

Formally Verifying Expert Conjectures on Extreme Scale Data

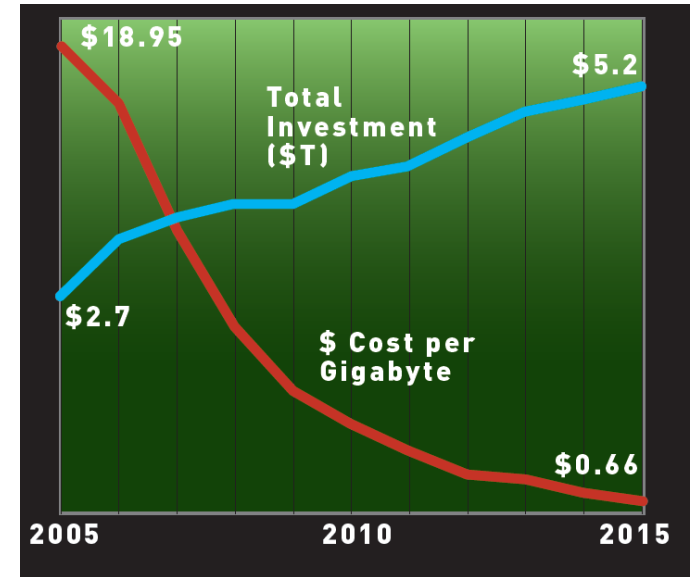


SUMIT K. JHA, UNIVERSITY OF CENTRAL FLORIDA, ORLANDO
RAJ G. DUTTA, UNIVERSITY OF CENTRAL FLORIDA, ORLANDO
EMILY SASSANO, UNIV. OF CENTRAL FLORIDA, ORLANDO



Global Data Challenge

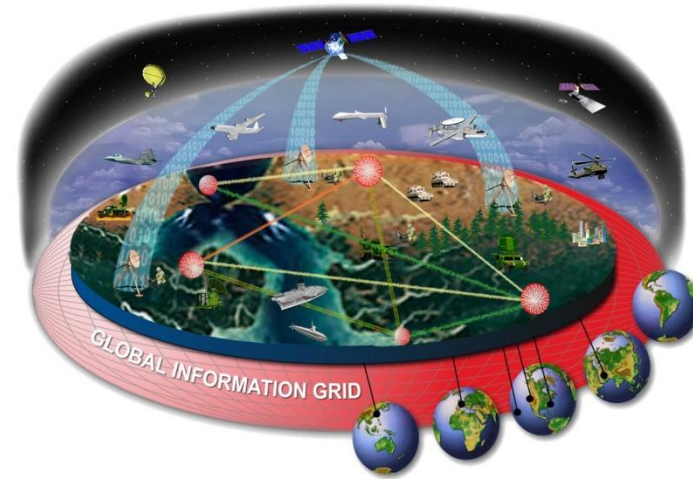
- 1.8 zettabytes (1.8 trillion gigabytes) stored in 500 quadrillion “files” in 2011
 - International Data Corporation (IDC) Digital Universe Study
- Amount of data is more than doubling every 2 years.
- **Digital Universe Paradox :**
 - Cost of creating, capturing, managing and storing information is going down since 2005
 - Investment to create, manage, store and derive information has gone up.



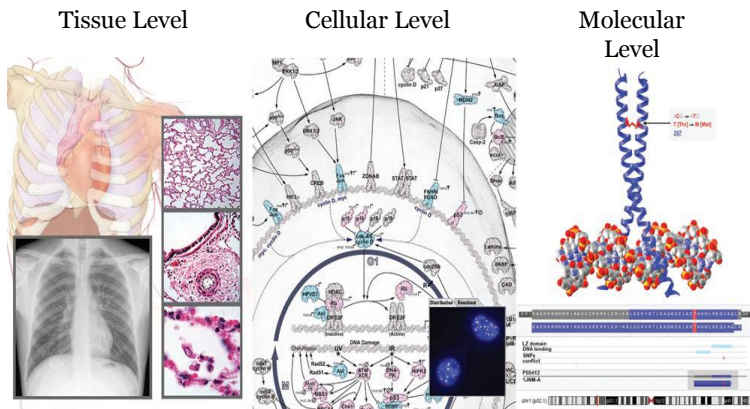
Digital Universe Paradox.
Source: IDC iView “ Extracting Value from Chaos”



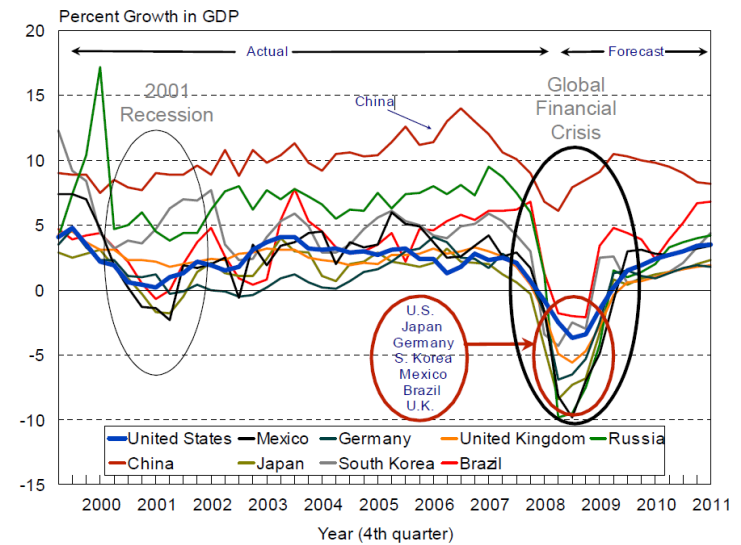
Cisco Forecast global growth of Mobile Data Traffic per month



Global Information Grid project of U.S DoD



Data at different scales from biologist
Source: "Visualizing Biological data – now and in future"



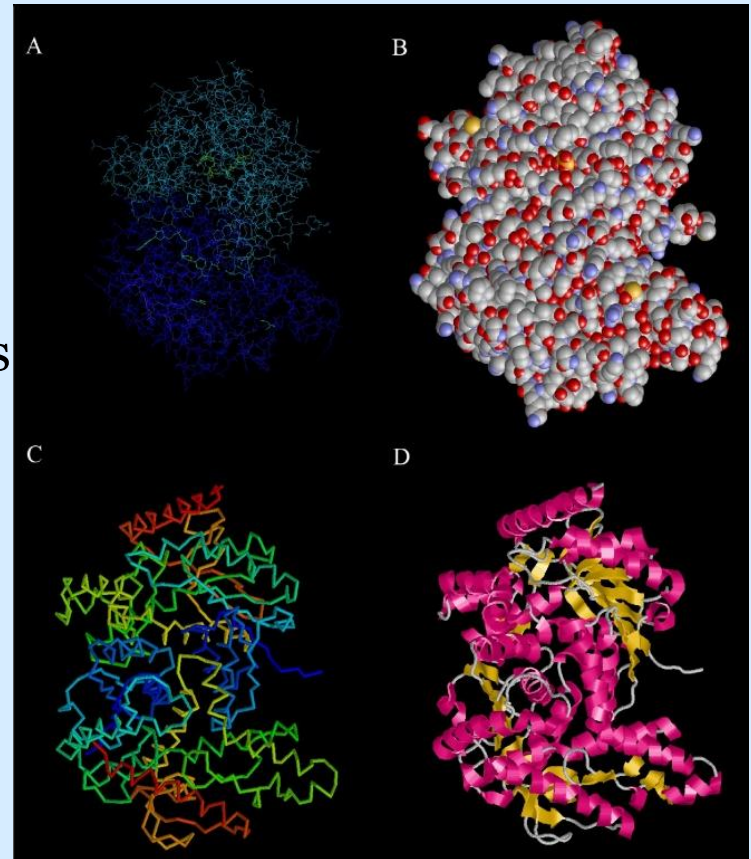
Economic Growth Rates
Source: Global Insight.



Current Approach to Extreme Scale Data



- Visualization
- Applied across domains
 - Biology e.g. Protein Structures
 - ✦ Shows an intuitive movie of events
 - ✦ Too many events
 - ✦ Many events hidden from view
 - Spatially (Projection)
 - Short time-scale events



Current Approach to Extreme Scale Data



- Visualization
- Applied across domains
 - Biological Data
 - Financial Data
 - ✦ On a recent visit to the trading floor of a large Wall Street firm, each trader had at least 6 monitors.
 - ✦ Each monitor had at least two plots!



Current Approach to Extreme Scale Data



- Visualization
- Applied across domains
 - Biological Data
 - Financial Data
 - Climate Data
 - ✦ Based on the same data,
 - Yes, its getting warmer!
 - No, let us produce more CO₂.



Current Approach to Extreme Scale Data



- Visualization
- Applied across domains
 - Biological Data
 - Financial Data
 - Climate Data
 - Homeland Security / Defense
 - ✦ Sensors are getting cheaper
 - ✦ Confluence of heterogeneous data
 - ✦ Global Network of Data Sources



Visualization of Data is not enough!



- Too much temporal data to observe.
 - protein structures
- Too much spatial data to visualize.
 - homeland security
- Data coming in too fast.
 - financial time series, astronomical data
- Observations subject to interpretation.
 - Climate Data
 - Most of wet lab experiment
 - ✦ We agree to disagree

Better Visualization Algorithms?



- **Limits of Human Perception and Cognition**
 - Built-in limit on our capability to perceive 1-D signals.
 - ✦ Mean is 2.6 bits, standard deviation is .6 bits.
 - Size, Brightness, and Hue
 - ✦ Should be 7.6, but measured at 4.1 bits.
 - Conjecture: We do compressed sensing!
- **No algorithm will change our perception capability**
 - Not enough resolution
 - Too few dimensions (three at most?)

Need a fundamental shift away from visualization



- unless we all turn into cyborgs
- and increase our abilities (hopefully?)
 - Perception
 - Cognition



Linear Temporal Logic:

- Language for specification of conjectures.
- Has been applied to
 - define semantics of temporal expression in natural languages,
 - verification of concurrent programs.
- Syntax of Linear Temporal Logic

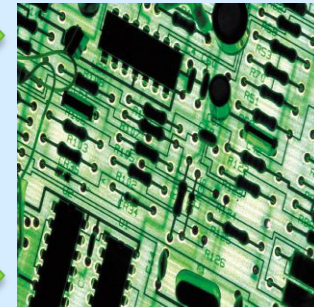
$$\phi ::= x_j \leq v \mid x_j \geq v \mid (\phi_1 \vee \phi_2) \mid \neg \phi_1 \mid (\phi_1 \mathbf{U} \phi_2)$$



Verify Conjectures on Extreme Scale Data



Data from
millions of
patients



Yes /
No

Spatio-
Temporal
Logic



Automated
Verification



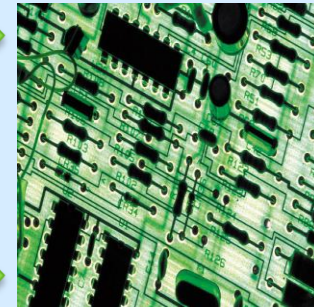
Biomarker Conjecture:

The level of protein X in the
blood goes high 4 to 6 months
before a cancer tumor is
observable

Verify Conjectures on Extreme Scale Data



Tick-by-tick
data for last
10 years



Yes /
No



Temporal
Logic



Automated
Verification



Trading Strategy Conjecture:
Pair trading future A against
future B with open and close
thresholds of 0.1 would produce
robust profits of \$1000 per trade.

Conjectures in Temporal Logic



- Not surprising!
- Temporal Logic captures tense in natural language.
 - not invented for hardware,
 - or software.
- Linear temporal logics to capture conjecture on data
 - derived from experiments
 - ✦ Natural,
 - ✦ Or computational,
 - ✦ or cyber-physical

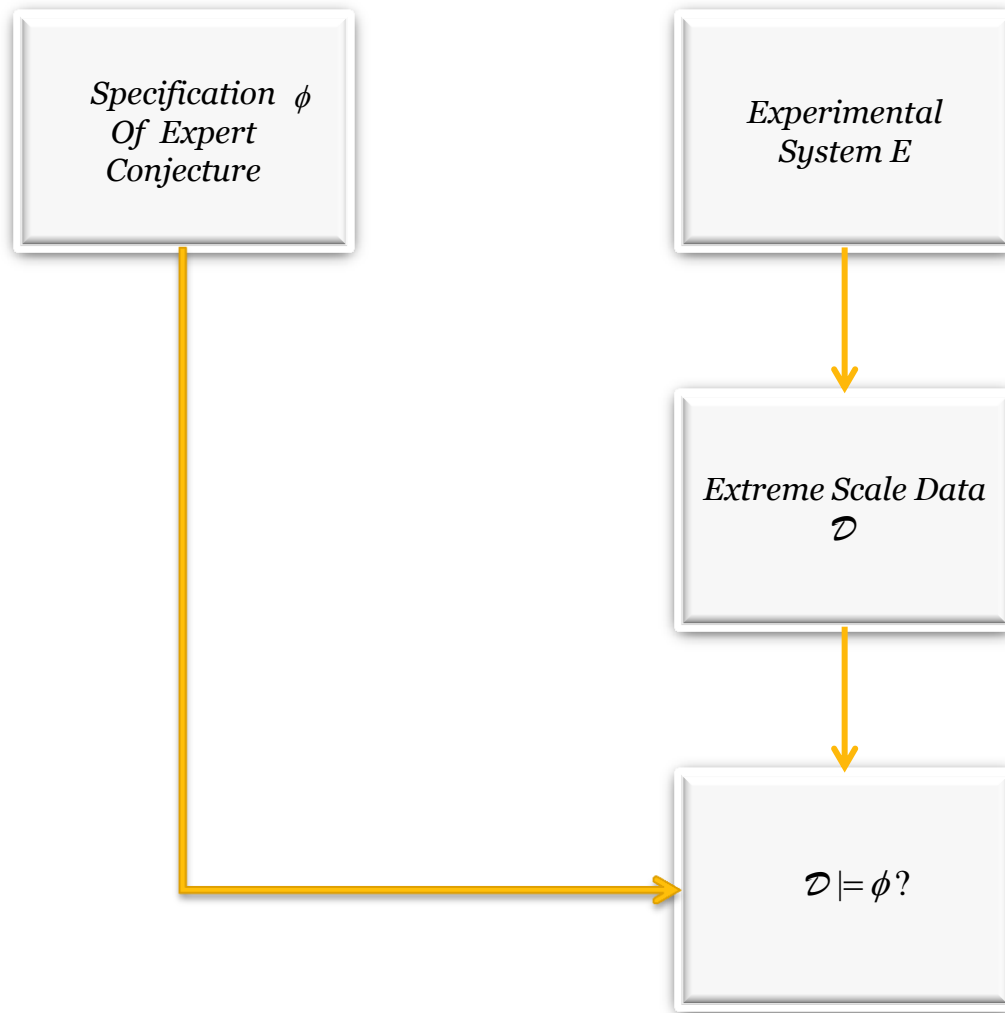


Figure 8: Verification of Extreme Scale Data Against Expert Conjecture

Difference from Traditional Verification



Model Verification

Formal specification known from design

Verifying Conjectures

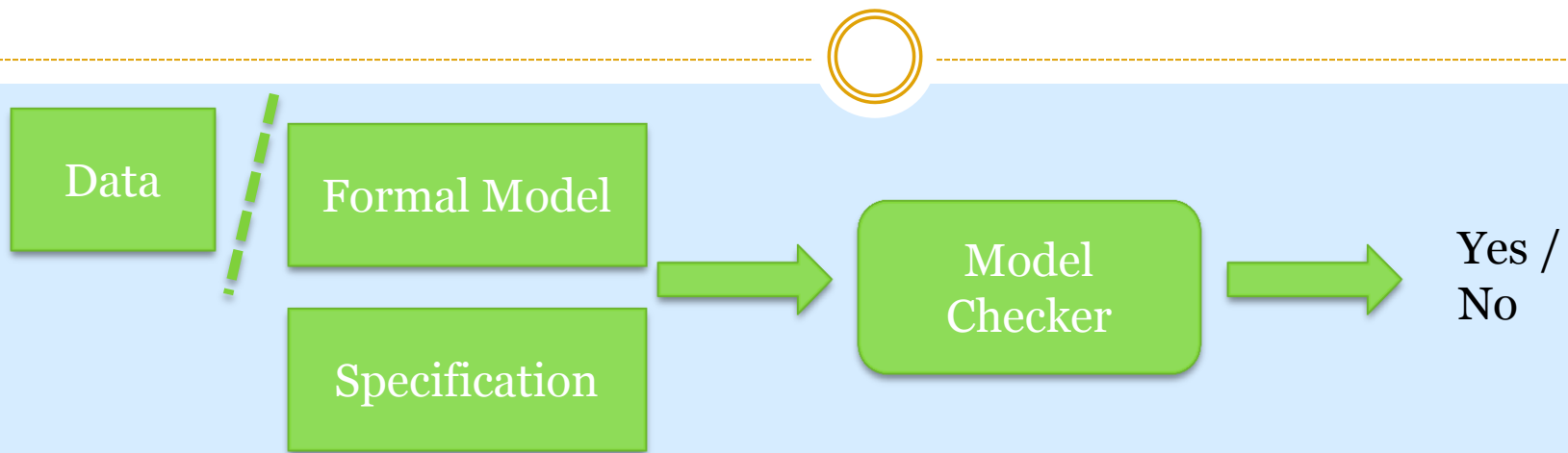
Set of conjectures not known apriori

Difference from Traditional Verification



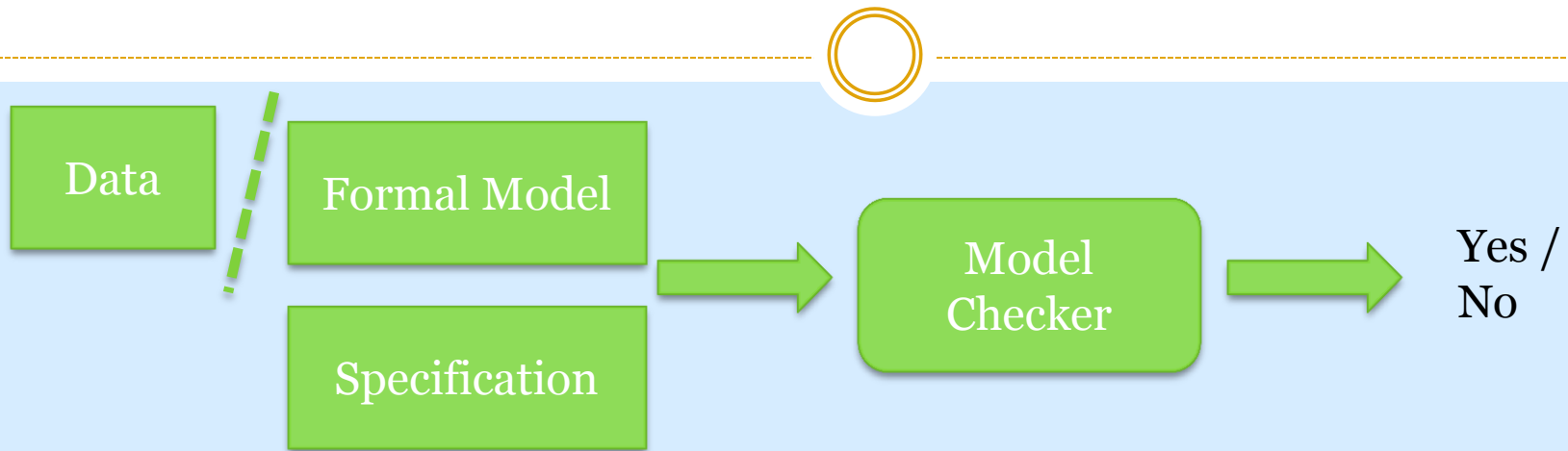
Model Verification	Verifying Conjectures
Formal specification known from design	Set of conjectures not known a priori
Succinct model – may be megabytes	Large data – may be petabytes

Difference from Traditional Verification



Model Verification	Verifying Conjectures
Formal specification known from design	Set of conjectures not known a priori
Succinct model – may be megabytes	Large data – may be petabytes
Model is perfect	Data is not perfect - noisy

Difference from Traditional Verification



Model Verification	Verifying Conjectures
Formal specification known from design	Set of conjectures not known a priori
Succinct model – may be megabytes	Large data – may be petabytes
Model is perfect	Data is not perfect - noisy
Specification is not necessarily robust	Conjectures must be robust

Verifying Extreme Scale Data against Robust Specifications



- Really extreme scale data!
 - Gigabytes (in our experiments) to Petabytes and Exabytes
- Robust conjectures
 - Specification would be true even if data was slightly perturbed
 - Not all of data is often crucial to prove the conjecture
- Next-generation exascale hardware
 - No hope of verifying petabyte of data on single processor
 - ✦ Energy needed per unit computation will not be safely dissipated.

Idea 1: Compress Data Before Verification



- Use metric projection to reduce the dimension of the data before verification starts
- Can reduce dimensions by losing information
 - Map 1000 dimensional data to 1 dimension
 - ✦ $F(x_1, \dots, x_{1000}) = x_1$
 - ✦ Useful in Iterative Relaxation Abstraction, HSCC 2007,2008
 - Not useful in general

Idea 1: Compress Data Before Verification



- Use metric projection to reduce the dimension of the data before verification starts
- Can reduce dimensions by losing information
 - Map 1000 dimensional data to 1 dimension
 - ✦ $F(x_1, \dots, x_{1000}) = x_1$
 - ✦ Useful in Iterative Relaxation Abstraction, HSCC 2007,2008
 - Not useful in general
 - Map 1000 dimensional data to 200 dimensions in complex nonlinear ways
 - ✦ $F(x_1, \dots, x_{1000}) = (x_1^2 + x_2^2 + \dots + x_{1000}^2)$
 - ✦ Too complicated to analyze
 - ✦ May be useful in future but we do not know

Linear Projection



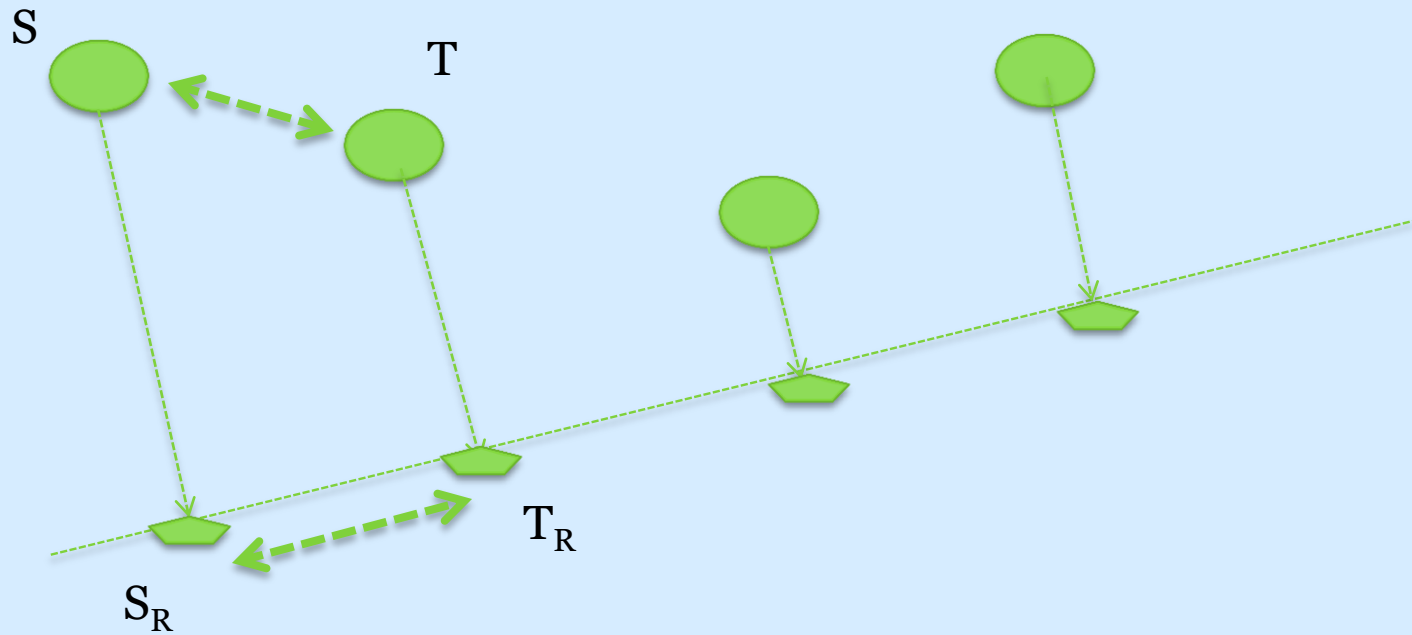
- Compress data by linear projection
- Take data D and a suitable matrix M
- Reduced data $D_R = M * D$
 - Matrix multiplication is easy to parallelize
- In general, no guarantees that relate D to D_R .

Randomized Linear Projections



- If M is chosen randomly from a set of well-formed distance preserving matrices, D_R and D are closely related.
- In particular, distances between points in D and distances between corresponding points in D_R are almost the same.

Distance Preserving Projections



$$(1-e) ||S-T||^2 \leq ||S_R-T_R||^2 \leq (1+e) ||S-T||^2$$

Distance between points are approximately preserved

What is a good choice for M?



- Any random matrix with entries
 - $\sqrt{3/n}$ with probability $1/6$
 - $-\sqrt{3/n}$ with probability $1/6$
 - 0 with probability $2/3$

- Where, n is the dimension of the original data.

Result



- To verify $D(i,j) < c$,
- It is sufficient to verify
 - $\sqrt{(1-e)} ||MD(i) - M(D(i)_{j/o})|| < \sqrt{(1+e)} ||M(D(i)_{j/c}) - M(D(i)_{j/o})||$
 - $D(i)_{j/c}$ denotes the replacement of the j th component of $D(i)$ by c .
 - Note that every term in the second expression is from the lower dimension data set MD

Proof Sketch



- To verify $D(i,j) < c$, it is sufficient to verify
 - $\sqrt{(1-e)} ||MD(i) - M(D(i)_{j/o})|| < \sqrt{(1+e)} ||M(D(i)_{j/c}) - M(D(i)_{j/o})||$
- Proof:
- $||MD(i) - M(D(i)_{j/o})||^2$
- $< (1+e) ||D(i) - (D(i)_{j/o})||^2$... Distance Preserving Projection
- $= (1+e)D(i,j)^2$... Choice of Projected Points

Proof Sketch



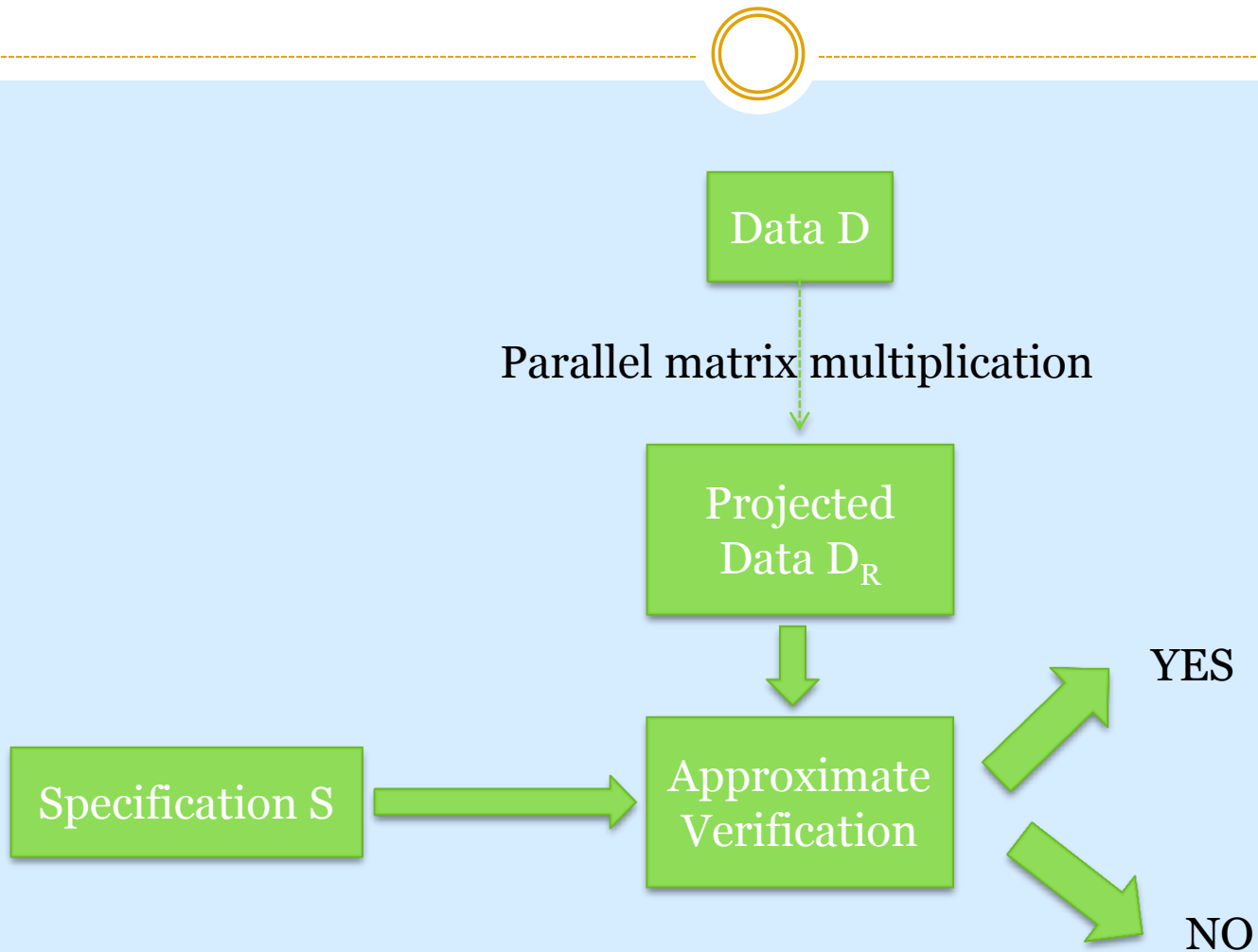
- To verify $D(i,j) < c$, it is sufficient to verify
 - $\sqrt{(1-e)} ||MD(i) - M(D(i)_{j/o})|| < \sqrt{(1+e)} ||M(D(i)_{j/c}) - M(D(i)_{j/o})||$
- Proof:
- $||MD(i)_{j/c} - M(D(i)_{j/o})||^2$
- $> (1-e) ||D(i)_{j/c} - (D(i)_{j/o})||^2$... Distance preserving projection
- $= (1-e) c^2$... Suitable choice of points

Proof Sketch

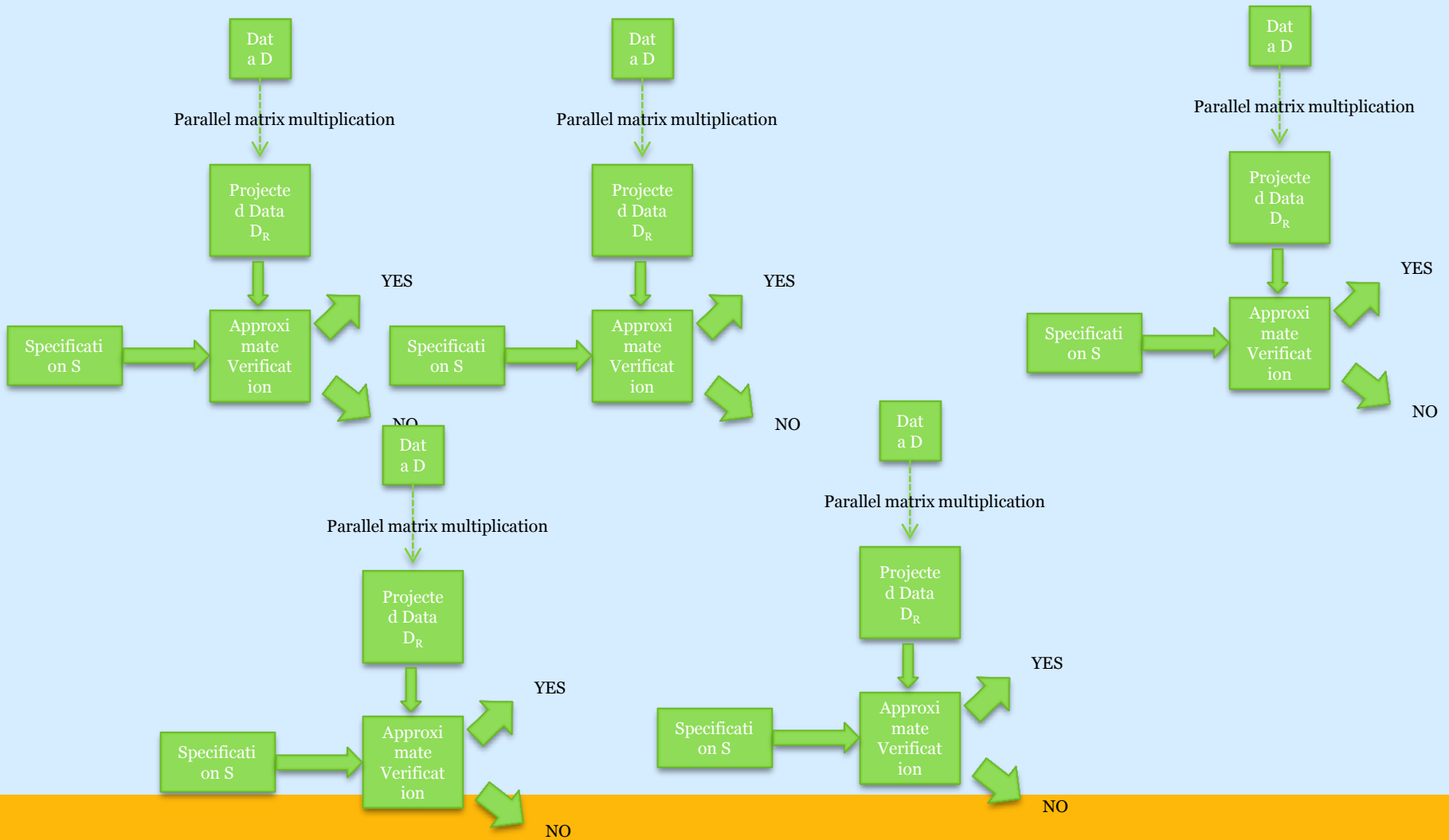


- To verify $D(i,j) < c$, it is sufficient to verify
 - $\sqrt{(1-e)} ||MD(i) - M(D(i)_{j/o})|| < \sqrt{(1+e)} ||M(D(i)_{j/c}) - M(D(i)_{j/o})||$
- Proof:
- $||MD(i) - M(D(i)_{j/o})||^2 < (1+e) ||D(i) - (D(i)_{j/o})||^2 = (1+e)D(i,j)^2$
- $||MD(i)_{j/c} - M(D(i)_{j/o})||^2 > (1-e) ||D(i)_{j/c} - (D(i)_{j/o})||^2 = (1-e) c^2$
- Thus, $\sqrt{(1-e)} ||MD(i) - M(D(i)_{j/o})|| < \sqrt{(1+e)} ||M(D(i)_{j/c}) - M(D(i)_{j/o})||$
- $\Rightarrow \sqrt{(1-e)} \sqrt{(1+e)} D(i,j) < \sqrt{(1+e)} \sqrt{(1-e)} c$
- $\Rightarrow D(i,j) < c$

Overall Flow



Parallel Algorithm



Nature of Parallelization



- No communication between processes
- Only need to communicate projected data to worker nodes
- Massively parallel
- Suitable for Exascale computing!
- Will not extend to crisp non-robust specifications 😊

Challenge Problem: Pairs Trading



- Find a pair of securities A and B that are “correlated”.
- When the price of A is lower than the price of B,
 - Buy A and sell B.
- When the price of A is higher than the price of B,
 - Buy B and sell A.

Several Design Choices

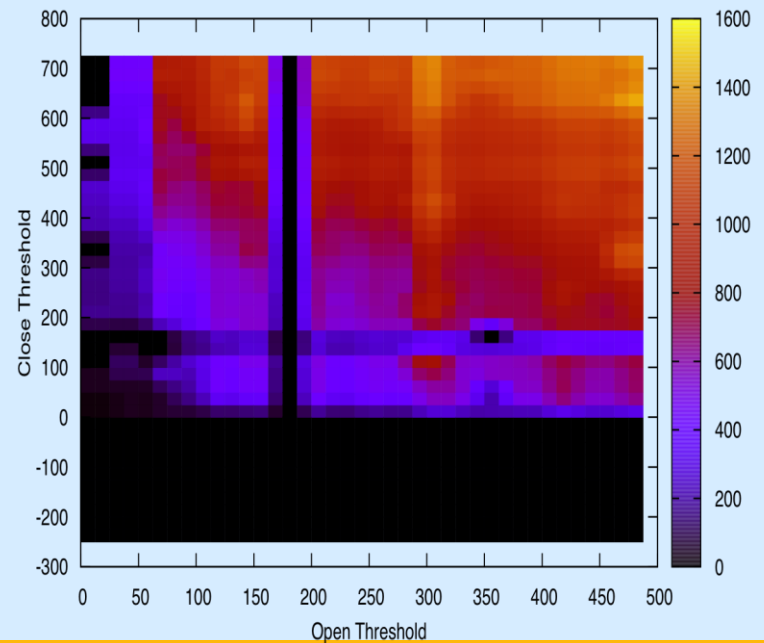


- Which pair of securities?
- How much should the price of A be higher than the price of B?
 - To sell A and buy B
- How much should the price of A be lower than the price of B?
 - To sell B and buy A
- To maximize profits (robustly)

Solution: Simulation



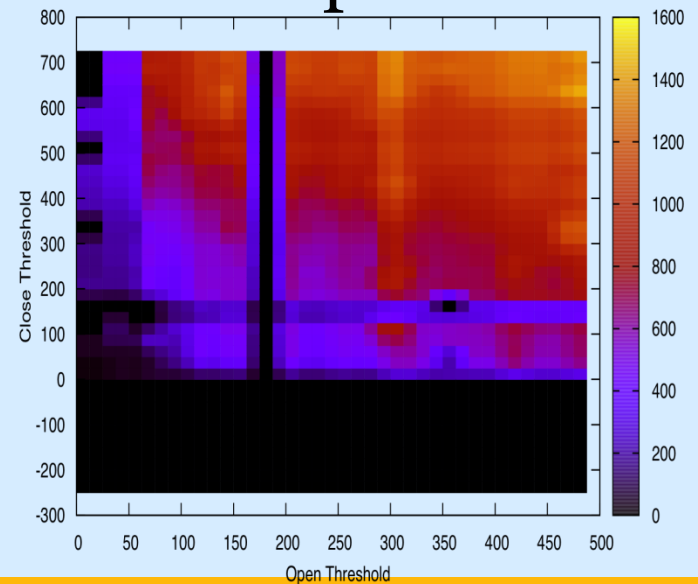
- Simulate the trading algorithm D with different choices
 - D(security1, security2, low, high)
- Observe simulation results and find the most profitable scenario



Solution: Simulation



- Simulate the trading algorithm D with different choices
 - D(security1, security2, low, high)
- Observe simulation results and find the most profitable scenario that is robust to small price fluctuations.



Challenge: Too Much Information



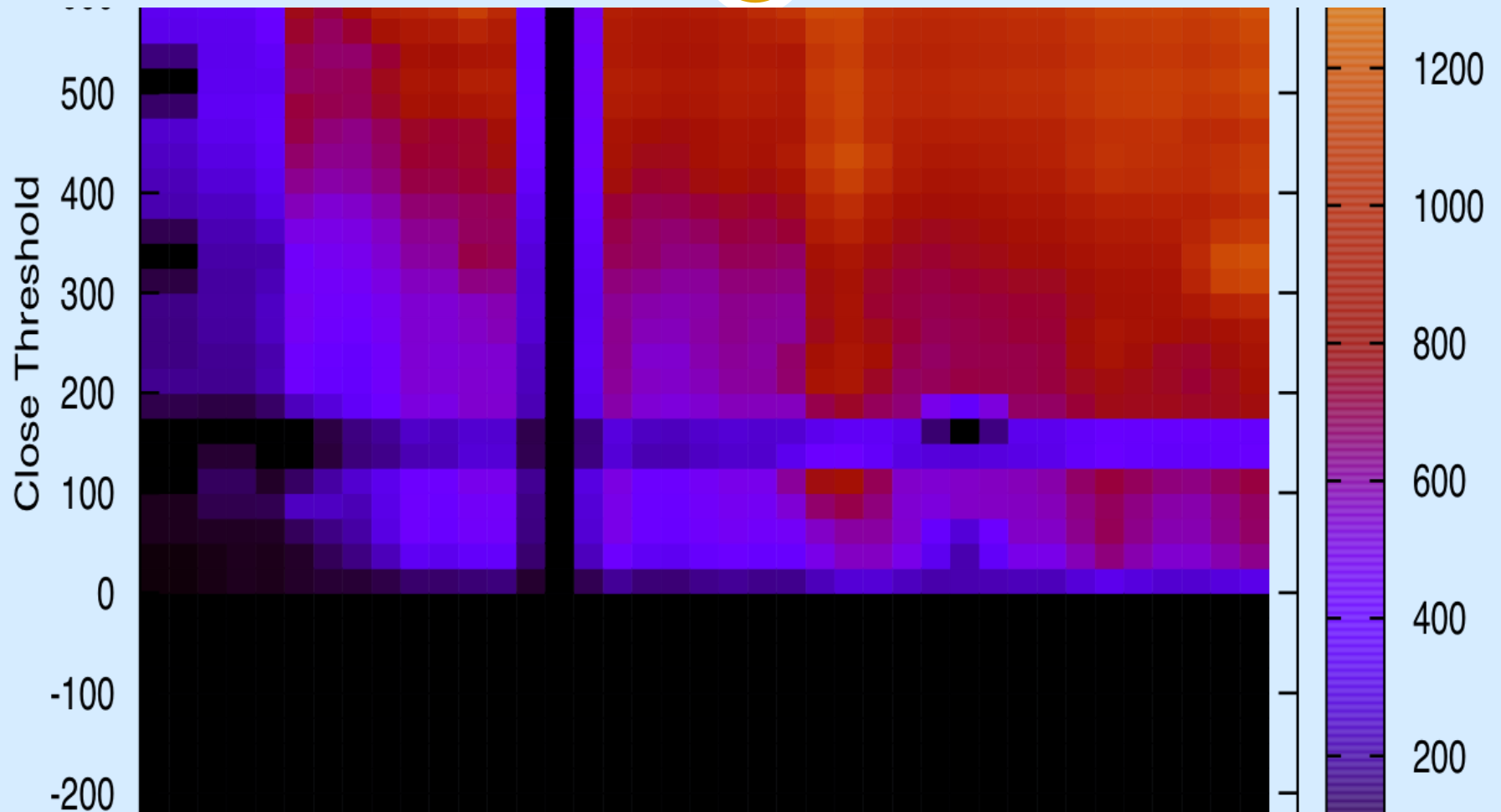
- NASDAQ, NYSE, AMEX: >5000 securities
- #Possible Trading Pairs: >25 million pairs!
- Each pair generates a two-dimensional plot
 - With 1.5 billion data points
 - ✦ Assuming 100 distinct open and close positions simulated
- Too much information to visualize!!
 - ✦ A report with all these images would have a size of about 2 GB!

Conjecture Verification



- We evaluated conjectures of the form
 - Strategy with future A and future B produces profit \$1,000 robustly when using an open and close threshold of X and Y respectively.
 - GPU based implementation of ordinary temporal logic monitoring algorithm.
 - ✦ Each pair of threshold given its own GPU
 - Can print a gigabyte PDF of returns for visualization!!
 - Used logic assertions to stop execution when conjecture was violated.
 - Obtained about 1,000 graphs to visualize and study!

Typical Visualization Obtained



Conclusion and Future Work



- Temporal Logics should be used to understand extreme scale data
- The problem is different than traditional verification
 - Data is noisy
 - Conjectures are often robust
 - Extreme scale data on exascale computing
- Future:
 - Extend theoretical results into an algorithm.
 - Evaluate on open source data
 - ✦ Financial data from global markets
 - ✦ And conjectures on trading strategies.





Presentation Agenda

- Current Dimension Reduction and Visualization Algorithms.
- Use of Temporal Logic to express Expert Conjectures
- Dimension Reduction
- Verification of Expert Conjecture
- Conclusion

Prior Work on Visualization and Dimensionality Reduction

- Dimensionality Reduction Algorithm:
 - Principal Component Analysis
 - Multidimensional Scaling
- Visualization Algorithm: 2-D or 3-D graphical display of multi dimensional data.

A PERIODIC TABLE OF VISUALIZATION METHODS

☼ ☼ ☼ € continuum		Data Visualization Visual representations of quantitative data in schematic form (either with or without axes)										Strategy Visualization The systematic use of complementary visual representations in the analysis, development, formulation, communication, and implementation of strategies in organizations.						☼ ☼ ☼ € graphic facilitation							
☼ ☼ ☼ Tb table		☼ ☼ ☼ Ga cartesian coordinates		Information Visualization The use of interactive visual representations of data to amplify cognition. This means that the data is transformed into an image, it is mapped to screen space. The image can be changed by users as they proceed working with it.										Metaphor Visualization Visual Metaphors position information graphically to organize and structure information. They also convey an insight about the represented information through the key characteristics of the metaphor that is employed.						☼ ☼ ☼ Me meeting trace	☼ ☼ ☼ Mm metro map	☼ ☼ ☼ Tm temple	☼ ☼ ☼ St story template	☼ ☼ ☼ Tr tree	☼ ☼ ☼ €t cartoon
☼ ☼ ☼ Pi pie chart		☼ ☼ ☼ L line chart		Concept Visualization Methods to elaborate (mostly) qualitative concepts, ideas, plans, and analyses.										Compound Visualization The complementary use of different graphic representation formats in one single schema or frame						☼ ☼ ☼ Co communication diagram	☼ ☼ ☼ Fp flight plan	☼ ☼ ☼ Cs concept skeleton	☼ ☼ ☼ Br bridge	☼ ☼ ☼ Fu funnel	☼ ☼ ☼ Ri rich picture
☼ ☼ ☼ B bar chart		☼ ☼ ☼ Ac area chart		☼ ☼ ☼ R radar chart cobweb	☼ ☼ ☼ Pa parallel coordinates	☼ ☼ ☼ Hy hyperbolic tree	☼ ☼ ☼ Cy cycle diagram	☼ ☼ ☼ T timeline	☼ ☼ ☼ Ve veins diagram	☼ ☼ ☼ Mi mindmap	☼ ☼ ☼ Sq square 90° oppositions	☼ ☼ ☼ Cc concentric circles	☼ ☼ ☼ Ar argument slide	☼ ☼ ☼ Sw swim lane diagram	☼ ☼ ☼ Gc gantt chart	☼ ☼ ☼ Pm perspectives diagram	☼ ☼ ☼ D dilemma diagram	☼ ☼ ☼ Pr parameter rater	☼ ☼ ☼ Kn knowledge map						
☼ ☼ ☼ Hi histogram		☼ ☼ ☼ Sc scatterplot		☼ ☼ ☼ Sa saankey diagram	☼ ☼ ☼ In information lense	☼ ☼ ☼ E entity relationship diagram	☼ ☼ ☼ Pt petri net	☼ ☼ ☼ Fl flow chart	☼ ☼ ☼ Cl clustering	☼ ☼ ☼ Lc layer chart	☼ ☼ ☼ Py minto pyramid technique	☼ ☼ ☼ Ce cause-effect chains	☼ ☼ ☼ Ti toulinin map	☼ ☼ ☼ Dt decision tree	☼ ☼ ☼ Cp cpm critical path method	☼ ☼ ☼ Cf concept fan	☼ ☼ ☼ Co concept map	☼ ☼ ☼ Ic iceberg	☼ ☼ ☼ Lm learning map						
☼ ☼ ☼ Tk tukey box plot		☼ ☼ ☼ Sp spectrogram		☼ ☼ ☼ Da data map	☼ ☼ ☼ Tp treemap	☼ ☼ ☼ Cn cone tree	☼ ☼ ☼ Sy system dyn./ simulation	☼ ☼ ☼ Df data flow diagram	☼ ☼ ☼ Se semantic network	☼ ☼ ☼ So soft system modeling	☼ ☼ ☼ Sn synergy map	☼ ☼ ☼ Fo force field diagram	☼ ☼ ☼ Ib ibn argumentation map	☼ ☼ ☼ Pr process event chains	☼ ☼ ☼ Pe pert chart	☼ ☼ ☼ Ev evocative knowledge map	☼ ☼ ☼ V vee diagram	☼ ☼ ☼ Hh heaves 'n' hell chart	☼ ☼ ☼ I infomural						

- ☼ ☼ ☼** **Cy** **Process Visualization**
- ☼ ☼ ☼** **Hy** **Structure Visualization**
- ☼ ☼ ☼** **☼ ☼ ☼** **Overview**
- ☼ ☼ ☼** **☼ ☼ ☼** **Detail**
- ☼ ☼ ☼** **☼ ☼ ☼** **Detail AND Overview**
- < >** **Divergent thinking**
- > <** **Convergent thinking**

Note: Depending on your location and connection speed it can take some time to load a pop-up picture.
 © Ralph Lengler & Martin J. Eppler, www.visual-literacy.org

version 1.5

☼ ☼ ☼ Su supply demand curve	☼ ☼ ☼ Pc performance charting	☼ ☼ ☼ St strategy map	☼ ☼ ☼ Oc organisation chart	☼ ☼ ☼ Ho house of quality	☼ ☼ ☼ Fd feedback diagram	☼ ☼ ☼ Ft failure tree	☼ ☼ ☼ Mq magic quadrant	☼ ☼ ☼ Ld life-cycle diagram	☼ ☼ ☼ Po porter's five forces	☼ ☼ ☼ S s-cycle	☼ ☼ ☼ Sm stakeholder map	☼ ☼ ☼ Is ishikawa diagram	☼ ☼ ☼ Tc technology roadmap
☼ ☼ ☼ Ed edgeworth box	☼ ☼ ☼ Pf portfolio diagram	☼ ☼ ☼ Sg strategic game board	☼ ☼ ☼ Mz mintsberg's organigraph	☼ ☼ ☼ Z zwicky's morphological box	☼ ☼ ☼ Ad affinity diagram	☼ ☼ ☼ De decision discovery diagram	☼ ☼ ☼ Bm bcg matrix	☼ ☼ ☼ Stc strategy canvas	☼ ☼ ☼ Vc value chain	☼ ☼ ☼ Hy hype-cycle	☼ ☼ ☼ Sr stakeholder rating map	☼ ☼ ☼ Ta taps	☼ ☼ ☼ Sd spray diagram

Figure 7: Periodic Table of Visualization algorithm by Visual-literacy