Open-universe probability models: Unifying logic and probability

Stuart Russell
Computer Science Division, UC Berkeley

Joint work with Brian Milch, David Sontag, Andrey Kolobov, Bhaskara Marthi, Lei Li, Siddharth Srivastava, Nimar Arora, Erik Sudderth, Paul Kidwell, David Moore, Kevin Mayeda, Steve Myers, Christopher Lin, Tony Dear, Ron Sun, Min Joon Seo

AI: intelligent systems in the real world

AI: intelligent systems in the real world

The world has things in it!!

AI: intelligent systems in the real world

The world has things in it!!

Good Old-Fashioned AI: first-order logic

Why did AI choose first-order logic?

- Provides a *declarative* substrate
  - Learn facts, rules from observation and communication
  - Combine and reuse in arbitrary ways

Why did AI choose first-order logic?

- Provides a *declarative* substrate
  - Learn facts, rules from observation and communication
  - Combine and reuse in arbitrary ways
- *Expressive* enough for general-purpose intelligence
  - It provides concise models, essential for learning

Why did AI choose first-order logic?

- Provides a *declarative* substrate
  - Learn facts, rules from observation and communication
  - Combine and reuse in arbitrary ways
- *Expressive* enough for general-purpose intelligence
  - It provides concise models, essential for learning
  - E.g., rules of chess (32 pieces, 64 squares, ~100 moves)

Why did AI choose first-order logic?

- Provides a *declarative* substrate
  - Learn facts, rules from observation and communication
  - Combine and reuse in arbitrary ways
- *Expressive* enough for general-purpose intelligence
  - It provides concise models, essential for learning
  - E.g., rules of chess (32 pieces, 64 squares, ~100 moves)
    - ~100 000 000 000 000 000 000 000 000 000 000 000 000 pages as a state-to-state transition matrix (cf HMMs, automata)

R.B.KB.RPPP..PPP..N..N…..PP….q.pp..Q..n..n..ppp..pppr.b.kb.r

Why did AI choose first-order logic?

- Provides a *declarative* substrate
  - Learn facts, rules from observation and communication
  - Combine and reuse in arbitrary ways
- *Expressive* enough for general-purpose intelligence
  - It provides concise models, essential for learning
  - E.g., rules of chess (32 pieces, 64 squares, ~100 moves)
    - ~100 000 000 000 000 000 000 000 000 000 000 000 000 pages as a state-to-state transition matrix (cf HMMs, automata)

R.B.KB.RPPP..PPP..N..N…..PP….q.pp..Q..n..n..ppp..pppr.b.kb.r

    - ~100 000 pages in propositional logic (cf circuits, graphical models)

WhiteKingOnC4@Move12

Why did AI choose first-order logic?

- Provides a *declarative* substrate
  - Learn facts, rules from observation and communication
  - Combine and reuse in arbitrary ways
- *Expressive* enough for general-purpose intelligence
  - It provides concise models, essential for learning
  - E.g., rules of chess (32 pieces, 64 squares, ~100 moves)
    - ~100 000 000 000 000 000 000 000 000 000 000 000 000 pages as a state-to-state transition matrix (cf HMMs, automata)

R.B.KB.RPPP..PPP..N..N…..PP….q.pp..Q..n..n..ppp..pppr.b.kb.r

    - ~100 000 pages in propositional logic (cf circuits, graphical models)

WhiteKingOnC4@Move12

    - 1 page in first-order logic

$\forall x,y,t,color,piece\ On(color,piece,x,y,t) \Leftrightarrow \ldots$

AI: intelligent systems in the real world

The world has things in it!!

Good Old-Fashioned AI: first-order logic

AI: intelligent systems in the real world

The world has things in it!!

The world is uncertain !!

Good Old-Fashioned AI: first-order logic

AI: intelligent systems in the real world

The world has things in it!!

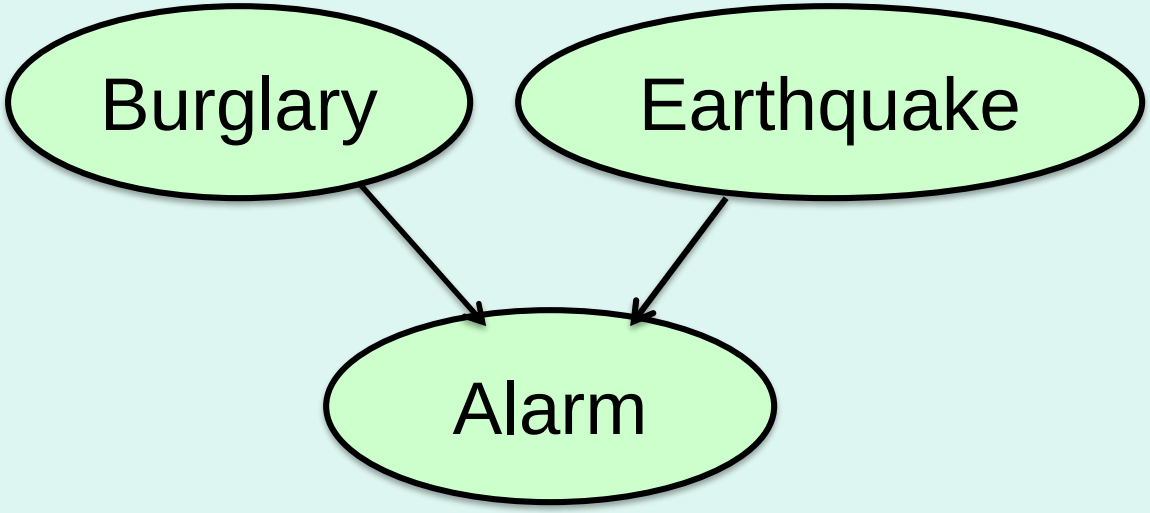
The world is uncertain!!

Good Old-Fashioned AI: first-order logic

Modern AI: probabilistic graphical models

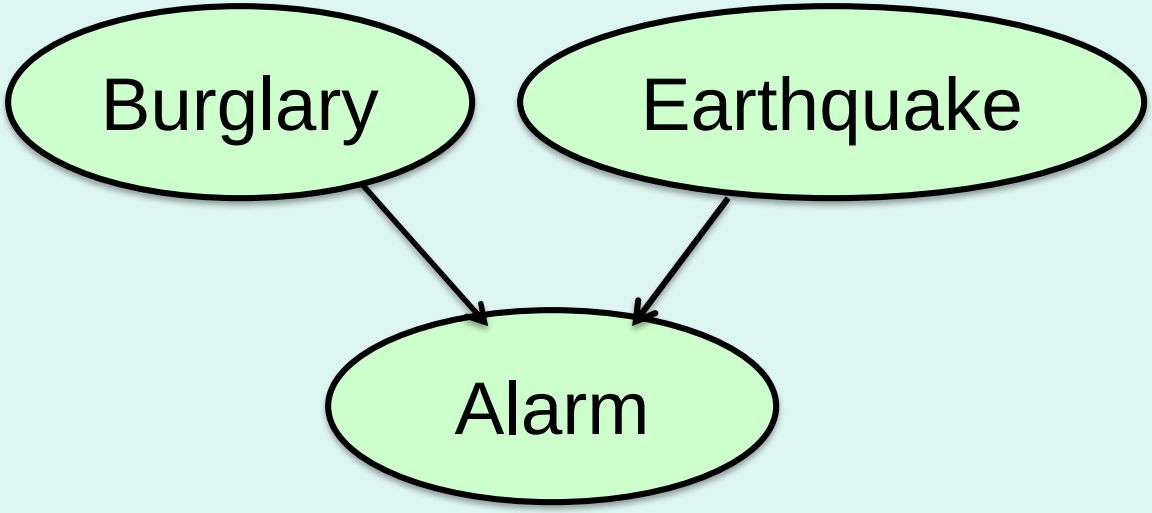
Bayesian networks

Define distributions on all possible ***propositional*** worlds



Bayesian networks

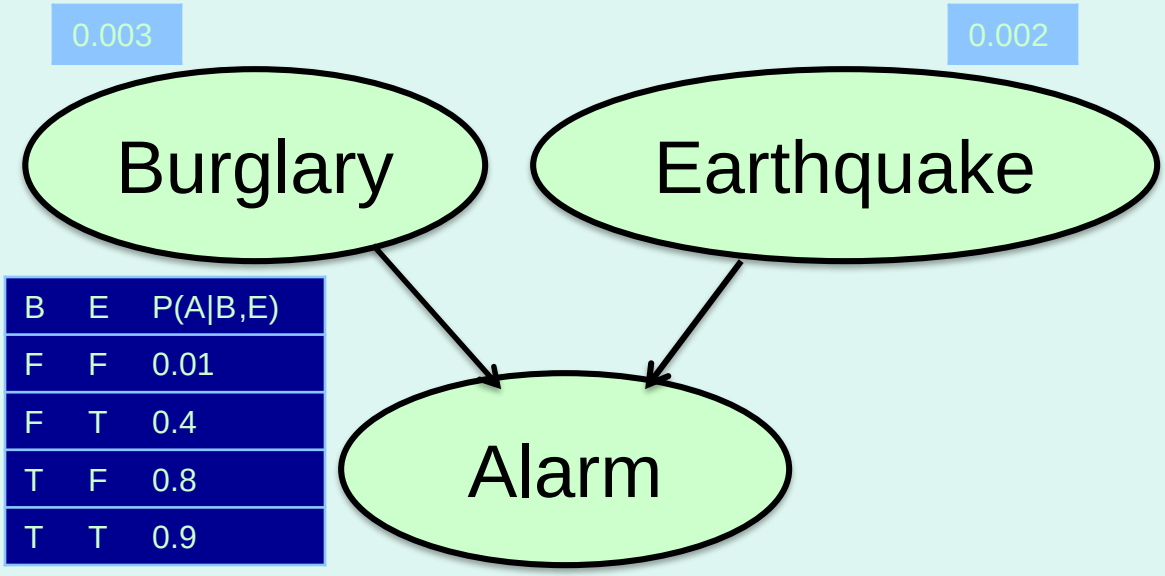
Define distributions on all possible ***propositional*** worlds



$$P(B,E,A) = P(B)\, P(E)\, P(A \mid B, E)$$

Bayesian networks

Define distributions on all possible ***propositional*** worlds

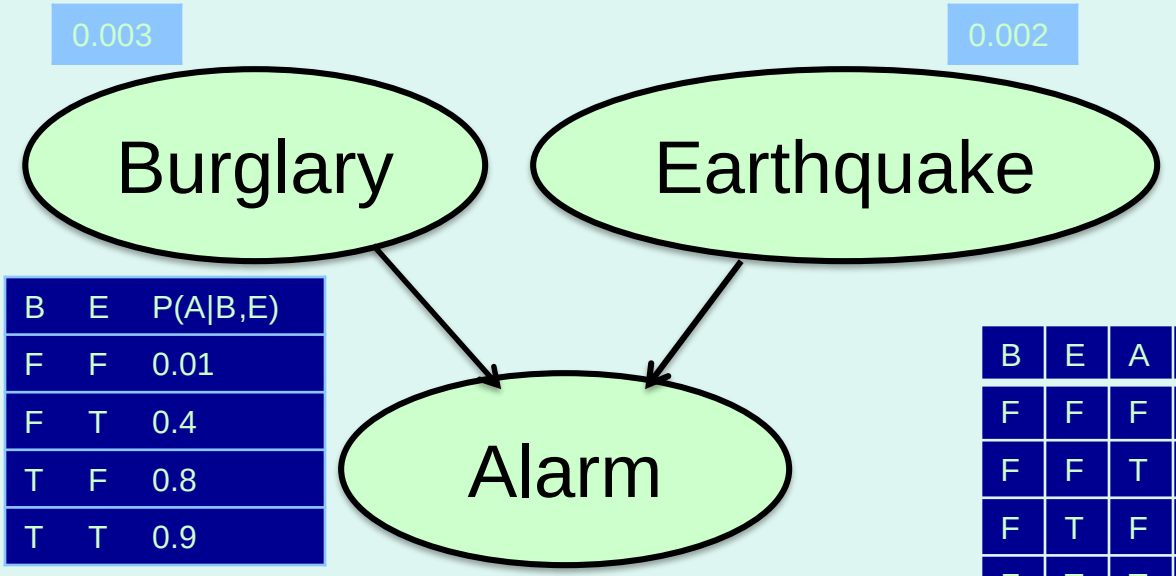


B	E	P(A|B,E)
F	F	0.01
F	T	0.4
T	F	0.8
T	T	0.9

$$P(B,E,A) = P(B)\ P(E)\ P(A \mid B, E)$$

Bayesian networks

Define distributions on all possible ***propositional*** worlds



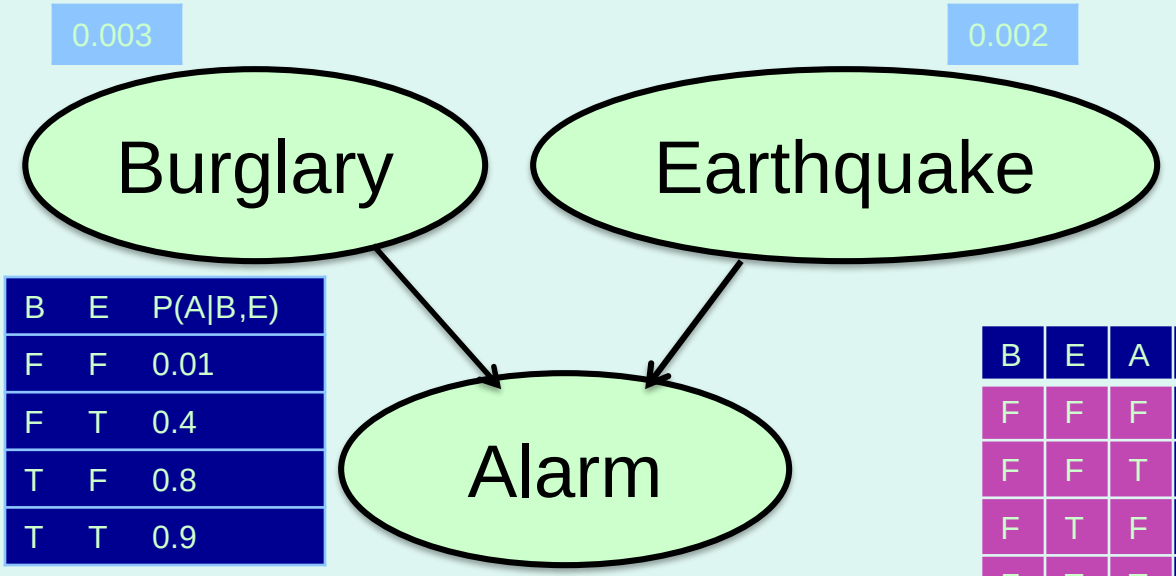
B	E	P(A|B,E)
F	F	0.01
F	T	0.4
T	F	0.8
T	T	0.9

B	E	A	P(B,E,A)
F	F	F	0.985
F	F	T	0.00995
F	T	F	0.0012
F	T	T	0.0008
T	F	F	0.0006
T	F	T	0.0024
T	T	F	0.0000006
T	T	T	0.0000054

$$P(B,E,A) = P(B)\, P(E)\, P(A \mid B, E)$$

Bayesian networks

Define distributions on all possible ***propositional*** worlds



B	E	P(A|B,E)
F	F	0.01
F	T	0.4
T	F	0.8
T	T	0.9

B	E	A	P(B,E,A)
F	F	F	0.985
F	F	T	0.00995
F	T	F	0.0012
F	T	T	0.0008
T	F	F	0.0006
T	F	T	0.0024
T	T	F	0.0000006
T	T	T	0.0000054

$$P(B,E,A) = P(B)\, P(E)\, P(A \mid B, E)$$

AI: intelligent systems in the real world

The world has things in it!!

The world is uncertain!!

Good Old-Fashioned AI: first-order logic

Modern AI: probabilistic graphical models

AI: intelligent systems in the real world

The world has things in it!!

The world is uncertain!!

Good Old-Fashioned AI: first-order logic

Modern AI: probabilistic graphical models

The world is uncertain!!

AI: intelligent systems in the real world

The world has things in it!!

The world is uncertain!!

Good Old-Fashioned AI: first-order logic

Modern AI: probabilistic graphical models

The world is uncertain!!

The world has things in it!!

AI: intelligent systems in the real world

The world has things in it!!

The world is uncertain!!

Good Old-Fashioned AI: first-order logic

Modern AI: probabilistic graphical models

The world is uncertain!!

The world has things in it!!

A New Dawn for AI™: first-order probabilistic languages

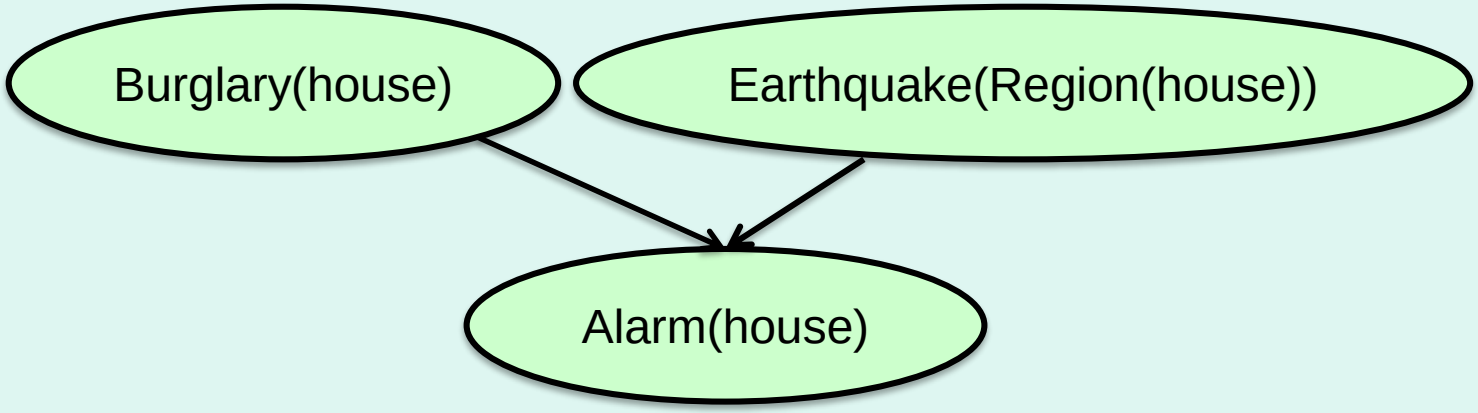
Anil Ananthaswamy, "***I, Algorithm: A new dawn for AI***," New Scientist, Jan 29, 2011

First-order probabilistic languages

- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea

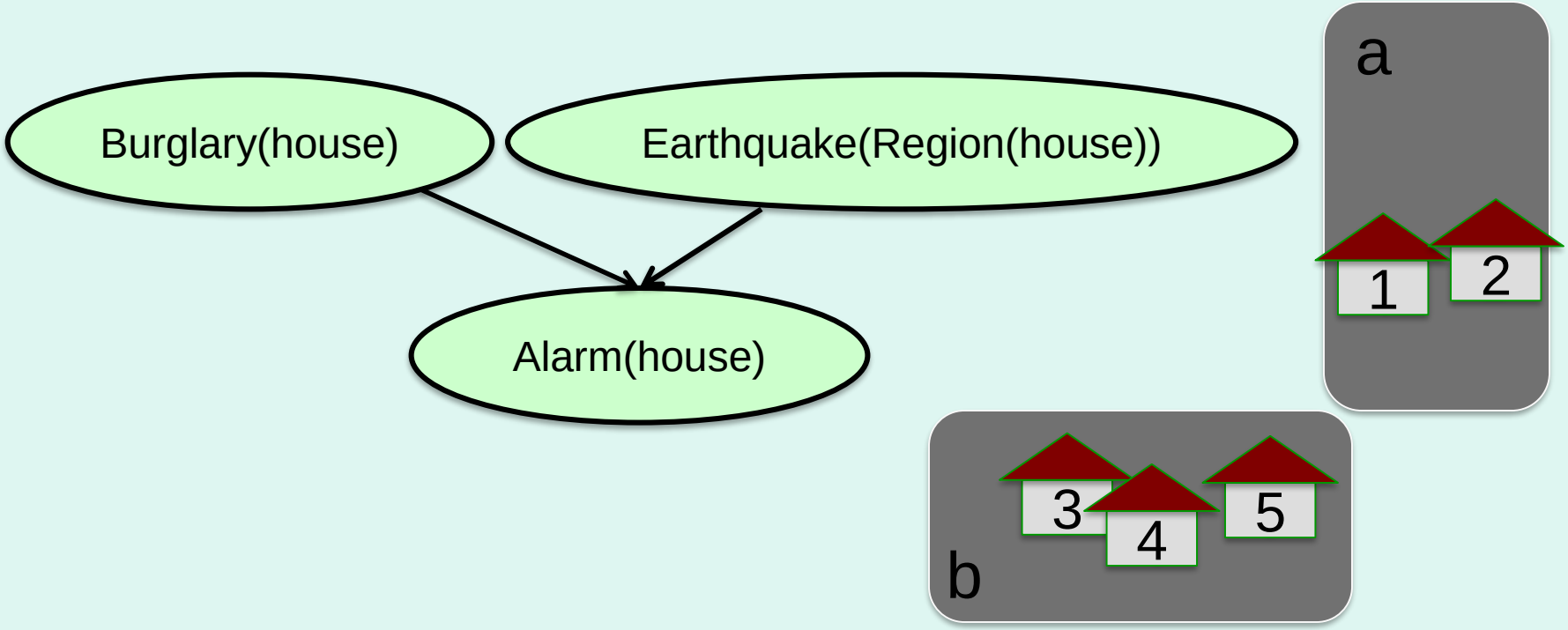
First-order probabilistic languages

- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea



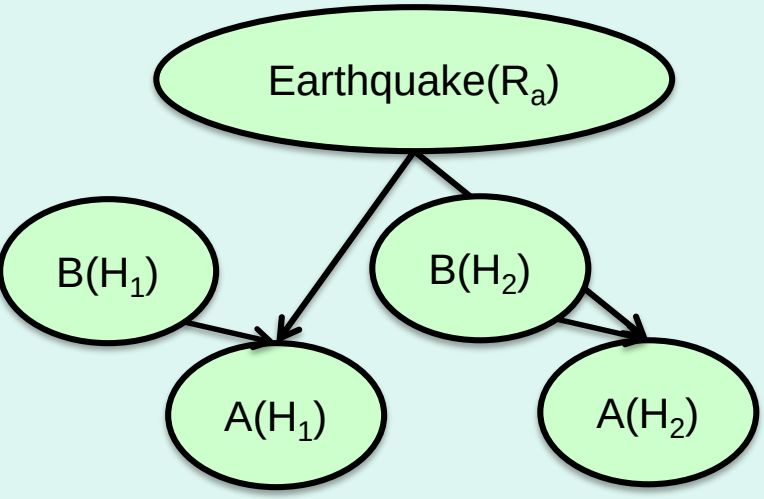
First-order probabilistic languages

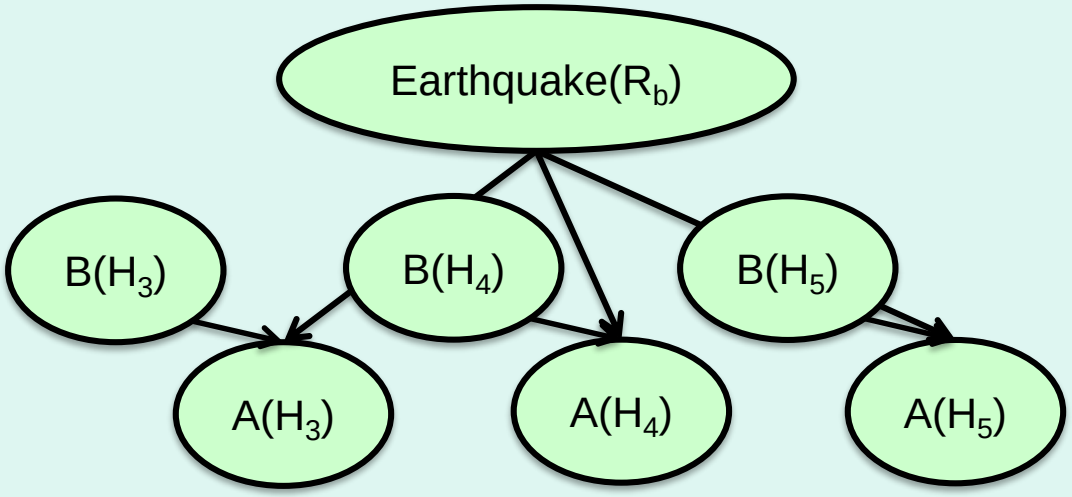
- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea



First-order probabilistic languages

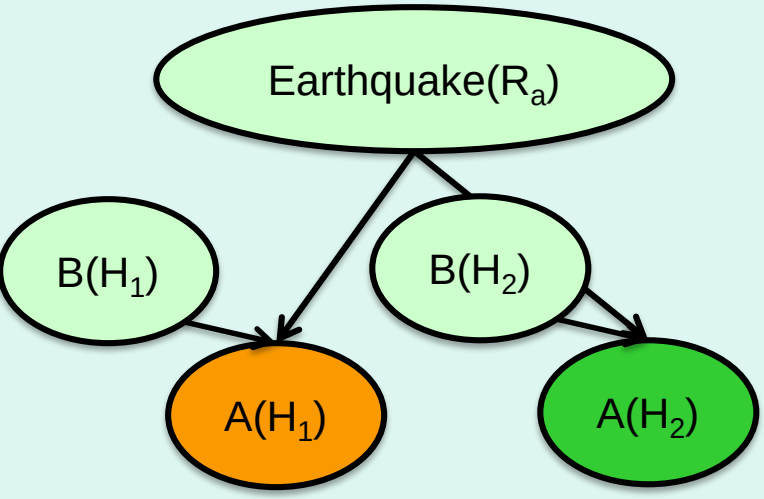
- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea

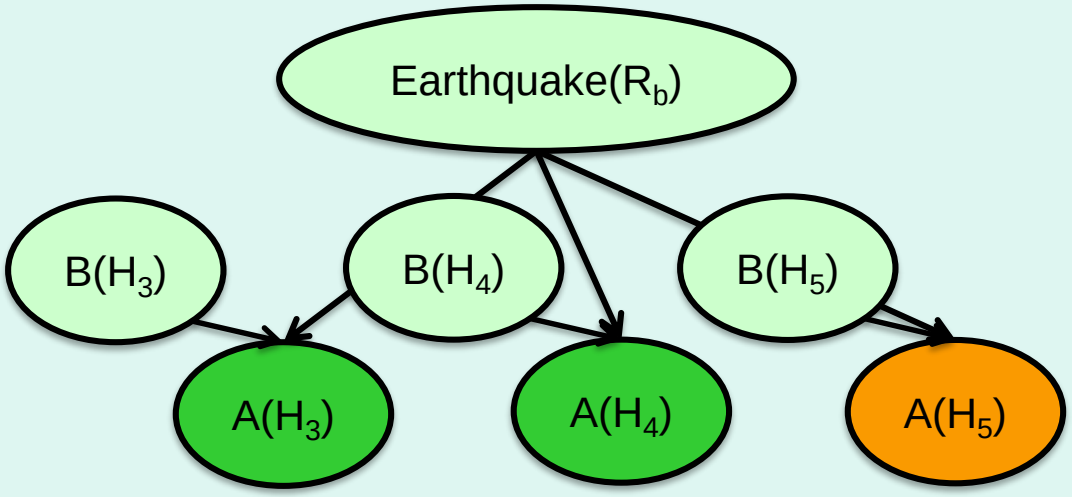




First-order probabilistic languages

- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea





AI: intelligent systems in the real world

The world is uncertain and has things in it!!

A New Dawn for AI™:
first-order probabilistic languages

AI: intelligent systems in the real world

The world has things in it and we don't know what they are!!

A Newer Dawn for AI™: first-order ***open-universe*** probabilistic languages

An important distinction in logic

- *Closed-universe* languages assume unique names and domain closure, i.e., known objects
  - Like Prolog, databases (Herbrand semantics)
  - Poole 93, Sato 97, Koller & Pfeffer 98, De Raedt 00, etc.
- *Open-universe* languages allow uncertainty over the existence and identity of objects
  - Like full first-order logic
  - BLOG (Milch & Russell 05): declarative OUPM language
  - Probabilistic programming (Koller, McAllester, Pfeffer 97, Pfeffer 01, Goodman et al 08): distribution on execution traces of stochastic programs

A little test

Given

Bill = Father(William) and Bill = Father(Junior)

How many children does Bill have?

Closed-universe (Herbrand) semantics:

2

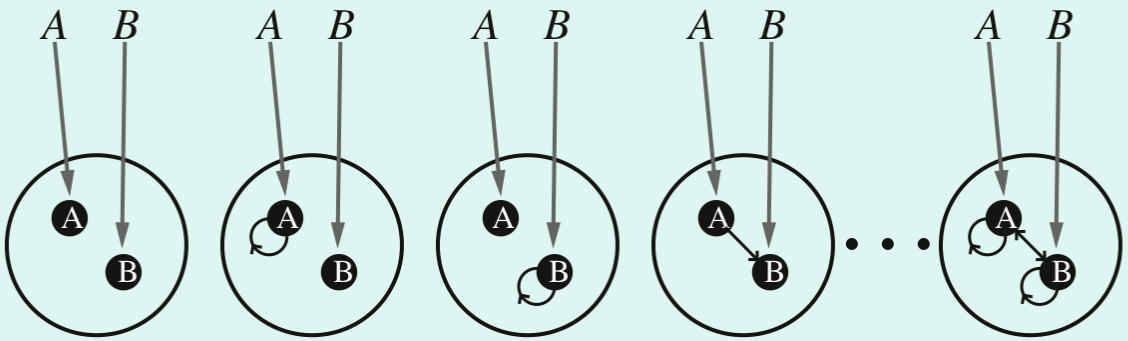
Open-universe (full first-order) semantics:

Between 1 and ∞

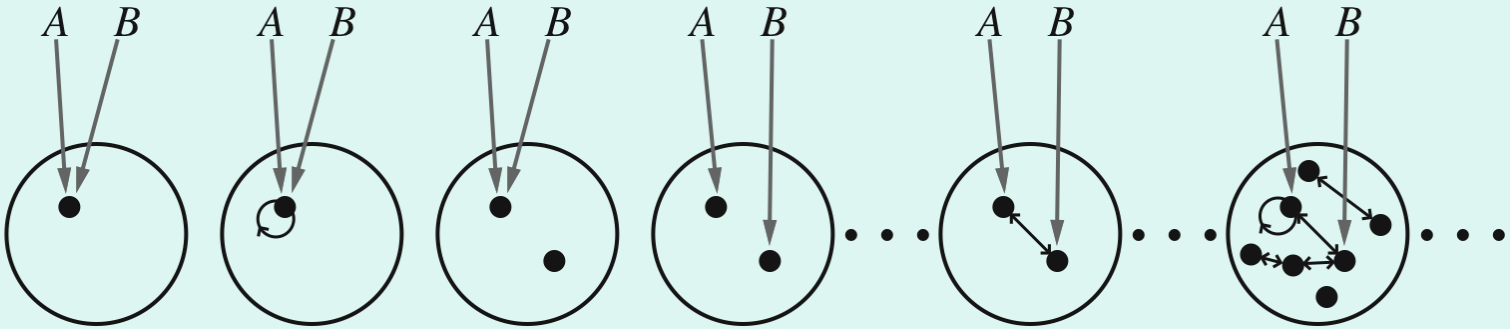
Open-universe semantics

Possible worlds for a language with two constant symbols A and B and one relation symbol

Closed-universe semantics

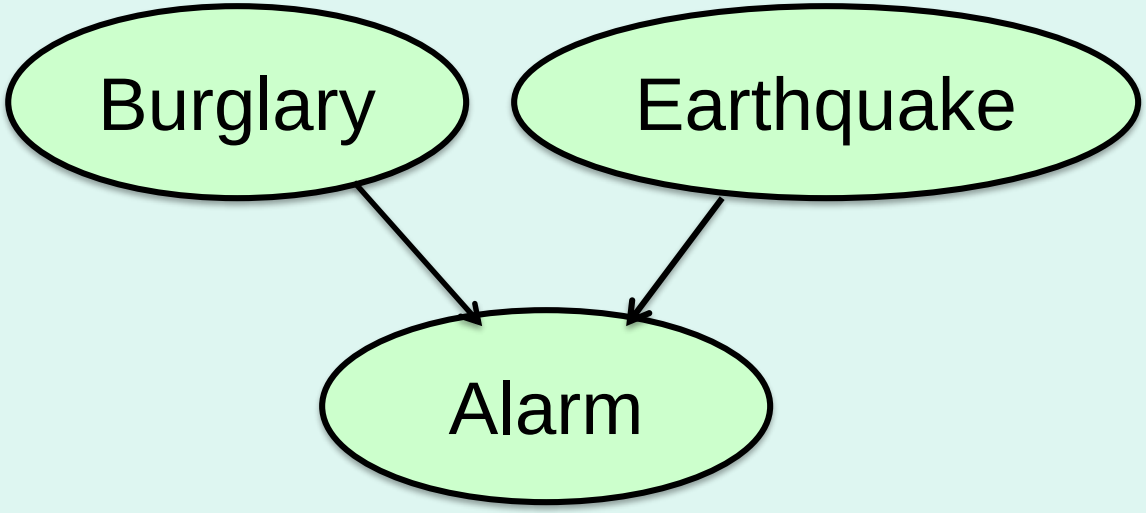


Open-universe semantics

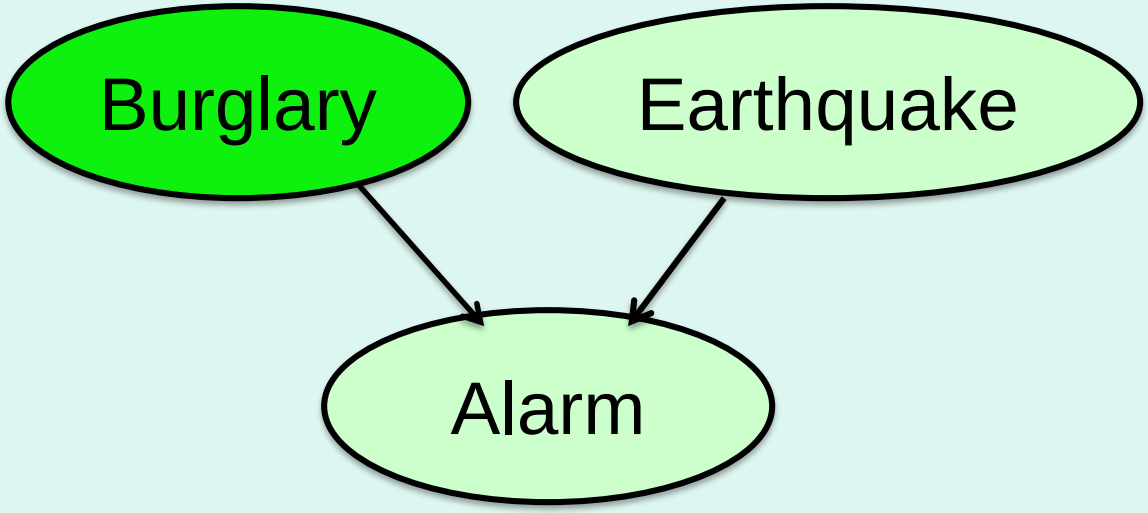


but how can we define P on Ω ??

Bayes nets build propositional worlds

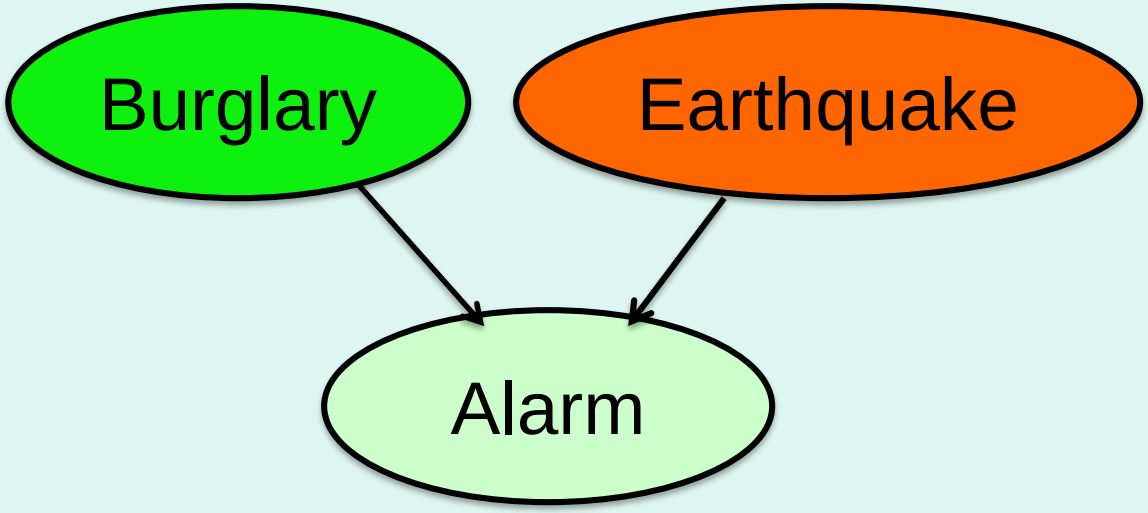


Bayes nets build propositional worlds



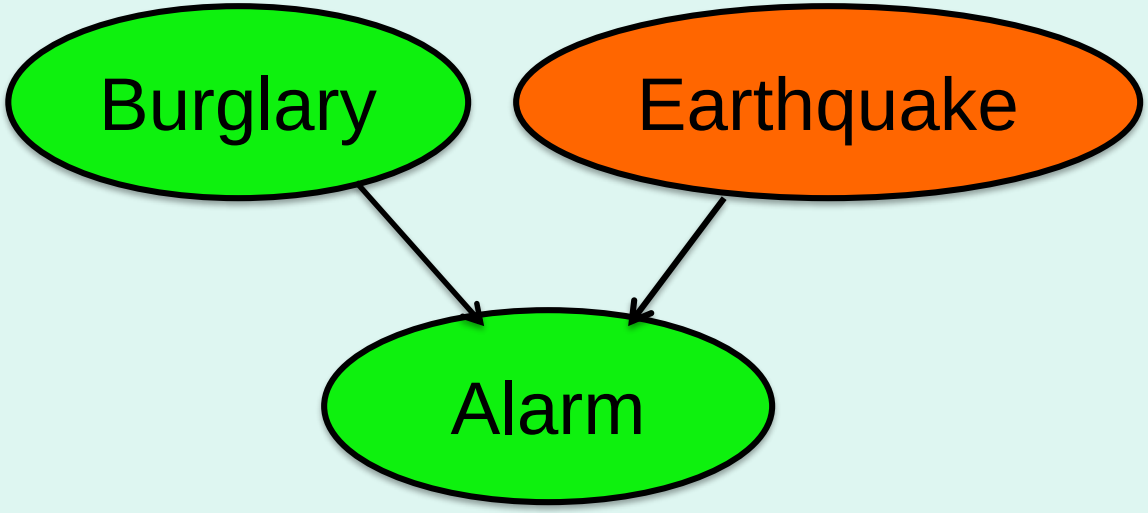
Burglary

Bayes nets build propositional worlds



Burglary
not Earthquake

Bayes nets build propositional worlds



Burglary
not Earthquake
Alarm

Open-universe models in BLOG

- Construct worlds using two kinds of steps, proceeding in topological order:
  - *Dependency statements*: Set the value of a *function or relation on its arguments*, conditioned on parent values
    - Alarm(h) ~ CPT[..](Burglary(h), Earthquake(Region(h)))
  - *Number statements*: *Add some objects to the world*, conditioned on what objects and relations exist so far
    - #GeologicalFaultRegion ~ Uniform{1…10}
    - #House(Region=r) ~ Poisson(50)

Origin function

Semantics

- *Objects* are defined by type, origin, number:
  - <GeologicalFaultRegion, ,3>
  - <House,<Region, <GeologicalFaultRegion, ,3>>, 97>
- Each *basic random variable* is a function or predicate symbol indexed by a tuple of objects:
  - $\text{Earthquake}_{\langle\langle \text{GeologicalFaultRegion}, ,3\rangle\rangle}(\omega)$
- Each *possible world* ω specifies values for all number variables and basic random variables
- *Probability* of ω is given by the product of conditional probabilities specified in the model

Semantics contd.

- *Probability* of ω is given by the product of conditional probabilities specified in the model
  - This holds because the generation path for any world is unique
  - Which holds because objects contain their generation history
  - (Which would not be true if we used the standard objects of model theory, o_1, o_2, o_3, o_4,…)

Semantics contd.

Every well-formed* BLOG model specifies a unique proper probability distribution over all possible worlds definable given its vocabulary

* No infinite receding ancestor chains; no conditioned cycles; all expressions finitely evaluable; functions of countable sets

Citation information extraction

- Given: a set of text strings from reference lists:
  - [Lashkari et al 94] Collaborative Interface Agents, Yezdi Lashkari, Max Metral, and Pattie Maes, Proceedings of the Twelfth National Conference on Articial Intelligence, MIT Press, Cambridge, MA, 1994.
  - Metral M. Lashkari, Y. and P. Maes. Collaborative interface agents. In Conference of the American Association for Artificial Intelligence, Seattle, WA, August 1994
- Decide:
  - What papers and researchers exist
  - For each paper
    - The real title
    - The real authors
    - The papers it cites

BLOG model (single-author)

```
#Researcher ~ LogNormal[6.9,2.3]();

Name(r) ~ CensusDB_NamePrior();

#Paper(Author=r) ~
    if Prof(r) then LogNormal[3.5,1.2]()
               else LogNormal[2.3,1.2]()

Title(p) ~ CSPaperDB_TitlePrior();

PubCited(c) ~ Uniform({Paper p});

Text(c) ~ NoisyCitationGrammar
      (Name(Author(PubCited(c))),
       Title(PubCited(c)));
```

BLOG model (single-author)

```
#Researcher ~ LogNormal[6.9,2.3]();

Name(r) ~ CensusDB_NamePrior();

#Paper(Author=r) ~
    if Prof(r) then LogNormal[3.5,1.2]()
               else LogNormal[2.3,1.2]()

Title(p) ~ CSPaperDB_TitlePrior();

PubCited(c) ~ Uniform({Paper p});

Text(c) ~ NoisyCitationGrammar
      (Name(Author(PubCited(c))),
       Title(PubCited(c)));
```

BLOG model (single-author)

```
#Researcher ~ LogNormal[6.9,2.3]();

Name(r) ~ CensusDB_NamePrior();

#Paper(Author=r) ~
    if Prof(r) then LogNormal[3.5,1.2]()
               else LogNormal[2.3,1.2]()

Title(p) ~ CSPaperDB_TitlePrior();

PubCited(c) ~ Uniform({Paper p});

Text(c) ~ NoisyCitationGrammar
      (Name(Author(PubCited(c))),
       Title(PubCited(c)));
```

BLOG model (single-author)

```
#Researcher ~ LogNormal[6.9,2.3]();

Name(r) ~ CensusDB_NamePrior();

#Paper(Author=r) ~
    if Prof(r) then LogNormal[3.5,1.2]()
               else LogNormal[2.3,1.2]()

Title(p) ~ CSPaperDB_TitlePrior();

PubCited(c) ~ Uniform({Paper p});

Text(c) ~ NoisyCitationGrammar
      (Name(Author(PubCited(c))),
       Title(PubCited(c)));
```

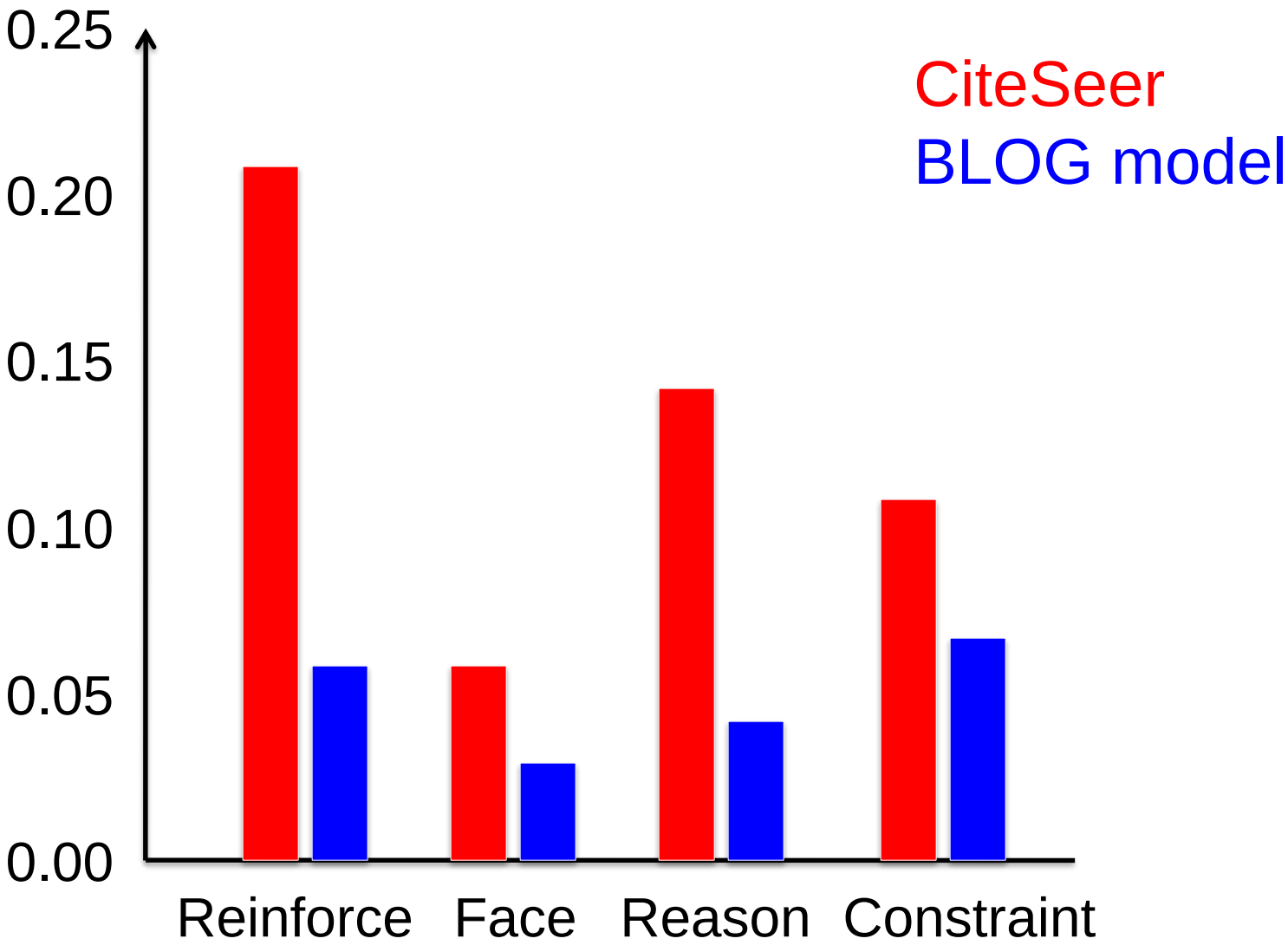
How to use BLOG

- Write model
- Supply data (e.g., citation strings)
- Ask query (e.g., what papers exist, who wrote them, what is the citation graph?)
- (wait a long time)

Fraction of citation clusters imperfectly recovered

Four data sets of ~300-500 citations, referring to ~150-300 papers

Fraction of citation clusters imperfectly recovered



Four data sets of ~300-500 citations, referring to ~150-300 papers

BLOG Example Library

1. PCFG for simple English
2. Simplified 3D vision
3. Hurricane prediction
4. Burglary
5. Balls and urns (counting)
6. Sybil attack (cybersecurity)
7. Students and grades
8. Topic models (LDA)
9. Citation information extraction
10. Competing workshops
11. Galaxy model
12. Infinite mixture of Gaussians
13. Monopoly (invisible opponent)
14. Blackjack
15. Multi-target tracking
16. HMM for genetic sequences
17. Weather forecasting
18. Video background subtraction
19. Financial volatility
20. Autoregression time series
21. Kalman filter
22. Infinite-state HMM

BLOG Example Library

1. PCFG for simple English
2. Simplified 3D vision
3. Hurricane prediction
4. Burglary
5. Balls and urns (counting)
6. Sybil attack (cybersecurity)
7. Students and grades
8. Topic models (LDA)
9. Citation information extraction
10. Competing workshops
11. Galaxy model
12. Infinite mixture of Gaussians
13. Monopoly (invisible opponent)
14. Blackjack
15. Multi-target tracking
16. HMM for genetic sequences
17. Weather forecasting
18. Video background subtraction
19. Financial volatility
20. Autoregression time series
21. Kalman filter
22. Infinite-state HMM

BLOG Example Library

1. PCFG for simple English
2. Simplified 3D vision
3. Hurricane prediction
4. Burglary
5. Balls and urns (counting)
6. Sybil attack (cybersecurity)
7. Students and grades
8. Topic models (LDA)
9. Citation information extraction
10. Competing workshops
11. Galaxy model
12. Infinite mixture of Gaussians
13. Monopoly (invisible opponent)
14. Blackjack
15. Multi-target tracking
16. HMM for genetic sequences
17. Weather forecasting
18. Video background subtraction
19. Financial volatility
20. Autoregression time series
21. Kalman filter
22. Infinite-state HMM

Multi-target tracking + data association

```
#Aircraft(EntryTime = t) ~ Poisson(λa);

Exits(a,t) if InFlight(a,t) then ~ Boolean(αe);

InFlight(a,t) = (t == EntryTime(a))
                |(InFlight(a,t-1) & !Exits(a,t-1));

X(a,t) if t = EntryTime(a) then ~ InitState()
       elseif InFlight(a,t) then
            ~ Normal(F*X(a,t-1),Σx);

#Blip(Source=a, Time=t)
  if InFlight(a,t) then
     ~ Bernoulli(DetectionProbability(X(a,t)));

#Blip(Time=t) ~ Poisson(λf);

Z(b) if Source(b)=null then ~ Uniform(R)
     else ~ Normal(H*X(Source(b),Time(b)),Σz);
```

BLOG Example Library

1. PCFG for simple English
2. Simplified 3D vision
3. Hurricane prediction
4. Burglary
5. Balls and urns (counting)
6. Sybil attack (cybersecurity)
7. Students and grades
8. Topic models (LDA)
9. Citation information extraction
10. Competing workshops
11. Galaxy model
12. Infinite mixture of Gaussians
13. Monopoly (invisible opponent)
14. Blackjack
15. Multi-target tracking
16. HMM for genetic sequences
17. Weather forecasting
18. Video background subtraction
19. Financial volatility
20. Autoregression time series
21. Kalman filter
22. Infinite-state HMM

BLOG Example Library

1. PCFG for simple English
2. Simplified 3D vision
3. Hurricane prediction
4. Burglary
5. Balls and urns (counting)
6. Sybil attack (cybersecurity)
7. Students and grades
8. Topic models (LDA)
9. Citation information extraction
10. Competing workshops
11. Galaxy model
12. Infinite mixture of Gaussians
13. Monopoly (invisible opponent)
14. Blackjack
15. Multi-target tracking
16. HMM for genetic sequences
17. Weather forecasting
18. Video background subtraction
19. Financial volatility
20. Autoregression time series
21. Kalman filter
22. Infinite-state HMM

Inference

Theorem: BLOG inference algorithms (rejection sampling, importance sampling, MCMC) converge* to correct posteriors for any well-formed model, for any finitely evaluable first-order query

Algorithms dynamically construct finite partial worlds with ground-atom variables directly relevant to query and evidence

Why MCMC?

- Works for any* model
- No model-specific mathematical work required
- Small space requirement
- Metropolis-Hastings step involves computing the acceptance ratio $\pi(x')q(x|x') / \pi(x)q(x'|x)$; everything cancels except local changes
- Query evaluation on states is also incrementalizable (cf DB systems)

Digression: Probabilistic programming

A probabilistic program is an ordinary stochastic program that defines a probability over each possible execution trace given its inputs

Since programs can construct objects, different executions may generate different numbers of objects; hence PPLs are “open-universe”

Most PPLs require the user program to build data structures for possible worlds

Digression: Probabilistic programming

```
(define num-regions (mem (lambda () (uniform 1 3))))
(define-record-type region (fields index))
(define regions (map (lambda (i) (make-region i))
                (iota (num-regions)))
(define num-houses (mem (lambda (r) (uniform 0 4))))
(define-record-type house (fields fault-region index))
(define houses (map (lambda (r)
                (map (lambda (i) (make-house r i))
                     (iota (num-houses r)))) regions)))
(define earthquake (mem (lambda (r) (flip 0.002))))
(define burglary (mem (lambda (h) (flip 0.003))))
(define alarm (mem (lambda (h)
  (if (burglary h)
      (if (earthquake (house-fault-region h))
          (flip 0.9) (flip 0.8))
      (if (earthquake (house-fault-region h))
          (flip 0.4) (flip 0.01))))
```

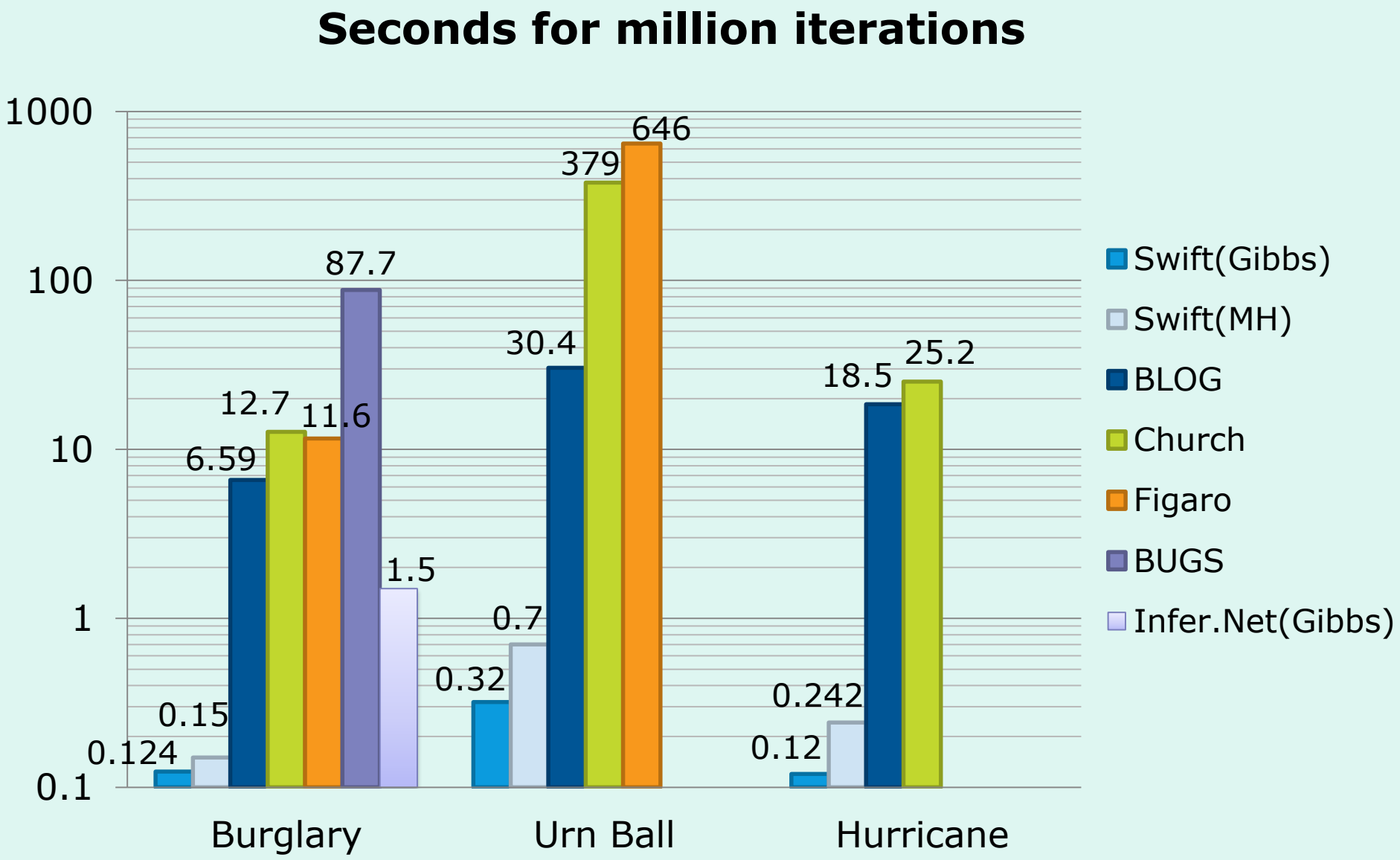
Digression: Probabilistic programming

```
(define num-regions (mem (lambda () (uniform 1 3))))
(define-record-type region (fields index))
(define regions (map (lambda (i) (make-region i))
                (iota (num-regions)))
(define num-houses (mem (lambda (r) (uniform 0 4))))
(define-record-type house (fields fault-region index))
(define houses (map (lambda (r)
                (map (lambda (i) (make-house r i))
                    (iota (num-houses r)))) regions)))
(define earthquake (mem (lambda (r) (flip 0.002))))
(define burglary (mem (lambda (h) (flip 0.003))))
(define alarm (mem (lambda (h)
  (if (burglary h)
      (if (earthquake (house-fault-region h))
          (flip 0.9) (flip 0.8))
      (if (earthquake (house-fault-region h))
          (flip 0.4) (flip 0.01))))
```

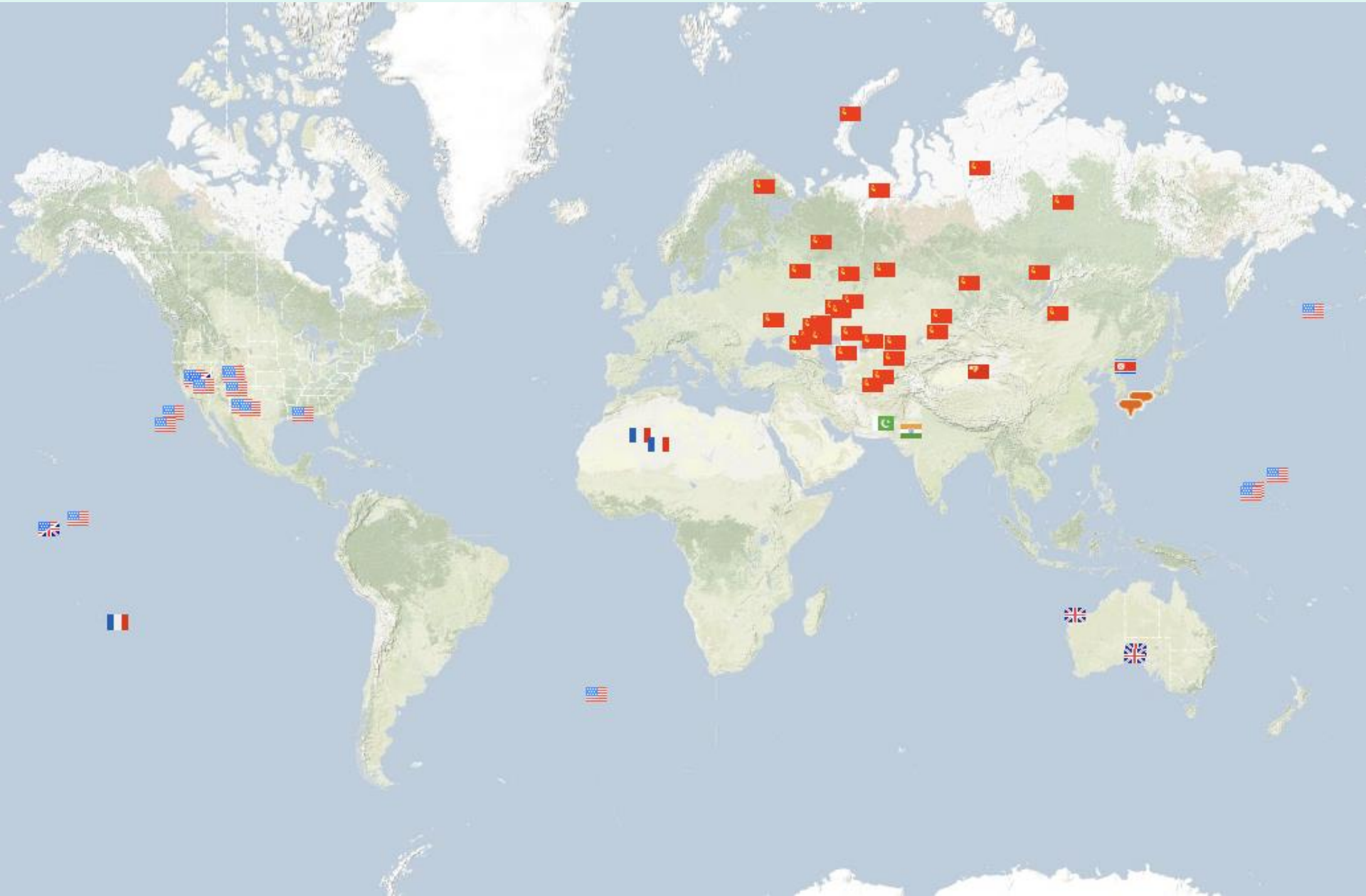
Efficient inference

- Real applications use special-purpose inference
- DARPA PPAML program is trying several solutions
  - Model-specific code generation reduces overhead
    - BLOG compiler gives 100x-300x speedup over original engine
      - 700x compared to widely used Bayes net packages
    - Partial evaluator independent of inference algorithm
  - Modular design with “plug-in” expert samplers
    - E.g., sample a parse tree given sentence + PCFG
    - E.g., sample $X_1,\ldots,X_k$ given their sum
  - Data and process parallelism, special-purpose chips
  - Lifted inference: an analogue of unification/resolution?
  - Adaptive MCMC proposals

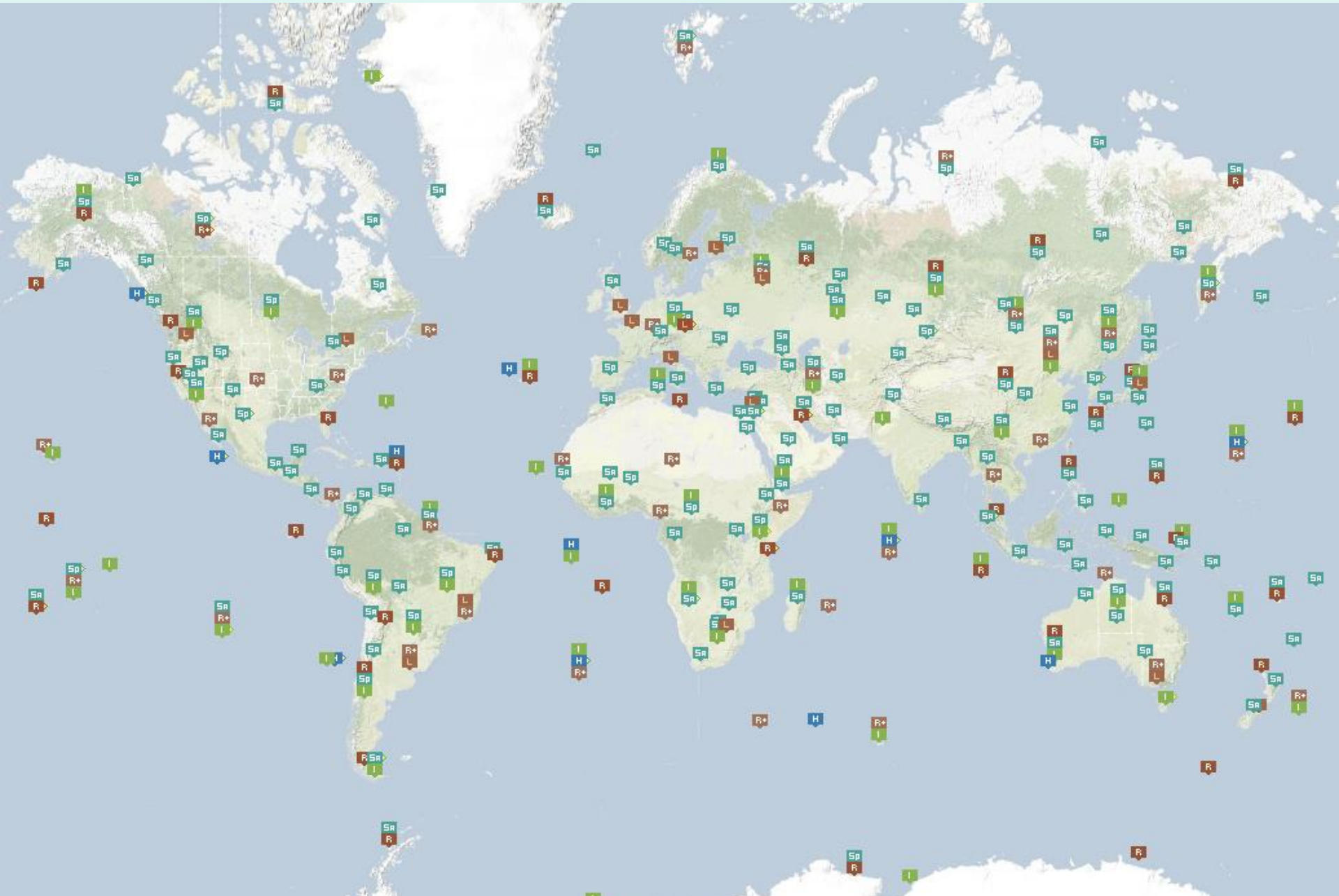
Markov Chain Monte Carlo

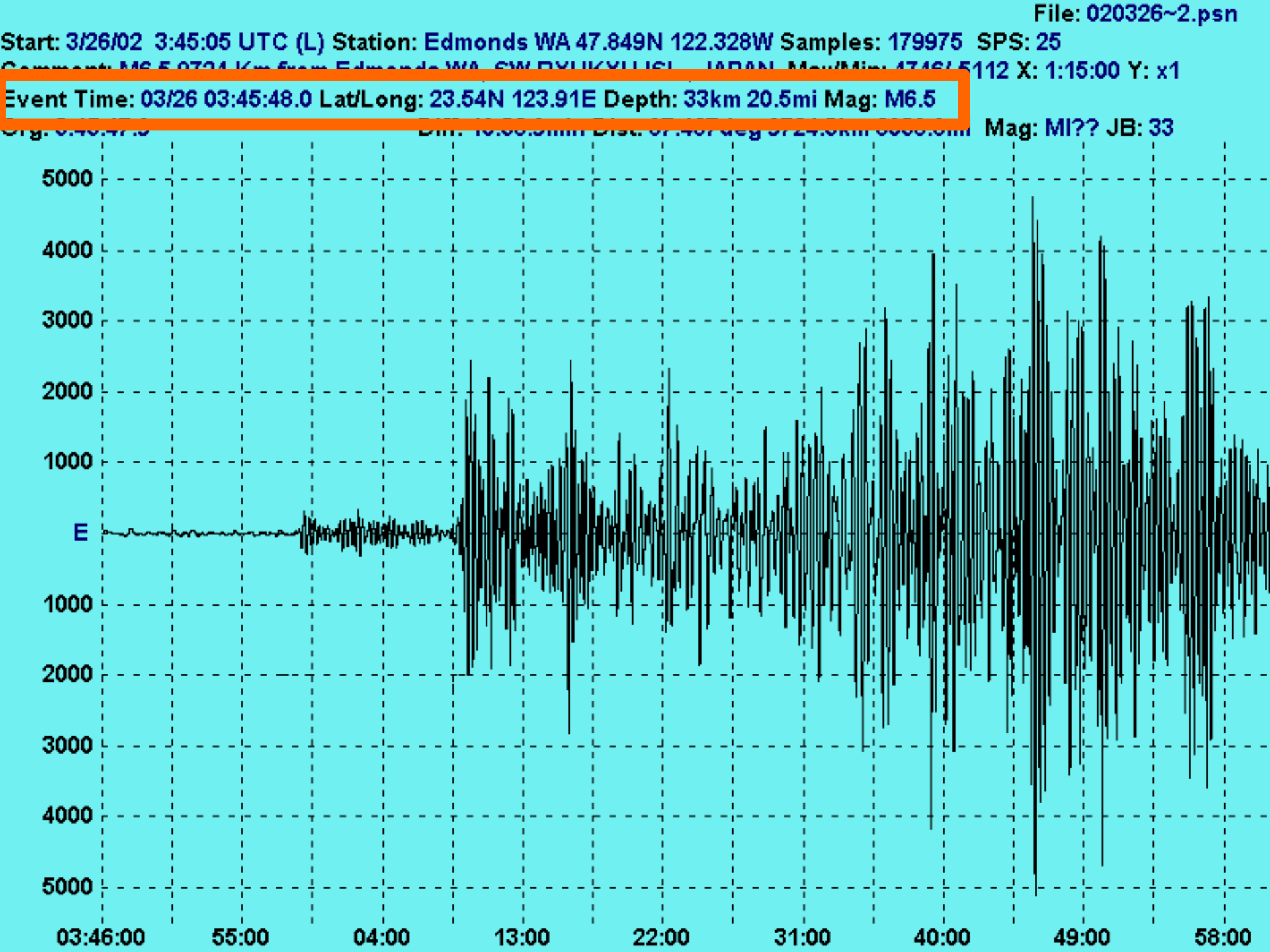


2055 nuclear explosions, 300K deaths



278 monitoring stations (147 seismic)




File: 020326~2.psn
Start: 3/26/02 3:45:05 UTC (L) Station: Edmonds WA 47.849N 122.328W Samples: 179975 SPS: 25
Event Time: 03/26 03:45:48.0 Lat/Long: 23.54N 123.91E Depth: 33km 20.5mi Mag: M6.5
Mag: Ml?? JB: 33
5000
4000
3000
2000
1000
E
1000
2000
3000
4000
5000
03:46:00
55:00
04:00
13:00
22:00
31:00
40:00
49:00
58:00

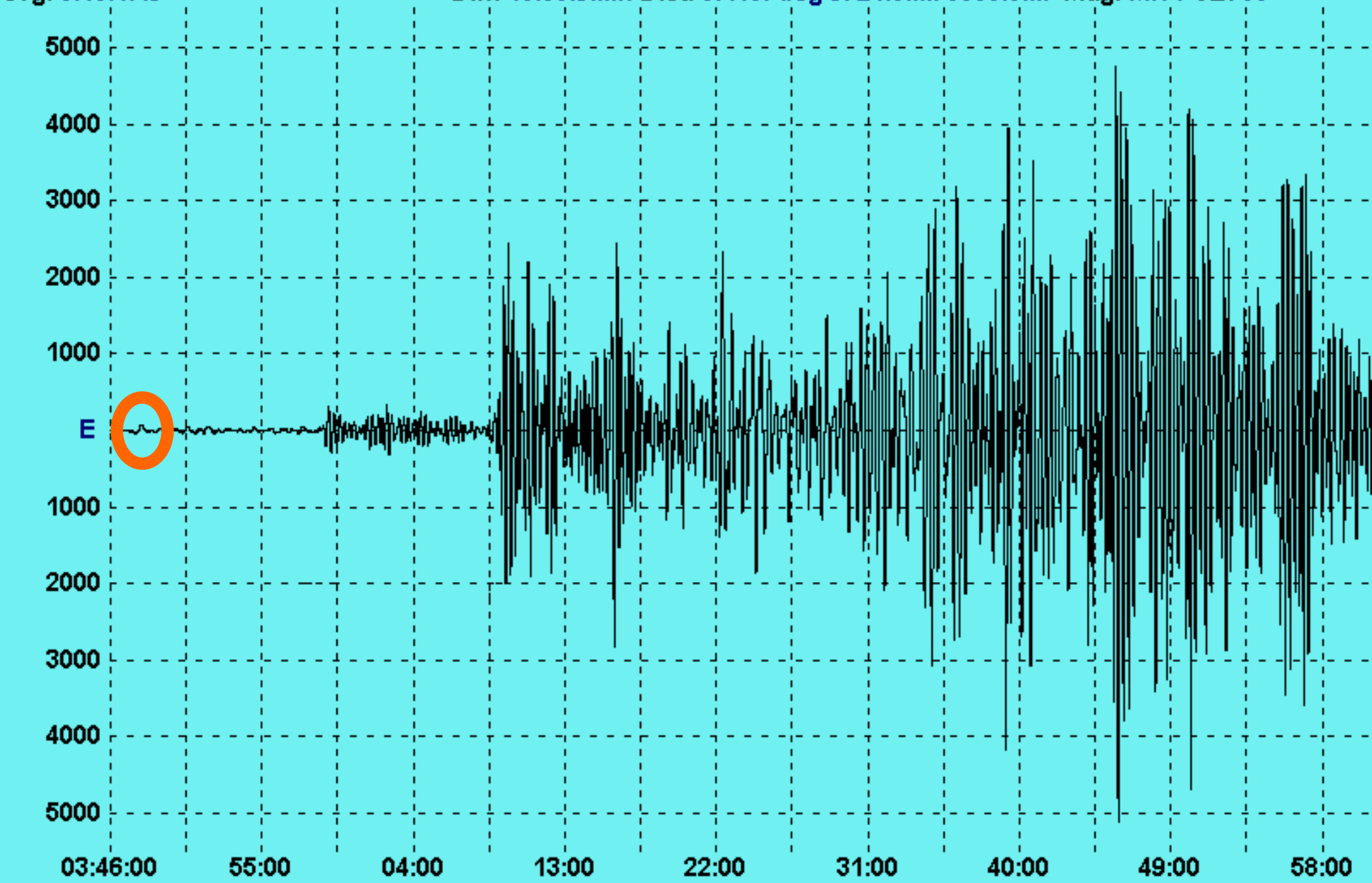
File: 020326~2.psn

Start: 3/26/02 3:45:05 UTC (L) Station: Edmonds WA 47.849N 122.328W Samples: 179975 SPS: 25

Comment: M6.5 9724 Km from Edmonds WA, SW RYUKYU ISL., JAPAN Max/Min: 4746/-5112 X: 1:15:00 Y: x1

Event Time: 03/26 03:45:48.0 Lat/Long: 23.54N 123.91E Depth: 33km 20.5mi Mag: M6.5

Org: 3:45:47.9 Diff: 10:36.9min Dist: 87.467deg 9724.3km 6038.8mi Mag: Ml?? JB: 33



Global seismic monitoring

- *Given*: continuous waveform measurements from a global network of seismometer stations
- *Output*: a *bulletin* listing seismic *events*, with
  - *Time*
  - *Location (latitude, longitude)*
  - *Depth*
  - *Magnitude*

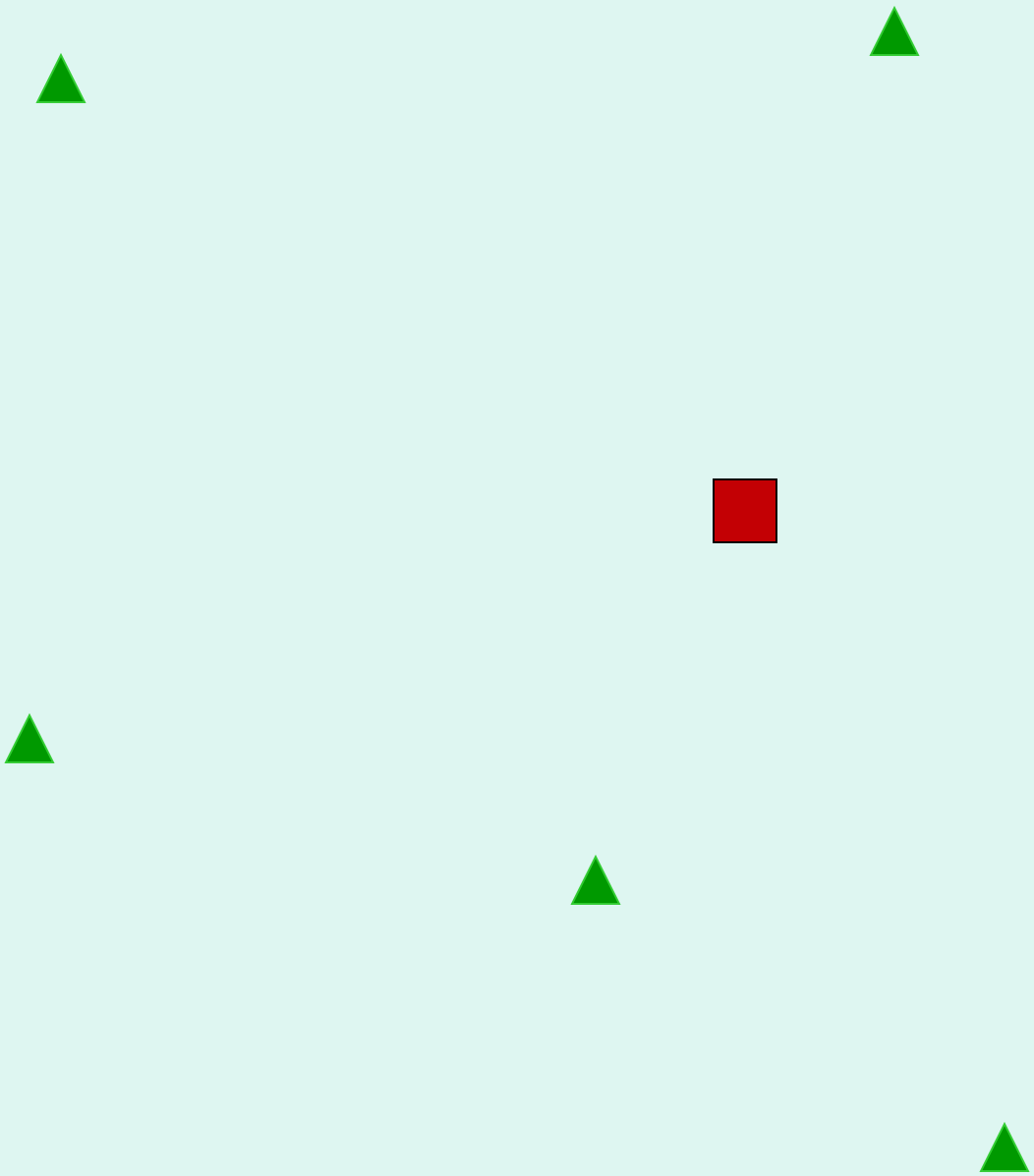
Is this a hard problem?

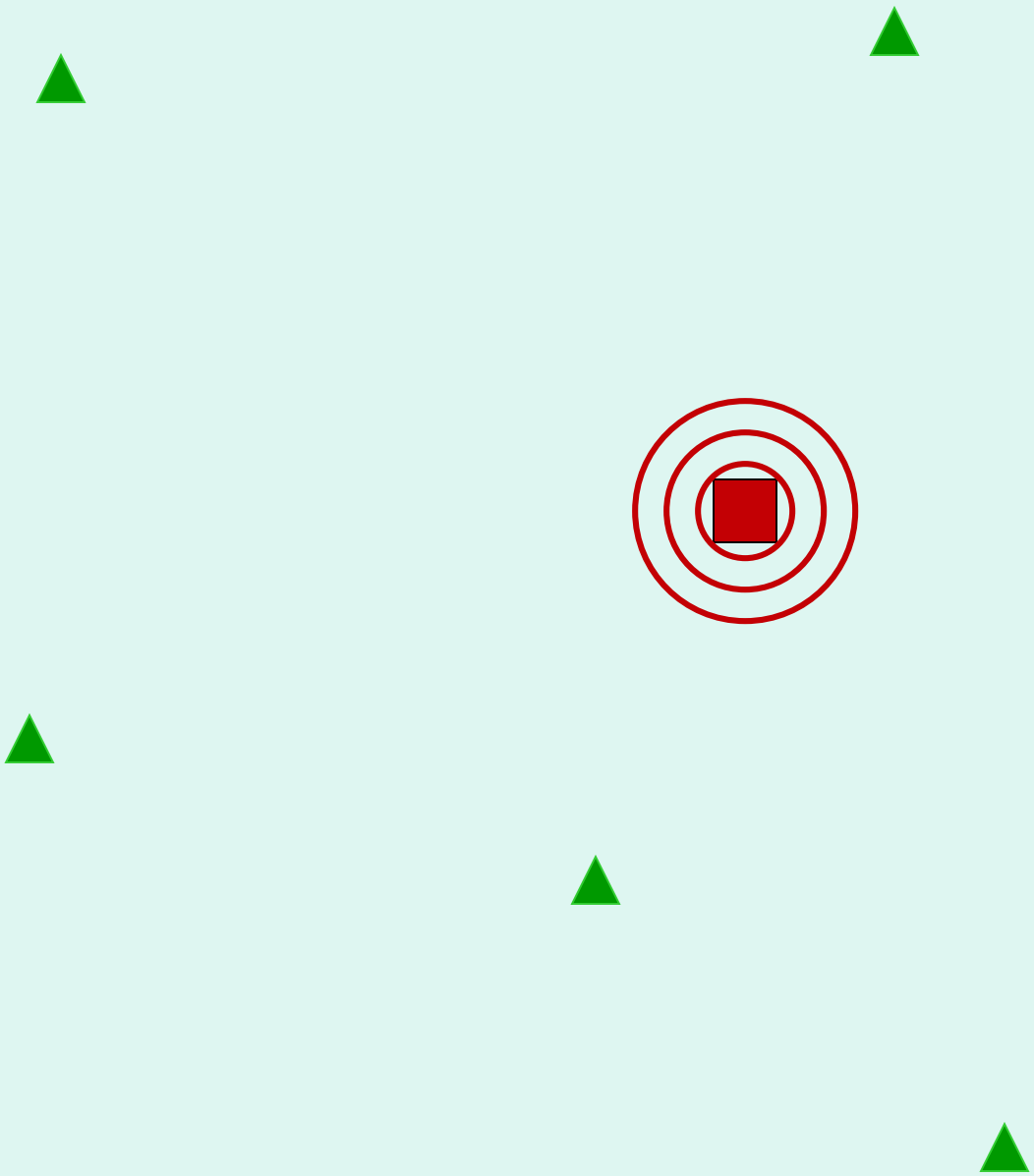
- CTBTO system (GA➔SEL3) developed over 10 years, $100M software plus $1B network
  - Recall 69%, precision 47%
  - 16 human analysts correct or discard SEL3 events, create new events, generate LEB (“ground truth”)
  - Unreliable below magnitude 4 (1kT)

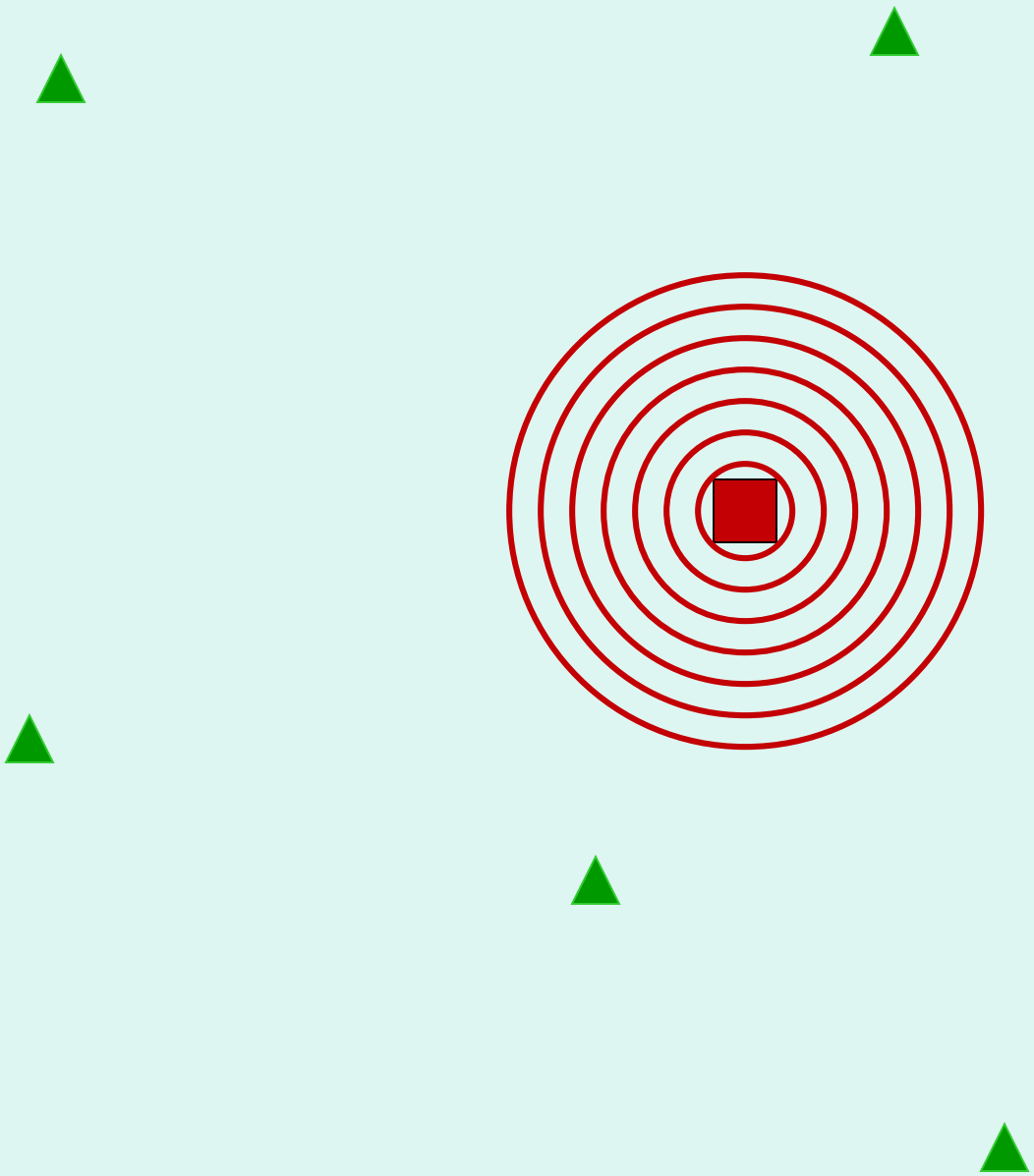
Why is this a hard problem?

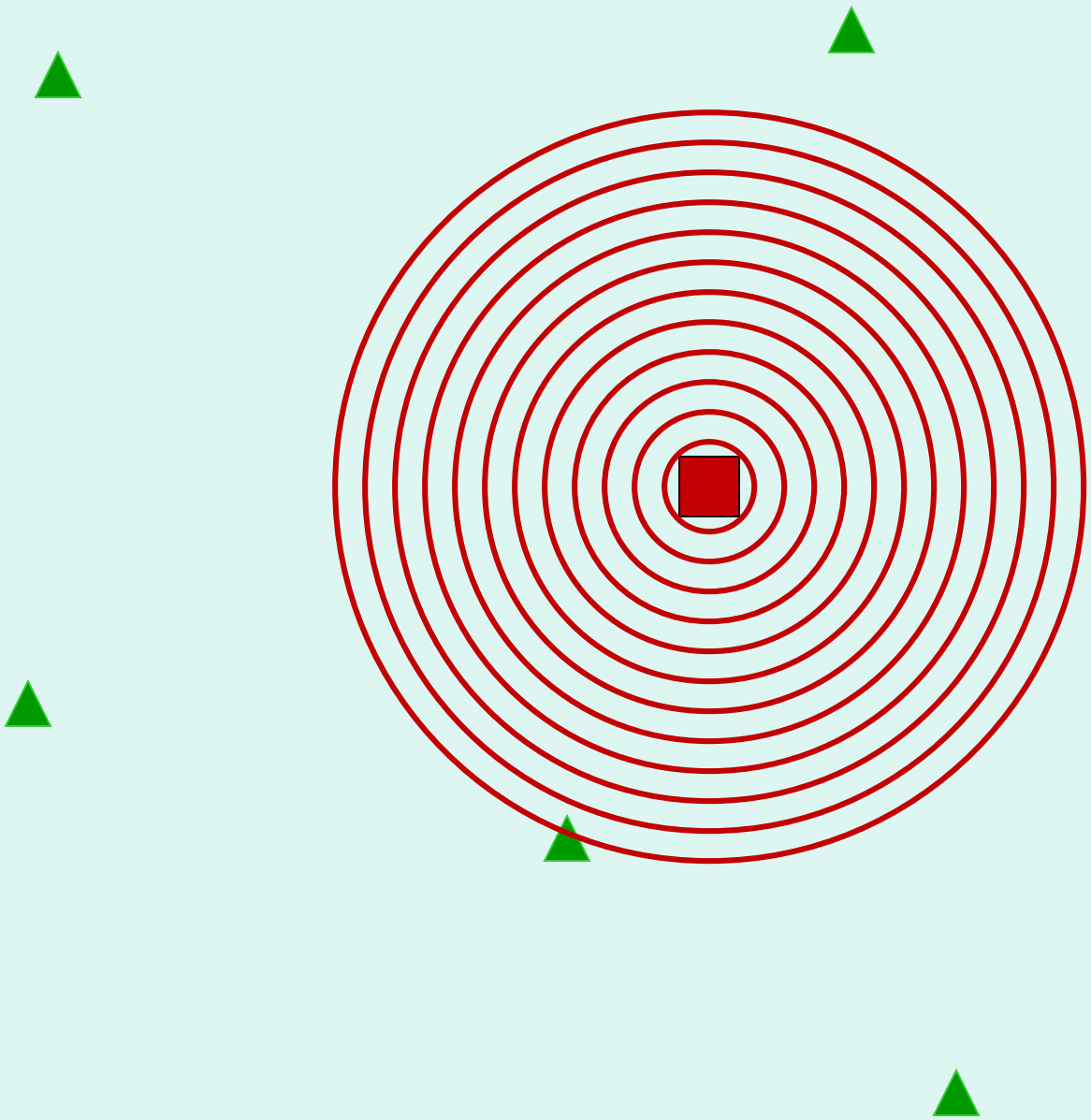
- ~10 000 “detections” per day, *90% false*
- Lots of background noise
- Events generate dozens of wave types (“phases”) with different velocities, paths
- Signals take 15 minutes to several hours to traverse the earth, so they are all mixed up

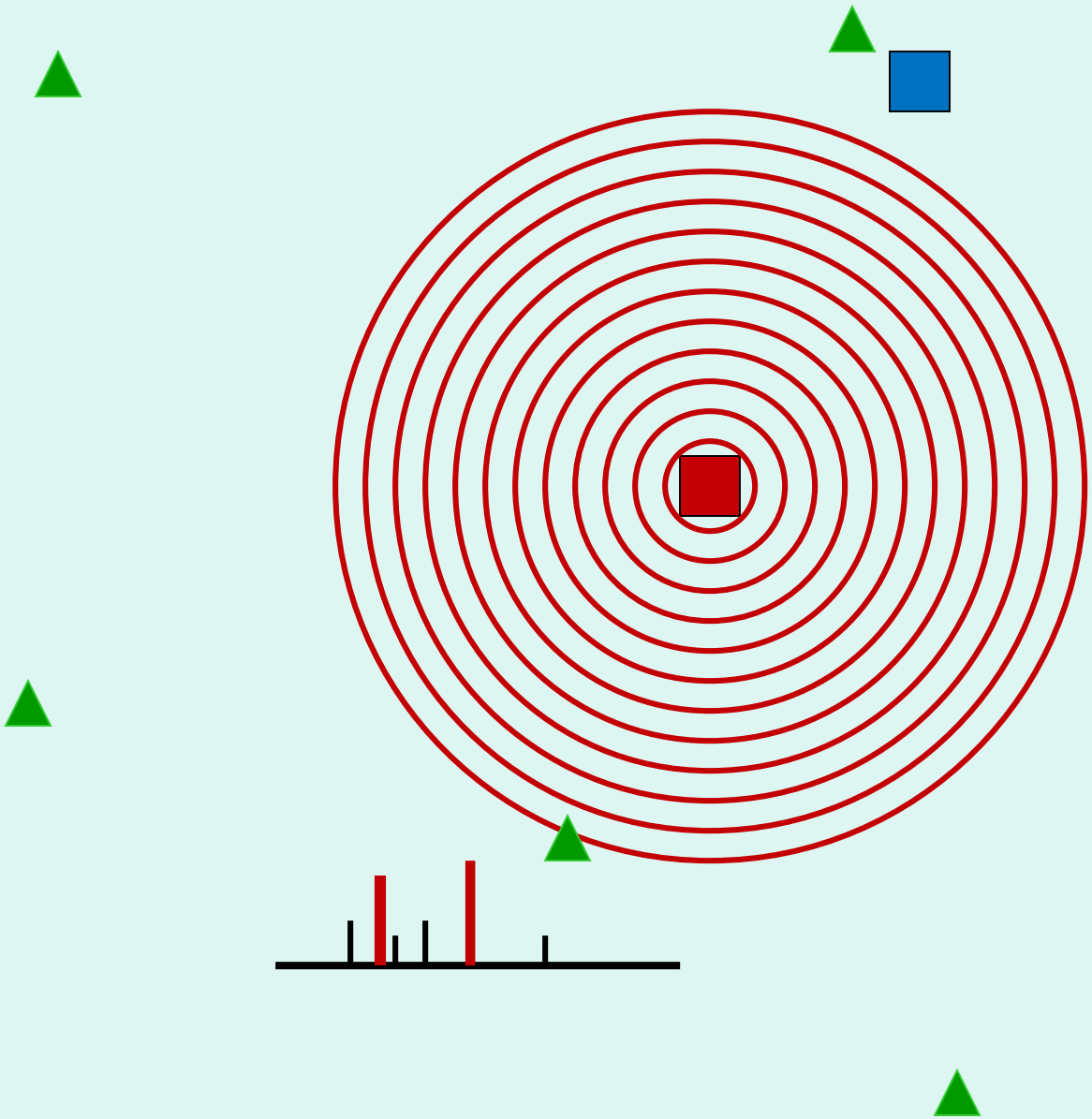
Very short course in seismology

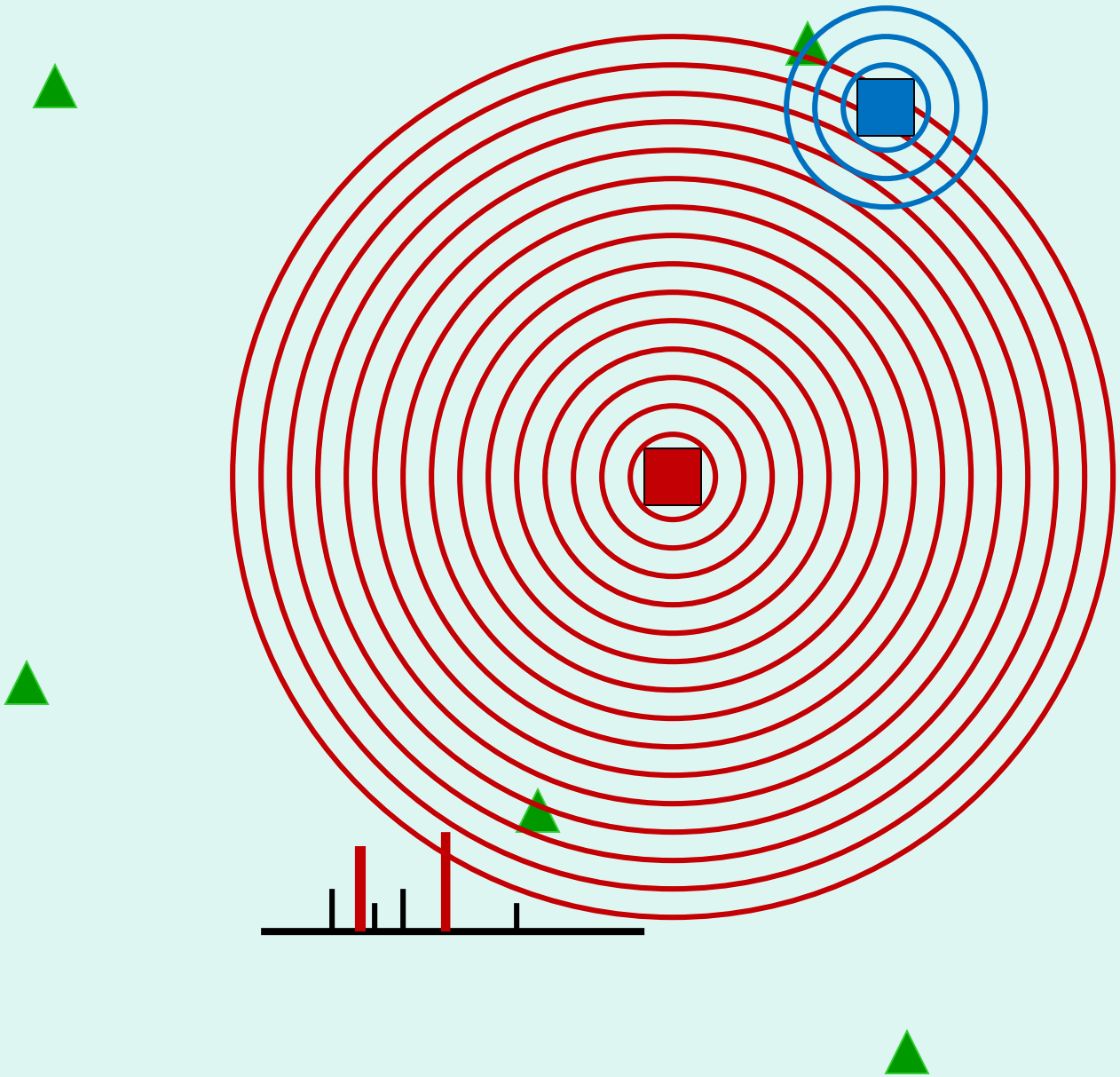


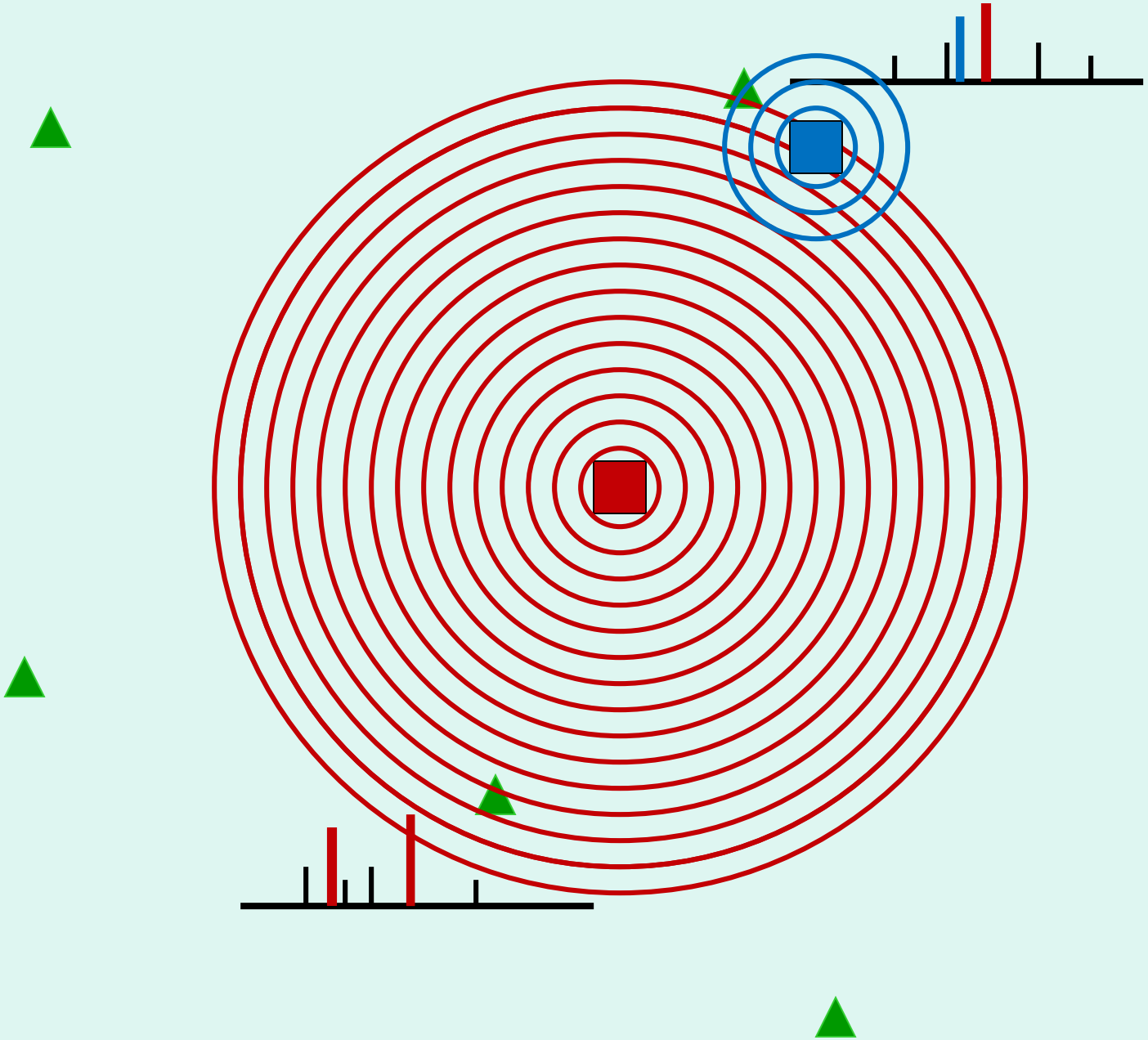


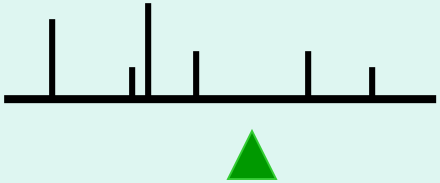


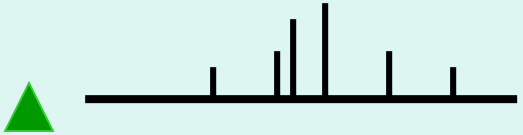


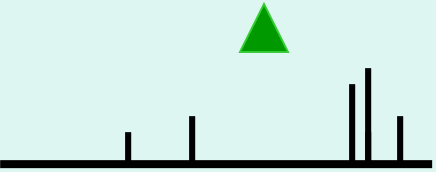


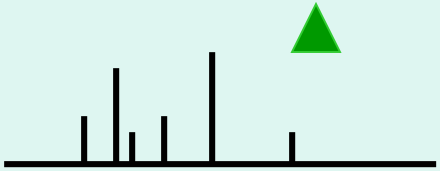


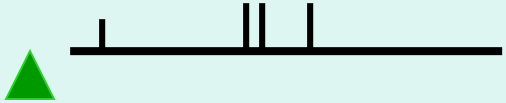












BLOG model for seismology

- Model describes
  - Event occurrence (frequency, location, magnitude)
  - Signal detection probability
  - Signal properties (time, amplitude, direction)
  - Measurement uncertainty
  - Noise processes producing false detections
- Advantages:
  - Prior knowledge, historical data => better results, happy geophysicists
  - Top-down inference disambiguates local signals

#SeismicEvents ~ Poisson[T*λ_e];
Time(e) **~** Uniform(0,T)
IsEarthQuake(e) ~ Bernoulli(.999);
Location(e) ~ if IsEarthQuake(e) then SpatialPrior() else UniformEarthDistribution();
Depth(e) ~ if IsEarthQuake(e) then Uniform[0,700] else 0;
Magnitude(e) ~ Exponential(log(10));
IsDetected(e,p,s) **~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**
#Detections(site = s) ~ Poisson[T*λ_f(s)];
#Detections(event=e, phase=p, station=s) **=** if IsDetected(e,p,s) then 1 else 0;
OnsetTime(a,s) **~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a))
+ Laplace(μ_t(s), σ_t(s))
Amplitude(a,s) **~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))
Azimuth(a,s) **~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
Slowness(a,s) **~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
ObservedPhase(a,s) **~** CategoricalPhaseModel(phase(a))

#SeismicEvents ~ Poisson[T*λ_e];

Time(e) **~** Uniform(0,T)

IsEarthQuake(e) ~ Bernoulli(.999);

Location(e) ~ if IsEarthQuake(e) then SpatialPrior() else UniformEarthDistribution();

Depth(e) ~ if IsEarthQuake(e) then Uniform[0,700] else 0;

Magnitude(e) ~ Exponential(log(10));

IsDetected(e,p,s) **~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**

#Detections(site = s) ~ Poisson[T*λ_f(s)];

#Detections(event=e, phase=p, station=s) = if IsDetected(e,p,s) then 1 else 0;

OnsetTime(a,s) **~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a))
+ Laplace(μ_t(s), σ_t(s))

Amplitude(a,s) **~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))

Azimuth(a,s) **~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

Slowness(a,s) **~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

ObservedPhase(a,s) **~** CategoricalPhaseModel(phase(a))

#SeismicEvents ~ Poisson[T*λ_e];

Time(e) **~** Uniform(0,T)

IsEarthQuake(e) ~ Bernoulli(.999);

Location(e) ~ if IsEarthQuake(e) then SpatialPrior() else UniformEarthDistribution();

Depth(e) ~ if IsEarthQuake(e) then Uniform[0,700] else 0;

Magnitude(e) ~ Exponential(log(10));

IsDetected(e,p,s) **~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**

#Detections(site = s) ~ Poisson[T*λ_f(s)];

#Detections(event=e, phase=p, station=s) **=** if IsDetected(e,p,s) then 1 else 0;

OnsetTime(a,s) **~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a))
+ Laplace(μ_t(s), σ_t(s))

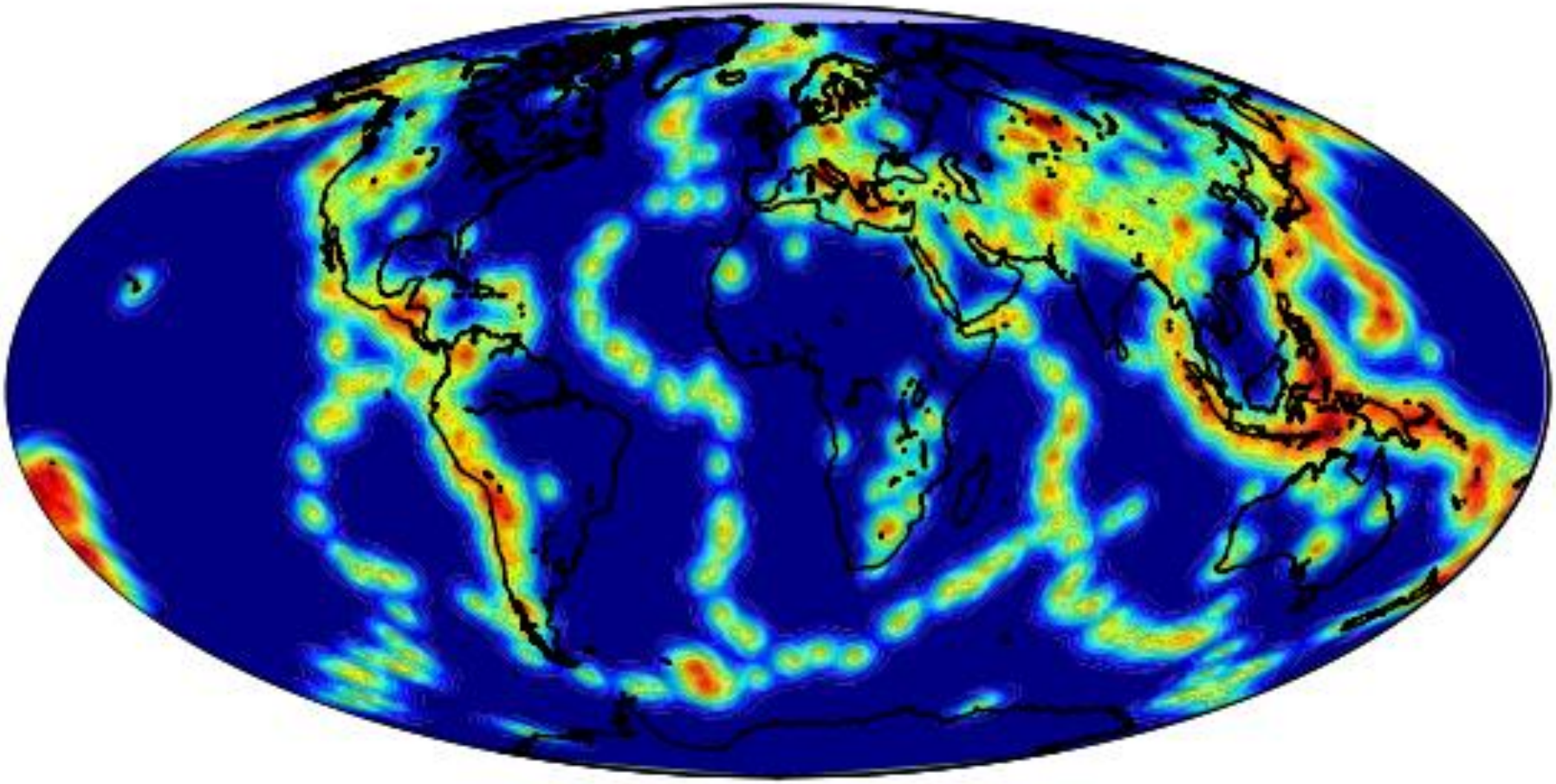
Amplitude(a,s) **~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))

Azimuth(a,s) **~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

Slowness(a,s) **~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

ObservedPhase(a,s) **~** CategoricalPhaseModel(phase(a))

Event location prior



#SeismicEvents ~ Poisson[T*λ_e];

Time(e) **~** Uniform(0,T)

IsEarthQuake(e) ~ Bernoulli(.999);

Location(e) ~ if IsEarthQuake(e) then SpatialPrior() else UniformEarthDistribution();

Depth(e) ~ if IsEarthQuake(e) then Uniform[0,700] else 0;

Magnitude(e) ~ Exponential(log(10));

IsDetected(e,p,s) **~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**

#Detections(site = s) ~ Poisson[T*λ_f(s)];

#Detections(event=e, phase=p, station=s) **=** if IsDetected(e,p,s) then 1 else 0;

OnsetTime(a,s) **~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a))
+ Laplace(μ_t(s), σ_t(s))

Amplitude(a,s) **~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))

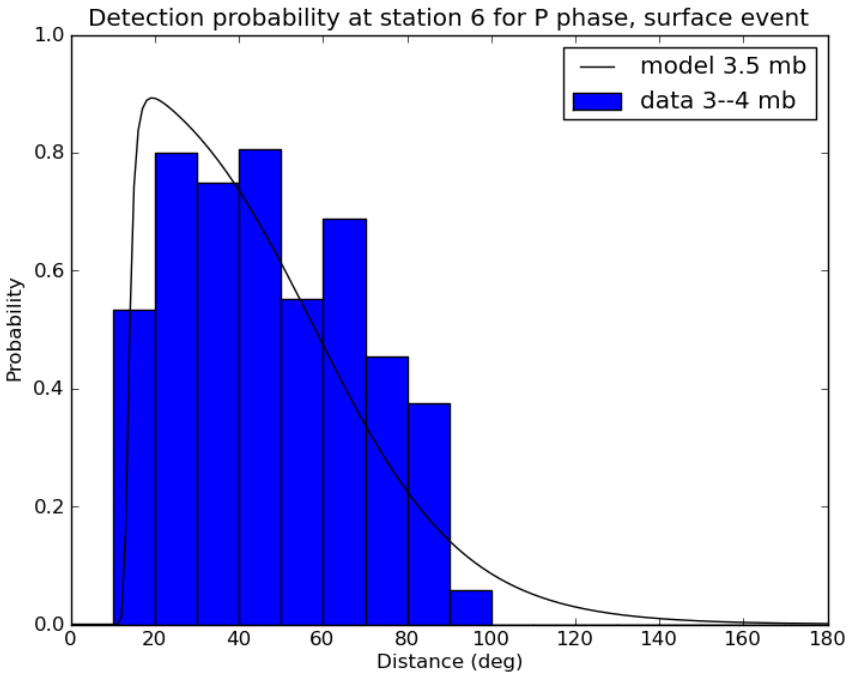
Azimuth(a,s) **~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

Slowness(a,s) **~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

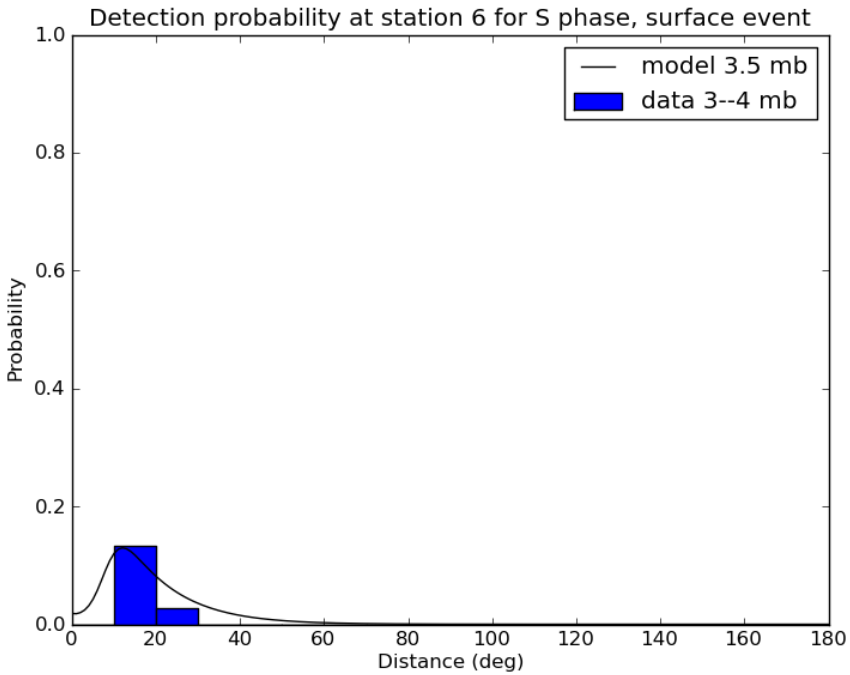
ObservedPhase(a,s) **~** CategoricalPhaseModel(phase(a))

Detection probability as a function of distance (station 6, m_b 3.5)

P phase



S phase



#SeismicEvents ~ Poisson[T*λ_e];

Time(e) **~** Uniform(0,T)

IsEarthQuake(e) ~ Bernoulli(.999);

Location(e) ~ if IsEarthQuake(e) then SpatialPrior() else UniformEarthDistribution();

Depth(e) ~ if IsEarthQuake(e) then Uniform[0,700] else 0;

Magnitude(e) ~ Exponential(log(10));

IsDetected(e,p,s) **~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**

#Detections(site = s) ~ Poisson[T*λ_f(s)];

#Detections(event=e, phase=p, station=s) **=** if IsDetected(e,p,s) then 1 else 0;

OnsetTime(a,s) **~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a))
+ Laplace(μ_t(s), σ_t(s))

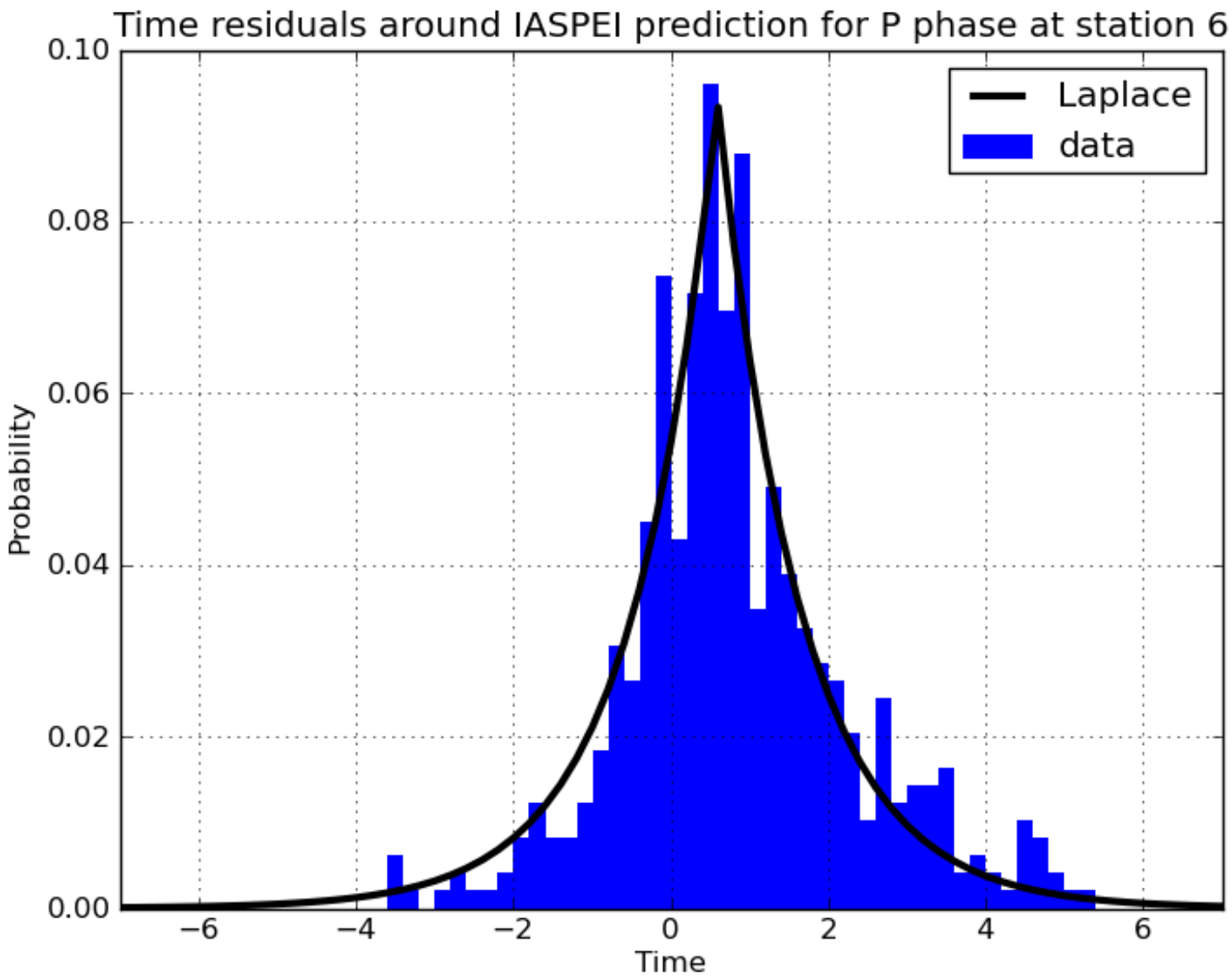
Amplitude(a,s) **~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))

Azimuth(a,s) **~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

Slowness(a,s) **~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

ObservedPhase(a,s) **~** CategoricalPhaseModel(phase(a))

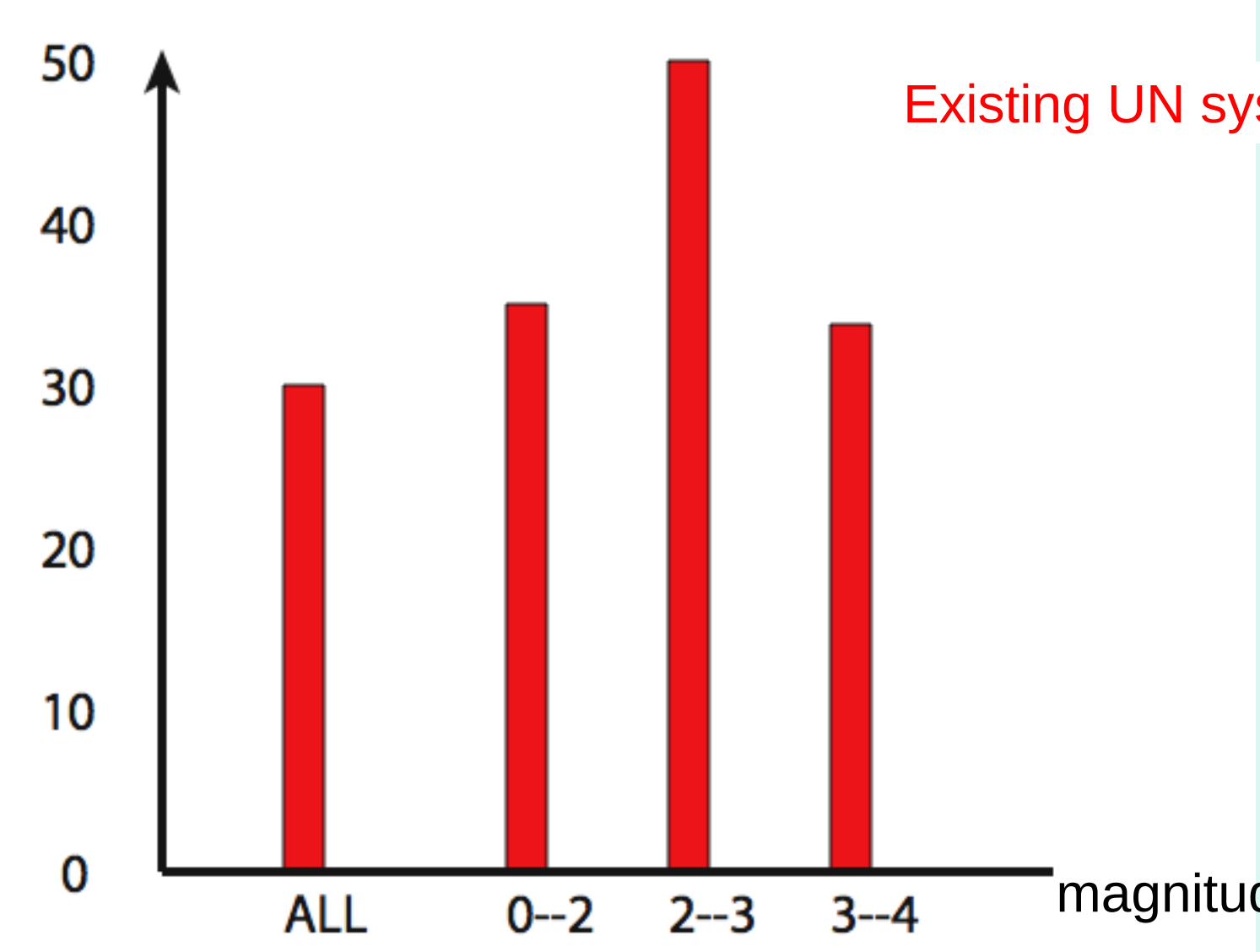
Travel-time residual (station 6)



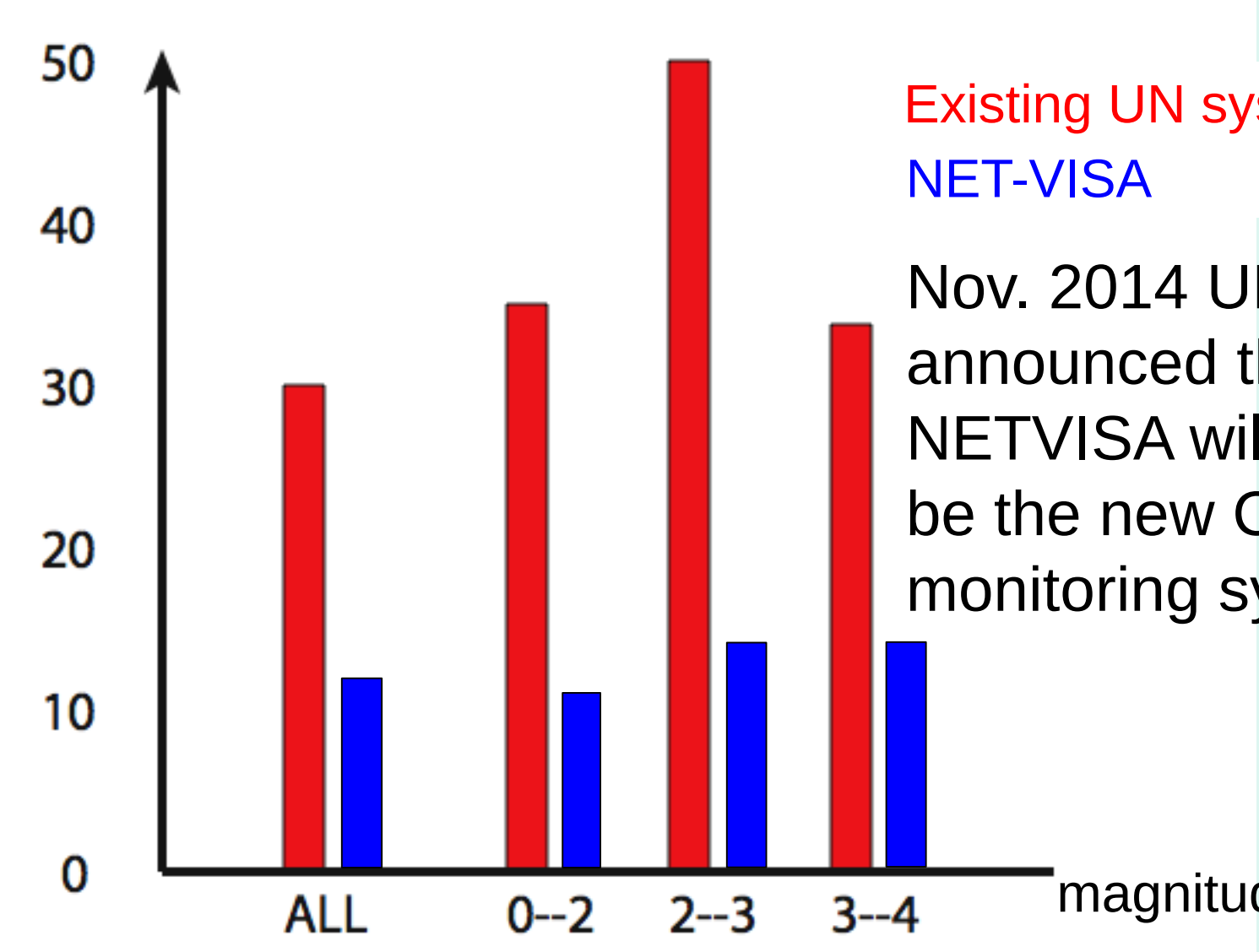
Evaluation

- 11 weeks / 1 week train/test split
- Evaluated using human-expert LEB as “ground truth”

Fraction of events missed

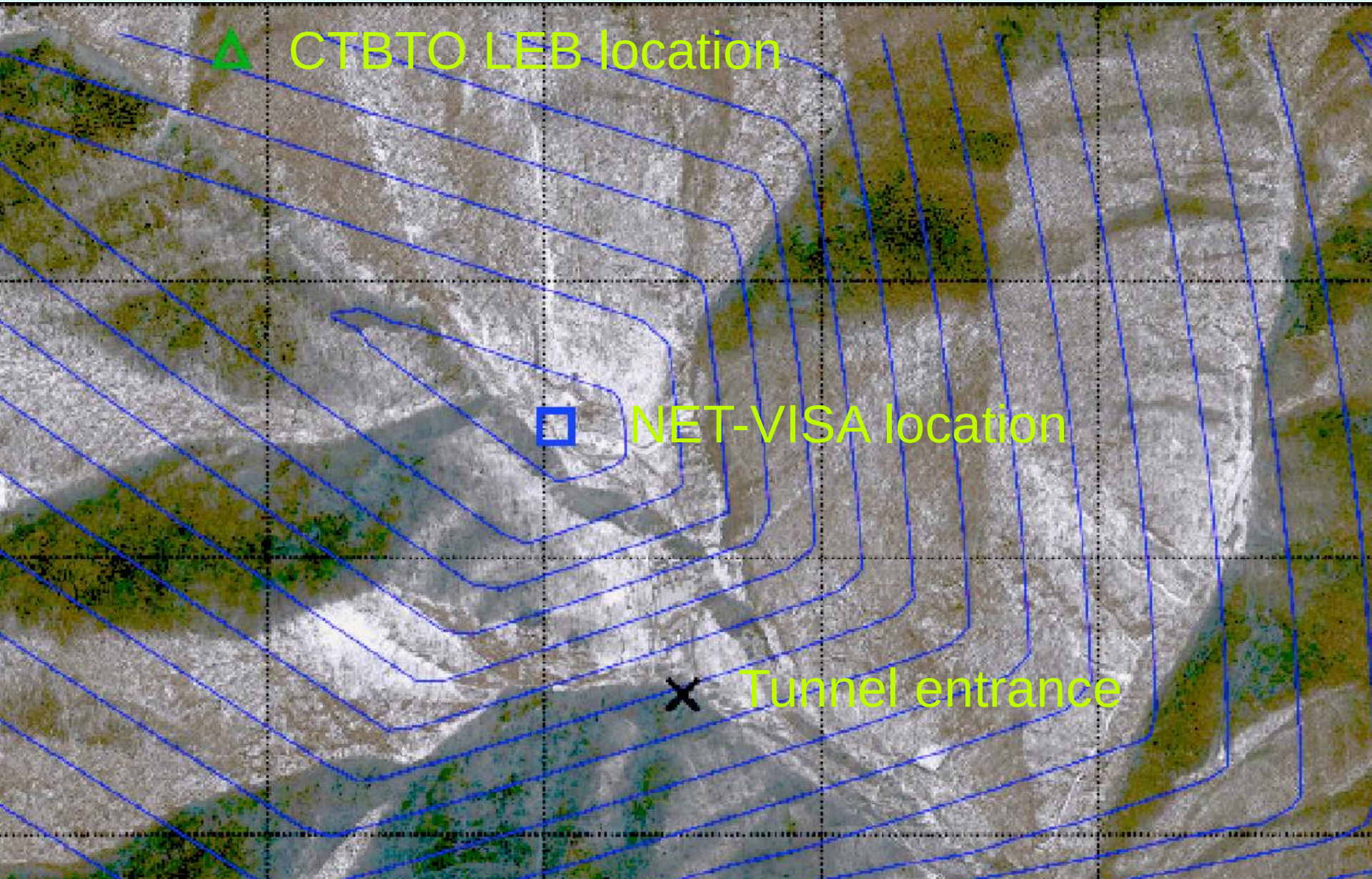


Fraction of events missed

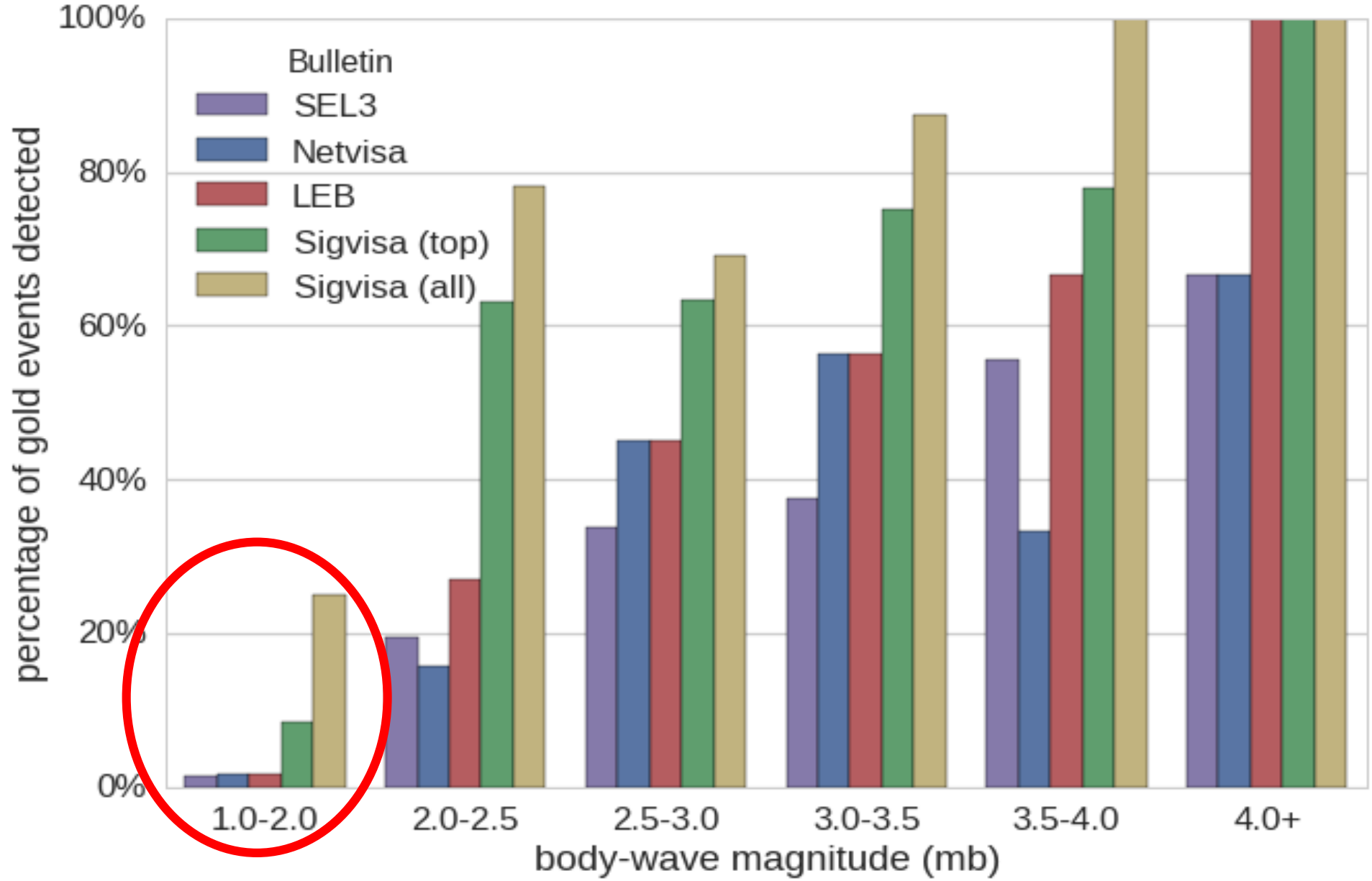


Nov. 2014 UN announced that NETVISA will be the new CTBT monitoring system

February 12, 2013 DPRK test



SIGVISA on Western US dataset



Tracking moving objects in video

- Standard model (Friedman & Russell, 1997; Stauffer & Grimson, 1998):
  - Each pixel sampled from 3-component mixture for background, foreground, shadow
  - Mixture parameters estimated online
- Improvement 1: temporal persistence for each pixel state: per-pixel HMM
- Improvement 2: spatial contiguity of pixel states within each frame: Ising potential

BLOG model (cleaned up a bit)

```
…
Intensity(x,y,t) ~
    MultivarGaussian(Mean(PixelState(x,y,t),x,y),
                     Variance(PixelState(x,y,t),x,y));
PixelState(x,y,t) ~
    Categorical(MixtureWeights(x,y,t));
…
```

BLOG model (cleaned up a bit)

```
…
Intensity(x,y,t) ~
    MultivarGaussian(Mean(PixelState(x,y,t),x,y),
                     Variance(PixelState(x,y,t),x,y));
PixelState(x,y,t) ~
    if t==0 then Categorical(MixtureWeights(x,y,0))
    else TransitionModel(x,y,PixelState(x,y,t-1));
…
```

Completely unsupervised text understanding

#Object ~ OM(3,1);

#Relation ~ OM(2,1)

Dictionary(r) ~ Dirichlet(α,StringList);

Sparsity(r) ~ Beta(10,1000);
Holds(r,x,y) ~ Boolean(Sparsity(r));

ChosenFact(s) ~ Uniform({f : Holds(f)})

Subject(s) = Arg1(ChosenFact(s))

Object(s) = Arg2(ChosenFact(s))

Verb(s) ~
Categ(Dictionary(Rel(ChosenFact(s))))

There are many objects in the world

There are quite a few relations

Relations are expressed by strings

Some (very few) objects are related to each other by any given relation

People somehow choose facts to say

Subject of sentence is 1st arg of fact

Object of sentence is 2nd arg of fact

Verb of sentence is the relation string

Evidence: unsupervised sentence data (NYT subset, from McCallum group)

Query: what is true in the world?

Relation [rel_46] : text patterns

appos->**unit**->prep->**of**->pobj
appos->**part**->prep->**of**->pobj
nn<-**unit**->prep->**of**->pobj
partmod->**own**->prep->**by**->pobj
rcmod->**own**->prep->**by**->pobj
appos->**subsidiary**->prep->**of**->pobj
rcmod->**part**->prep->**of**->pobj
rcmod->**unit**->prep->**of**->pobj
poss<-**parent**->appos
appos->**division**->prep->**of**->pobj
pobj<-**of**<-prep<-**office**->appos->**part**->prep->**of**->pobj
pobj<-**of**<-prep<-**unit**->appos->**part**->prep->**of**->pobj
nn<-**division**->prep->**of**->pobj
appos->**unit**->nn
nsubjpass<-**own**->prep->**by**->pobj
nn<-**office**->prep->**of**->pobj

Relation [rel_46] : extracted facts

rel_46(ABC, Walt Disney Company)
rel_46(American Airlines, AMR Corporation)
rel_46(American, AMR Corporation)
rel_46(Arnold Worldwide, Arnold Worldwide Partners division)
rel_46(BBDO Worldwide, Omnicom Group)
rel_46(Bozell Worldwide, Bozell)
rel_46(Chicago, DDB Worldwide)
rel_46(Conde Nast Publications, Advance Publications)
rel_46(DDB Needham Worldwide, Omnicom Group)
rel_46(DDB Worldwide, Omnicom Group)
rel_46(Eastern, Texas Air Corporation)
rel_46(Electronic Data Systems Corporation, General Motors Corporation)
rel_46(Euro RSCG Worldwide, Havas Advertising)
rel_46(Euro RSCG Worldwide, Havas)
rel_46(Fallon Worldwide, Publicis Groupe)
rel_46(Foote, True North Communications)
rel_46(Fox, News Corporation)
rel_46(Goodby, Omnicom Group)
rel_46(Grey Worldwide, Grey Global Group)
rel_46(Hughes, General Motors Corporation)
rel_46(J. Walter Thompson, WPP Group)
rel_46(Kellogg Brown & Root, Halliburton)
rel_46(Kellogg, Halliburton)
rel_46(Kraft General Foods, Philip Morris Cos.)
rel_46(Lorillard Tobacco, Loews Corporation)
rel_46(Lowe Group, Interpublic Group of Companies)
rel_46(McCann-Erickson World Group, Interpublic Group of Companies)
rel_46(NBC, General Electric Company)
rel_46(New York, BBDO Worldwide)
rel_46(New York, Hill)
rel_46(Ogilvy & Mather Worldwide, WPP Group)
rel_46(Saatchi & Saatchi, Publicis Groupe)
rel_46(Salomon Smith Barney, Citigroup)
rel_46(San Francisco, Foote)
rel_46(Sears Receivables Financing Group Inc., Sears)
rel_46(TBWA Worldwide, Omnicom Group)
rel_46(United, UAL Corporation)
rel_46(United, UAL)
rel_46(Young & Rubicam, WPP Group)

Lots more to do

- Question answering
- Structural learning
- Semantics and inference with functions of continuous arguments
- Experiment with many applications
- Tackle some harder problems: real vision, real NLP

Thank you!

Research funded by DARPA, CTBTO, DTRA, ANR, and Région d’Île de France

Questions?