the designer of the mathematical model can prefer to aggregate some of the ***molecular entities*** into a pool and this can be assumed to be an aggregated metabolite. For this reason, in our dynamic pathway schema, we will assume that each reactant, product, enzyme, cofactor, inhibitor, activator of a reaction are pools of metabolites. Each may contain only one metabolite in which case no metabolites are aggregated or it may contain more than one metabolite in which case these metabolites are aggregated and considered as one metabolite (or one variable of the mathematical model) in the corresponding dynamic pathway schema. Thus, the first component in our schema definition is ***Metabolite*** component which corresponds to a single ***molecular entity*** in the static pathway schema. The second basic component is the ***MetabolitePool*** which is basically a collection of the ***Metabolite*** components. The next component ***ReactionChain***, represents a set of reactions. As noted in section 2.2, the mathematical model may have a linearly chained set of reactions as one reaction in the dynamic pathway. This raises the need for a component which represents a set of reactions rather than a single process. Therefore, we define the component ***ReactionChain*** which represents a set of reactions which are linearly linked (i.e., the substrates of a reaction are the products of the previous reaction in the chain) in the corresponding static model. The final component which is in the highest level of the hierarchy is a ***Pathway*** component. It

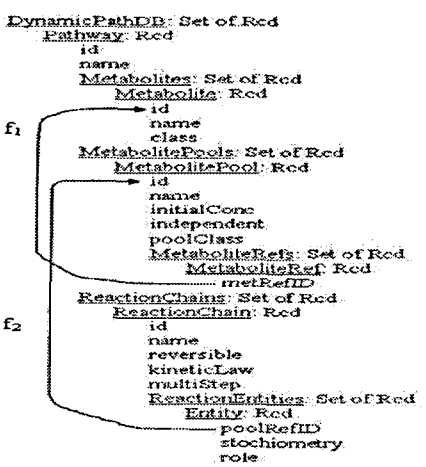

Figure 3: The Dynamic Pathway Database schema

includes the sub-component definitions. In our dynamic pathway schema, the basic building blocks of a ***Pathway*** component are ***ReactionChains*** and ***MetabolitePools*** instead of ***processes*** and ***molecular entities***, respectively. Moreover, each ***ReactionChain*** has a mathematical representation for its flux which is basically the rate of the reaction in terms of the concentration of its reactants and modifiers. In order to capture, the dynamic properties of this reaction, we need to represent the flux of each ***ReactionChain*** in the schema as a formula. This is accomplished by using the MathML 2.0 [10], which is a meta-language used for representing mathematical formulas in machine readable format.

We use a nested relational data model, which is also used in [6], to represent our dynamic pathway schema. The model includes a set of atomic types denoted by T, set types of the form SetOf[T] and record types of the form Rcd[A_1: T_1, ..., A_n: T_n], where each T_i represents either an atomic, set or record type. Each symbol A_i is called label or element. The dynamic pathway database schema is presented in Figure 3. In this figure, the arrows represent the foreign key constraints of the relations. For instance, the reference f_1 denotes that a ***MetabolitePool*** can only be composed of metabolites which are listed under the ***Metabolite*** relation.

3.1 Component Definitions

Pathway: A pathway structure is the highest level for our models. It contains all the other components inside it and represents a single metabolic pathway.

Metabolite: This component refers to the species that can take part in chemical reactions of the pathways as reactants, products, enzymes, activators, inhibitors or cofactors. These metabolites can be simple ions (such as calcium, sodium), simple molecules (such as glucose, ADP), large molecules (such as RNA, polysaccharides, and proteins), and others. In the definition of a metabolite, (see Figure 3), there is a *class* attribute which refers to the type of the molecular entity. It can have the values SIMPLE_MOLECULE, PROTEIN, GENE and RNA.

MetabolitePool: From our previous discussion, we know that in the mathematical model two or more species can be treated as a single variable. Thus, we generalize this strategy and represent the basic building blocks of a reaction as a set of metabolites. For this purpose, we have a component of ***MetabolitePool*** which consists of one or more metabolites.

This component has the *initialConc* attribute for the initial concentration of the metabolite pool. The *independent* attribute is of type Boolean indicating whether the variable corresponding to this metabolite is assumed to have a constant concentration during the simulation of the chemical reaction. The *poolClass* attribute defines the class of the aggregate metabolites. The class of each metabolite that is aggregated to form a ***MetabolitePool*** should be same so that they can be aggregated. It means that metabolites from different classes can not be aggregated. Thus, the *poolClass* for a metabolite pool will be same as the class of the metabolites which form this pool. While defining the list of metabolites that participate in this metabolite pool, we only make a reference to the previously defined ***Metabolite*** component. This is accomplished by the ***MetaboliteRef*** structure.

ReactionChain: In section 2.2, we have discussed that several linear chains of reactions can be represented as a single reaction in the mathematical model. Thus, we have to have a component which represents a linear chain of reactions of the static model. This component is called ***ReactionChain***. Since a linear chain

of reactions is treated as a single reaction in the mathematical model, only one kinetic law definition is allowed to define the flux for the set of aggregated reactions.

The ***ReactionChain*** component has a *kineticLaw* attribute which is an expression that represents the mathematical formula of flux of the aggregated reactions. This attribute is represented using MathML 2.0 [10], the XML standard for describing mathematics in machine-readable format. The attribute *reversible* defines whether the aggregated reaction list is reversible. Since ***ReactionChain*** represents a set of reactions, we define another attribute *multiStep* representing whether this reaction chain consists of more than one reaction or it is just a single step. If the value of this attribute is *true*, this chain has more than one reaction steps, else it is composed of a single reaction.

Furthermore, the *ReactionEntities* label defines a list of entities, each of which can be a reactant, product, modifier or an enzyme of the current reaction chain. Every ***MetabolitePool*** that is the substrate of the first reaction in the ***ReactionChain*** or the product of the last reaction in the chain should be listed in the *ReactionEntities* list. *ReactionEntities* is a list of ***Entity*** components. In Figure 3, one can see the attributes of ***Entity*** component. The *poolRefId* makes a reference to a previously defined ***MetabolitePool***. Thus, *poolRefId* is a foreign key referencing the *id* attribute of ***MetabolitePool*** component. This reference guarantees that the entity used in this reaction is previously defined as a ***MetabolitePool*** in the pathway. The role of this entity in this reaction is defined by the attribute *role*. The role of an entity in a reaction can be: *Reactant, Product, Activator, Inhibitor, Regulator, Enzyme or Unkown*. For a ***MetabolitePool*** to have a role as an *Enzyme* in a reaction, it should be of class *Protein*. The attribute *stochiometry* defines the stochiometry of the referenced *MetabolitePool* in the current reaction.

Moreover, the list of entities for a reaction chain defines two different groups of ***ReactionChain*** components. If any of the reaction entities defined under the *ReactionEntities* label has a poolRefId value which is set to UNSPECIFIED, or the reaction chain is multi-step then we say that this is a *Partially-specified* reaction chain. Otherwise, we call it a *Fully-specified* reaction chain. The multi-step reaction chains are *Partially-specified* by their nature because only the substrates of their first step and products of their last step are specified in the model. The entities for the intermediate reaction steps are not explicitly specified. As an example, consider the single-step reaction chain represented by V_{aprt} in Figure 2. This single-step reaction chain has the substrate PRPP and Ade and the aggregation of Ado, AMP, ADP and ATP as its single product and the product of this reaction also functions as an inhibitor. Since all of the reaction entities of this reaction chain are specified in the model, it is a *Fully-specified* ***ReactionChain***. On the other hand, a single-step reaction can also be *Partially-specified*. For instance, the reaction chain V_{adna} which has IMP as its substrate and aggregation of HX, Ino and dIno as its product is a *Partially-specified* single-step reaction chain since there is one more product yielded by the reaction but not specified in the model. It is just shown as sub-arrow underneath the main arrow, V_{adna}.

4. MAPPING COMPONENTS OF THE DYNAMIC SCHEMA IN THE STATIC SCHEMA

This section explains how one can match the components of our dynamic pathway schema with a given schema of a static pathway representation. In [3], an integrated system, called *Pathways Database System* is presented, with a set of software tools for modeling, storing, analyzing, visualizing, and querying biological pathways data at different levels of detail. These two schemas define two different data sources but each of these data sources represents the different aspects of the same pathway data. However, there should be some sort of integration between these two data sources since a user of the database may want to see the components of a dynamic pathway in the corresponding static pathway diagram.

In the following subsections, we define how we match each basic building block of a dynamic pathways database in the corresponding static pathways database of [3]. The schema of the database of [3] and our schema are presented in Figure 4 We suppose that our database schema is the source schema and the static pathways schema is the target. For mapping, we use simple element correspondences defined between the components of two database schemas. These correspondences can be defined by experts of the domain or by automated techniques for schema matching [11]. For our application, we define the inter-schema correspondences m_1 and m_2 between the names of the corresponding components.

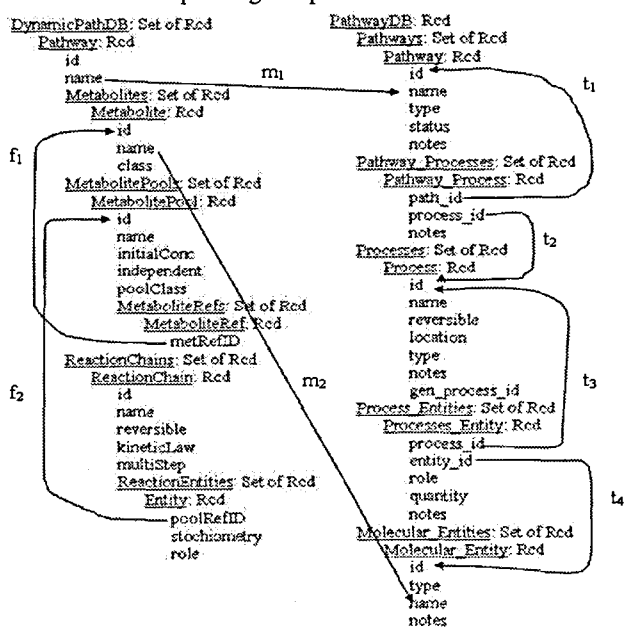

Figure 4: Source and target schemas and the correspondences

For instance, the correspondence m_2 that is defined between the *name* entity of ***Metabolite*** component in the dynamic schema and the name entity of ***Molecular_Entity*** component in the pathways schema means that for each ***Metabolite*** in the dynamic pathway database, there must be a ***Molecular_Entity*** with the same name in the pathways database.

4.1 Matching *Metabolite* Component

Given a ***Metabolite***, (say *M*), in a dynamic pathway, (suppose that pathway has an id of *p*), our goal is to find the corresponding ***Molecular Entity*** in the target schema. From the ***Metabolite*** table, it is possible to find the "name" attribute of the metabolite *M*. With the help of the correspondence, m_1, between the target (static pathway) schema and the source (dynamic pathway) schema, we find the static pathway with id *p'* that has the same name with the dynamic pathway with id *p*. Next, we join the tables ***Pathway_Process***, ***Process_Entities*** and the ***Molecular_Entity***. In the relation resulting from the join, we select the tuples with "Pathway_Process.*path_id= p'*" and then project the remaining tuples on the attributes (*Molecular_Entity.id, Molecular_Entity. name*). Now, by using the inter-schema correspondence m_2, we select the tuple in the final relation which satisfies "Molecular_Entity.*name* = *M*.name". This tuple represents the matching ***Molecular Entity*** in the target schema.

4.2 Matching *MetabolitePool* Component

Suppose that we are given a ***MetabolitePool*** , (say *MP*), in a dynamic pathway, (with id *p*), and our aim is finding the corresponding ***Molecular_Entity*** tuples in the target schema.

In the dynamic pathways database, we first find the list of ***Metabolite*** references in the ***MetabolitePool*** *MP*. Next, we follow these links (i.e., foreign key constraint f_1) and we obtain the names of all the metabolites in the pool *MP*, call the set of these names as *T*. With the help of the correspondence, m_1, we find the static pathway with id *p'* that has the same name with the dynamic pathway with id *p* (here we use the mapping m_1). In the target schema, we join the tables ***Pathway_Process***, ***Process_Entities*** and the ***Molecular_Entity***. Then in the obtained relation, we select the tuples with "Pathway_Process.*path_id= p'* ". After that, we project the remaining tuples on the attributes (*Molecular_Entity.id, Molecular_Entity. name*). Now, we select the tuples from the final relation which has a *Molecular_Entity.name* that exist in the set *T* (here we use the mapping m_2). These tuples will be the set of ***Molecular_Entities*** which correspond to the given ***MetabolitePool***.

4.3 Matching *ReactionChain* Component

When matching a ***ReactionChain*** component (say *RC*) in the static pathway database, there can exist three possible cases:

(i) ***RC* can be exactly matched with a set of reactions in the static pathway database**:

This case can be valid for just fully-specified one-step reaction chains (In a case where *RC* is a multi-step reaction chain we can not exactly determine the steps of the *RC* but only the possible ones. Because there can be many different paths from the substrates of *RC* to its products). For instance consider the fully-specified single-step reaction chain (represented by V_{impd} in Figure 2) between the metabolite pool IMP and the metabolite pool XMP. In [1], it is indicated that this reaction is a single step reaction and catalyzed by enzyme with the EC number 1.1.1.205. It is possible to match this reaction with an exact match to the reaction (Figure 1) with substrate IMP, product XMP and the catalyzing enzyme having an EC number of 1.1.1.205.

(ii) ***RC* can be approximately matched with a set of reactions in the static pathway database**:

There are several sub-cases in which we can observe approximate matchings. First, there are some fully-specified single step reaction chains in Figure 2 which have two or more substrates (or products) but the corresponding reaction in Figure 1 has less products or substrates. This indicates an approximate matching. For instance consider the fully-specified single-step reaction chain, V_{gprt}, having substrates PRPP metabolite pool and the metabolite pool formed by the aggregation of Gua, Guo and dGuo and the product metabolite pool formed by the aggregation of GMP, GDP and GTP. This reaction is catalyzed by enzyme 2.4.2.8 [1]. The reaction that best matches to this single-step ***ReactionChain*** in Figure 1 is the reaction between Guanine and GMP with the enzyme 2.4.2.8. As you can see that the matched reaction has one less substrate, PRPP. Thus we say that there is an approximate match.

Second, it can be the case that the substrates and the products exactly match but the catalyzing enzymes do not. As an example consider the fully-specified single-step reaction chain (represented by V_{gmps} in Figure 2) which has substrates XMP metabolite pool and the product metabolite pool formed by the aggregation of GMP, GDP and GTP. The reaction is catalyzed by enzyme 6.3.4.1 [1]. In the static representation (see Figure 1), the reaction that best matches this single-step reaction chain is between XMP and GMP with the catalyzing enzyme 6.3.5.2. Since the substrates and products are matched but the catalyzing enzyme does not, we say that this is an approximate matching.

Third, we observe the approximate matchings in the case of the multi-step ***ReactionChain*** components. Consider the multi-step reaction chain represented by V_{den} in Figure 2. This reaction chain has the substrate PRPP and product IMP. Since the reaction is multi-step and the first reaction in the chain has substrate PRPP, the second reaction should have the products of the first reaction as substrates and so on. The last reaction has the IMP as substrate. For matching, we have to search the Figure 1 for all the paths starting from the substrates (PRPP in this case) and ending with products (IMP in this case). In this case, the best possible matching path is shown as a solid line in Figure 5. There is only one possible path in this case but it is also possible that there may be more than one possible path from substrates to the products in other pathways data. Therefore, we can not say that we can find an exact matching path for a multi-step reaction chain. We can only output the possible paths matching the given ***ReactionChain***.

Moreover, approximate matchings are possible in case of partially-specified single-step reactions. For example, consider the reaction chain V_{ada} having the metabolite pool formed by aggregation of Ado, AMP, ADP and ATP as its substrate. It yields the metabolite pool which is formed by the aggregation of HX, Ino and dIno. It is catalyzed by enzyme 3.5.4.4 [1]. But there is one more product of the reaction which is not defined as a metabolite pool in the model (see the sub-arrow on V_{ada} in Figure 2). This non-specification of a product can happen due to

the irrelevance of this metabolite to the other parts of the model. The reaction that best matches this single-step reaction chain is between Adenosine and Inosine with the catalyzing enzyme 3.5.4.4. This is assumed to be an approximate match because we do not know the non-specified product and can not match it.

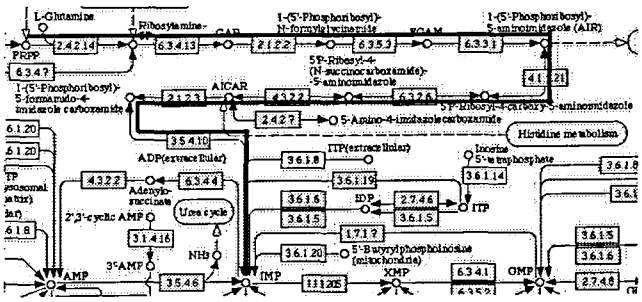

Figure 5: The best possible matching path for V_{den}

There is one more case left for approximate matching. It is possible that, any of the substrate, product or modifier of the current reaction chain is not defined as a part of the pathway which is under consideration. As an example, consider the fully-specified single step reaction chain, V_{mat}, in Figure 2. This reaction chain has ATP as its substrate and S-Adenosyl-L-methionine (SAM) as its product. It is catalyzed by the enzyme with EC number 2.5.1.6. However, SAM is not defined as a process entity in the corresponding static representation (see Figure 1) of the purine metabolism. In the Kegg database [2], this metabolite is defined in Methionine metabolism, Arginine and proline metabolism and Propanoate metabolism. The best matching reaction to the given reaction chain is found under Methionine metabolism pathway. It has L-methionine as its substrate and SAM as its product. Additionally, this reaction is catalyzed by the enzyme with the EC number 2.5.1.6. Thus, it is possible to make such an approximate match in which the substrates, products or both are found in different pathways.

(iii) **RC can not be matched with any reactions in the static pathway database:**

It is also possible that the given ***ReactionChain*** can not be matched to any reaction or reactions in the static representation. This can happen when we do not have any information about either any of the products or any of the substrates of a reaction chain. For instance, consider the partially-specified single-step ReactionChain, V_{ade}. It has Ade as its substrate, and is catalyzed by enzyme 1.2.1.37 but the product of that reaction is not specified in [1]. Since we do not know the product of the reaction, it is not possible to match it with any reactions in the static representation.

With these cases in hand, our ***ReactionChain*** matching algorithm uses a Depth First Search (DFS) approach which is appropriately modified for handling the hyper-graph data and taking care of the different matching criteria which are summarized in the cases (i), (ii) and (iii) (i.e., exact matching, approximate matching, no matching). To match a reaction chain, our idea is first finding the corresponding process entities to the metabolite pools as explained in Section 4.2. Next, we find all the paths from the substrates to the products in a DFS manner considering the three cases summarized in previous paragraphs.

5. CONCLUSION

Recently, mathematical modeling and simulation of biochemical reaction networks gained much importance since they can provide us clues to understand the general working and organization principles of the metabolic pathways. Due to this importance, the storage, information exchange and efficient querying of these mathematical models become a topic that needs investigation.

In this paper, we present a novel nested relational data schema to store a pathway with its dynamic properties. We then propose methods to match the corresponding components of a dynamic model in the static pathway schema. For this purpose, we use the foreign key constraints defined in the schemas and the inter-schema correspondences which are defined between the components of two schemas.

It is evident that the functionality of matching dynamic schema components in the static representation can provide invaluable insights for the systems biologists when updating mathematical models for pathway dynamics to reflect the new knowledge on the structure and the components of pathways structure into the existing dynamic models as the current knowledge on pathways evolve.

REFERENCES

[1] Curto, R., E.O. Voit, A. Sorribas, and M.Cascante: "Mathematical models of purine metabolism in man". Math. Biosci. 151, 1-49, 1998

[2] Kyoto Encyclopedia of Genes and Genomes, available online at http://www.genome.jp/dbgetbin/get_pathway?org_name=hsa&mapno=00230

[3] Krishnamurthy, L., J. Nadeau, G. Ozsoyoglu, Z.M. Ozsoyoglu, G, Schaeffer, M. Tasan, and W. Xu, "Pathways Database System: An integrated set of tools for biological pathways", ACM Symposium on Applied Computing, Melbourne, FL, March, 2003

[4] Enzymes and Metabolic Pathways database, available online at http://emp.mcs.anl.gov

[5] SoyBean metabolic pathways source, available online at http://cgsc.biology.yale.edu/metab.html

[6] L. Popa, Y. Velegrakis, M. Hernandez, R. J. Miller, and R. Fagin, Translating Web Data", in 28th International Conference for Very Large Databases (VLDB 2002), August 2002

[7] Savageau, M.A.: "Biochemical systems analysis, I. Some mathematical properties of the rate law for component enzymatic reactions". J. Theor. Biol. 25, 365-9, 1969

[8] Savageau, M.A.: "Biochemical systems analysis, II. The steady-state solutions for an n-pool system using a power-law approximation". J. Theor. Biol. 25, 370-9, 1969

[9] Savageau, M.A.: "Biochemical systems analysis, III. Dynamic solutions using a power law approximation". J. Theor. Biol. 26, 215-26, 1970

[10] W3C's math home page, available via online at http://www.w3.org/Math/

[11] E. Rahm, P.A. Bernstein. "On Matching Schemas Automatically". VLDB Journal, 10(4), Dec. 2001