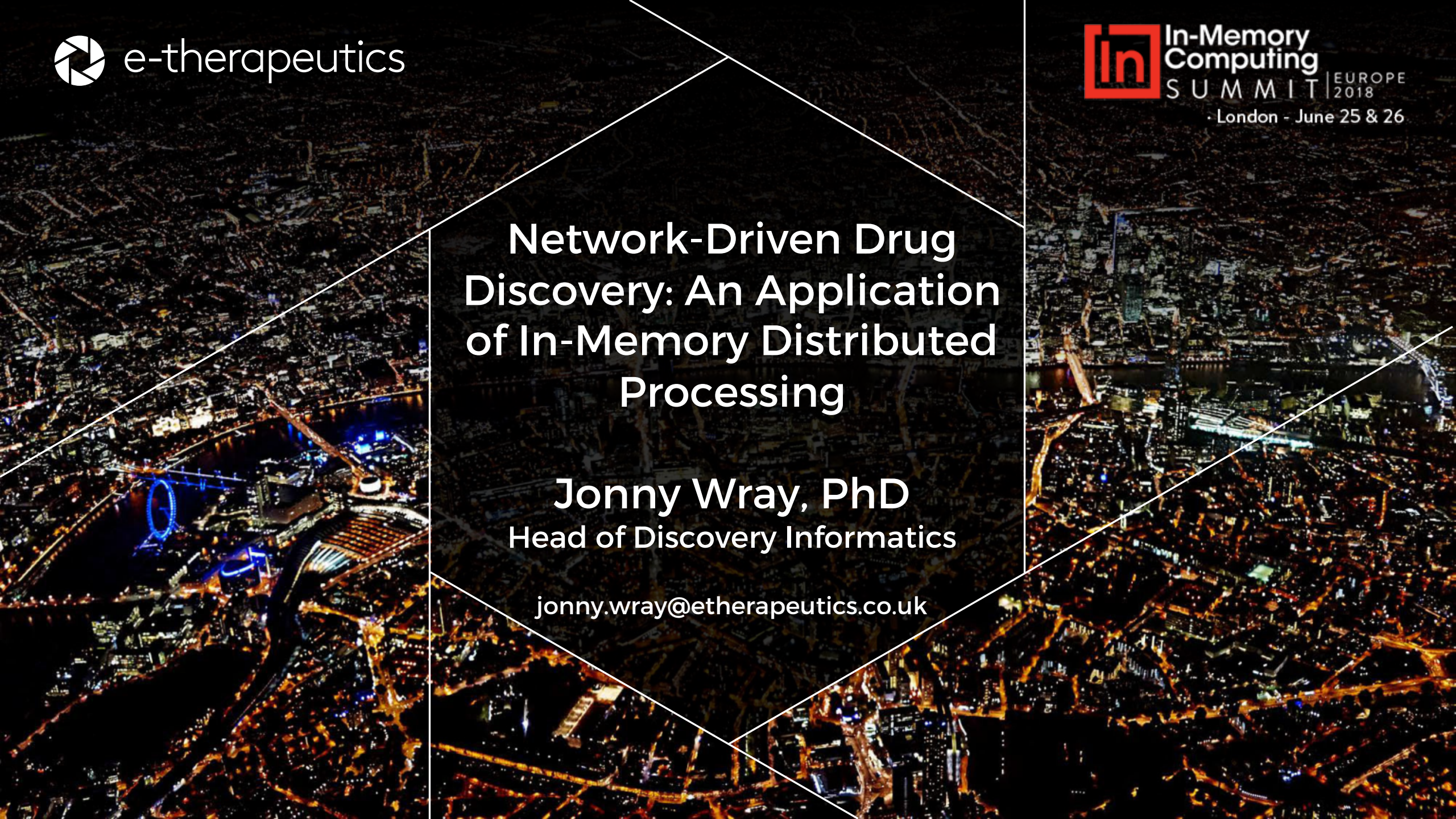

e-therapeutics
In-Memory Computing SUMMIT EUROPE 2018
London - June 25 & 26
Network-Driven Drug Discovery: An Application of In-Memory Distributed Processing
Jonny Wray, PhD
Head of Discovery Informatics
jonny.wray@etherapeutics.co.uk

Who We Are

e-therapeutics

Pioneers of the next frontier in drug discovery

A unique drug discovery company headquartered in Oxford, UK, and listed on the AIM market in London (ETX.L.)

Achieve diverse and high-performing drug hits quickly and cost efficiently

Demonstrated success in 12 diverse areas of biology, from oncology to immunology and neurodegeneration

Architects of an original, proprietary NETWORK-DRIVEN DRUG DISCOVERY platform

A suit of powerful, custom computational tools that tap into large-scale, proprietary databases

Applies network science to tackle complex diseases

Employs data mining, machine learning, artificial intelligence, optimisation and network analysis

A professional business partner: collaborations or out-licensing self-discovered assets

Current focus on preclinical discovery programmes in immuno-oncology

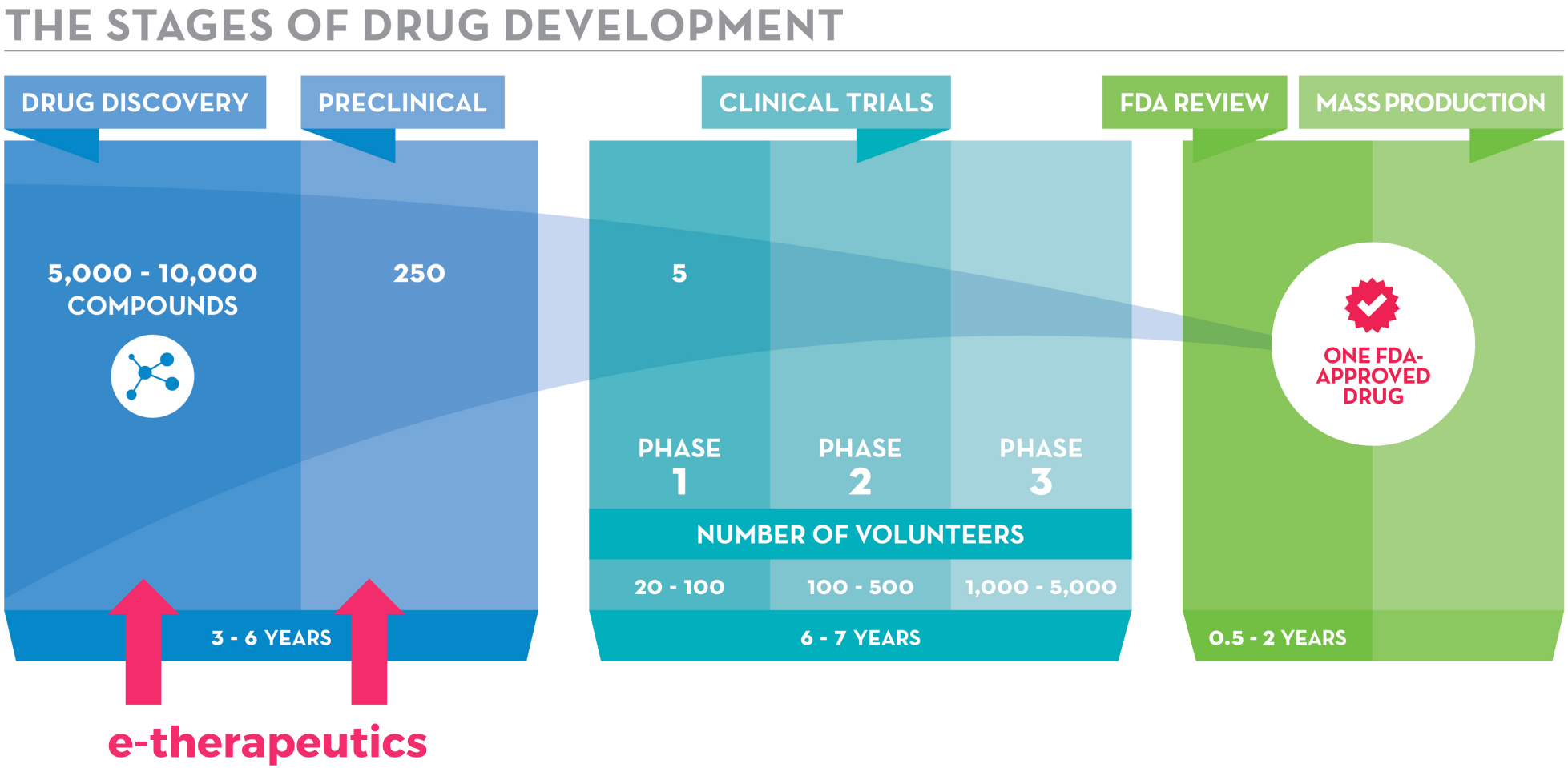
Offering a Hedgehog pathway modulation programme for out-licensing

Seeking collaborations to apply our Network-Driven Drug Discovery platform to disease areas of mutual interest

Drug Discovery and Development

Where e-therapeutics Operates

e-therapeutics

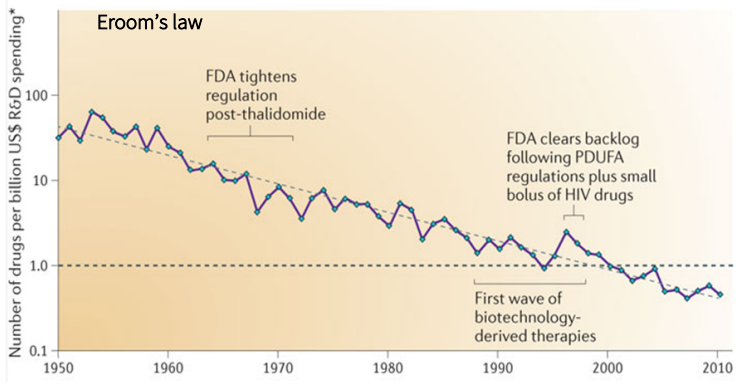


Drug Discovery Process Analysis

An Industry Ripe for Innovation

e-therapeutics

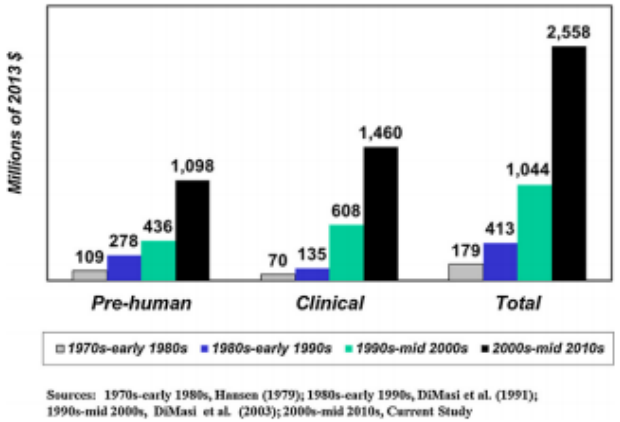
Industry productivity is decreasing



*Inflation-adjusted

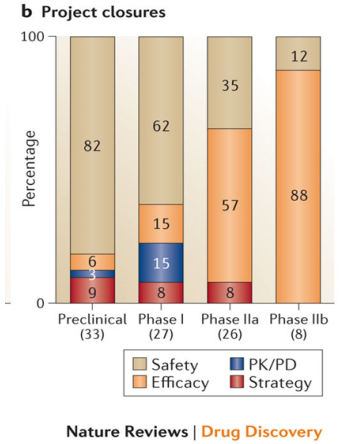
Source: Jack W. Scannell et. al., *Nature Reviews Drug Discovery* 11, 191-200 (March 2012).

Costs are massive and increasing



Source: DiMasi et. al., *Journal of Health Economics* 47, 20-33 (2016)

Late stage failures due to efficacy



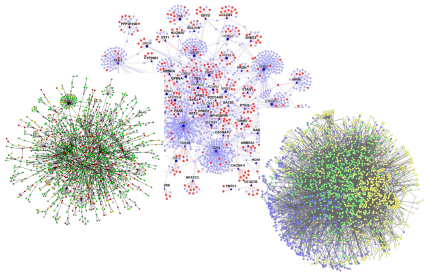
Source: Cook et. al., *Nature Reviews Drug Discovery* 13, 419-431 (2014)

Network Biology

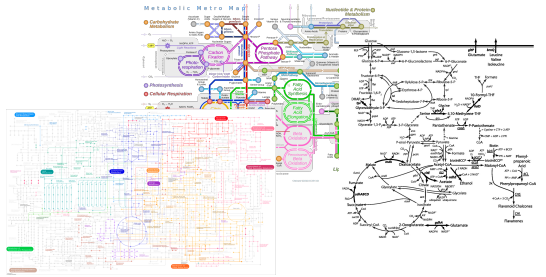
The Cell as a Network

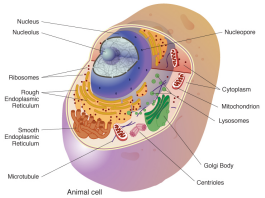
e-therapeutics

Protein-Protein Interaction Network

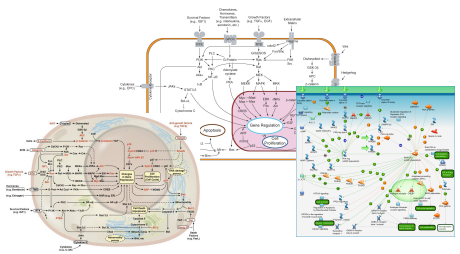


Metabolic Network

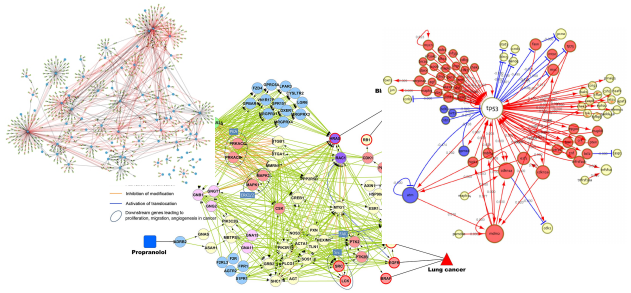




Signal Transduction Pathways



Gene Regulatory Network

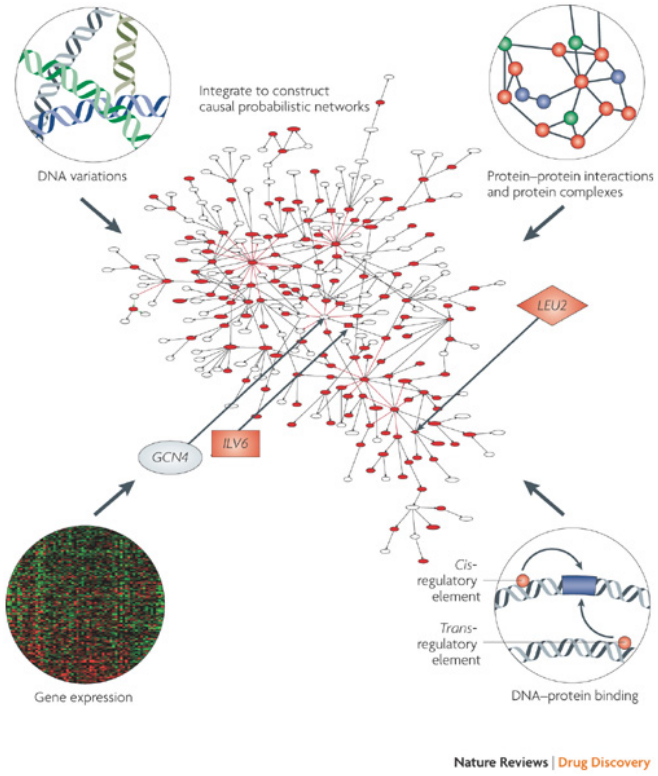


Network Biology

Disease Behavior is an Emergent Property of Molecular Networks

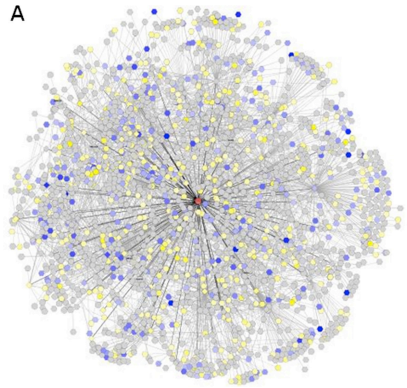
e-therapeutics

Dysregulated network module identification



Source: Schadt, E., et al. *Nature Reviews Drug Discovery* (2009)

Pathological interaction identification in Huntington's disease

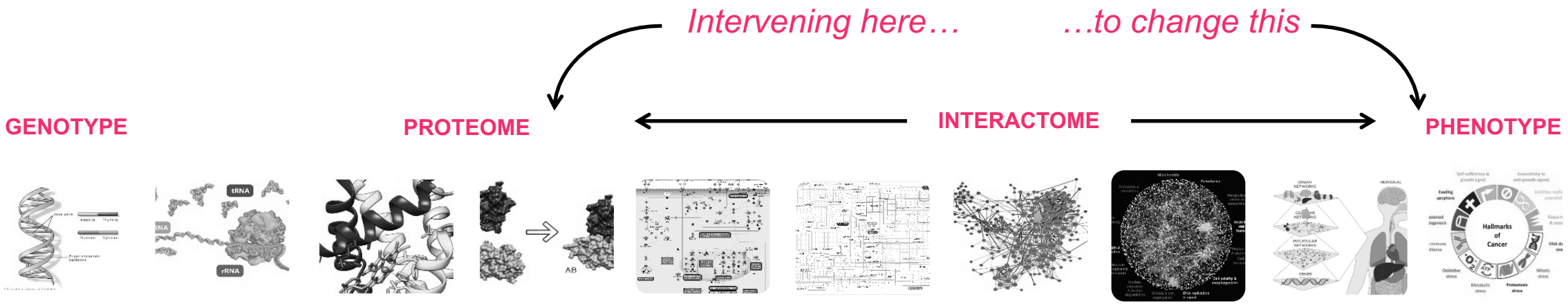


Source: Tourette, C., et al. *Journal Biological Chemistry* (2014)

Network Biology

Drugs Need to Alter Phenotype

e-therapeutics

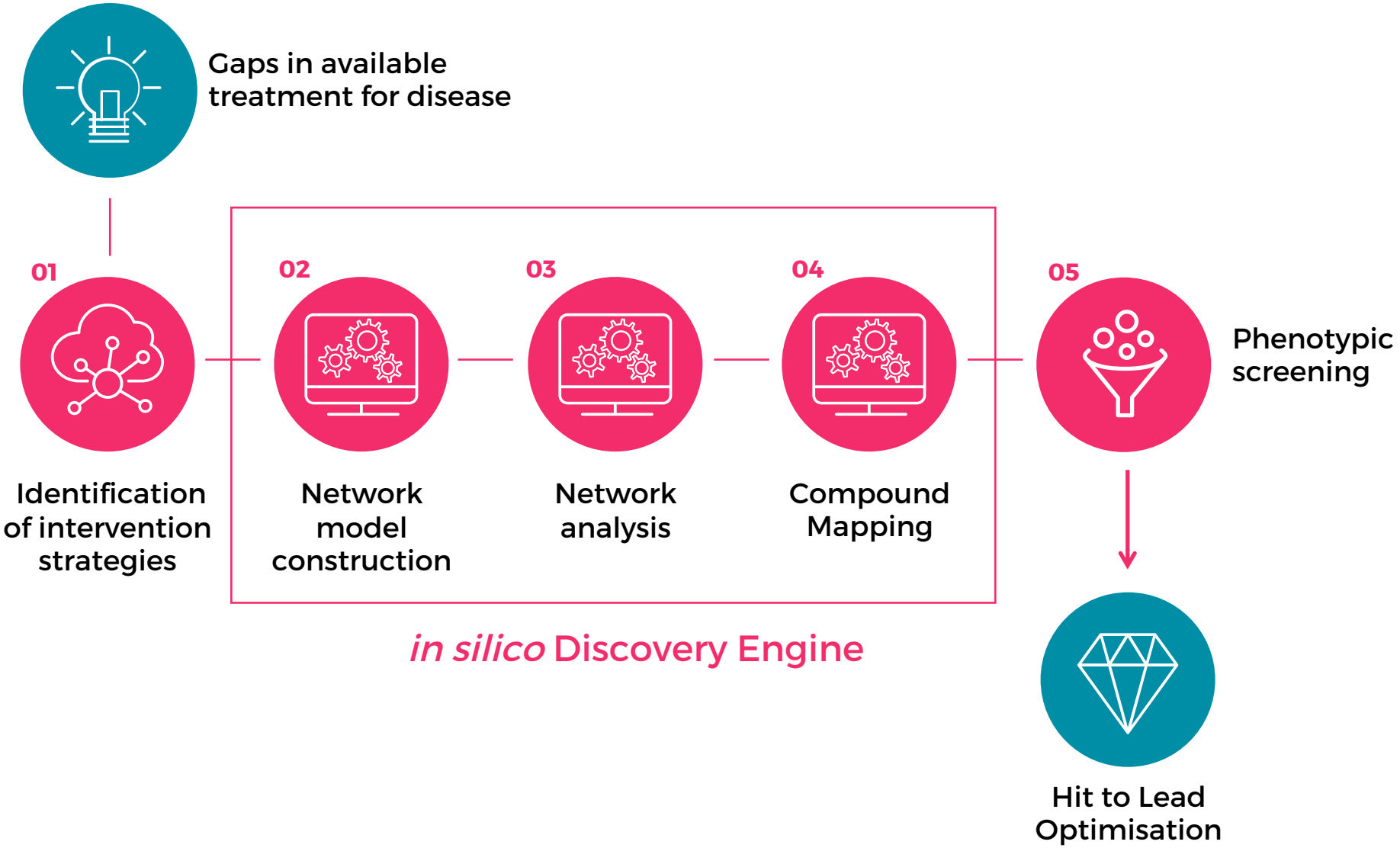


- Phenotype is an emergent property of cellular networks
- Networks can be viewed as the mechanistic bridge between the molecular and the phenotype

Network-driven Drug Discovery Process

From Hypothesis to Compound Testing in 9 Months

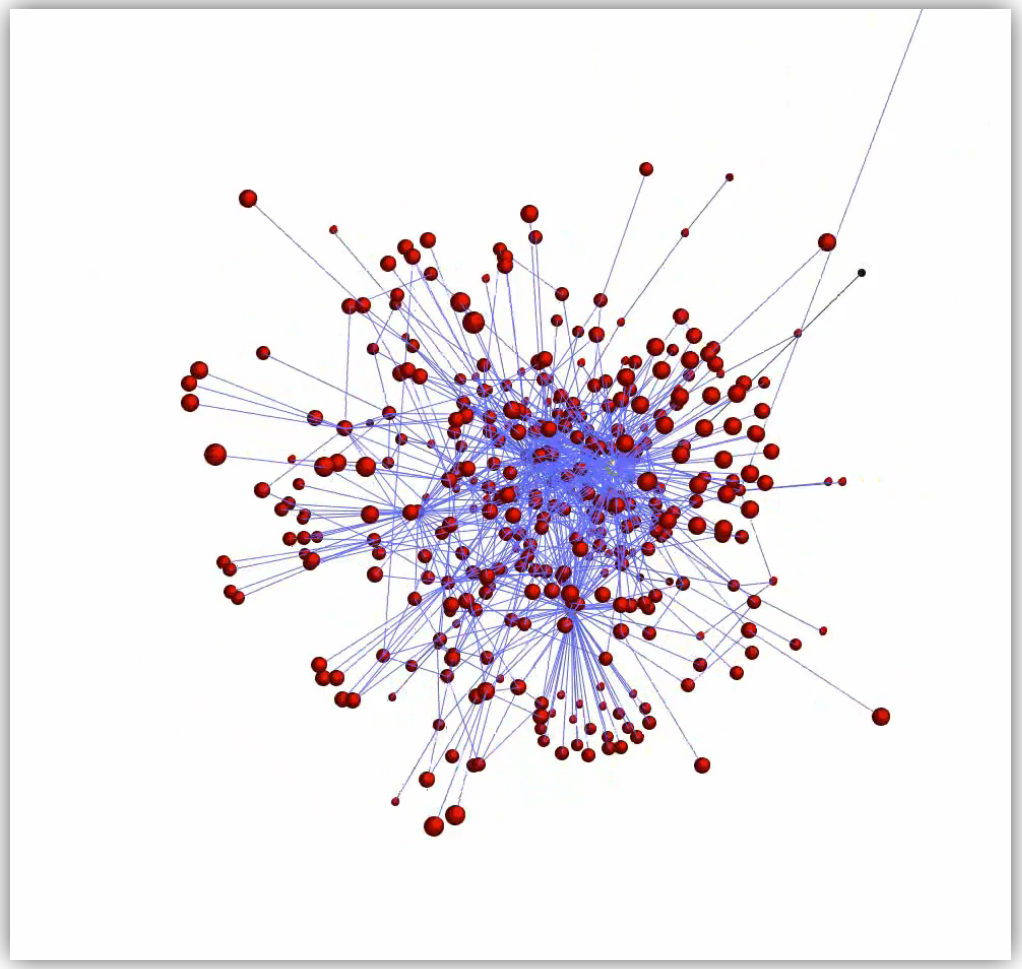
e-therapeutics



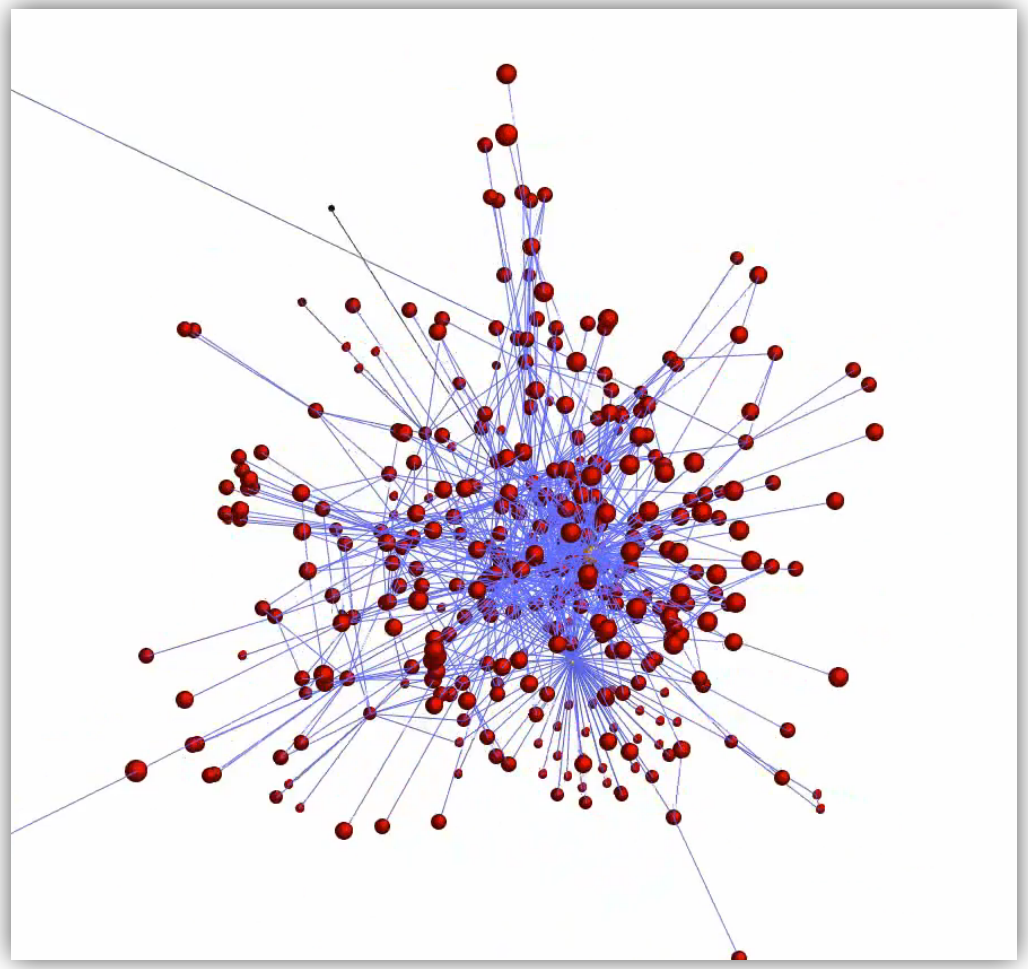
Disease Network Perturbation Analysis

Core Foundation of Discovery Process

e-therapeutics



Networks are robust to random perturbation…



… but susceptible to targeted perturbation

Random Perturbation: YouTube Video

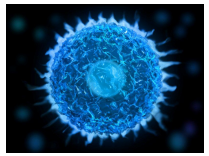
Targeted Perturbation: YouTube Video

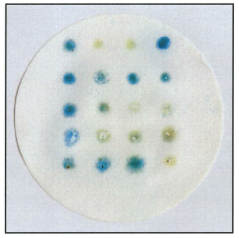
Network Model Construction

Biological Inverse Problem

e-therapeutics

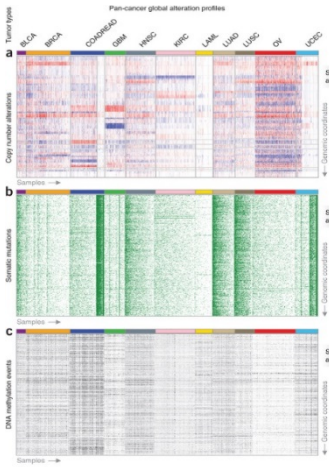
Cells

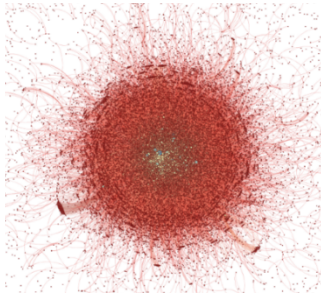




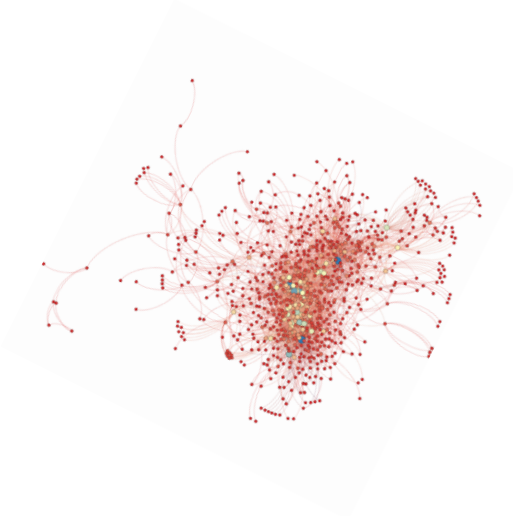
Healthy
Vs
Diseased

Measurements





Network Model of Disease



Network Model Construction

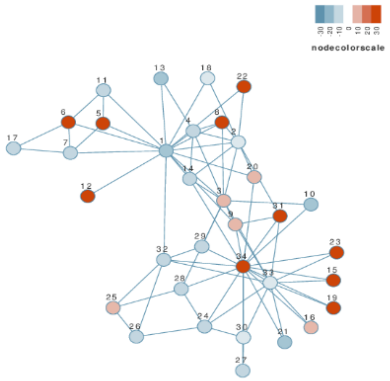
Computational Issues

‘Active Module’ Detection: Integration of molecular profiles with cellular interactions

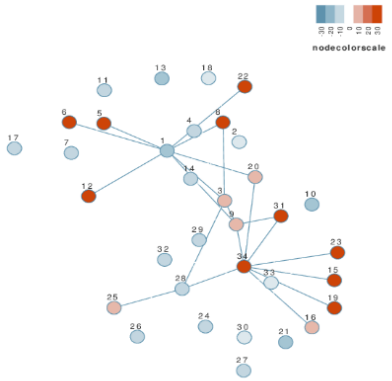
- Formulated as an optimization problem – find high scoring sub-network
- Heuristic approaches: greedy search
- Exact approach: Prize-collecting Steiner tree formulated as linear programming problem

Maximum weight connected subgraph problem

Random Scores Across Graph

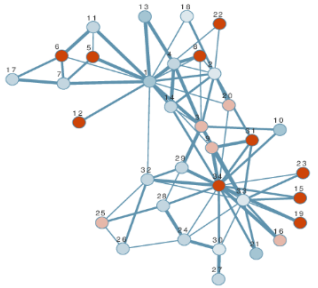


Maximum Scoring Subgraph

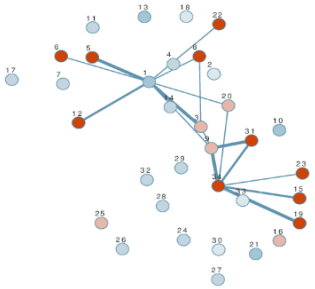


Prize-collecting Steiner tree problem

Random Profits And Edge Costs Across Graph



Prize Collecting Steiner Tree Solution

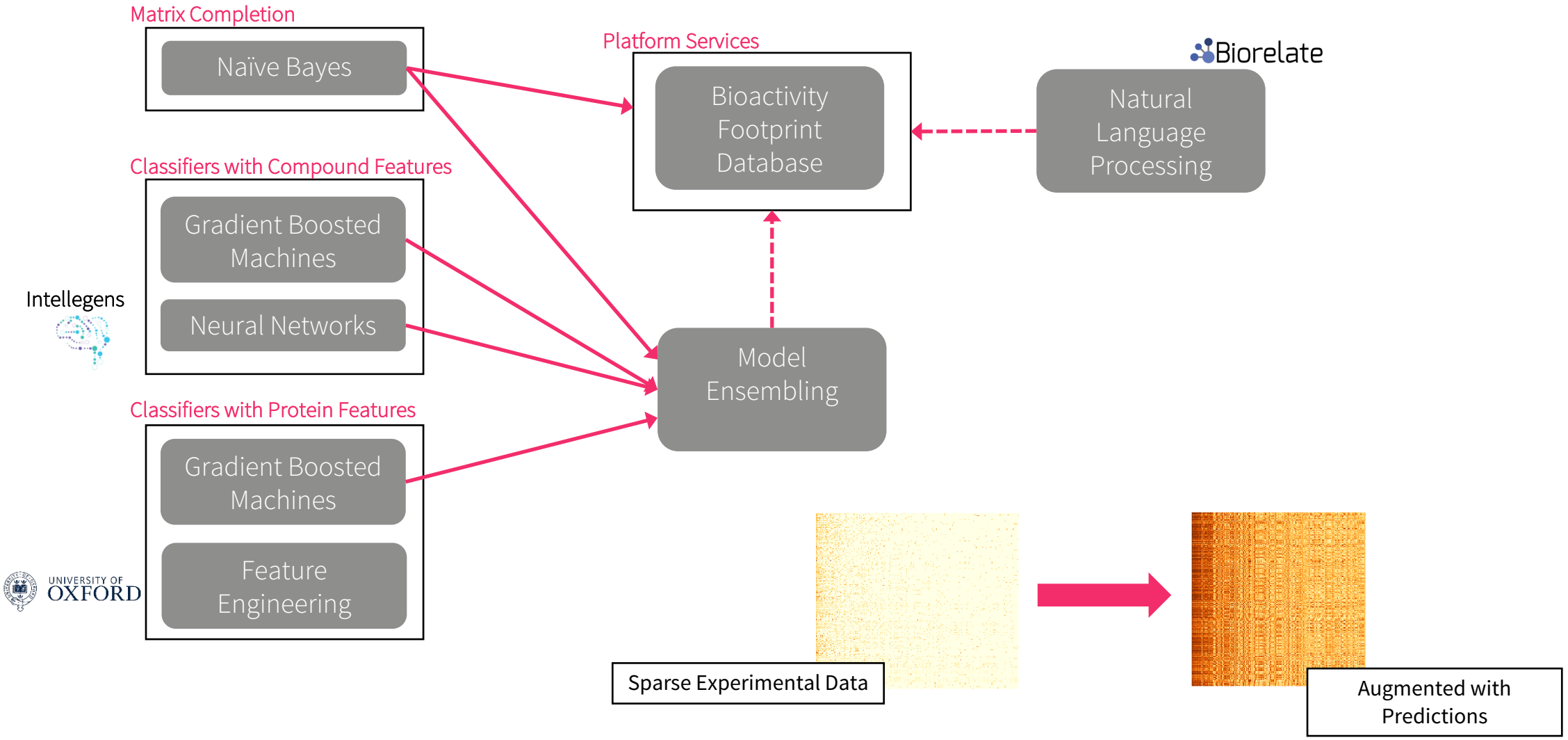


- Computationally expensive to solve: We use IBM CPLEX Optimizer
- Multiple optimal, and suboptimal, solutions: Steiner Forests
- Future challenges: move from gene based (22k) to protein based (250k – 1.5M) networks

Compound Mapping

Data Augmentation With Machine Learning

e-therapeutics



Compound Mapping

Computational Issues

e-therapeutics

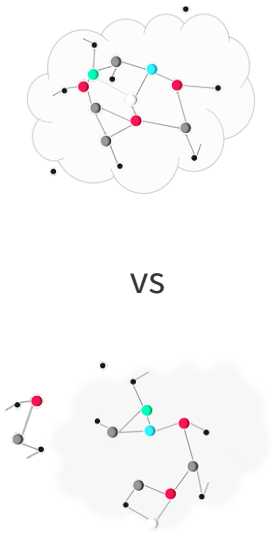
- **Requirements**
  - Heterogenous data: hard to make sampled data set results generalize to full data set
  - Speed: slow training times kill exploratory development of machine learning solutions
  - In memory requirements
    - Full matrix: 15M (compounds) x 20k (proteins)
    - ~1200G with Java float
    - Sensible data filtering: ~300G

- **Solution Used**
  - H20.ai:
    - “H2O is an open source, in-memory, distributed, fast, and scalable machine learning and predictive analytics platform that allows you to build machine learning models on big data and provides easy productionalization of those models in an enterprise environment.”

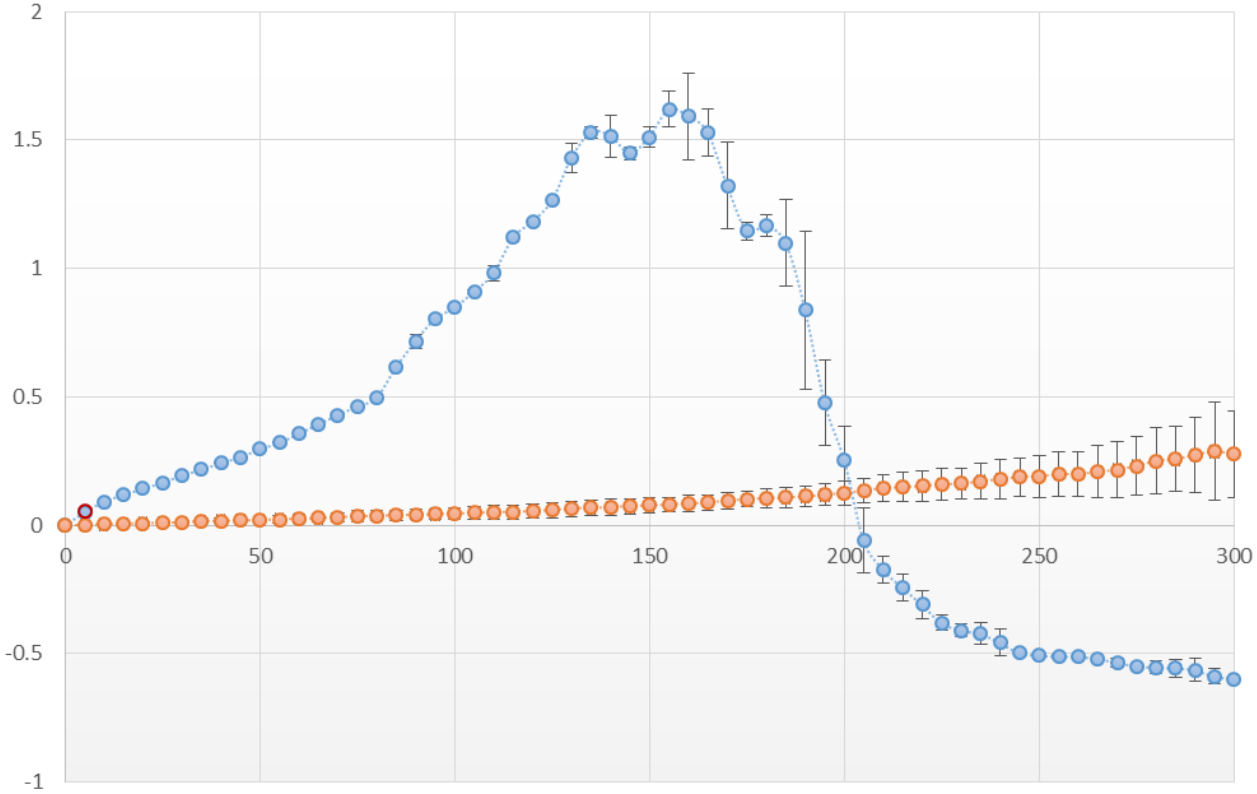
  - Can deal with machine learning on full data set in-memory on our hardware (distributed 512G grid)
  - Required algorithms implemented
  - Data scientists prefer the environment over Spark

Network Analysis

Error vs Attack Tolerance: Biological Networks are Robust



$$Impact = \Delta(Avg.Shortest\ Path)$$



- Albert, R., H. Jeong, and A. L. Barabasi. 2000. “Error and Attack Tolerance of Complex Networks.” *Nature* 406 (6794): 378–82.

Network Analysis

Algorithms

e-therapeutics

Core algorithms used in drug discovery process

- All can be formulated as embarrassingly parallel problems

- Perturbation Analysis
  - Sequentially remove nodes from a network and measure change in network structure
  - Generate data for random vs targeted comparison
  - Used to calibrate other analysis for specific networks – identifies region of random effect

- Impact Maximization
  - Find the optimal set of nodes (proteins) that maximally disrupt a network

- Compound Impact Ranking
  - Rank all entries in our compound database by their impact on a network

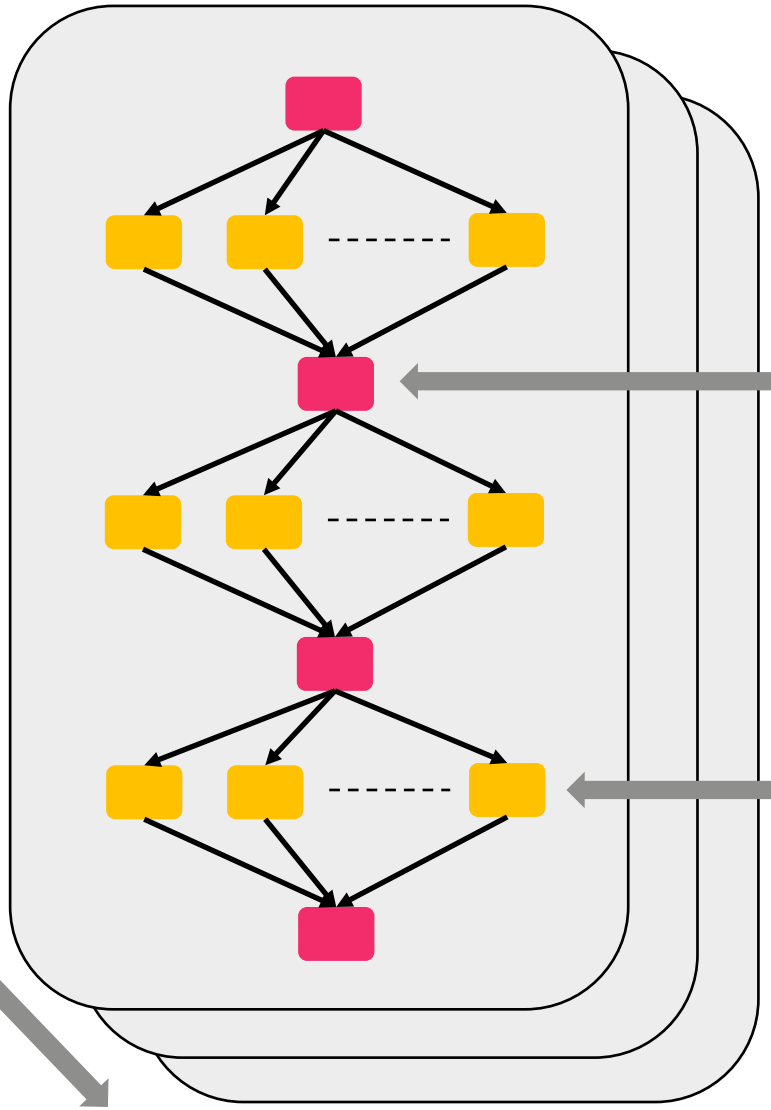
GridGain (Ignite) compute grid

- Infrastructure for parallel **distributed** compute
- Map-reduce or fork-join extended from multiple threads to multiple JVMs and physical machines
- Hadoop:
  - Standard map-reduce framework (when we implemented)
  - Focused on massive data sets - not in-memory – which isn't our situation
  - Batch focused – key requirement was for on-line, user triggered processing

Distributed Fork-Join or Simple Map-Reduce

Generic Algorithm

e-therapeutics



Master node

Worker nodes – distributed across multiple machines

Compute task:

- divide into multiple jobs
- collate results from multiple jobs

Compute jobs: perform calculations on isolated data

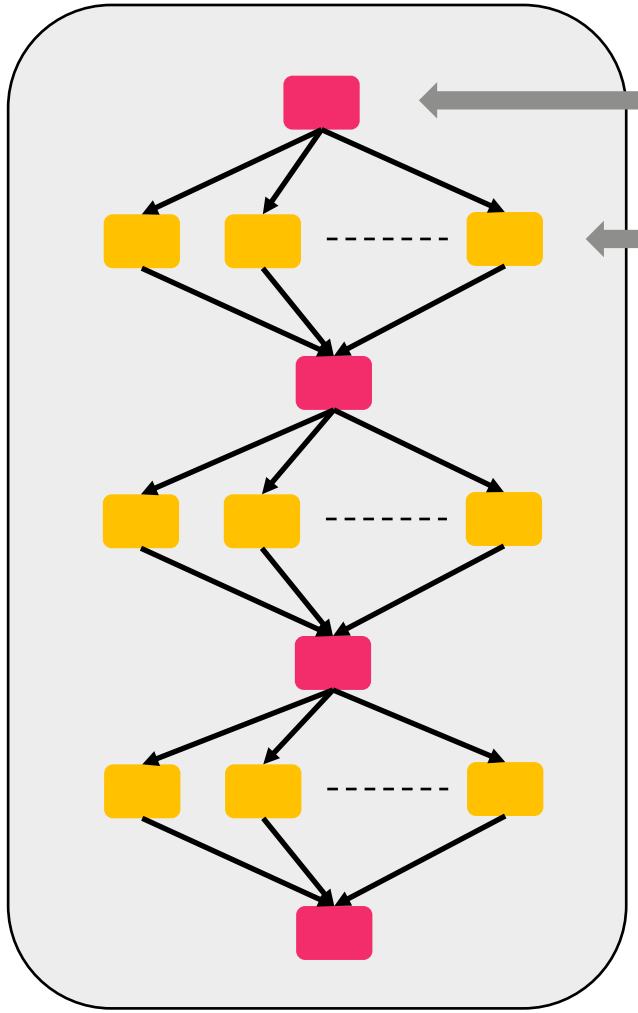
Multiple concurrent analysis runs from multiple users

Network Analysis

Perturbation

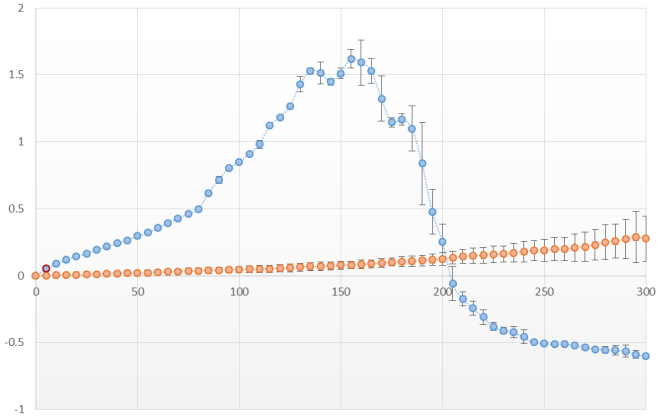
Goal: characterize network robustness behavior via perturbation

Total repeats



- One compute task per repeat
- One compute job
  - Calculate impact for a specific node set size
- All jobs:
  - impact calculations for node sets of all sizes
- Example below
  - 300 network calculations per repeat
  - Error bars generated by repeats

Generated data:



Network Analysis

Impact Maximization

e-therapeutics

Goals:

- Find protein sets that have a large effect on network structural coherence and so on the targeted biological process
- Robustness properties of biological networks mean the vast majority of protein sets have little effect
- Compound mapping to those protein sets finds potential therapeutics

Algorithmic Approach

- Exhaustive approach unfeasible due to combinatoric explosion: $C_{20}^{1000} \approx 3.4 * 10^{41}$
- Stochastic approximation or metaheuristics
- Stochastic aspect facilitates the exploration of solution space: more likely to find global maxima

- Genetic algorithm
  - Specific, **population based** stochastic approximation approach
  - Based (very loosely) on natural selection
  - Population based ⇒ embarrassingly parallel

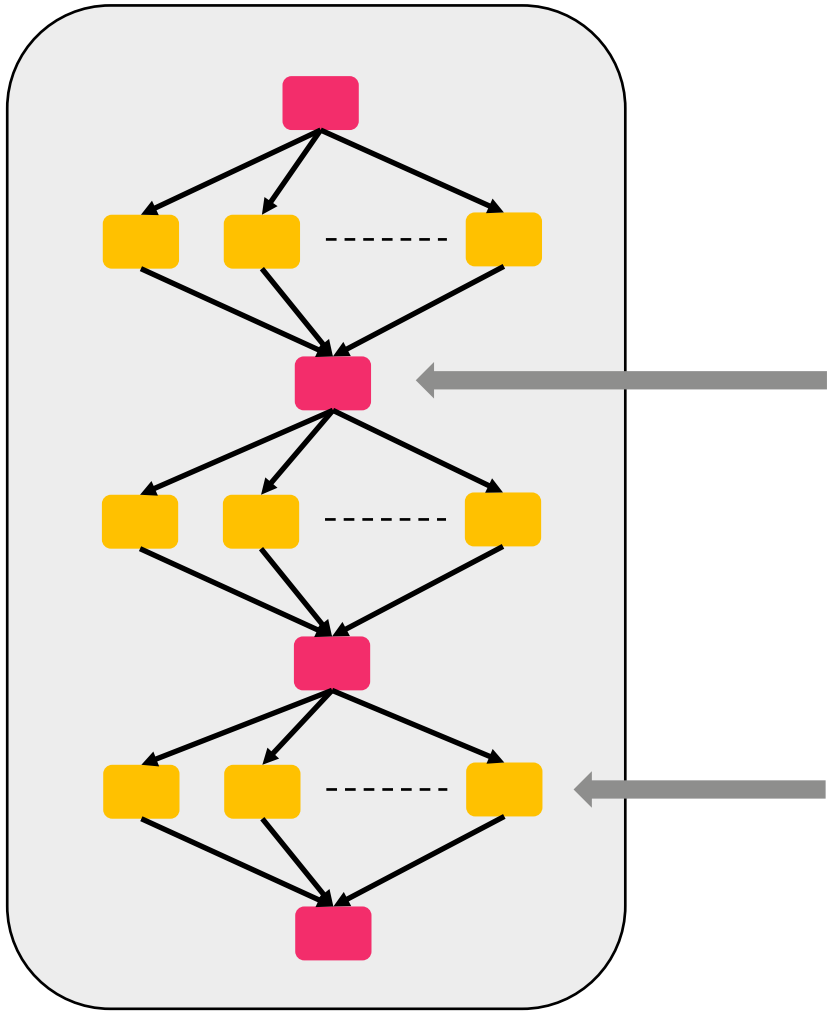
Network Analysis

Impact Maximization via Genetic Algorithm

e-therapeutics

Goal: find protein set(s) that maximize network impact

asymptotic convergence



- One compute task per “generation”
- Generates population of potential solutions (nodes to remove)
  - Initially randomly
  - Then by “breeding” best solutions of previous generation

- Compute job: evaluation of one member of population
- All jobs: evaluation of whole population
- Evaluation: quantification of the effect of node removal

Implementation Lessons

1. Minimize Data Distribution

e-therapeutics

Naïve (first) implementation

- Master node generates population of perturbed networks
- Networks are distributed to worker nodes
- Worker nodes perform network calculations (e.g. shortest path analysis)

- Parallel distributed implementation was slower than serial
- Cost of data distribution swamped gain due to parallel calculations

Current Solution

- Full, intact network is distributed to all worker nodes once at the start
- Master node generates population of bit vectors indicating which nodes to remove
- Bit vectors are distributed to worker nodes

- Intact network is shared between worker nodes and multiple threads on each worker node
  - Immutable data structure for network
  - Percolation operation is construction of new network not removal of nodes from intact network.

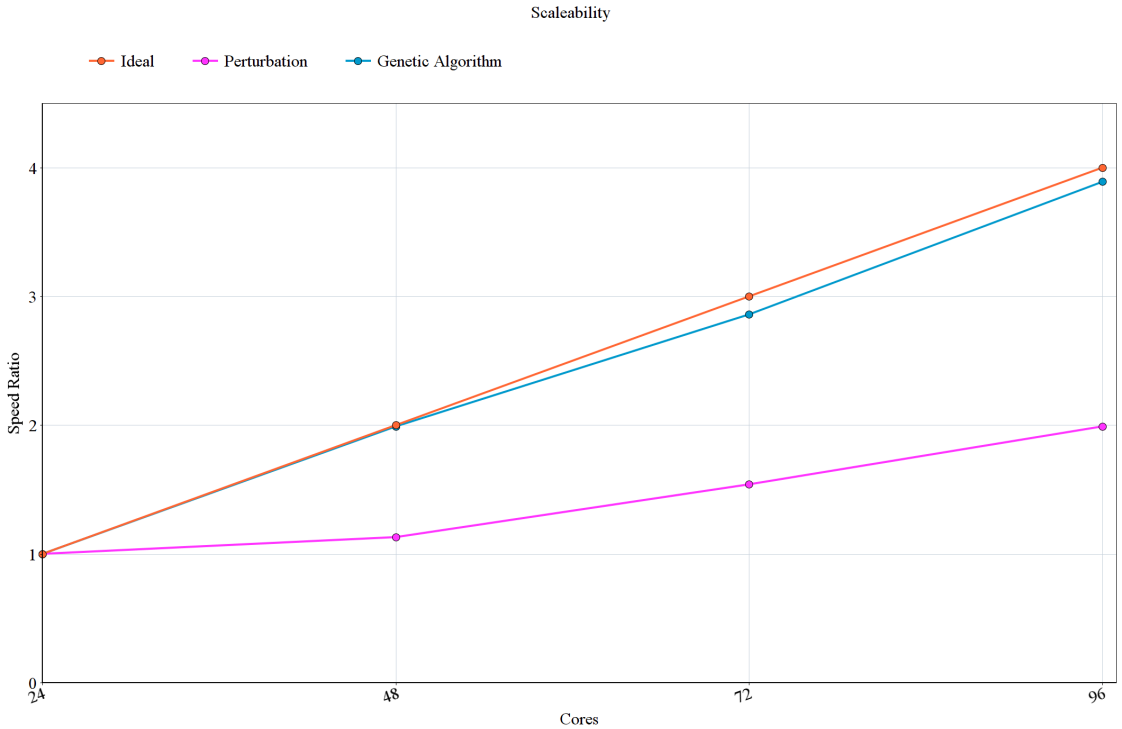
Implementation Lessons

2. Scalability Depends on Compute Job Homogeneity

e-therapeutics

Scalability Measurements

- Measure time taken for one actual compute run on grids of different sizes
  - Minimum: 1 physical machine with 24 cores
  - Maximum: 4 physical machines with 96 cores
- Ratio of time taken relative to minimum grid size



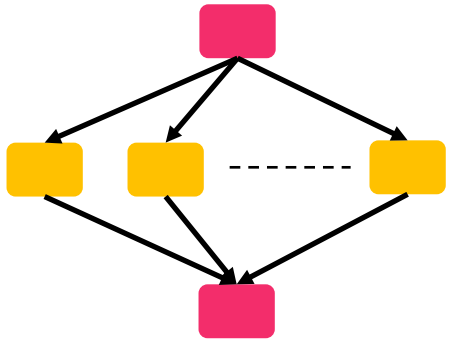
Implementation Lessons

2. Scalability Depends on Compute Job Homogeneity

e-therapeutics

Genetic Algorithm

- Excellent scalability
- Scalability generalizes across compute job parameters
- Homogenous jobs within a task
  - Removing the same number of nodes from the same network
  - Calculating the same network statistics



Fastest possible task time is slowest job

Perturbation Analysis

- Scalability is poor
- Jobs within a task are much more heterogenous
  - Each job removes a different number of nodes from the network
  - Tuning the task and job boundaries for this analysis is hard

Future

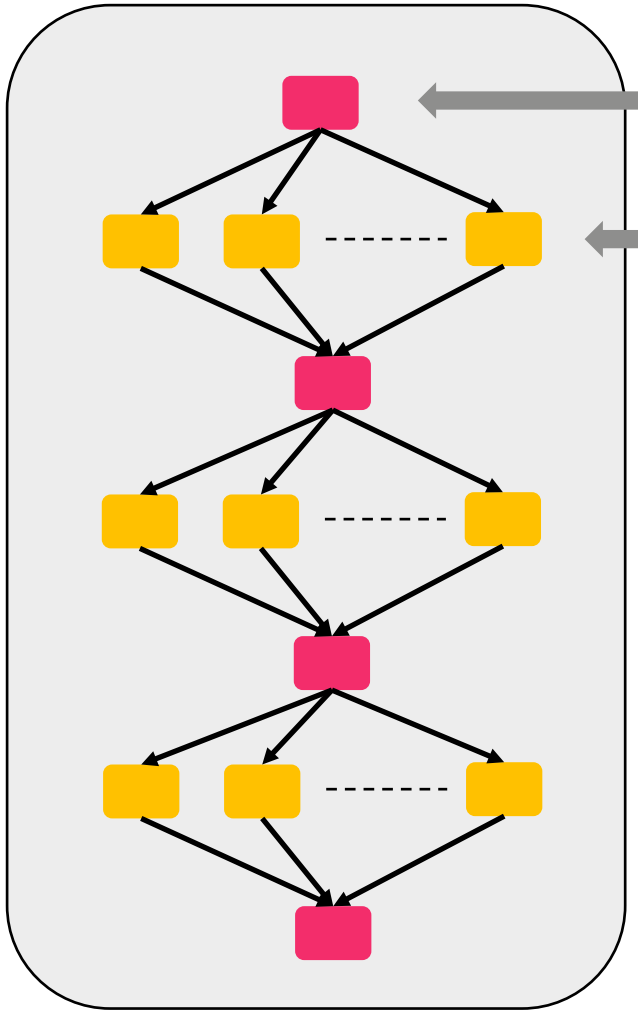
- Job stealing SPI: potentially allows redistribution of jobs when some are slow and others are fast

Network Analysis

Compound Impact Ranking

- Goal: evaluate network impact of every compound in our database

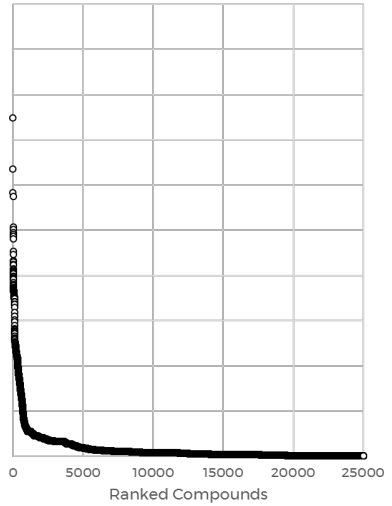
all compounds in virtual screening library (~13M)



- Compute task: set of compounds (from database)
- Multiple tasks: full compound set pagination

- Compute job: evaluation of a subset of compounds
  - Set size determined by hardware knowledge

Generate data:

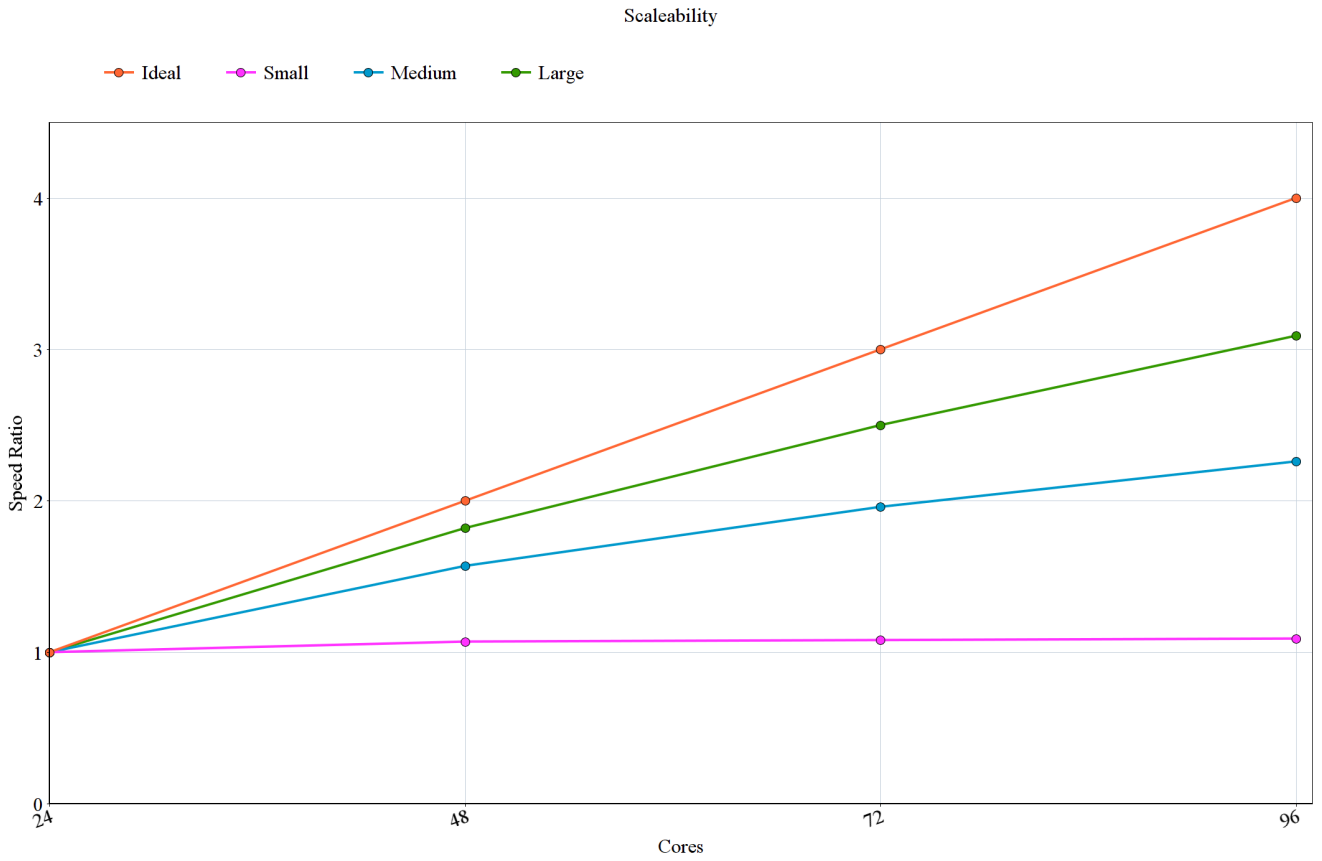


Implementation Lessons

3. Data access can dominate

e-therapeutics

Scalability Measurements: Networks of Different Size

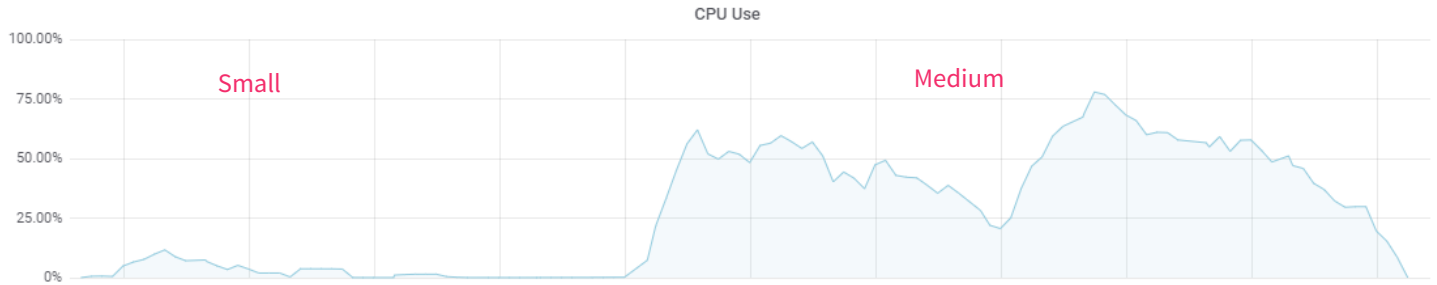


Implementation Lessons

3. Data access can dominate

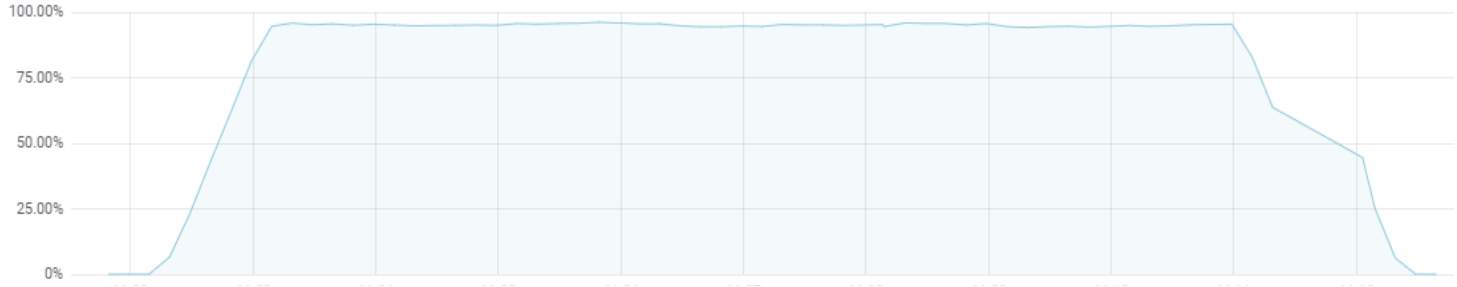
e-therapeutics

Compound Impact Ranking
Small and Medium Network



Compound Impact Ranking
Large Network

Genetic Algorithm



Implementation Lessons

3. Data access can dominate

e-therapeutics

Scalability depends on network size

Small networks

- compute time swamped by database access time

Larger networks

- database access still reduces CPU utilization

We expected network calculations to dominate data access

- Measure, don’t assume

Future

- Integrate in-memory data grid
- Job heterogeneity still an issue although currently dominated by database access

Summary

e-therapeutics

Three different business processes in production for over four years

Technical advantages of using GridGain

- Removes the need to implement parallel distributed processing infrastructure
- Facilitates development focus on business problems
- Lessons learnt
  - Implications of parallel distributed processing do not disappear – see Sun's "fallacies of distributed computing"
  - Powerful, easy to use API can lead to naïve solutions
  - Minimize data transfer from master to workers
  - Need to be very aware of how parameters affect compute job homogeneity
  - Database access can affect even very CPU intensive jobs

Business advantages of solution

- Remove computational parts as bottlenecks in full process
- Change working model from batch driven to real-time and exploratory
  - Disease biologists have definitely noticed and it has changed the way they work
  - Increased ability to explore more hypotheses
  - We can still improve
- Algorithm choice can be driven ease of mapping to fork-join/map-reduce

Future

e-therapeutics

Migrate from old GridGain version to Apache Ignite

Investigate new capabilities to improve speed

- Job stealing to deal with heterogenous job distributions
- In-memory data grid to improve IO bound compute

On-demand compute grid architecture

- Our use patterns are very spikey – *in silico* is only a small part of full discovery process
- Investigate use of cloud platforms to provide compute grid as and when needed
- Combine with general platform migration to Kubernetes

Future

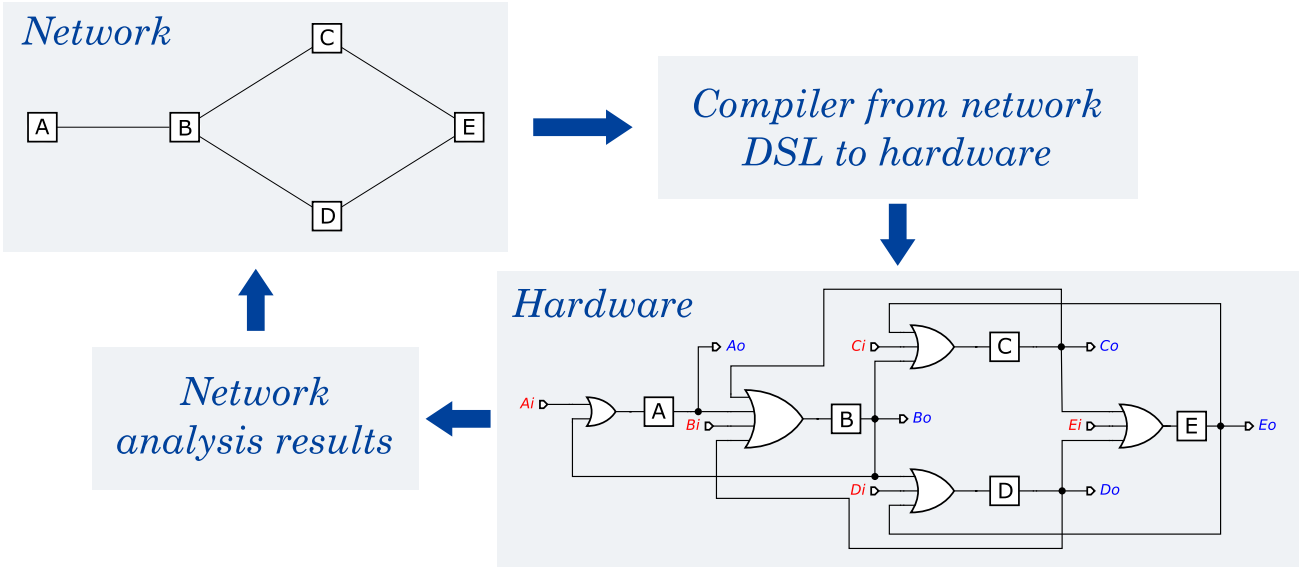
General to Specific Purpose Computing

e-therapeutics

Academic collaborations:

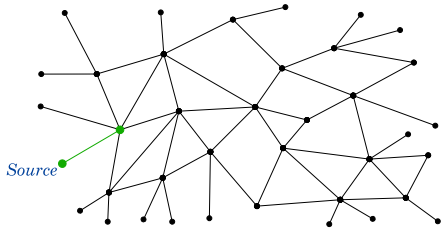
- FANTASI (Fast Network Analysis in Silicon)
  - co-funded EPSRC project with µSystems Research Group (Andrey Mokhov) at Newcastle University
  - Investigate hardware approaches to network analysis using FPGAs
- POETS (Partially Ordered Event Triggered Systems)
  - EPSRC funded project involving Cambridge, Imperial College, Newcastle, and Southampton Universities
  - Investigate compute architectures consisting of extremely large number of small cores

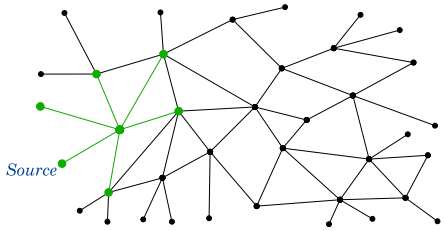
FANTASI

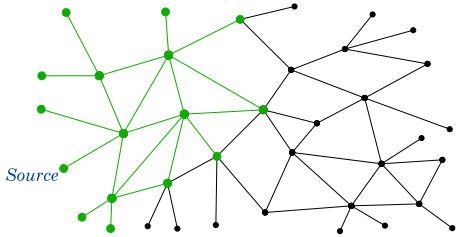


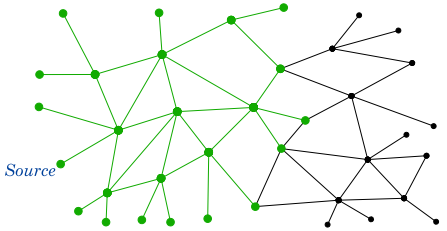
Future

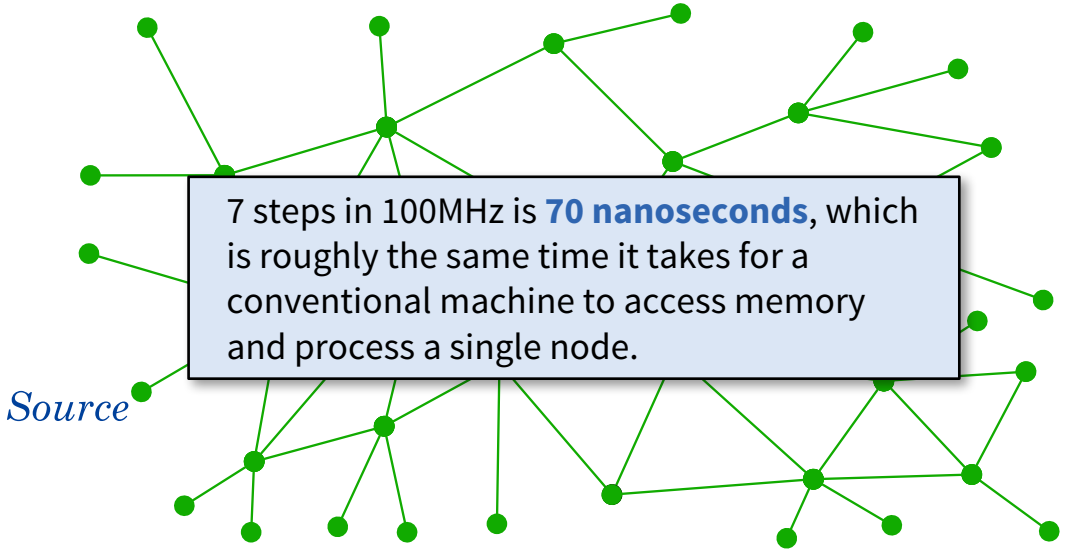
FANTASI: Shortest Path Analysis











- Successfully implemented on FPGA
- Acceleration factors of over three orders of magnitude
  - Over 2500x for network of 3500 nodes
- Network size limited by hardware and layout algorithms
  - POETS project for larger networks
- Algorithms limited to those that can be mapped to hardware


e-therapeutics
Thank you
jonny.wray@etherapeutics.co.uk

Implementation Lessons

CPU Utilization: IO can dominate

e-therapeutics