Managing large-scale scientific hypotheses as uncertain data with support for predictive analytics

Bernardo Gonçalves
LNCC – National Laboratory for Scientific Computing
Petrópolis, Brazil
bgonc@lncc.br

Fabio Porto
LNCC – National Laboratory for Scientific Computing
Petrópolis, Brazil
fporto@lncc.br

Abstract—In the era of data-intensive science and big data, much of the scientific thinking has been shifting to the data analysis phase of the research life cycle. To meet this paradigm shift, the vision of Τ-DB abstracts deterministic scientific hypotheses as a kind of uncertain data. It comprises a probabilistic database design methodology for the systematic construction and management of U-relational hypothesis databases, viz., Τ-DBs. The methodology of Τ-DB allows scientists and engineers to manage and evaluate (rate/rank) scientific hypotheses ‘as uncertain data.’ In this paper we show the applicability of Τ-DB in a real-world scenario. We present use cases in Computational Hemodynamics derived from the Physiome project.

I. INTRODUCTION

As part of the paradigm shift that makes science ever more data-driven, the so-called fourth paradigm [1], high-throughput technology and large-scale experiments have been providing scientists with lots of empirical data that has to be extracted, transformed and loaded (the so-called ETL pipeline) before it is ready for analysis. Science’s ETL has been distinguished by its unfrequent, incremental-only updates and by having large raw files as data sources [2]. Data management challenges for enabling a direct, efficient access to such raw data have been documented and pointed as key to some important use cases of exploratory data analysis ‘in situ’ in the raw data [3].

Simulation (raw) data, nonetheless, being generated from first principles or deterministic hypotheses, is a kind of theoretical data that poses additional data management challenges as pointed out recently [4], [5]. Considering the five v’s associated to the notion of big data, scientific data in general can be said to qualify to value, because of its role in advancing science and technology; and volume, because of the large scale of modern scientific problems. While empirical data may be more related to velocity (e.g., in streaming applications), theoretical data is strongly associated with the two v’s of veracity, due to its uncertainty; and variety, due to its structural heterogeneity. Both have to do with the user alternative (external) views of the world. This paper shows why coping with challenges of such theoretical data can be in turn key for enabling hypothesis management and predictive data analysis. This is for the sake of decision making at industrial scale directly within a probabilistic database (p-DB) [6].

Hypothesis management. Big Computational Science research programs such as the Human Brain Project or Cardiovascular Mathematics are highly-demanding applications challenged by such theoretical big data. Users need to analyze results of thousands of data-intensive simulation trials, each providing a possible alternative to explain the studied phenomenon. Direct access to data as in exploratory analytics is not enough here, since the trial datasets must be analyzed altogether in a controlled and unified way. We introduce then into data an epistemological dimension under which all possible, mutually inconsistent theoretical alternatives are keyed according to classical Functional Dependency (FD) theory [7].

Predictive analytics. Moreover, the last decades have seen the emergence of model repositories in computational sciences as an attempt to deal with reproducibility and large-scale model integration and sharing [8], [9], [10], [11]. They are growing relatively fast, but with limited integrity since they lack data management support for the evaluation (rating/ranking) of competing models. We introduce then the concept of data-intensive evaluation studies. These are meant to allow models that have structural- and data-level overlapping to be set into a common reference frame for analysis according to Bayesian statistics [12] by means of U-relational p-DB technology [13].

Our approach to hypothesis management and predictive analytics unfold from the Τ-DB vision [5]. This is, essentially, an abstraction of hypotheses as uncertain data. It comprises a p-DB design methodology for the systematic construction and management of U-relational hypothesis databases, viz., Τ-DBs. U-relations and probabilistic world-set algebra compose the data model of MayBMS [13], a state-of-the-art p-DB management system [2] built on top of MayBMS, the methodology of Τ-DB allows scientists and engineers to manage and evaluate (rate/rank) scientific hypotheses as theoretical data.

In this paper we show the applicability of Τ-DB in a real-world scenario. For details on the Τ-DB methodology from a database research perspective, we refer the reader to [5]. In [11] as the background of this paper, we present the vision of Τ-DB in brief, and introduce the Physiome research program as providing a testbed for Τ-DB in such a real scenario. Then we show in [11] the construction and management of Τ-DB as applied to Physiome use cases in Computational Hemodynamics. In [11] we present initial experiments on Τ-DB. In [11] we discuss related work and, finally, in [11] we conclude the paper.

MayBMS is as a backend extension of PostgreSQL. It offers all of the traditional query capabilities in addition to the uncertain and probabilistic.
A. The Vision of Υ-DB in a Nutshell

Scientific hypotheses are tentative, testable explanations of phenomena [13]. In the fourth paradigm, hypotheses are [3]: (i) formed as principles or ideas, (ii) then mathematically expressed and (iii) implemented ‘in silico’ as a program that is run to give (iv) their decisive form of data (see Fig. 1).

By observing that, in Υ-DB, hypotheses (whether single laws or complex mathematical models) are flattened into a data perspective. Note that the semantic structure of item iv (Fig. 1) can be expressed by FD \( t \rightarrow v s \). This is typical semantics assigned to scientific data for exploratory data analysis. A “physical” dimension (like time \( t \)) is used as a key to observables (like velocity \( v \) and position \( s \)). For hypothesis management, however, we need two special attributes to compose the epistemological dimension of keys to observables: \( \phi \), identifying the studied phenomena; and \( v \), identifying the hypotheses aimed at explaining them. That is, we shall leverage the semantics of item (iv) to \( \phi \rightarrow v \rightarrow s \). This leap is a core abstraction in the Υ-DB vision [5]. We refer to Example 1 for a first illustration of the Υ-DB methodology.

Example 1: A research is conducted on the effects of gravity on a falling object in the Earth’s atmosphere. Scientists are uncertain about the precise object’s density and its predominant state as a fluid or a solid. Three hypotheses are then considered as alternative explanations of the fall (Figs. 2-3).

The methodology of Υ-DB consists in a pipeline (Fig. 4). In its first step, hypotheses are encoded into certain (i.e., classical) H-relations by processing their mathematical structure as provided in a MathML-based file (Fig. 2 to ii). The system then extracts the FDs hidden in the model equations. For example, equation \( v(t) = -g t + v_0 \) (Fig. 1 item ii) allows us to infer that \( v \) is a prediction variable functionally dependent on \( t \) (the physical dimension), \( g \) and \( v_0 \) (the parameters). The key point here is that Υ-DB’s design methodology comes with a technique for hypothesis encoding based on its mathematical equations [5]. Once a hypothesis has been defined and at least one simulation trial dataset (raw data, Fig. 4 item i) has been loaded into the system, the user is able to query it as usual in a traditional relational DB (Fig. 4 item ii). Phenomena, in turn, are encoded by the incremental addition of empirical (observational or experimental) datasets at anytime.

Moreover, simulation trial datasets (hypotheses) can be analyzed against empirical data (phenomena). Based on the structure of both hypotheses and phenomena, the system supports the user in the setup of data-intensive evaluation studies (Fig. 4 item iii). Given a phenomenon of interest, the user shall choose and rate/rank a selection of competing hypotheses whose predictive power is detected to be fit for the phenomenon. Once the user assigns a prior probability distribution (uniform by default) to the competing hypotheses, the system analyzes them against empirical data (phenomena). Based on the structure of both hypotheses and phenomena, the system supports the user in the setup of data-intensive evaluation studies (Fig. 4 item iii). These are encoded as uncertain relations. The key point here is that Υ-DB’s design methodology comes with algorithmic techniques for uncertainty introduction [5]. Bayesian inference can then be applied to get a posterior distribution, possibly re-ranking the hypotheses in face of the evidence available for the phenomenon.

We shall walk diligently through the Υ-DB methodology in [13]. Before we proceed, query Q1 illustrates how the data transformation step ii to iii is performed in the Υ-DB system in terms of MayBMS’ p-DB query language, see Fig. 5. Y-relations like \( Y[Exp] \) are uncertain and then have in their schema a set of pairs \( (V_i, D_i) \) of condition columns to map each discrete random variable \( x_i \), to one of its possible values (e.g., \( x_1 \rightarrow 1 \) [5]. World table \( W \) stores their marginal
Fig. 5: Explanation table seen under possible-world semantics.

### B. The Physiome Project as a Testbed for Y-DB

The Physiome project is an initiative to seriously address the problems of reproducibility, model integration and sharing in Computational Physiology [10], [11]. It essentially comprises:

- a curated repository of 370+ computational physiology models available online for researchers;
- the Mathematical Modeling Language (MML) to allow models to be written in declarative form and then exported into a number of interoperable XML formats;
- a problem-solving environment called JSim to allow researchers to code their MML models straightforwardly, run them under different parameter and solver settings and build customized data plots as result.

From the point of view of Y-DB, Physiome provides a very interesting testbed replete with real scenarios. As an external data source for Y-DB, we extract Physiome models into Y-DB by means of a wrapper that can read XMMML files (JSim’s XML encoding of MML models). Simulation trial datasets are exported from JSim as well. Currently, Y-DB’s Physiome wrapper can read ‘.par’ text files in order to load parameter setting (model input) as well as MAT or CSV files to load computed predictions (model output) for each simulation trial.

Physiome does not keep records of phenomena in a repository, but it does have empirical (observational or experimental) data attached to some of the entries of the model repository. Such models appear in the filter ‘models with data,’ meaning that they have one or more empirical datasets and plots showing how the model data fits to the empirical data. We shall make use of model entries containing empirical data in the real scenarios presented in this paper.

### III. Y-DB Application to Real-World Scenario

Computational models of physiology may account for diverse effects that take place at different levels from the organ to the cellular and molecular levels [11]. Example 2 presents a Physiome scenario gathering alternative models of a reference human blood vessel.

**Example 2:** The relevant information of this example is shown in Fig. 6. We consider the Physiome model entries displayed in relation HYPOTHESIS, which have been chosen in view of their overlapping. This can be precisely detected by the Y-DB system from their mathematical structure (cf. [11-B]) and is taken into account in the definition of data-intensive evaluation studies. In this example, the user defines two phenomena (see relation PHENOMENON) and perform two studies (see relation EXPLANATION), one on each phenomenon. These studies evaluate a subset of hypotheses to compete unbiasedly as phenomenon explanations.

#### A. Hypothesis Management

In the first data transformation step of the Y-DB pipeline (Fig. 4), phenomenon and hypothesis entities are defined. The entities shown in Fig. 5 have been defined for this example. Following their creation by insertion of the descriptive textual data (Fig. 5), their (resp.) empirical and simulation trial datasets (the “raw” data) are loaded into Y-DB.

**Phenomenon Data Definition.** The user creates a phenomenon entry and loads empirical datasets (CSV or MAT files) associated to it at anytime. Fig. 6 shows the phenomenon entries and Table I shows their associated empirical datasets as available in Physiome for Example 2. Fig. 7 shows a relation which is synthesized into Y-DB as result of loading empirical dataset ‘Hill_Fig1_P’. Phenomenon structure is redefined incrementally as each empirical dataset is inserted.

| PHENOMENON | HYPOTHESIS | FD (Key Constraint) |
|------------|------------|---------------------|
| 1          | Davis_Nikes_Fig3_Myo_DigData | 9 Pressure → Diameter |
| 1          | Bund_Fig1Men_Da | 12 Pressure → Diameter |
| 2          | Hill_Fig1_P | 15 Time → Pressure |
| 2          | Hill_Fig1_P | 38 Time → Diameter |

**TABLE I:** Empirical datasets loaded into Y-DB from CSV files. Column ‘N’ is the number of samples.
### Hypothesis Data Definition

The user creates a hypothesis entry, loads its MML code (mandatory), and then loads simulation trial datasets associated to it from MAT or CSV files. Fig. [6] shows the hypothesis entries as available in Physiome's model repository and Table [II] shows their associated simulation trial datasets.

The structure of each simulation trial dataset is validated against the hypothesis structure (extracted from its MML code). First, its primitive FD schema \( \Sigma \) is extracted from the MML code, and then a lossless, non-redundant FD schema \( \Sigma' \) is derived from the former to synthesize the hypothesis 3NF relations [5]. Fig. [8] shows both the primitive and derived FD schemes of hypothesis \( v = 89 \). Fig. [9] shows its synthesized relations loaded with simulation raw data. Note that, except for extra attribute \( \text{tid} \), the synthesized H-relations match the FDs in \( \Sigma_{89} \) (H89_KEY2 is not shown). That special attribute is added by default to H-relations in order to identify each simulation trial and "pretend" consistency until uncertainty is properly introduced in the data definition of evaluation studies [5]. The derivation of \( \Sigma' \) from \( \Sigma \) and the synthesis procedure as a whole are based on an original pseudo-transitive closure (PTC) algorithm for p-DBs in the context of hypothesis management applications (ibid.).

Once phenomena and hypotheses are defined, their management as data is then possible by means of all classical relational DB query capabilities. This is a result of \( \Sigma' \)-DB's systematic methodology for hypothesis encoding. We discuss that in perspective w.r.t. related work in [6].

#### TABLE II: Simulation trial datasets loaded into \( \Upsilon\)-DB from MAT files. Columns ‘tid’ and ‘N’ are (resp.) the trial ID and number of samples. Column ‘FD’ shows the constraint on the attributes which are relevant for Example 2.

| \( v \) | \( \phi \) | Name | Pub | Data | Description |
|---|---|---|---|---|---|
| 60 | \( R \) | Myogenic_Compliant_Vessel | N | N | This model simulates the flow through a passive and actively responding vessel driven by a sinusoidal pressure input. |
| 89 | \( R \) | Myo_Dyn_Resp_wFit | N | Y | This model describes the dynamic response of a vessel after a step increase in intraluminal pressure. |
| 113 | \( R \) | Vessel_Mechanics | N | Y | This model describes how a microvessel responds to changes in intraluminal pressure in the steady state. This change in vessel diameter to pressure is known as the myogenic response. |
| 186 | \( R \) | Regulatory_Vessel | Y | Y | This model describes the steady state regulatory vessel response to changes in pressure across and shear stress on the vessel wall. |

#### Fig. 6: Descriptive (textual) data of Example 2 with ids \( v \) from Physiome’s model repository (http://www.physiome.org/Models/modelDB/). Columns ‘Pub’ and ‘Data’ inform whether the model is associated with (resp.) a published paper and some empirical dataset.

#### Fig. 7: Phenomenon ‘as data’ relation loaded with data from ‘Hill_Fig1_P’ (cf. Table [I]).

#### Fig. 8: Primitive and derived FD schemes of hypothesis \( v = 89 \).
Query Q2 illustrates a typical user request for a hypothesis prediction: find the blood vessel’s pressure and diameter as predicted by $\upsilon = 89$ for $\phi = 2$ at, say, “nearby” $t = 5$ secs, see Fig. 10. Note in this result set of Q2 that trial tid=2 has finer resolution than tid=1 in the time domain.

**B. Predictive Analytics**

There are a few methodological steps to be processed in $\Upsilon$-DB’s pipeline (in between items ii and iii, Fig 4) before an evaluation study is completely defined for predictive analytics. The data definition of an evaluation study starts with the user selection of a phenomenon and a subset of its observables (e.g., $\phi = 1$, $\{\text{Pressure, Diameter}\}$). The system then carries out the processing steps just mentioned as filters to the hypotheses which are admissible to the evaluation study under definition.

**Simulation Trials.** Although hypotheses are meant to be general enough to be applied to different phenomena, its predictions are meant to be rather specific [14]. Thus, every simulation trial of a hypothesis is set to make predictions about a specific phenomenon. This is why the phenomenon id, $\phi$, is expected to be functionally related to the values of parameters when no parameter uncertainty is present (e.g., see relation H89_KEY1 in Fig 9). Once a phenomenon is chosen by the user for the study, the system then filters the hypotheses admissible to those that have at least one trial identified for that phenomenon.

**Semantic Mappings.** The data definition of evaluation studies for the purpose of predictive analytics may raise issues of model integration, which translates to data integration in $\Upsilon$-DB’s vision of hypotheses ‘as data.’ We approach that by considering phenomenon structure as global and hypothesis structure as local. Then, the semantic precision of the mathematical discourse allows us to handle occasional heterogeneities syntactically. We assume that if two (case-sensitive) symbols are equal, then they stand for the same physical quantity. This approach to semantic issues is arguably seamless to the common practice of the user. Thus, it is only expected from the user in view of the evaluation studies to insert some symbol mappings into the system! The system then keeps an updated summary $\mathcal{M}$ of mappings from phenomenon attribute symbols (global) to hypothesis attribute symbols (local). For Example 2, the set $\mathcal{M}$ of necessary mappings is as follows.

$$\mathcal{M} = \{ \text{Pressure} \rightarrow \text{P Pin}, \text{Diameter} \rightarrow \text{D}, \text{TotalTension} \rightarrow \text{Ttot TenTot}, \text{Time} \rightarrow \text{t} \}.$$

**Structural Overlapping.** Once the necessary semantic mappings have been set by the user, the data definition of the evaluation study must take into account the structural overlapping between phenomena and hypotheses. Table III shows the FDs involved in the overlapping of Example 2 which can be detected as long as we compute mappings $\mathcal{M}$ first.

**Sample Reference Frame.** The final processing step for the data definition of an evaluation study is the alignment of the competing hypotheses to the chosen phenomenon observation, i.e., the selection of predictive samples from the hypotheses (as data) in alignment with the observed sample from the phenomenon. It requires the setup of a (common) sample

---

1This is intrinsic to the mathematical modeling practice. We note that the precise specification of the phenomenon of interest is abstracted in terms of a parameter setting, including initial and boundary conditions.

2As opposed to other approaches in which the user is required, say, to adhere strictly to some domain ontology.

3Physical unit mappings are not necessary for the examples presented in this paper and then are left to future work.
Prior Probability Distribution. The data definition of the evaluation study is now ready to be accomplished. Certain relation \textit{EXPLANATION} (Fig. 6) gives rise to uncertain relation \textit{Y}[Exp] (Fig 12) under a uniform prior probability distribution just as shown by query Q1 in [II-A]. Similar data transformation queries produce (uncertain) \textit{Y}-relations from their (certain) \textit{H}-relations counterparts.\footnote{This involves a technique for learning parameter uncertainty and its correlations from the hypothesis “raw” data and propagating them into the predictive data [5].} The probability of, say, hypothesis \(\nu=89\) (which has parameter uncertainty, cf. \textit{Y89[tid]}) is split uniformly into its simulation trials. Query Q4 then puts all predictive samples into the sample reference frame set by Q3 and compute their probabilities (see Fig 12 \textit{Y[D]}, for the result).

The prior probability of a prediction, say, last row of \textit{Y[D]}, is always the product of the marginal probabilities of the hypothesis and of its trial as stored in world table \(W\).

Bayesian Inference. From such a prior probability distribution, the \textit{Y}-DB system allows the user to carry out one or more Bayesian inference steps that update at each step that prior to a posterior. Since we have discrete random variables mapped to the possible values of prediction attributes (e.g., diameter \(D\)) whose domain are continuous, we apply Bayes’ theorem for normal mean with a discrete prior \([12]\) \footnote{This involves a technique for learning parameter uncertainty and its correlations from the hypothesis “raw” data and propagating them into the predictive data [5].}. The procedure uses normal density function (1) with standard deviation \(\sigma\) to get the likelihood \(f(y | \mu_i)\) for each alternative prediction as mean \(\mu_i\), given observed value \(y\). Then Bayes’ rule (2) is applied to get the posterior \(p(\mu_i | y)\) from prior \(p(\mu_i)\).

\[
f(y | \mu_i) = \frac{1}{\sqrt{2\pi}\sigma^2} e^{-\frac{1}{2\sigma^2}(y-\mu_i)^2}
\]

\[
p(\mu_i | y) = f(y | \mu_i) p(\mu_i) / \sum_{i=1}^{m} f(y | \mu_i) p(\mu_i)
\]

In general, as for Example 2 we shall have not a single observation but a sample of independent observations \(y_1, \ldots, y_n, \text{viz., the observations of} \ D \text{at} t \in \{14.8, 30.5, 43.7\}\). Then we have to compute the likelihood function, \(f(y_1, \ldots, y_n | \mu_i)\), for each prediction \(\mu_i\), as a product \(\prod_{j=1}^{n} f(y_j | \mu_i)\) of the individual likelihoods \(f(y_j | \mu_i)\). Bayes’ rule is then given by Eq. 3.

\[
p(\mu_i | y_1, \cdots, y_n) = \frac{\prod_{j=1}^{n} f(y_j | \mu_i) p(\mu_i)}{\sum_{i=1}^{m} \prod_{j=1}^{n} f(y_j | \mu_i) p(\mu_i)}
\]

Performing the Bayesian inference in our example with a standard deviation of, say, \(\sigma = 5\), gives posterior distribution \(p(\mu_i | y_1, \cdots, y_n)\) for each alternative prediction (see \textit{Y[D]}, Fig. 12). Note that it leads to the elimination of hypothesis \(\nu = 60\) from the study and to imputing higher confidence on trial \textit{tid}=2 of hypothesis \(\nu = 89\) on the basis of observed data.

In case more empirical datasets become available, Bayesian inference may be applied over at anytime given the current probability as a prior until eventually a final posterior distribution is achieved. As a Bayesian approach to data-intensive science, the hypothesis probabilities shall not be interpreted at all as degrees of corroboration \([14]\). Essentially, the methodology of \textit{Y}-DB brings forward a tool for data-driven predictive analytics and decision making to the scientist users.

Finally, still in Example 2 we refer to the data definition of a second study \(\phi = 1, \{\text{Diameter}\}\). Through this study we highlight that \textit{Y}-DB allows the user to apply data analysis methods such as, say, interpolation, in the setup of sample reference frames as much representative as possible. Users shall prefer in general to (re-)set one or more simulation trials in view of their alignment to empirical datasets already available. Yet, some datasets may have irregular sample points and then require, say, some interpolation before a precise alignment to a frame in common with the hypotheses is possible.

For the study on phenomenon \(\phi = 1\), we apply a cubic-spline interpolation on dataset ‘Davis_Sikes_Fig3_Myo_DigData’ (Table 1) to setup a 24-size sample reference frame in common with competing hypotheses \(\nu \in \{113, 186\}\). This sample has FD (key constraint) \textit{Pressure} \(\rightarrow\) \textit{Diameter} with values of Pressure in the real interval \([15.0, 130.0]\), sampled with a skip of \(\Delta = 5.0\). Fig. 13 shows the result of applying Bayesian inference with standard deviation \(\sigma = 10\) for this study. Hypothesis \(\nu = 113\) gets near elimination from the study in face of the evidence available for the phenomenon.
C. Discussion

The hypotheses’ ratings resulting from both studies shown in the context of Example 2 in fact coincide with the results of model tuning as described in the Physiome model entries and their related publications. That validates the applicability of the Y-DB methodology as a tool for data-driven analysis in this real scenario.

The current practice in Computational Science for model evaluation and comparison in the presence of empirical data is somewhat handcrafted: model agreement is often assessed only qualitatively by referring to curve shapes in data plots. The Y-DB methodology offers a p-DB tool to perform such analysis semi-automatically directly ‘in the data.’ It is, therefore, a step towards higher standards of reproducibility and scalability for data-driven, industrial-scale science.

IV. Experiments

The efficiency and scalability of the U-relational representation system and its p-WSA query algebra have been extensively demonstrated [16]. Y-DBs, as U-relational hypothesis DBs, shall therefore be in general as efficient and scalable as any arbitrary U-relational DB. In these experiments we study the performance behavior of the Y-DB methodology in the particular context of our real-world Physiome scenario.

Setup. These experiments were performed on a 2.3GHz/4GB Intel Core i5 running Mac OS X 10.6.8 and MayBMS (a PostgreSQL 8.3.3 extension).

Metrics. We measure (procstime) the mean processing time (from ten runs) spent on (i) the main steps of the Y-DB pipeline (P_ETL and P_U-intro, cf. Fig. 2) and on (ii) representative queries Q_Y-scan and Q_Y-conf (hypothesis full scan and confidence querying). The procstime covers from early algorithmic query synthesis till its finished execution, and it is measured always for a given hypothesis. A hypothesis can be varied along the following metrics: hstructsize, (size of the hypothesis structure in attribute symbols), hdatasize (size of the hypothesis as data on disk in MB), ntrials (number of its alternative simulation trials). Example 3 settles the Physiome hypotheses used in these experiments.

Example 3: We add to hypotheses from the previous example larger Physiome hypotheses, $Y = \{186, 89, 60\} \cup \{75, 70, 120, 91, 153, 154\}$ (Fig. 14). We need to use, say, such five hypotheses to study how procstime relates to hstructsize. This is because structure is intrinsic to the hypothesis (it cannot be modified arbitrarily). The other metrics are easier to control. They can be set arbitrarily for any given hypothesis by modifying parameters of the model in JSim, e.g., generating a number of simulation trials (parameter sweeps), or larger datasets by setting the physical domains (e.g., time) in longer extension and/or finer resolution. Thus in order to study all the other metrics we shall fix one hypothesis, viz., $v = 154$, which is the larger w.r.t. hstructsize, having 600+ embarrassingly coupled mathematical equations.

Performance (Fig. 15). We start by evaluating hstructsize. We split the P_ETL (extract-transform) pipeline component apart from P_ETL because data loading is irrelevant to assess hstructsize (orthogonal to hdatasize). Considering the hypotheses listed in Fig. 14 we have P_ETL’s procstime varying along hstructsize as shown in in Fig. 15 (top, left).

The relationship between hdatasize and uncertainty (given by ntrials in our metrics) and its effects on procstime constitute a major notion of scalability for any p-DB. MayBMS has been thoroughly tested in that sense by introducing uncertainty into the TPC-H benchmark [16]. For the purpose of this specific real scenario, we have measured (for hypothesis $v = 154$) (i) the impact of hdatasize on P_ETL’s procstime; and then (ii) kept hdatasize roughly fixed to study how the P_ETL, Q_Y-scan and Q_Y-conf’s procstime vary along ntrials. Those are variants of queries Q_1, Q_2 (on Y-relations) and Q_4, respectively, without restrictive selectivity (in order to apply to full Y-relations).

Fig. 15 (top, right) shows how P_ETL’s procstime scales with hdatasize (in comparison with MATLAB just for the sake of having a useful reference). We learn that Y-DB’s ETL is competitive and in fact pays off as it enables plenty of benefits offered by p-DB technology, whether classical (e.g., data integrity) or advanced (e.g., uncertainty management). Fig. 15 shows also P_U-intro, Q_Y-scan and Q_Y-conf’s procstime performances. Our goal here is to assess the impact of uncertainty in performance. Note that it shall have higher qualitative significance when applied to multiple hypotheses (cf. Example 2). For these initial experiments, however, it is more convenient (easier to control experimentally) to exploit parameter uncertainty within a fixed hypothesis—it is irrelevant to MayBMS’s query engine whether uncertainty comes from multiple hypotheses and/or multiple trials on each.

V. Related Work

The NoDB approach argues to eliminate such ETL phase and enable a direct access to data in the raw data files [3]. Y-DB is not claimed for exploratory analytics. As we have seen, predictive analytics settles quite a different, also important use case that justifies loading multiple simulation trial datasets into a p-DB for a principled analysis according to the scientific method. Once this need to manage simulation data in a p-DB is motivated, one has to cope with the problem of systematic DB design—a long-lasting source of
As the first work on encoding hypotheses of mathematically-expressed hypotheses, the methodology in turn consists in the U-relational encoding of sequence and genome analysis hypotheses. The loading of scientific reports.

(iii) differs from the former (i-ii) in that each hypothesis is empirical evidence (RDF-encoded gene/protein data). SW AN (RDF-encoded) that are meant to support it on the basis of some empirical evidence (RDF-encoded gene/protein data). SW AN includes RDF-encoded empirical data. HyBrow is likewise, but hypotheses are formulated by the user and they are biological events.

The Robot Scientist relies on rule-based logic programming analytics to automatically generate and test RDF-encoded hypotheses of the kind ‘gene G has function A’ against RDF-encoded empirical data. HyBrow is likewise, but hypotheses are formulated by the user and they are biological events. SW AN in turn disfavors hypothesis evaluation analytical techniques and focus on descriptive aspects: hypotheses are high-level natural language statements retrieved from publications. Each ‘hypothesis’ is associated with lower-level ‘claims’ (both RDF-encoded) that are meant to support it on the basis of some empirical evidence (RDF-encoded gene/protein data). SW AN includes RDF-encoded empirical data. HyBrow is likewise, but hypotheses are formulated by the user and they are biological events.

Beyond traditional DB design, Y-DB’s methodology addresses the challenging problem of large-scale hypothesis encoding. Some recent initiatives address hypothesis encoding into the RDF data model in the field of Bioinformatics: (i) the Robot Scientist is a knowledge-base system (KBS) for automated generation and testing of hypotheses about what genes encode enzymes in the yeast organism; (ii) HyBrow is a KBS for scientists to test their hypotheses about events of the galactose metabolism also of the yeast organism; and (iii) SW AN is a KBS for scientists to share hypotheses on possible causes of the Alzheimer disease.

The Robot Scientist relies on rule-based logic programming analytics to automatically generate and test RDF-encoded hypotheses of the kind ‘gene G has function A’ against RDF-encoded empirical data. HyBrow is likewise, but hypotheses are formulated by the user and they are biological events. SW AN in turn disfavors hypothesis evaluation analytical techniques and focus on descriptive aspects: hypotheses are high-level natural language statements retrieved from publications. Each ‘hypothesis’ is associated with lower-level ‘claims’ (both RDF-encoded) that are meant to support it on the basis of some empirical evidence (RDF-encoded gene/protein data). SW AN includes RDF-encoded empirical data. HyBrow is likewise, but hypotheses are formulated by the user and they are biological events.

All of them (i-ii-iii), though, consist in an RDF encoding of sequence and genome analysis hypotheses. The Y-DB methodology in turn consists in the U-relational encoding of mathematically-expressed hypotheses. To the best of our knowledge, this is the first work on encoding hypotheses from mathematical equations into a relational (probabilistic) data model. As for hypothesis evaluation and comparison analytics, Y-DB is distinguished in terms of its Bayesian inference approach. The latter has been pointed out as a major direction for the improvement of the Bioinformatics’ initiatives just mentioned (cf. [18, p. 13]), and it is in fact an influential model of decision making for hypothesis evaluation [14, p. 220].

VI. CONCLUSIONS

In this paper we have presented the vision of Y-DB, which addresses hypothesis management as a new challenge in view of supporting large-scale predictive analytics. The latter differs from exploratory analytics in some fundamental ways, opening a new use case which is explored in this paper. We have presented the methodology Y-DB as providing a principled approach to manage large-scale hypotheses and analyze them as uncertain predictive data in face of empirical data.

The applicability of Y-DB has been shown in a real-world scenario derived from the Physiome research project. We have introduced our technique for hypothesis encoding in brief and detailed the process of building an Y-DB with representative models from Physiome’s model repository. That qualitative assessment is followed by experiments that provide a concrete feel on the performance behavior of Y-DB for models with up to 600+ mathematical variables. We have studied in particular the overhead incurred in its ETL process that enables hypothesis management. Future work includes further experimentation as well as expanding hypothesis encoding and the Y-DB methodology into other areas of science.

ACKNOWLEDGMENTS

This research has been supported by the Brazilian funding agencies CNPq (grants n° 141838/2011-6 and 309494/2012-5) and FAPERJ (grants INCT-MACC E-26/170.030/2008 and ‘Nota10’ E-26/100.286/2013). We thank IBM’s Ph.D. Fellowship 2013-2014, and we are grateful to Paulo Trenhago for his invaluable contribution to this applicability study as a user.
REFERENCES

[1] T. Hey et al., The Fourth Paradigm: Data-Intensive Scientific Discovery. Microsoft Research, 2009.
[2] A. Ailamaki, V. Kantere, and D. Dash, “Managing scientific data,” Comm ACM, vol. 53, no. 6, pp. 68–78, 2010.
[3] I. Alagiannis, R. Borovica, M. Branco, S. Idreos, and A. Ailamaki, “NoDB: Efficient query execution on raw data files,” in ACM SIGMOD, 2012, pp. 241–52.
[4] P. Haas, P. Maglio, P. Selinger, and W. Tan, ‘Data is dead... without what-if models,” PVLDB, vol. 4, no. 12, pp. 1486–9, 2011.
[5] B. Gonçalves and F. Porto, “Υ-DB: Managing scientific hypotheses as uncertain data,” PVLDB, vol. 7, no. 11, 2014. (postprint available at arXiv/CoRR 1405.4607).
[6] D. Suciu, D. Olteanu, C. Ré, and C. Koch, Probabilistic Databases. Morgan & Claypool Publishers, 2011.
[7] J. Ullman, Principles of Databases and Knowledge-Base Systems. Computer Science Press, 1988.
[8] M. Hines, T. Morse, M. Migliore, N. Carnevale, and G. Shepherd, “ModelDB: A database to support computational neuroscience,” J Comput Neurosci, vol. 17, no. 1, pp. 7–11, 2004.
[9] V. Chelliah, C. Laibe, and N. Le Novère, “BioModels Database: A repository of mathematical models of biological processes,” Methods in Molecular Biology, no. 1021, pp. 189–99, 2013.
[10] J. B. Bassingthwaighte, “Strategies for the Physiome Project,” Ann Biomed Eng, vol. 28, pp. 1043–58, 2000.
[11] P. J. Hunter and T. K. Borg, “Integration from proteins to organs: the Physiome Project.” Nat Rev Mol Cell Biol., vol. 4, no. 3, pp. 237–43, 2003.
[12] W. M. Bolstad, Introduction to Bayesian Statistics, 2nd ed. Wiley-Interscience, 2007.
[13] C. Koch, MayBMS: A system for managing large uncertain and probabilistic databases. In C. Aggarwal (ed.), Managing and Mining Uncertain Data, chapter 6. Springer-Verlag, 2009.
[14] J. Losee, A historical introduction to the philosophy of science, 4th ed. Oxford University Press, 2001.
[15] E. Rahm and H. Hai Do, “Data cleaning: Problems and current approaches,” IEEE Data Engineering Bulletin, vol. 23, no. 4, pp. 3–13, 2001.
[16] L. Antova, T. Jansen, C. Koch, and D. Olteanu, “Fast and simple relational processing of uncertain data,” in Proc. 24th IEEE ICDE, 2008, pp. 983–92.
[17] A. Badia and D. Lemire, “A call to arms: Revisiting database design,” SIGMOD Record, vol. 40, no. 3, pp. 61–9, 2011.
[18] L. N. Soldatova and A. Rzhetsky, “Representation of research hypotheses,” J. Biomed. Sem., vol. 2, no. S2, 2011.
[19] R. D. King et al., “The automation of science,” Science, vol. 324, no. 5923, pp. 85–9, 2009.
[20] S. Racunas et al., “HyBrow: a prototype system for computer-aided hypothesis evaluation,” Bioinformatics, vol. 20, no. 1, pp. 257–64, 2004.
[21] Y. Gao et al., “SWAN: A distributed knowledge infrastructure for alzheimer disease research,” J. Web Semantics, vol. 4, no. 3, pp. 222–8, 2006.
[22] A. de Waard et al., “Hypotheses, evidence and relationships: The HypER approach for representing scientific knowledge claims,” in Proc. of ISWC, Workshop on Semantic Web Applications in Scientific Discourse, 2009.