Model-Data Ecosystems: Challenges, Tools, and Trends

Peter J. Haas
IBM Almaden Research Center
Great Progress in Analytics by the Database Community

- Transactions & Reports, IMS
- OLAP
- Semi-structured & Unstructured text
- Data mining
- Web data
- Streaming data
- Statistical analysis
- Text analytics
- Machine learning
- Relational model & SQL
- Massive Data/Cloud DB
- Uncertain data
- Semantic data
Great Progress in Analytics by the Database Community

BUT: Why do enterprises care about (big) data in the first place?
Because Enterprises Need to Make DECISIONS

“Analytics is...a complete [enterprise] problem solving and decision making process”

**Descriptive Analytics**: Finding patterns and relationships in historical and existing data

**Predictive analytics**: predict future probabilities and trends to allow what-if analysis

**Prescriptive analytics**: deterministic and stochastic optimization to support better decision making
Shallow Versus Deep Predictive Analytics

Extrapolation of 1970-2006 median U.S. housing prices

3.3 Eulerian Dynamical Core
\[ \frac{\partial \zeta}{\partial t} = k \cdot \nabla \left( \frac{n}{\cos \phi} \right) + F_{\zeta}, \]
\[ \frac{\partial \theta}{\partial t} = \nabla \cdot \left( \frac{n}{\cos \phi} \left( E + \Phi \right) \right) + F_{\theta}, \]
\[ \frac{\partial T}{\partial t} = \frac{-1}{a \cos^2 \phi} \left[ \frac{\partial}{\partial \lambda} \left( UT \right) + \cos \phi \frac{\partial}{\partial \phi} \left( VT \right) \right] + T \delta - \eta \frac{\partial T}{\partial \eta} + \frac{R_{\nu} \omega}{c_p} \frac{\partial}{\partial t}, \]
\[ \frac{\partial q}{\partial t} = \frac{-1}{a \cos^2 \phi} \left[ \frac{\partial}{\partial \lambda} \left( Uq \right) + \cos \phi \frac{\partial}{\partial \phi} \left( Vq \right) \right] + q \delta - \eta \frac{\partial q}{\partial \eta} + S, \]
\[ \frac{\partial \eta}{\partial t} = \int_{\eta}^{r} \nabla \cdot \left( \frac{\partial}{\partial \eta} \mathbf{V} \right) \, d\eta. \]
Data is dead… Without What-If Models

Descriptive and shallow predictive analytics are last resorts for decision making in complex systems…

When you can’t find the domain experts…

…but they are the main focus of most database and IM technology and research!

Need to supplement data with first-principles simulation models

…The notion that quantitative, numerical data are the only type of information needed to build an accurate model is flawed. In fact, I believe that the typical business obsession with numeric data can do more damage than good.

- Eric Bonabeau
Confluence of Research on (Big) Data Management & Predictive Analytics

Today: An idiosyncratic whirlwind tour of

Simulation and information integration
- Information integration via agent-based simulation
- Fusing real & simulated data (data assimilation)

Data-intensive simulation
- Composite simulation models
  - Data transformation between models
  - Query optimization → simulation-run optimization
- Incorporating simulation into DB systems and vice versa

Goal: Some interesting examples to stimulate your thinking
π-shaped presentation, additional topics in paper (metamodeling)
Simulation and information integration

- Information integration via agent-based simulation
- Fusing real & simulated data (data assimilation)
Information Integration via Agent-Based Simulation
II via Agent-Based Simulation: Marketing Example

[Bonabeau, WSC 2013]

Many drivers of consumer behavior...

Non-overlapping datasets studied in isolation...

...are now integrated

Source: Eric Bonabeau
II via Agent-Based Simulation: Marketing Example

[Bonabeau, WSC 2013]

Many drivers of consumer behavior...

Non-overlapping datasets studied in isolation...

Key problem is model calibration (see paper)
- Maximum likelihood
- Method of simulated moments
- Machine learning?

...are now integrated

Source: Eric Bonabeau
Fusing Real and Simulated Data (Data Assimilation)
Fusing Real and Simulated Data: Data Assimilation

Integrate real and simulated data via particle filtering [Xue et al., 2012]

Classical Monte Carlo estimation of density \( \pi_n(x_{1:n}) = \gamma_n(x_{1:n}) / Z_n \)

\[
\hat{\pi}_n(x_{1:n}) = \sum_{i=1}^{N} \frac{1}{N} \delta_{x_{1:n}}(x_{1:n})
\]

so that

\[
E[g(X_{1:n})] = \int g(x_{1:n})\pi_n(x_{1:n}) \, dx_{1:n} 
\approx \int g(x_{1:n})\hat{\pi}_n(x_{1:n}) \, dx_{1:n} = \frac{1}{N} \sum_{i=1}^{N} g(X_{1:n})
\]

- Can fail when \( n \) is large and/or \( \pi_n \) is complex (\( Z_n \) is often the culprit)

Importance sampling

- Sample from an “easier” importance density \( q_n \) and correct:

\[
w_n(x_{1:n}) = \gamma_n(x_{1:n}) / q_n(x_{1:n})
\]

\[
\pi_n(x_{1:n}) = w_n(x_{1:n}) q_n(x_{1:n}) / Z_n \quad \text{and}
\]

\[
Z_n = \int w_n(x_{1:n}) q_n(x_{1:n}) \, dx_{1:n}
\]

- So draw \( N \) i.i.d. samples (particles) from \( q_n \) and insert MC approx. for \( q_n \) above:

\[
\hat{\pi}_n(x_{1:n}) = \sum_{i=1}^{N} W^i_N \delta_{x_{1:n}}(x_{1:n})
\]

where \( W^i_N = w_n(x_{1:n}^i) / \sum_{j=1}^{N} w_n(x_{1:n}^j) \)

\( Z_n \) not needed to compute weights
Sequential importance sampling (SIS)

- Importance sampling where \( q_n(x_{1:n}) = q_1(x_1) \prod_{k=2}^n q_k(x_k \mid x_{1:k-1}) \)
- Recursive formula for weights:

\[
W_n(x_{1:n}) = W_{n-1}(x_{1:n-1}) \alpha(x_{1:n}) \quad \text{where} \quad \alpha_n = \frac{\gamma_n(x_{1:n})}{\gamma_{n-1}(x_{1:n-1}) q_n(x_n \mid x_{1:n-1})}
\]

SIS with resampling (SI SR)

- Stabilize SIS by resampling according to \( W_n^1, W_n^2, \ldots, W_n^N \) at each step
- This is a sample from \( \hat{\pi}_n \) --- set all new weights equal to \( 1/N \)

Particle filtering (SI SR for hidden Markov models)

- Discrete time Markov chain \( \{X_n\}_{n \geq 1} \) with transition probability density \( p_n(x_n \mid x_{n-1}) \)
- Observation process \( \{Y_n\}_{n \geq 1} \) with probs \( p_n(y_n \mid x_n) \) of observation given true state
- Take \( \gamma_n(x_{1:n}) = p_n(x_{1:n}, y_{1:n}) \) so \( \pi_n(x_{1:n}) = p_n(x_{1:n} \mid y_{1:n}) \)
- Optimal importance density (minimizes variance of weights):

\[
q_n^*(x_n \mid x_{n-1}, y_{n-1}) \propto p_n(x_n \mid x_{n-1}) p_n(y_n \mid x_n)
\]
Data Assimilation, Continued

Application to data assimilation [Xue et al., 2013]

- DEVS-FIRE model
  - Models stochastic progression of wildfire over gridded terrain
  - State ∈ {unburned, burned, burning-intensity}
  - Merge model data x and sensor data y: \( p_n(x_n | y_n) \)

- Gaussian sensor model: \( p_n(y_n | x_n) \)

- Original importance density: \( p_n(x_n | x_{n-1}), n \geq 1 \)
  - To sample from importance density (step 6), run simulation for 1 time step
  - Analytical expressions (step 8) reduce to sensor model
  - Ignores sensor reading recall: \( q_n(x_n | x_{n-1}, y_{n-1}) \propto p_n(x_n | x_{n-1})p_n(y_n | x_n) \)

- Improved sensor-aware importance density under development
  - Model and sensors weighted according to “confidence”
  - Kernel density estimation used to obtain analytical expressions (step 8)
Data-intensive simulation

- Composite simulation models
  - Data transformation between models
  - Query optimization → simulation-run optimization
- Incorporating simulation into DB systems and vice versa
Composite Simulation Models
Composite Simulation Models: Overview

Motivation:
- Domain experts have different worldviews
- Use different vocabularies
- Sit in different organizations
- Develop models on different platforms
- Don’t want to rewrite existing models!

Composite modeling approach
- Combines data integration with simulation
- Loose coupling via data exchange
- Metadata for detection and semi-automatic correction of data mismatches
- Ex: Splash prototype [Tan et al., IHI 2012]

Advantages
- Model curation and re-use
- Flexibility
- No need for “universal” platform, API, etc.
Composite Simulation Models: Splash Example

SADL metadata language

Kepler adapted for model composition

Design-time components

Run-time components:
- Kepler adapted for model execution
- Experiment Manager
  (sensitivity analysis, metamodeling, optimization)

Data transformation tools:
- Clio++
- Time Aligner (MapReduce algorithms)
- Templating mechanism

© 2012 IBM Corporation
Composite Simulation Models: Data transformation I

- Algebra of “gridfields” [Howe and Maier, VLDBJ 2005]
  - Grid: collection of cells (of various dimension) + incidence relation
    - $x \preceq y$ if
      - $\dim(x) = \dim(y)$ and $x = y$; or
      - $\dim(x) < \dim(y)$ and $x$ “.touches” $y$
  - Gridfield = grid + mappings from cells to data values
  - Key operation on gridfields: regrid
    - Maps set $S$ of source cells to a given target cell
    - Applies aggregation functions to $S$ to compute associated data values
  - Restrictions (a kind of selection) commute with regrid $\rightarrow$ optimizations

Source: Howe & Maier

![Fig. 1 Datasets bound to the nodes and polygons of a 2-D grid](image)
Massive scale time alignment

- Common Splash time alignment operation:
  Interpolating (massive) time-series data

- Parallelize on Hadoop

- Linear interpolation: easy

- Cubic spline interpolation: hard
  - Computing spline constants = solving massive tri-diagonal linear system
  - Solution: distributed stochastic gradient descent algorithm (see paper)

\[ A = \begin{bmatrix}
\frac{h_0 + h_1}{3} & \frac{h_1}{6} & 0 & \ldots & 0 & 0 & 0 \\
\frac{h_1}{6} & \frac{h_1 + h_2}{3} & \frac{h_2}{6} & \ldots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & \frac{h_{m-3}}{6} & \frac{h_{m-3} + h_{m-2}}{3} & \frac{h_{m-2}}{6} \\
0 & 0 & 0 & \ldots & 0 & \frac{h_{m-2}}{6} & \frac{h_{m-2} + h_{m-1}}{3}
\end{bmatrix} \quad b = \begin{bmatrix}
\frac{d_2 - d_1}{h_1} - \frac{d_1 - d_0}{h_0} \\
\frac{d_3 - d_2}{h_2} - \frac{d_2 - d_1}{h_1} \\
\vdots \\
\frac{d_m - d_{m-1}}{h_{m-1}} - \frac{d_{m-1} - d_{m-2}}{h_{m-2}}
\end{bmatrix} \]

Solve: \( Ax = b \)
Motivating example: Two models in series, 100 reps

- Naïve approach: execute composite model (i.e., Models 1 & 2) 100 times
- A better approach:
  - Execute Model 1 once and cache result
  - Read from cache when executing Model 1

Question: Can result-caching idea be generalized?
Optimizing Simulation Runs (Continued)

Result-Caching: General Method for Two Models [Haas, WSC 2014]

- Running example: Two models in series
  - Model 1 → \( Y_1 \) → Model 2 → \( Y_2 \)
  - Stochastic → Stochastic

- Goal: Estimate \( \theta = E[Y_2] \) based on \( n \) replications

- Result-caching approach:
  1. Choose \( \alpha \in (0,1) \) (the re-use factor)
  2. Generate \( \lceil \alpha n \rceil \) outputs from Model 1 and cache them
  3. To execute Model 2, cycle through results
  4. Estimate \( \theta \) by \( \theta_n = n^{-1} \sum_{i=1}^{n} Y_{2;i} \)
Result-Caching: Optimizing the Re-Use Factor

Budget-constrained setting [Glynn & Whitt 1992]
- Cost of producing $n$ outputs from Model 2 is $C_n = \sum_{j=1}^{[\alpha n]} \tau_{1;j} + \sum_{j=1}^{n} \tau_{2;j}$ (random)
- Under (large) fixed computational budget $c$:
  - Number of Model 2 outputs produced is $N(c) = \max\{n \geq 0 : C_n \leq c\}$
  - Estimator is $U(c) = \theta_{n(c)}$ (average of a random # of dependent variables)

Key result: a central limit theorem

Suppose that $E[\tau_1 + \tau_2 + Y_2^2] < \infty$. Then $U(c)$ is asymptotically $N(\theta, g(\alpha) / c)$.

where $r_\alpha = \lfloor 1 / \alpha \rfloor$ and

$$g(\alpha) = (\alpha E[\tau_1] + E[\tau_2]) \left\{ \text{Var}[Y_2] + (2r_\alpha - \alpha r_\alpha (r_\alpha + 1)) \text{Cov}[Y_2, Y_2'] \right\}$$

(expected cost per obs.) x (variance per obs.)

- Thus, minimize $g(\alpha)$ [or maximize asymptotic efficiency $= 1 / g(\alpha)$]
Result-Caching: The Optimal Re-Use Factor

Optimal solution

- Assume that $\text{Cov}[Y_2, Y'_2] \geq 0$
- Approximate $r_\alpha$ by $1 / \alpha$

$$
\alpha^* \approx \left( \frac{E[\tau_2] / E[\tau_1]}{\left( \frac{\text{Var}[Y_2]}{\text{Cov}[Y_2, Y'_2]} - 1 \right)} \right)^{1/2} \wedge 1
$$

Observations

- If Model 1 cost is large relative to Model 2, then high re-use of output
- If Model 2 insensitive to Model 1 ($\text{Cov} \ll \text{Var}$), then high re-use
- If Model 1 is deterministic ($\text{Cov} = 0$), then total re-use

Ongoing work

- Generalize to > 2 models (math similar to sampling-based join-size estimation)
- Develop techniques to compute/approximate needed statistics
- In general: Extend query optimization to “simulation-run optimization”
Incorporating Simulation into DB Systems
Incorporating Simulation into DB I: MCDB [Jampani et al., TODS 2011]

CREATE TABLE optionVal (opID, val) AS
  FOR EACH o IN option
    WITH oVal AS optionSim(
      VALUES(o.initVal, o.r, o.sigma, o.k, o.m, o.T))
  SELECT o.opID, v.VALUE FROM oVal v

Random DB = D

Schema
VG Functions
Parameter Tables

Q(D) =
  Select c.opID, SUM(…)
  ...

optionSim = Value generation (VG) function

Generator

Q(d_1)
Q(d_2)
...
Q(d_n)

Instantiations
(possible worlds)

Estimator

\hat{E} [ \text{totVal} ]
\hat{\text{Var}} [ \text{totVal} ]
q_{0.01} [ \text{totVal} ]
Histogram
Error bounds

- Implementation uses “tuple bundle” techniques, parallel DB & MapReduce execution
- Challenges: extreme quantiles, threshold queries (>2% decline in sales with prob > 50%)
Incorporating Simulation into DB II: SimSQL

- Re-implementation and extension of MCDB [Cai et al., SIGMOD 2013]
  - Database sequence: D[0], D[1], D[2], …
  - VG function for D[i] can be parameterized on any table in D[i-1]
  - I.e., Can simulate database-valued Markov chains

- Potential application to massive-scale agent-based simulations [Wang et al., VLDB, 2010]

<table>
<thead>
<tr>
<th>ID</th>
<th>LocX</th>
<th>LocY</th>
<th>DState</th>
<th>Vaccinated?</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent1</td>
<td>2.34</td>
<td>2.48</td>
<td>Infected</td>
<td>N</td>
</tr>
<tr>
<td>agent2</td>
<td>3.57</td>
<td>3.72</td>
<td>recovered</td>
<td>N</td>
</tr>
<tr>
<td>agent3</td>
<td>50.20</td>
<td>80.9</td>
<td>susceptible</td>
<td>Y</td>
</tr>
</tbody>
</table>

- Agent-based simulation = sequence of self-joins
  - Often, only nearby agents interact, so can exploit parallel processing
  - Not really explored in SimSQL setting
Incorporating DB Systems into Simulation
Incorporating DB into Simulation: Indemics

- Indemics system for simulating epidemics [Bisset et al., ACM TOMACS 2014]
  - Uses HPC for compute-intensive tasks (disease propagation), DBMS for data-intensive tasks (state assessment and intervention)
  - Observer can stop simulation, input an intervention, then resume

Source: Bisset et al.
Example intervention strategy:

```
Algorithm 1 Vaccinate preschoolers if more than 1% are sick
CREATE TABLE Preschool(pid) AS
  (SELECT pid FROM Person WHERE 0 ≤ age ≤ 4);
/* Based on demographic data */
DEFINE nPreschool AS (SELECT COUNT(pid) FROM Preschool);
for day = 1 to 300 do
  /* Based on demographic and disease dynamic data */
  WITH InfectedPreschool (pid) AS
    (SELECT pid FROM Preschool, InfectedPerson
     WHERE Preschool.pid = InfectedPerson.pid);
  DEFINE nInfectedPreschool AS
    (SELECT COUNT(pid) FROM InfectedPreschool);
  if nInfectedPreschool > 1% × nPreschool then
    Apply vaccines to SELECT( pid FROM Preschool);
  /* Intervention subpopulation and action */
  end if
end for
```

Formal model of system:
- Coevolving Graphical Discrete Dynamical System (CGDDS)
- Partially observable Markov decision process (POMDP)
Conclusions

- Intertwining of data management and simulation — both are needed
- Many problems in early stages, need formalization
- Lots of room for interesting research!

Data-intensive simulation

- Composite simulation models
  - Data transformation between models
  - Query optimization → simulation-run optimization
  - Incorporating simulation into DB systems and vice versa

Simulation and information integration

- Information integration via agent-based simulation
- Fusing real & simulated data (data assimilation)
Model-Data Ecosystems:
Challenges, Tools, and Trends

Peter J. Haas
IBM Almaden Research Center
phaas@us.ibm.com

PODS 2014