Unifying logic and probability

A "New Dawn" for Artificial Intelligence?

Stuart Russell
Computer Science Division, UC Berkeley
LIP6, Université Pierre et Marie Curie

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

      On(*color*,*piece*,*x*,*y*,*t*)

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!!

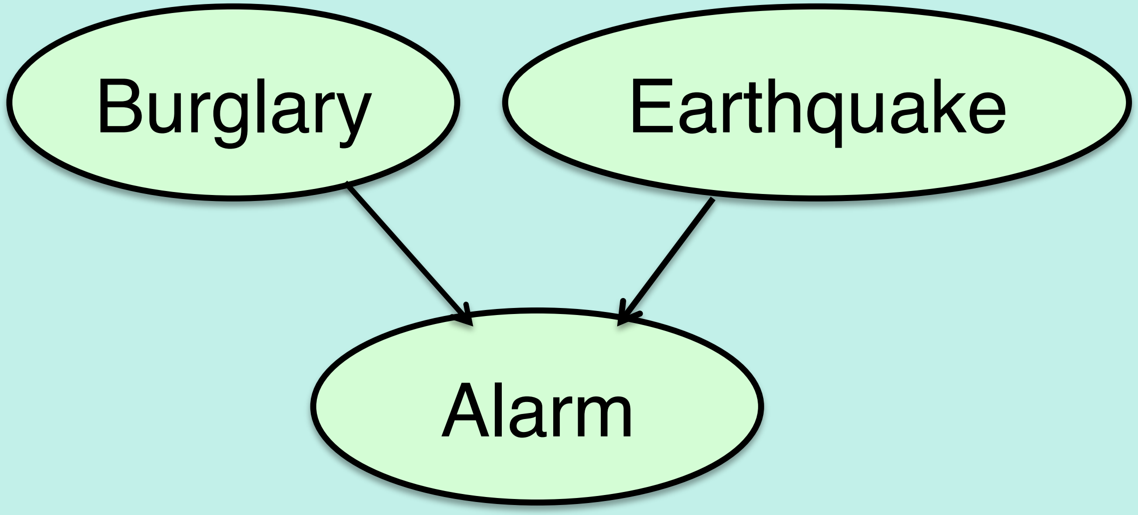
The world is uncertain!!

Good Old-Fashioned AI:
first-order logic

Modern AI:
probabilistic graphical models

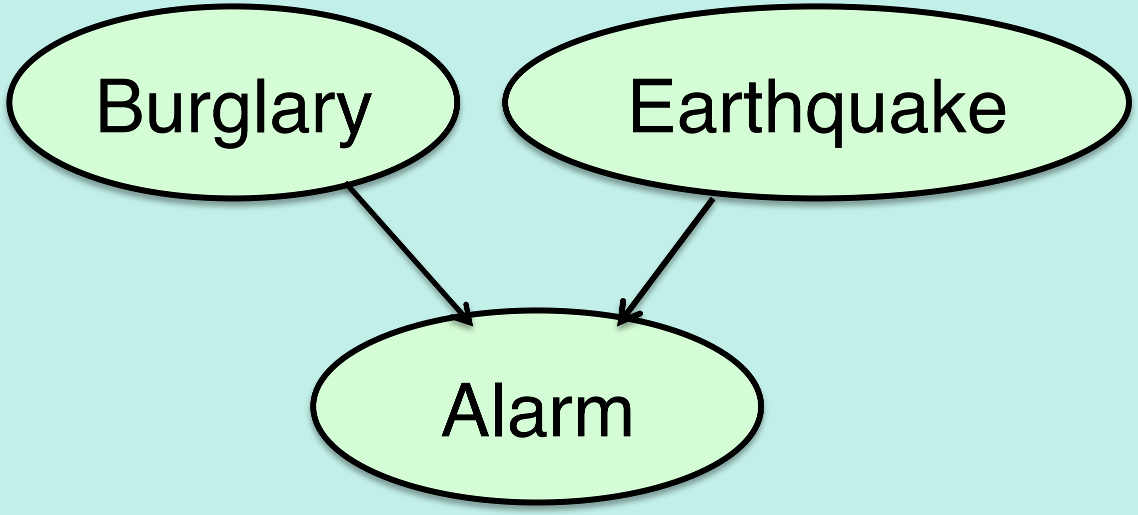
Bayesian networks

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



Bayesian networks

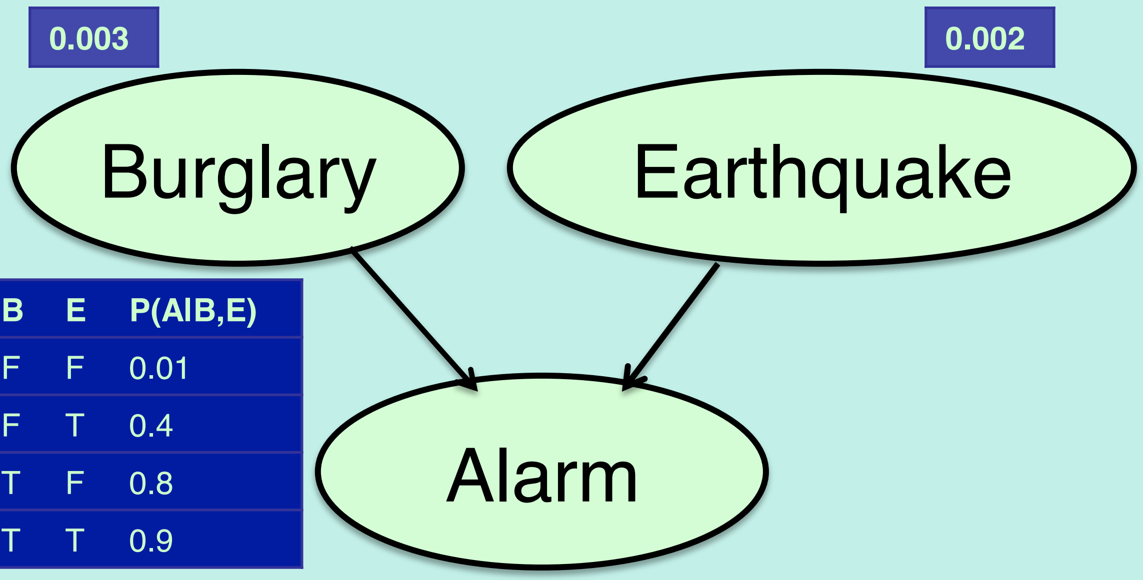
Define distributions on all possible ***propositional*** worlds



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

Bayesian networks

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

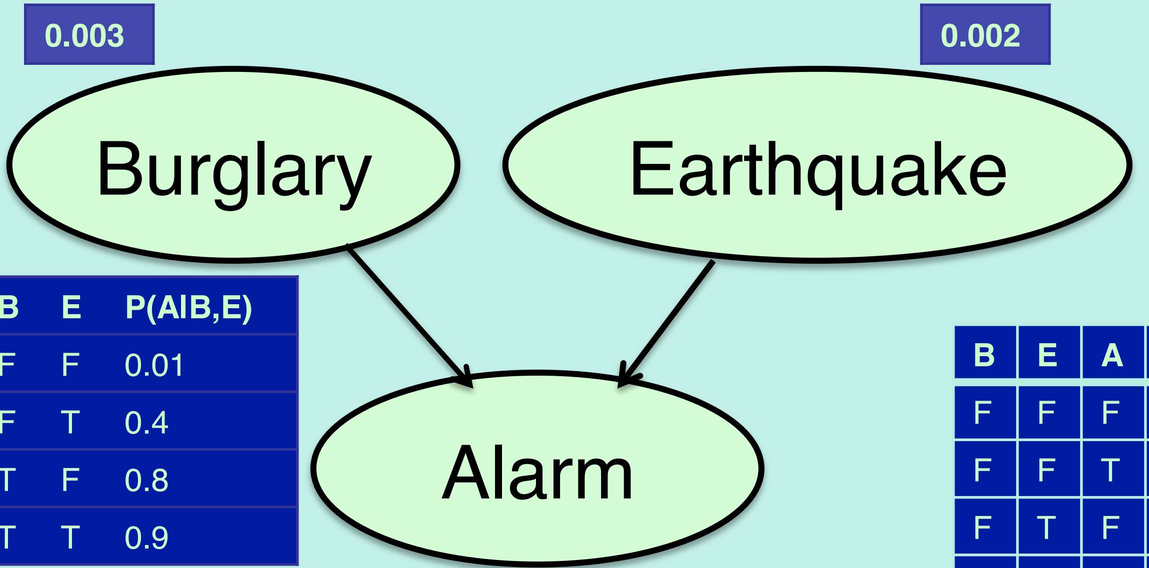


B	E	P(AIB,E)
F	F	0.01
F	T	0.4
T	F	0.8
T	T	0.9

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

Bayesian networks

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



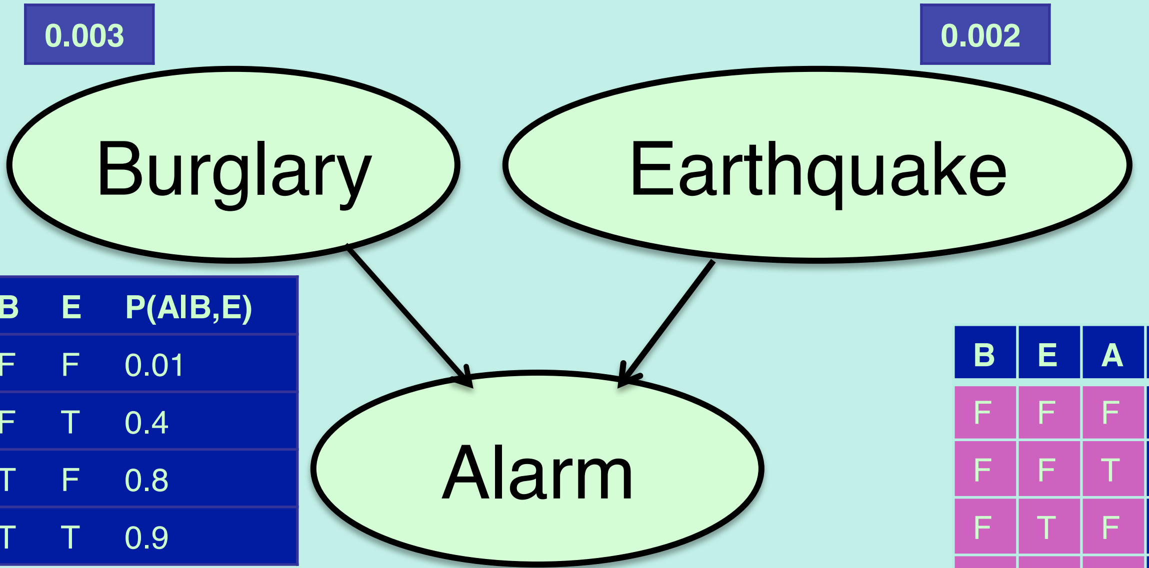
B	E	P(AIB,E)
F	F	0.01
F	T	0.4
T	F	0.8
T	T	0.9

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

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

Bayesian networks

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



B	E	$P(A \mid B,E)$
F	F	0.01
F	T	0.4
T	F	0.8
T	T	0.9

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

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

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

“AI is in bloom again … At last, artificial intelligences are thinking along human lines.”

~~“AI is in bloom again … At last, artificial intelligences are thinking along human~~ lines.”

~~"AI is in bloom again … At last, artificial intelligences are thinking along human lines."~~

"A technique [that] combines the logical underpinnings of the old AI with the power of statistics and probability … is finally starting to disperse the fog of the long AI winter."

First-order probabilistic languages

- Gaifman [1964]: we can unify logic and probability by defining distributions over possible worlds that are ***first-order model structures*** (objects and relations)

First-order probabilistic languages

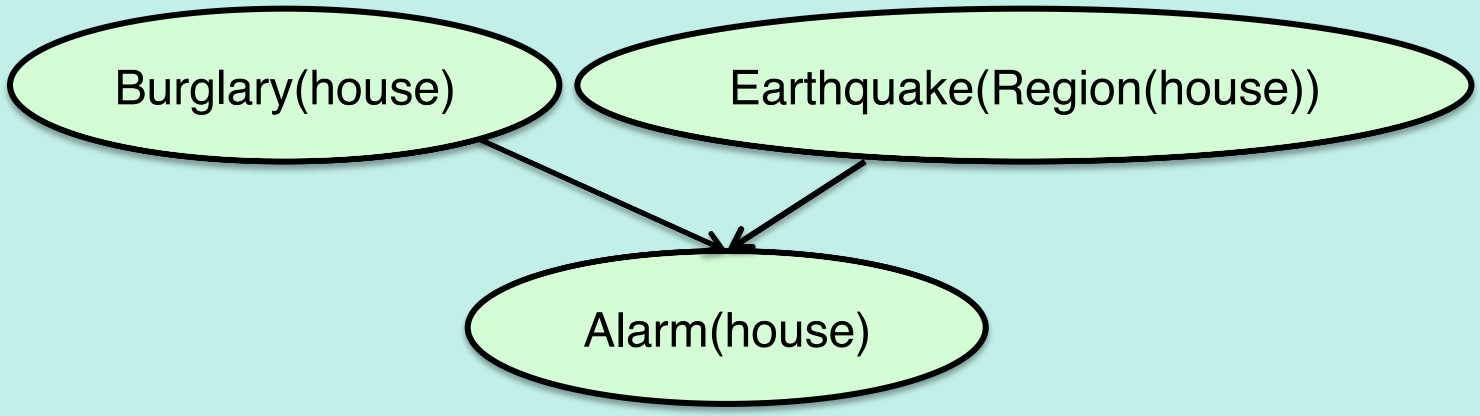
- Gaifman [1964]: we can unify logic and probability by defining distributions over possible worlds that are ***first-order model structures*** (objects and relations)
  - Not obvious how to do it – infinitely many parameters??

First-order probabilistic languages

- Gaifman [1964]: we can unify logic and probability by defining distributions over possible worlds that are ***first-order model structures*** (objects and relations)
  - Not obvious how to do it – infinitely many parameters??
- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea

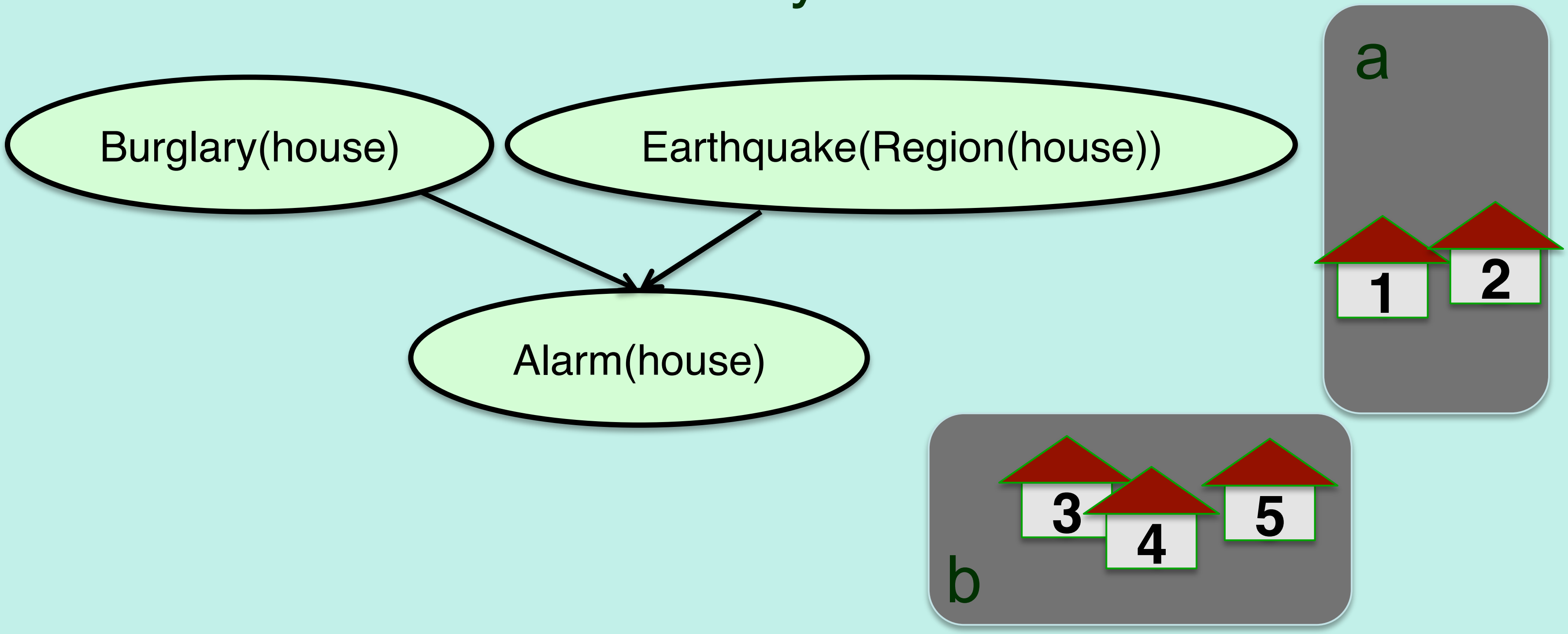
First-order probabilistic languages

- Gaifman [1964]: we can unify logic and probability by defining distributions over possible worlds that are ***first-order model structures*** (objects and relations)
  - Not obvious how to do it – infinitely many parameters??
- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea



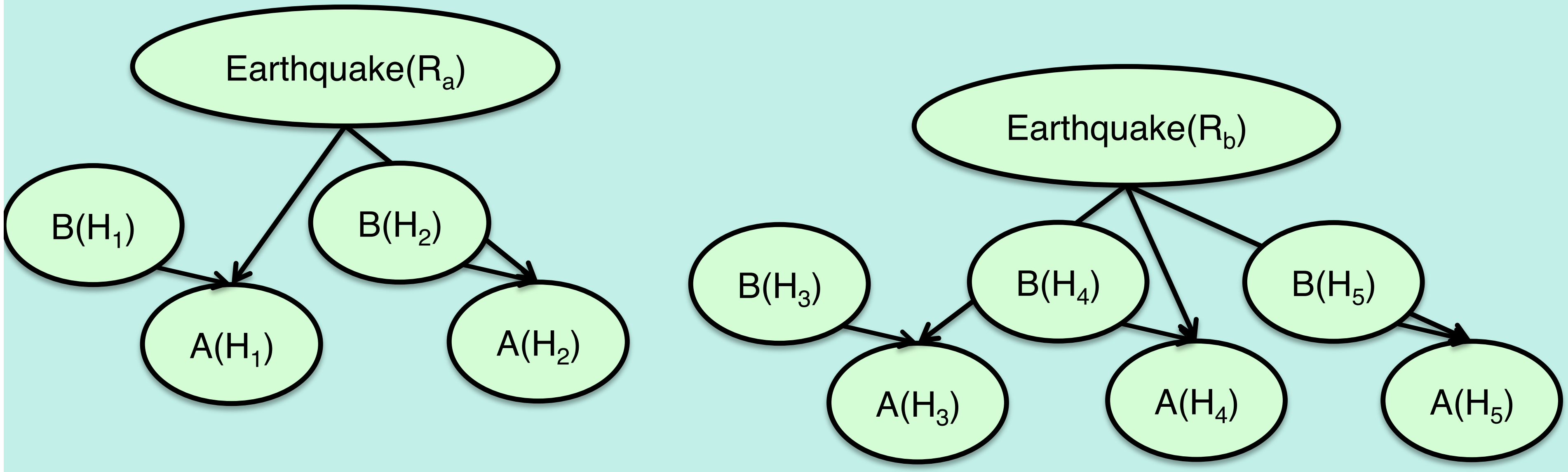
First-order probabilistic languages

- Gaifman [1964]: we can unify logic and probability by defining distributions over possible worlds that are ***first-order model structures*** (objects and relations)
  - Not obvious how to do it – infinitely many parameters??
- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea



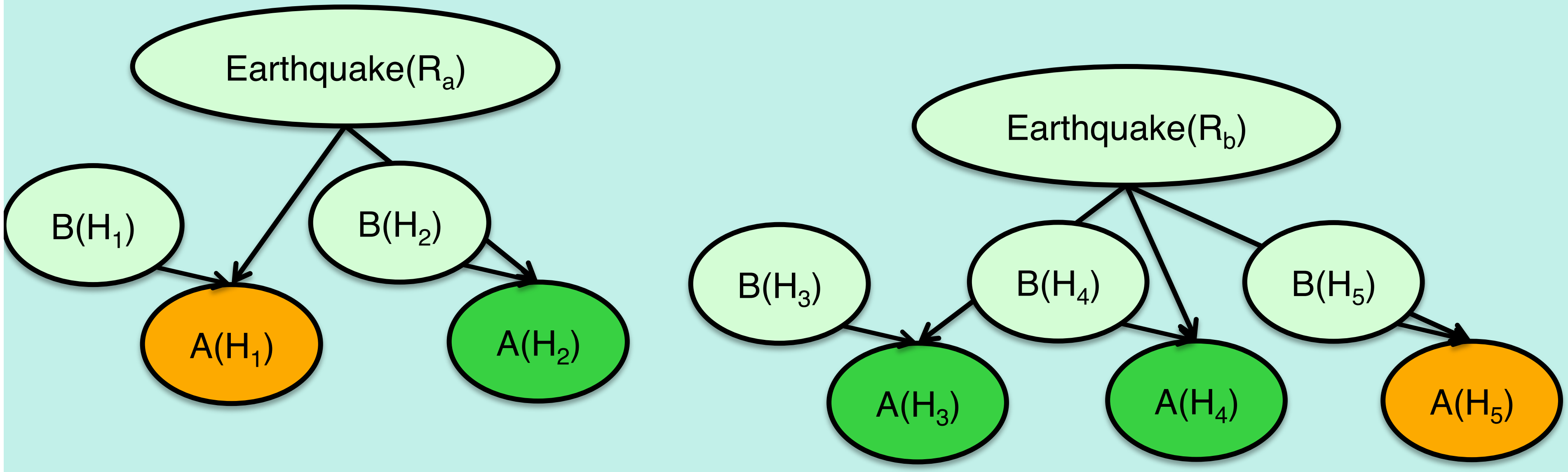
First-order probabilistic languages

- Gaifman [1964]: we can unify logic and probability by defining distributions over possible worlds that are ***first-order model structures*** (objects and relations)
  - Not obvious how to do it – infinitely many parameters??
- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea



First-order probabilistic languages

- Gaifman [1964]: we can unify logic and probability by defining distributions over possible worlds that are ***first-order model structures*** (objects and relations)
  - Not obvious how to do it – infinitely many parameters??
- Simple idea (1990s): combine logical notation for random variables with Bayes net factorization idea



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 (Pfeffer 03, Goodman et al 08): distribution on execution traces of stochastic programs

Closed vs open universes

Given

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

How many children does Bill have?

Closed vs open universes

Given

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

How many children does Bill have?

Closed-universe (Herbrand) semantics:

2

Closed vs open universes

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 ∞

AI: intelligent systems in the real world

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

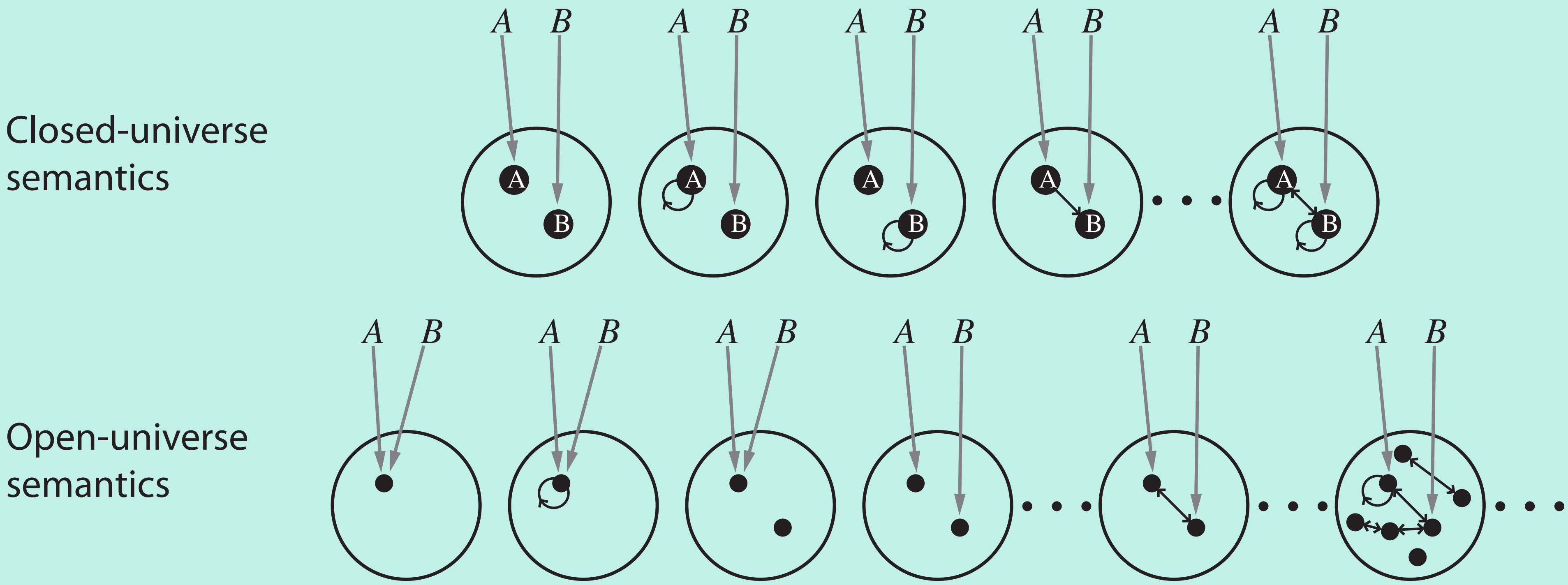
A New Dawn for AI™:
first-order probabilistic languages

Key idea

- Given:
  - An open-universe probability model
  - Evidence from observations
- Apply: Bayesian updating
- Output: beliefs about what objects exist, their identities, and their interrelations

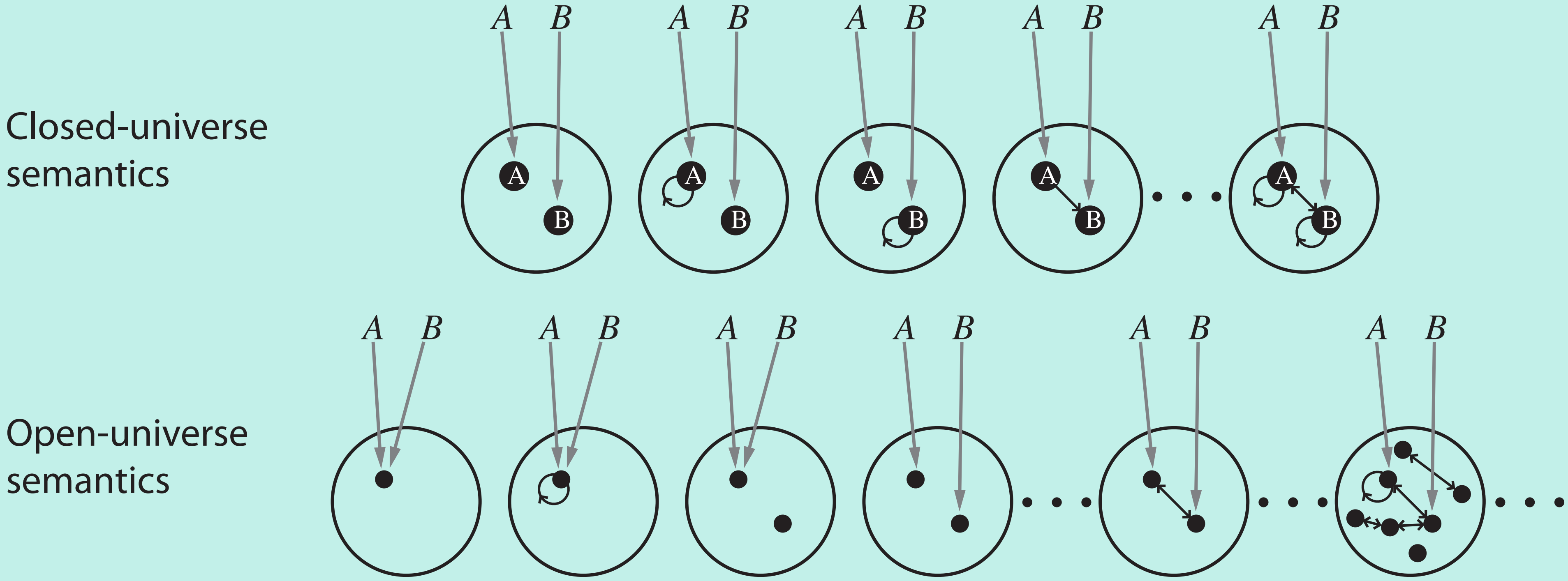
Open-universe semantics

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



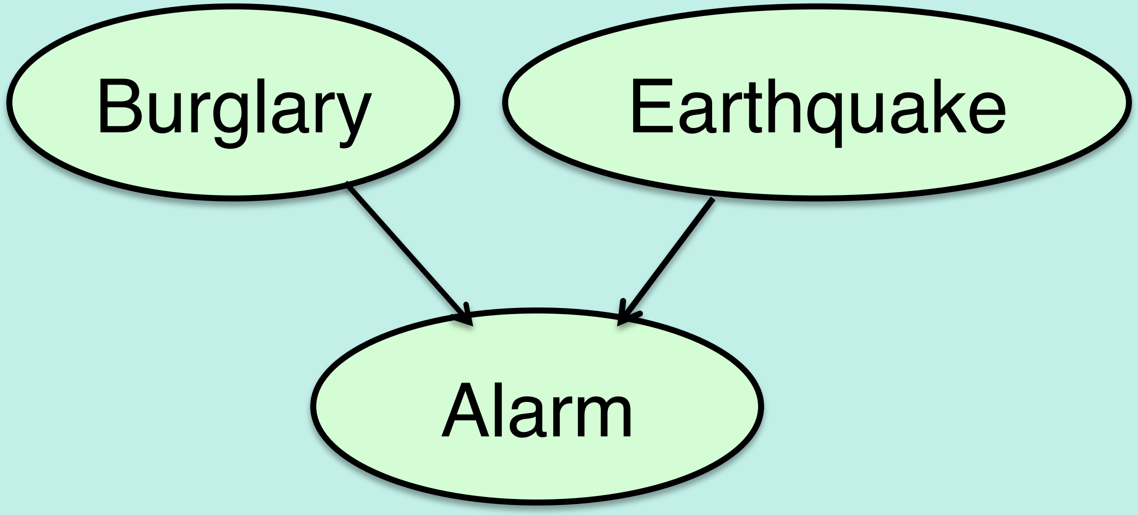
Open-universe semantics

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

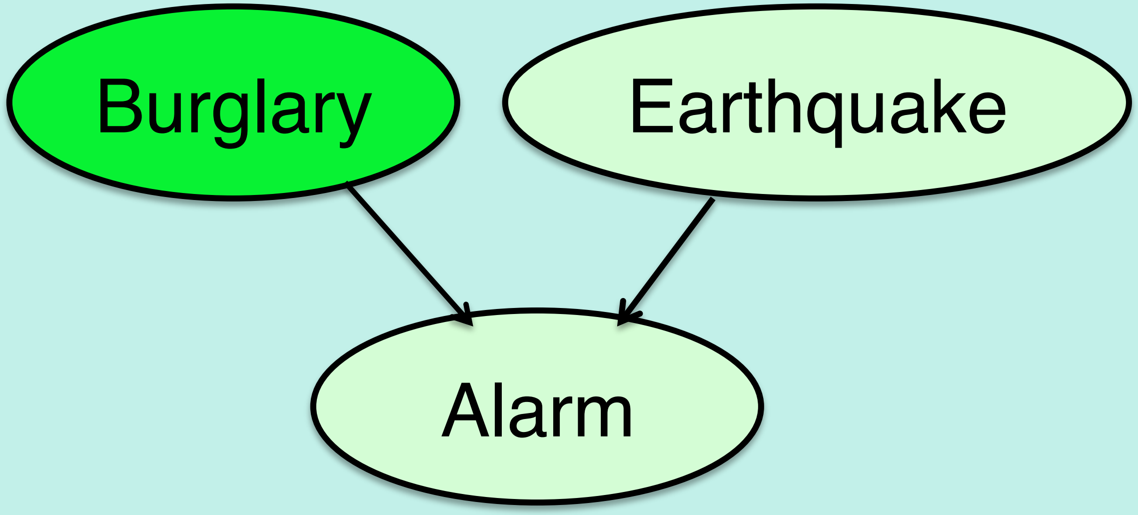


but how can we define P on Ω ??

Bayes nets build propositional worlds

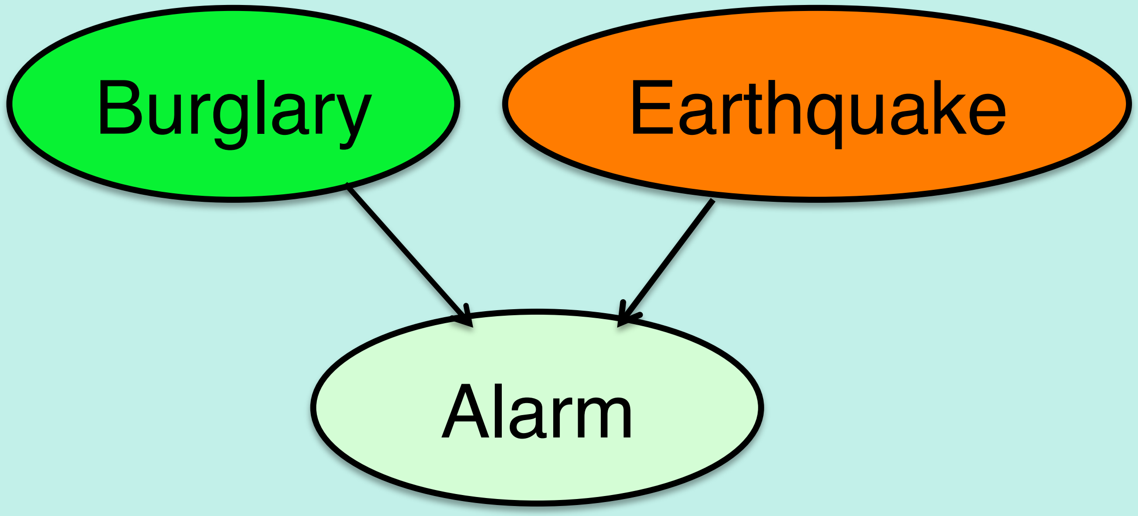


Bayes nets build propositional worlds



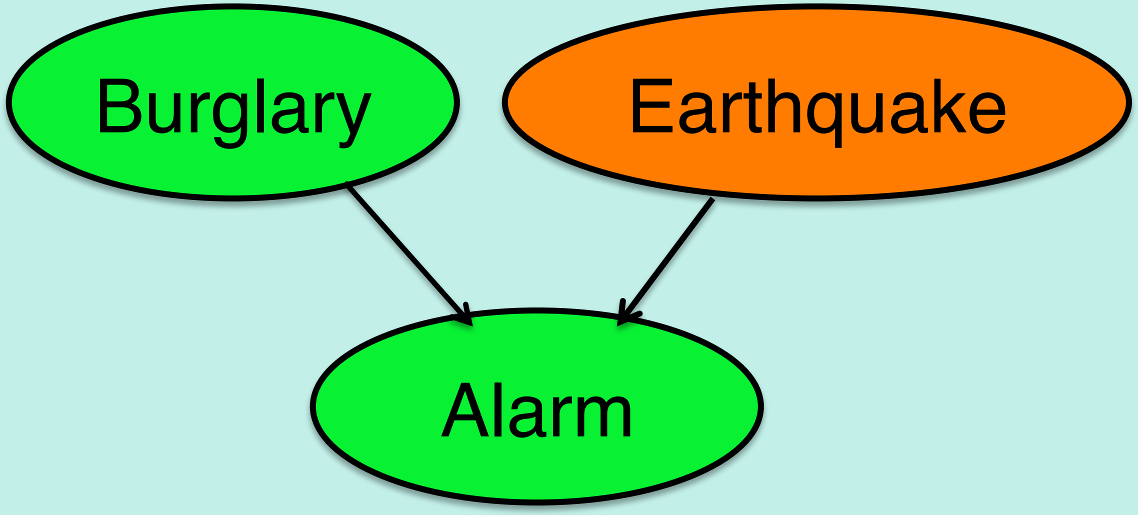
Burglary

Bayes nets build propositional worlds



Burglary
not Earthquake

Bayes nets build propositional worlds



Burglary
not Earthquake
Alarm

Open-universe models in BLOG

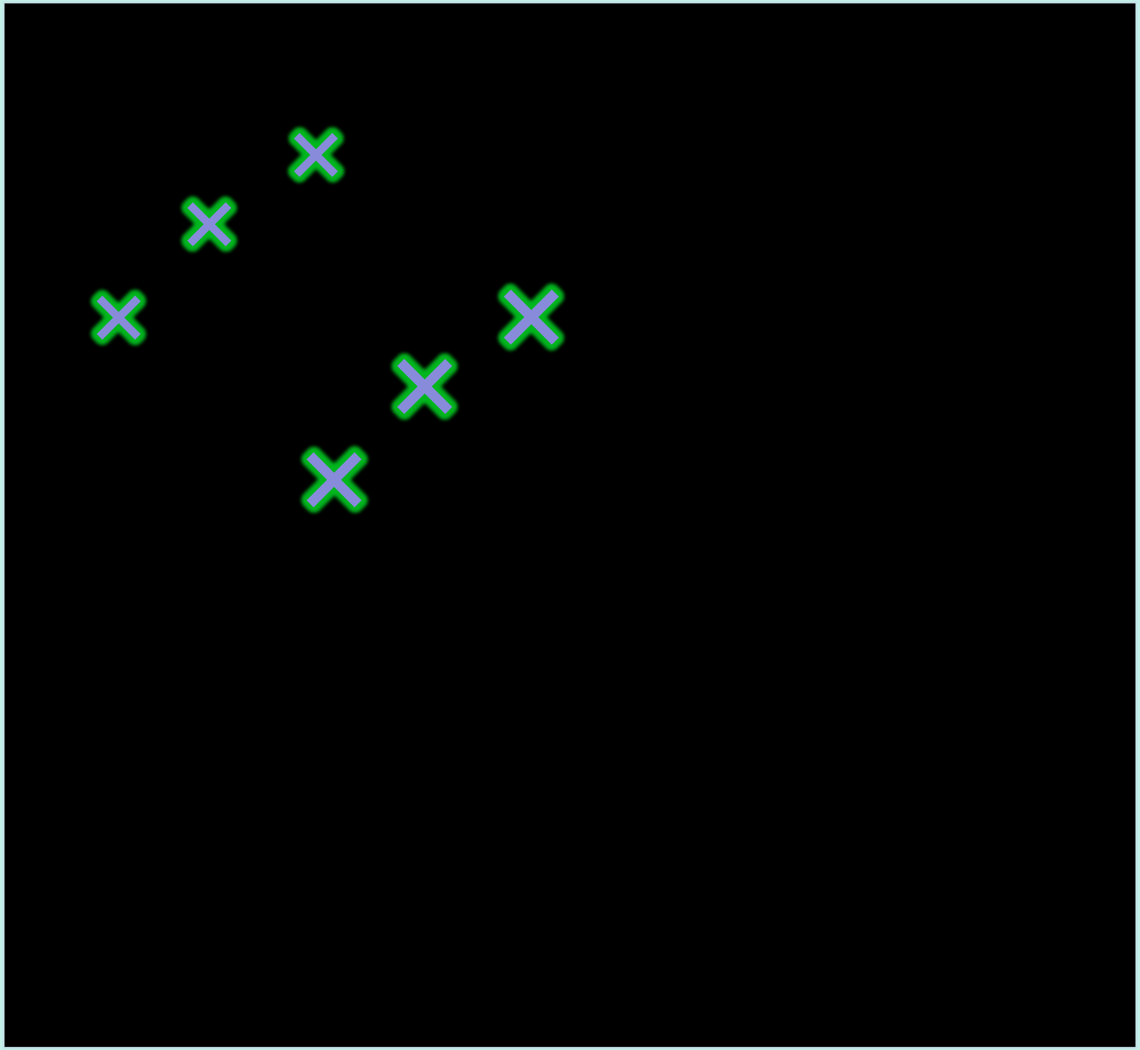
- Construct worlds using two kinds of steps, proceeding in topological order:
  - ***Dependency statements***: Set the value of a *function or relation on a tuple of (quantified) arguments*, conditioned on parent values
    - Alarm(h) ~ CPT[..](Burglary(h), Earthquake(Region(h)))

Open-universe models in BLOG

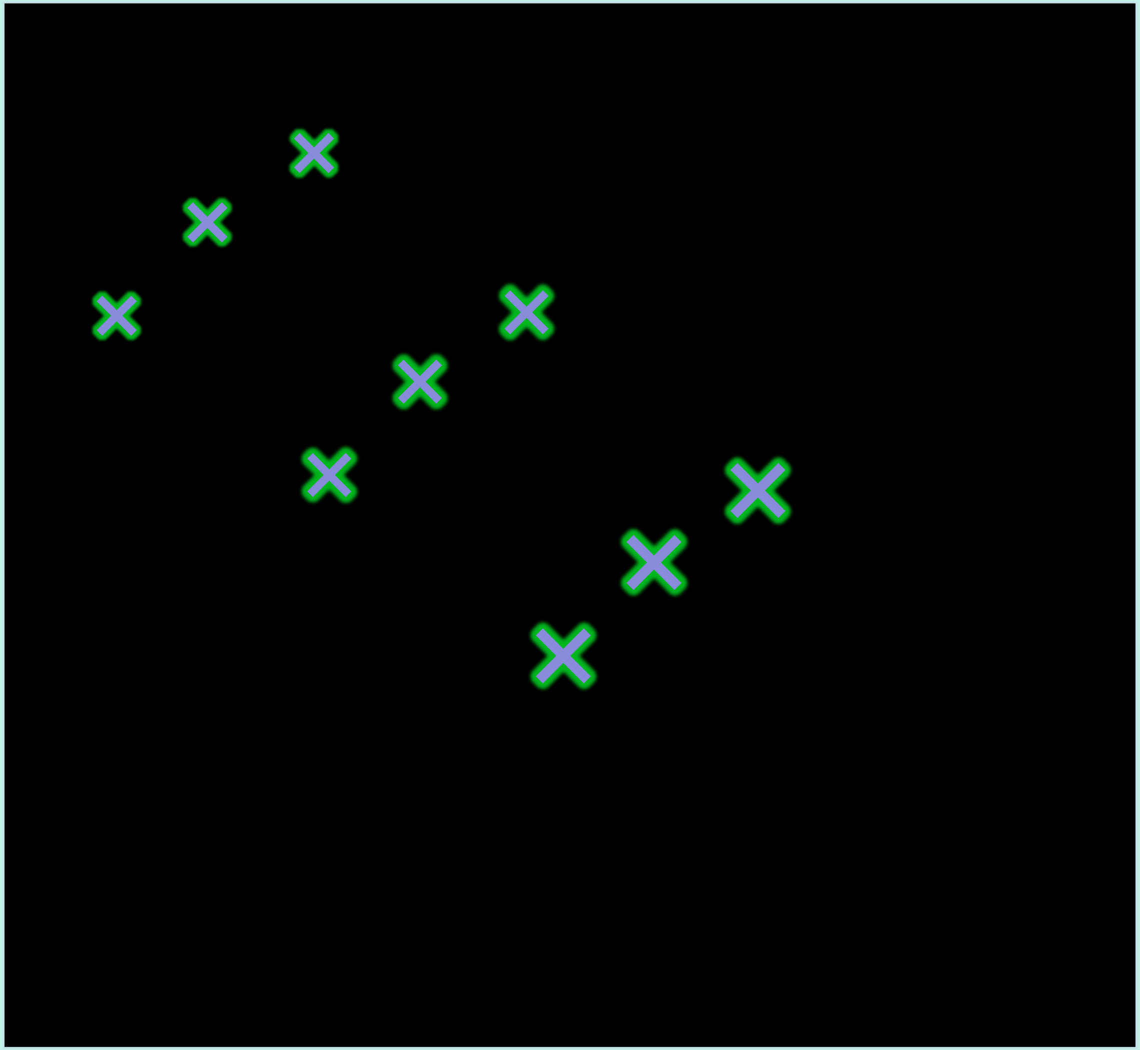
- Construct worlds using two kinds of steps, proceeding in topological order:
  - ***Dependency statements***: Set the value of a *function or relation on a tuple of (quantified) arguments*, conditioned on parent values
    - Alarm(h) ~ CPT[..](Burglary(h), Earthquake(Region(h)))
  - ***Number statements***: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

Example: Multi-target tracking on radar

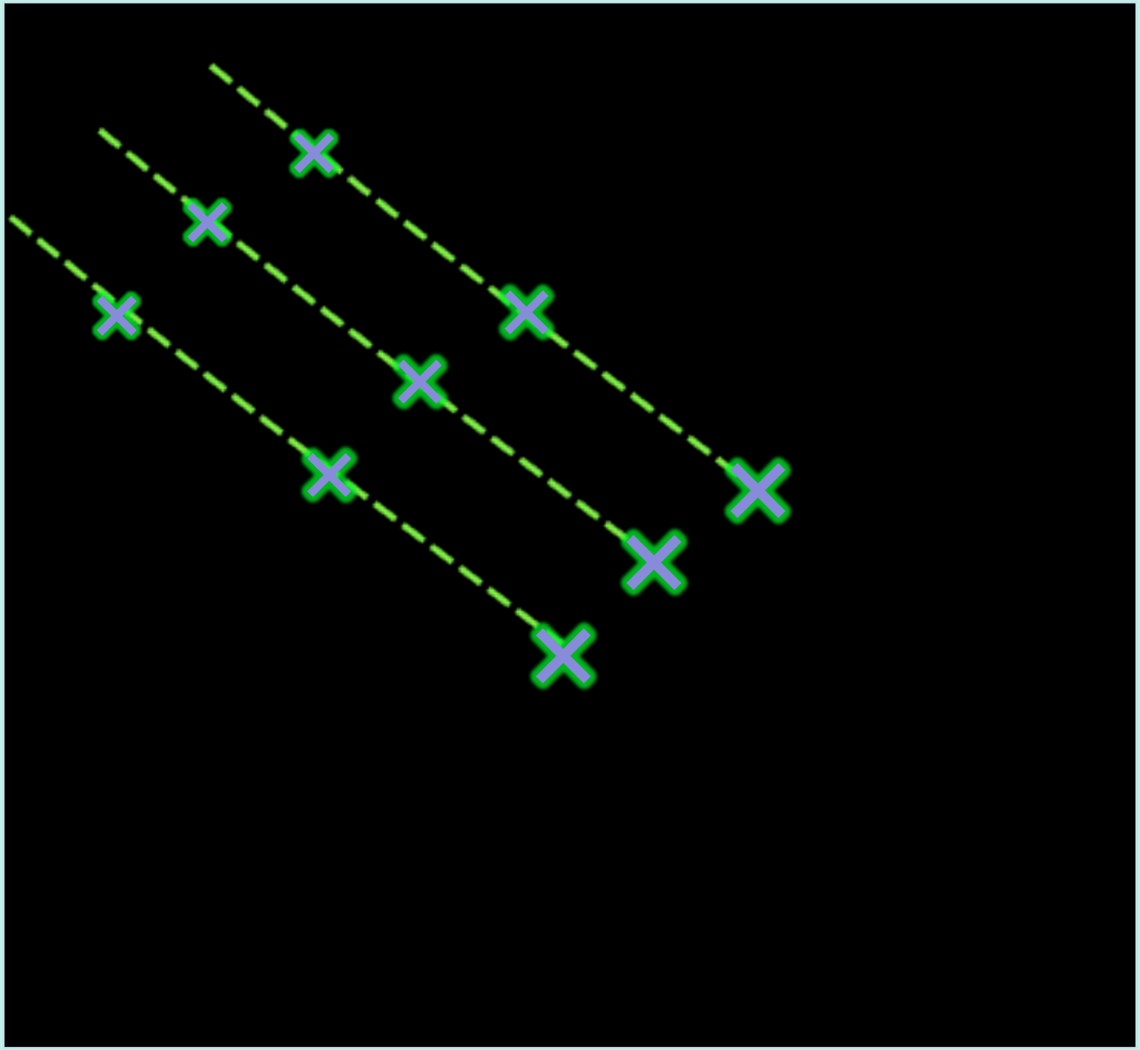
Example: Multi-target tracking on radar



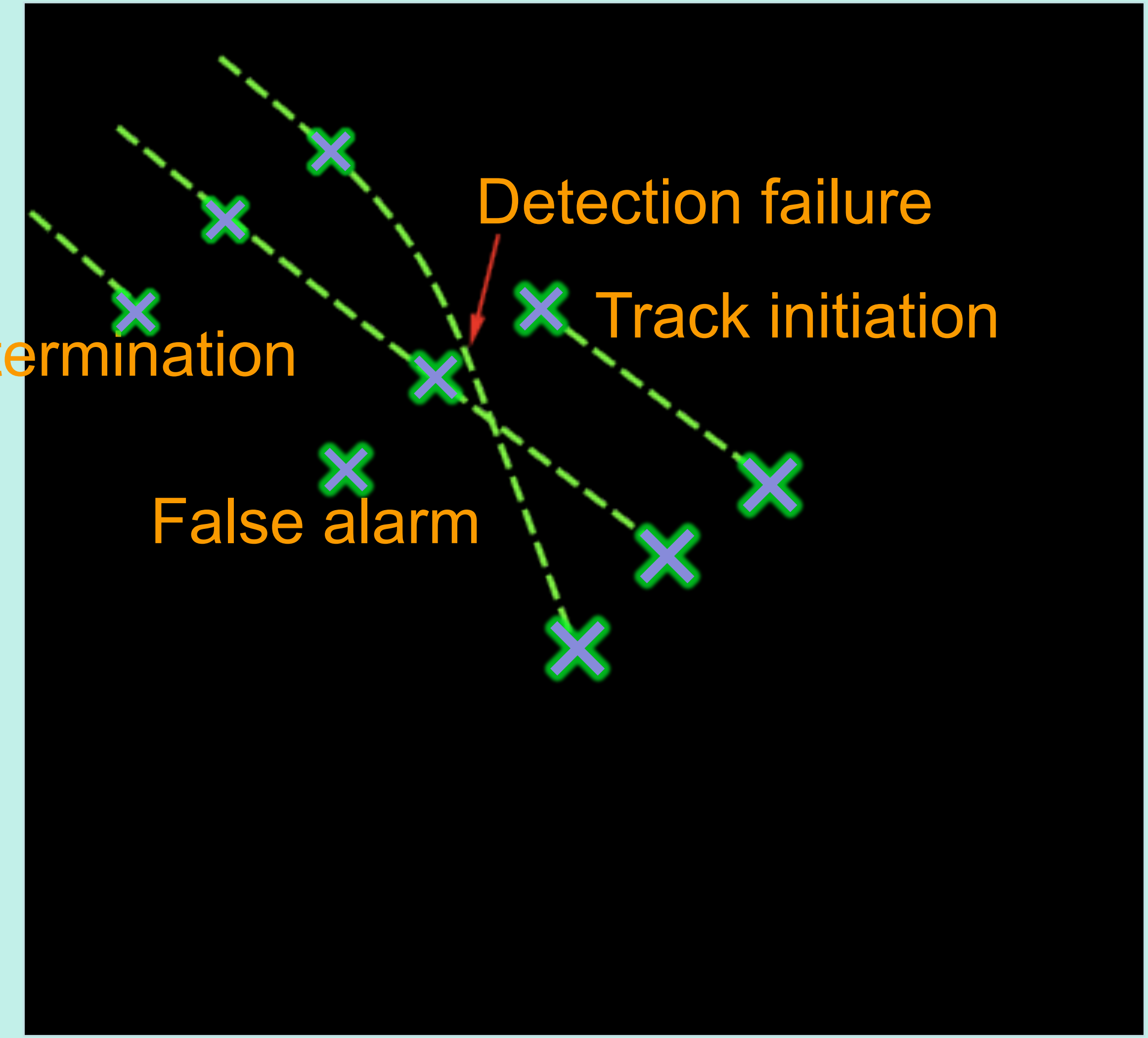
Example: Multi-target tracking on radar



Example: Multi-target tracking on radar



Example: Multi-target tracking on radar



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

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

InFlight(a,t)
  if t < EntryTime(a) then = false
  elseif t = EntryTime(a) then = true
  else = (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]();
```

```
#Aircraft(EntryTime = t) ~ Poisson[λ_a]();

Exits(a,t)
  if InFlight(a,t) then ~ Boolean[α_e]();

InFlight(a,t)
  if t < EntryTime(a) then = false
  elseif t = EntryTime(a) then = true
  else = (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),Σ_x]();

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

#Blip(Time=t) ~ Poisson[λ_f]();

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

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

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

InFlight(a,t)
  if t < EntryTime(a) then = false
  elseif t = EntryTime(a) then = true
  else = (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]();
```

```
#Aircraft(EntryTime = t) ~ Poisson[λ_a]();

Exits(a,t)
   if InFlight(a,t) then ~ Boolean[α_e]();

InFlight(a,t)
   if t < EntryTime(a) then = false
   elseif t = EntryTime(a) then = true
   else = (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),Σ_x]();

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

                         Poisson[λ_f]();

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

Origin function

Semantics

- *Objects* are defined by type, origin, number:
  - <Aircraft,<EntryTime,<TimeStep,5>>,2>
  - <Blip,<Source, <Aircraft,<EntryTime,<TimeStep,5>>,2>, <Time,<TimeStep,7>>,1>
- Each *basic random variable* is a function or predicate symbol indexed by a tuple of objects:
  - InFlight$_{\langle \text{Aircraft},\langle \text{EntryTime},\langle \text{TimeStep},5\rangle\rangle,2\rangle,\langle \text{TimeStep},7\rangle}(\omega)$
- 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

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

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

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 exist
    - Their titles and authors
    - For each paper, the papers it cites

(Simplified) BLOG model

```
#Researcher ~ NumResearchersPrior();

Name(r) ~ NamePrior();

#Paper(FirstAuthor = r) ~
        NumPapersPrior(Position(r));

Title(p) ~ TitlePrior();

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

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

(Simplified) BLOG model

```
#Researcher ~ NumResearchersPrior();

Name(r) ~ NamePrior();

#Paper(FirstAuthor = r) ~
        NumPapersPrior(Position(r));

Title(p) ~ TitlePrior();

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

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

(Simplified) BLOG model

```
#Researcher ~ NumResearchersPrior();

Name(r) ~ NamePrior();

#Paper(FirstAuthor = r) ~
        NumPapersPrior(Position(r));

Title(p) ~ TitlePrior();

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

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

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

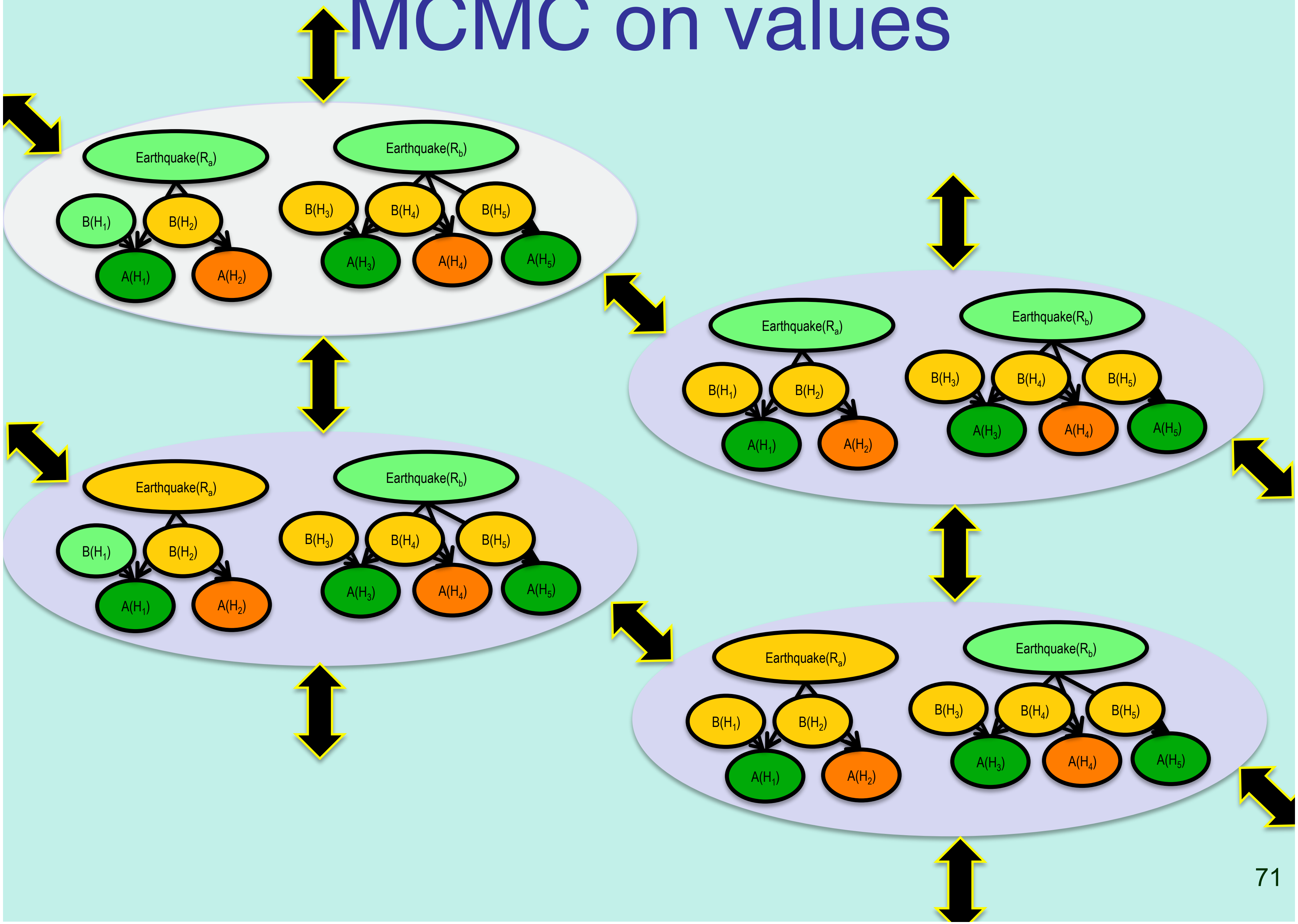
1. PCFG for simple English
2. Simplified 3D vision
3. Hurricane prediction
4. Burglary
5. Balls and urns (counting)
6. Sybil attack (cybersecurity)
7. Students and grades
8. Topic models (LDA)
9. Citation information extraction
10. Competing workshops
11. Galaxy model
12. Infinite mixture of Gaussians
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

Inference

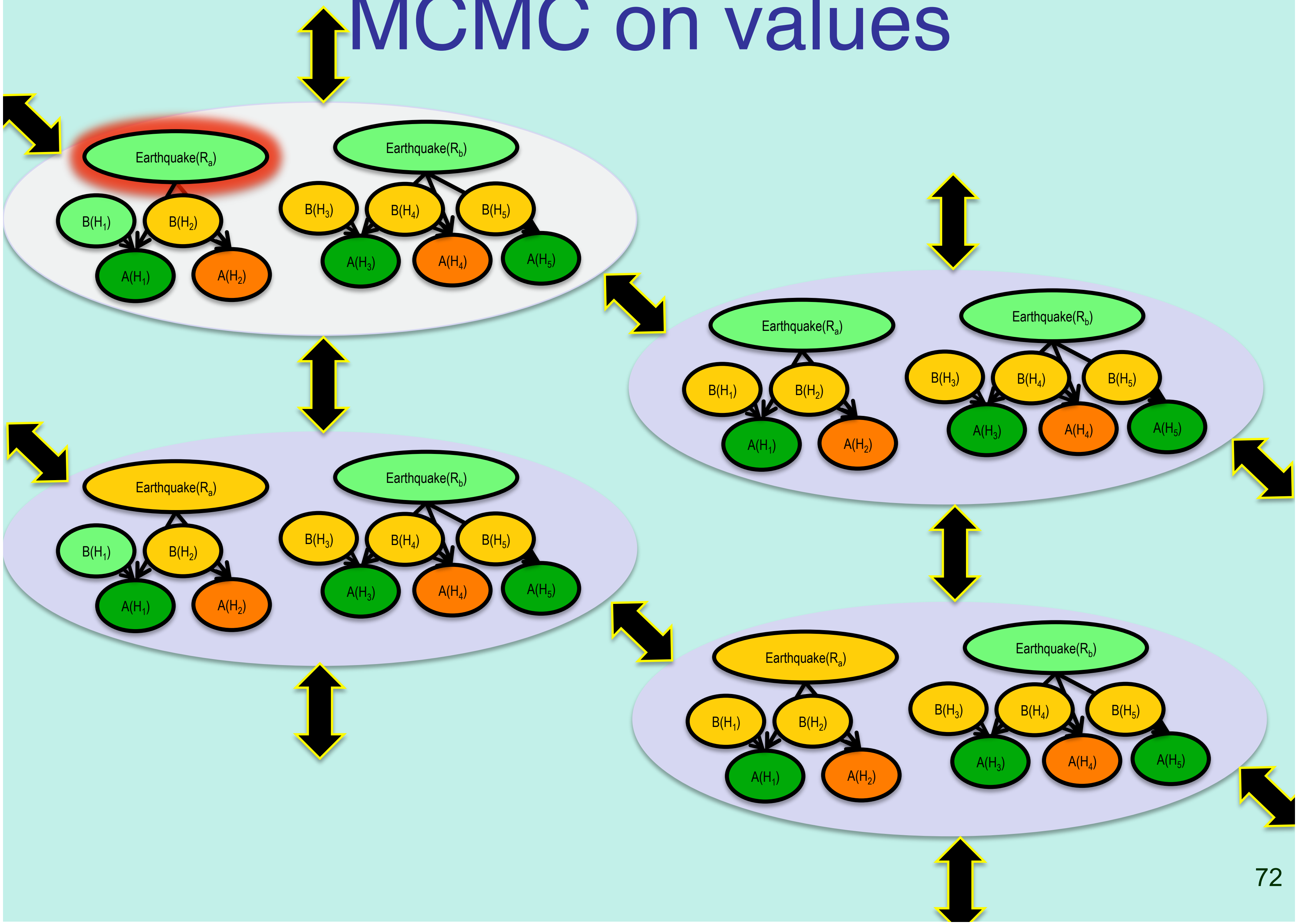
Theorem: BLOG inference algorithms (rejection sampling, importance sampling, MCMC) converge* to correct posteriors for any well-formed model, for any finitely evaluable first-order query

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

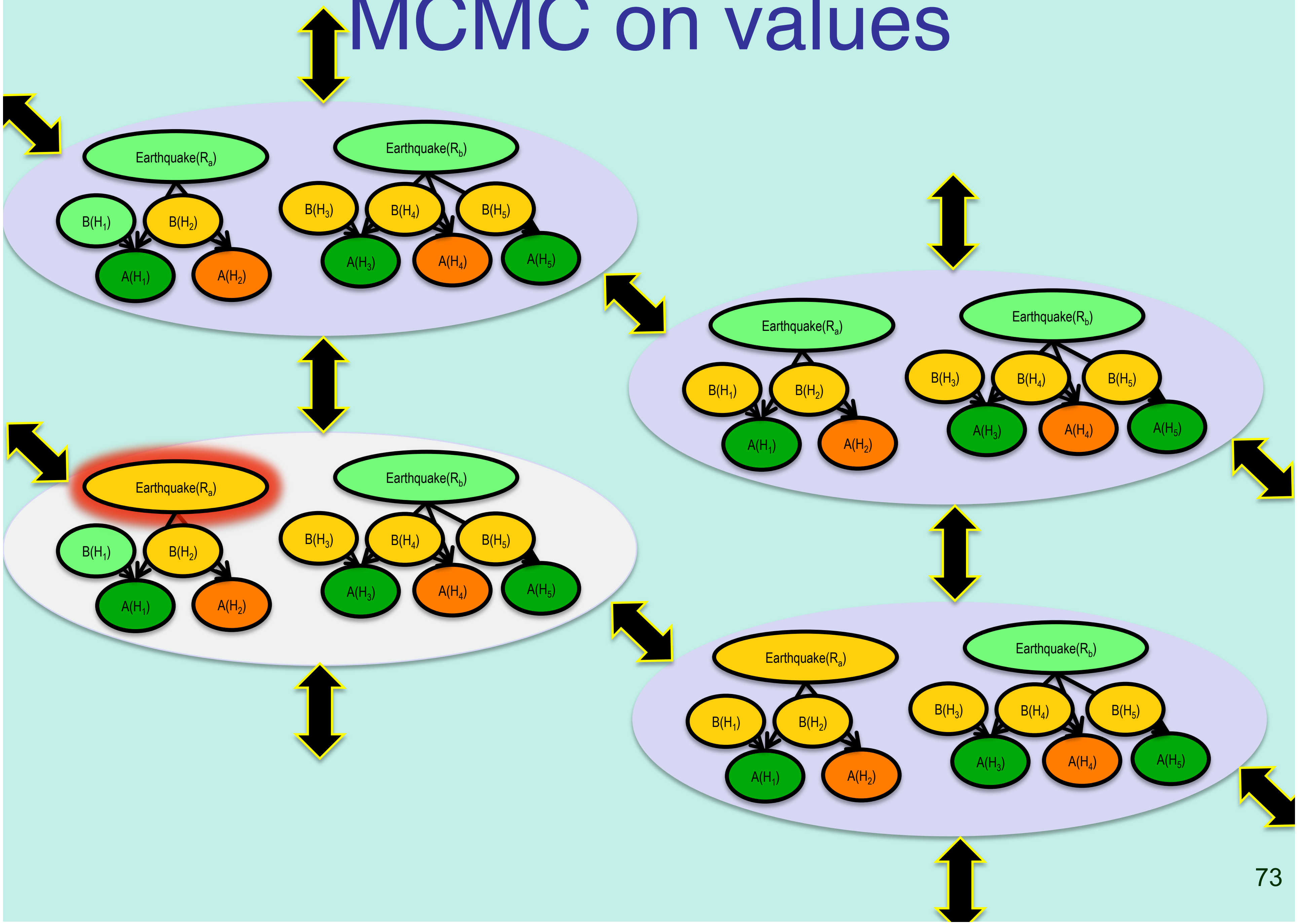
MCMC on values



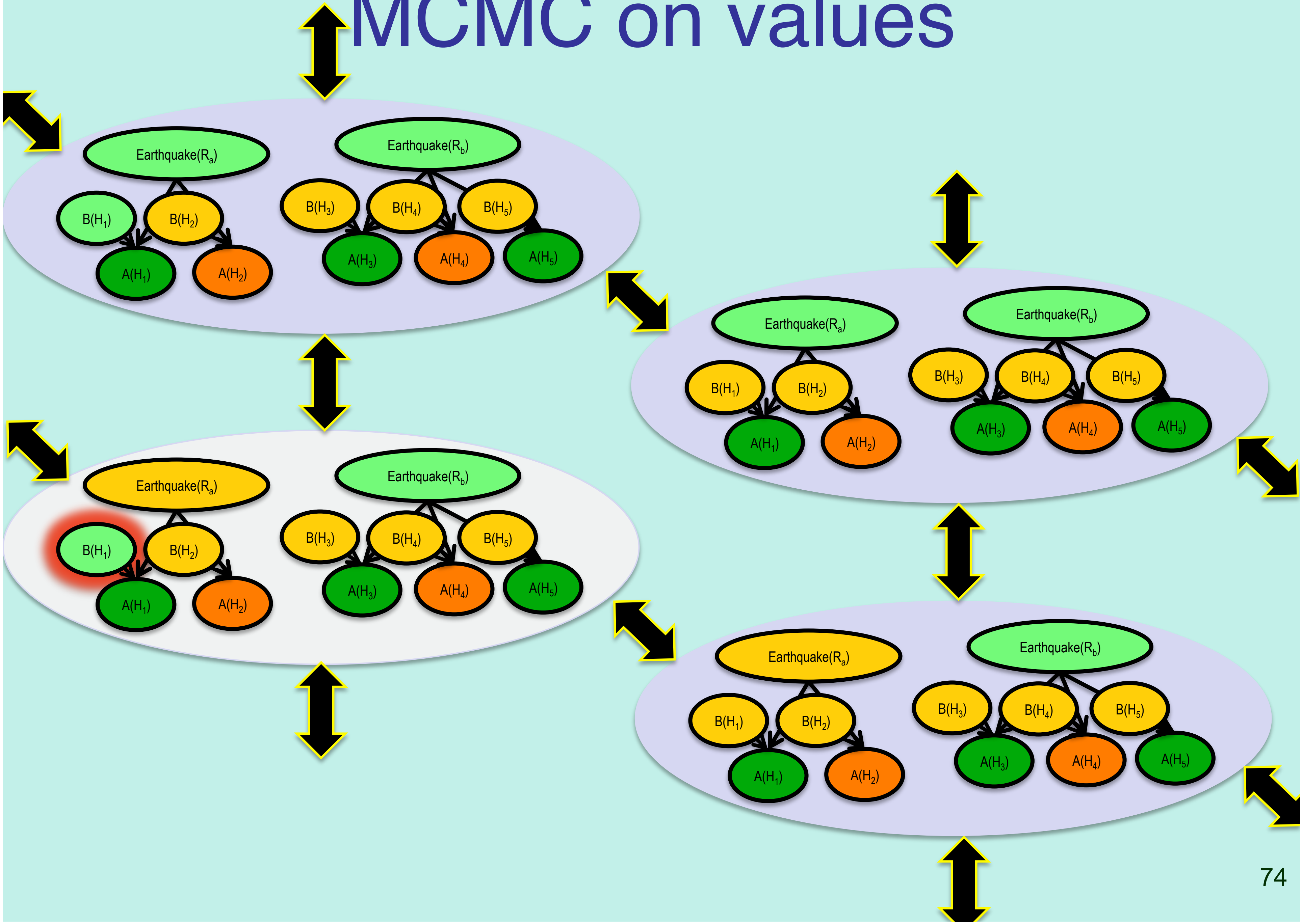
MCMC on values



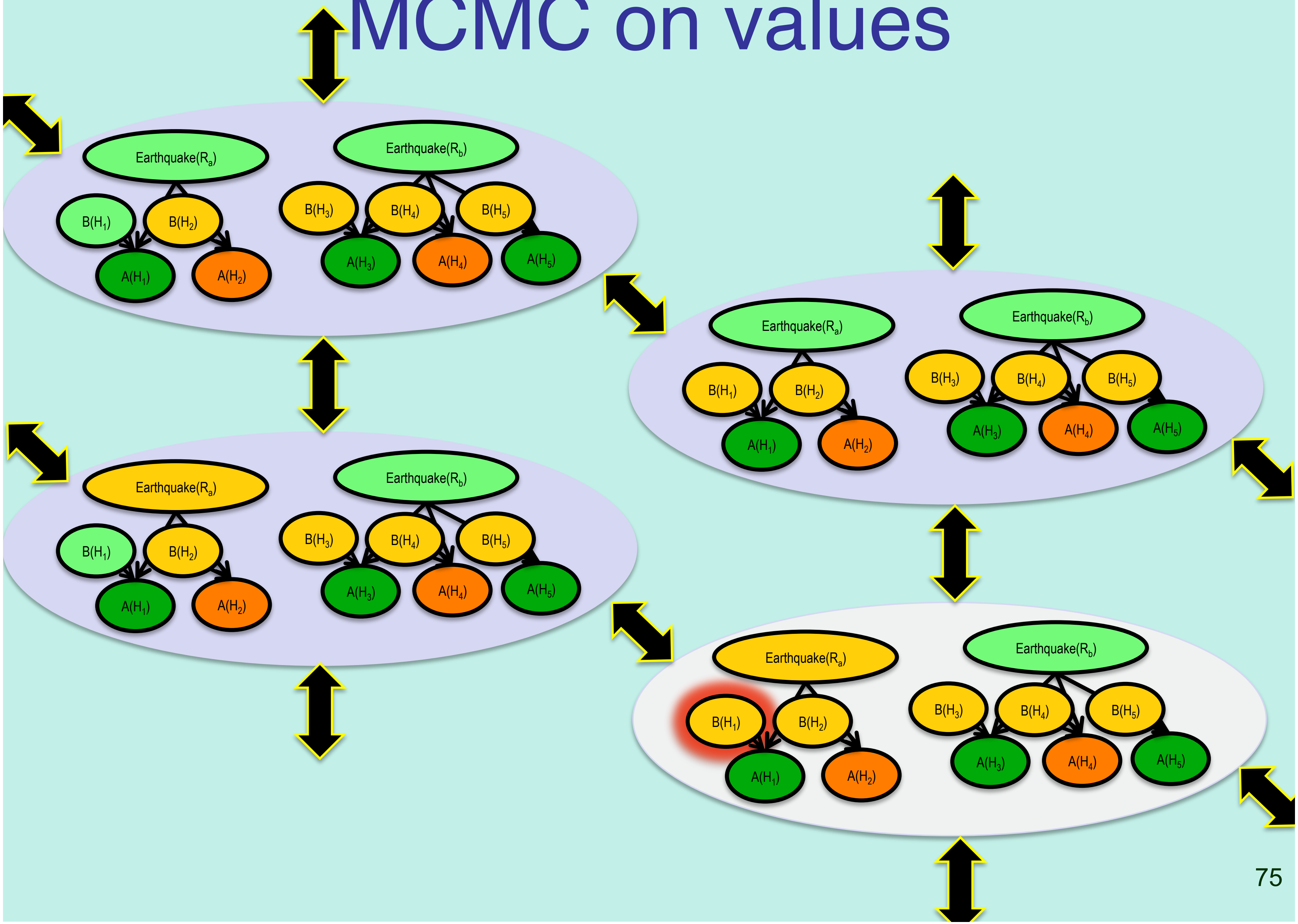
MCMC on values

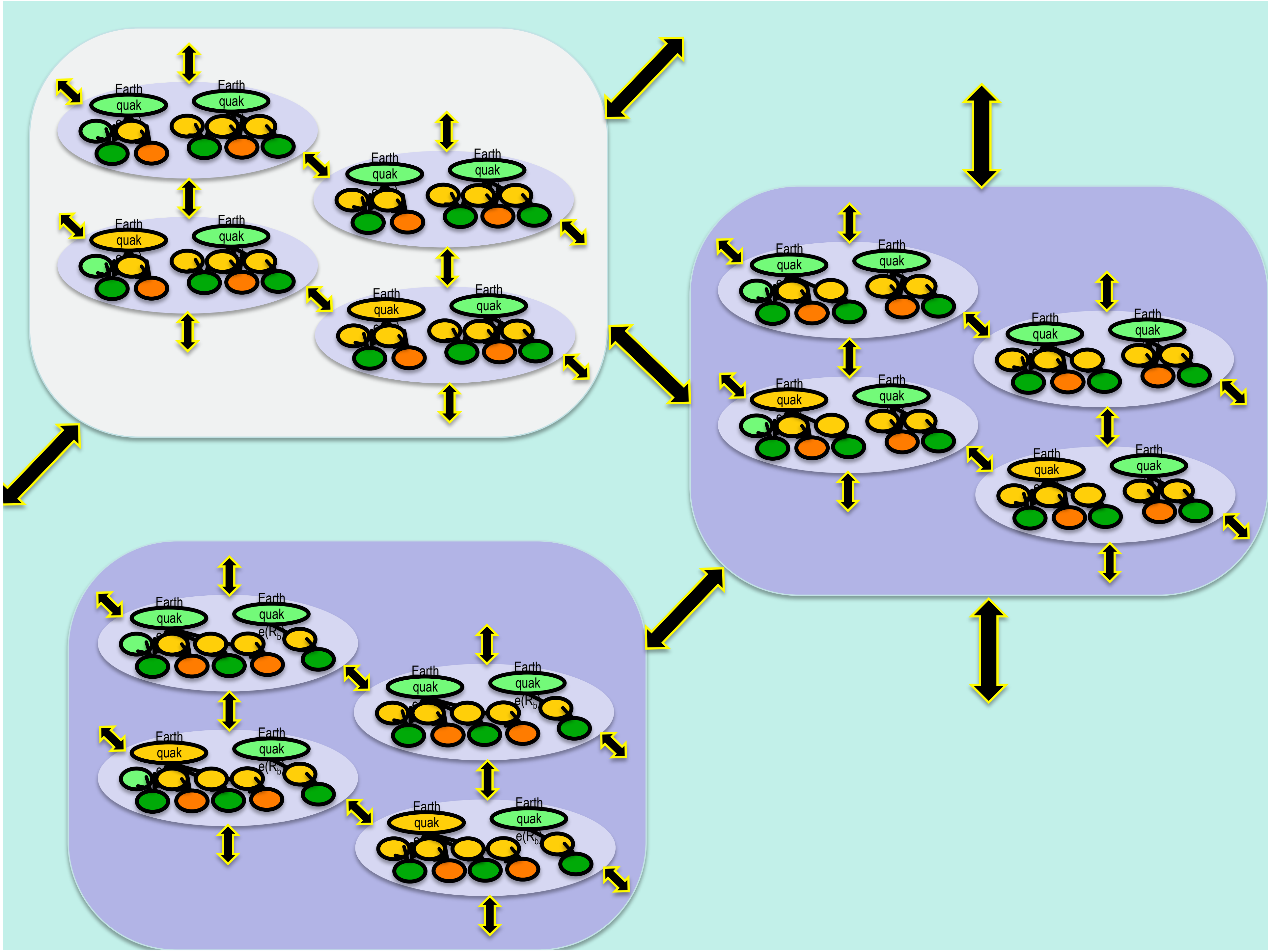


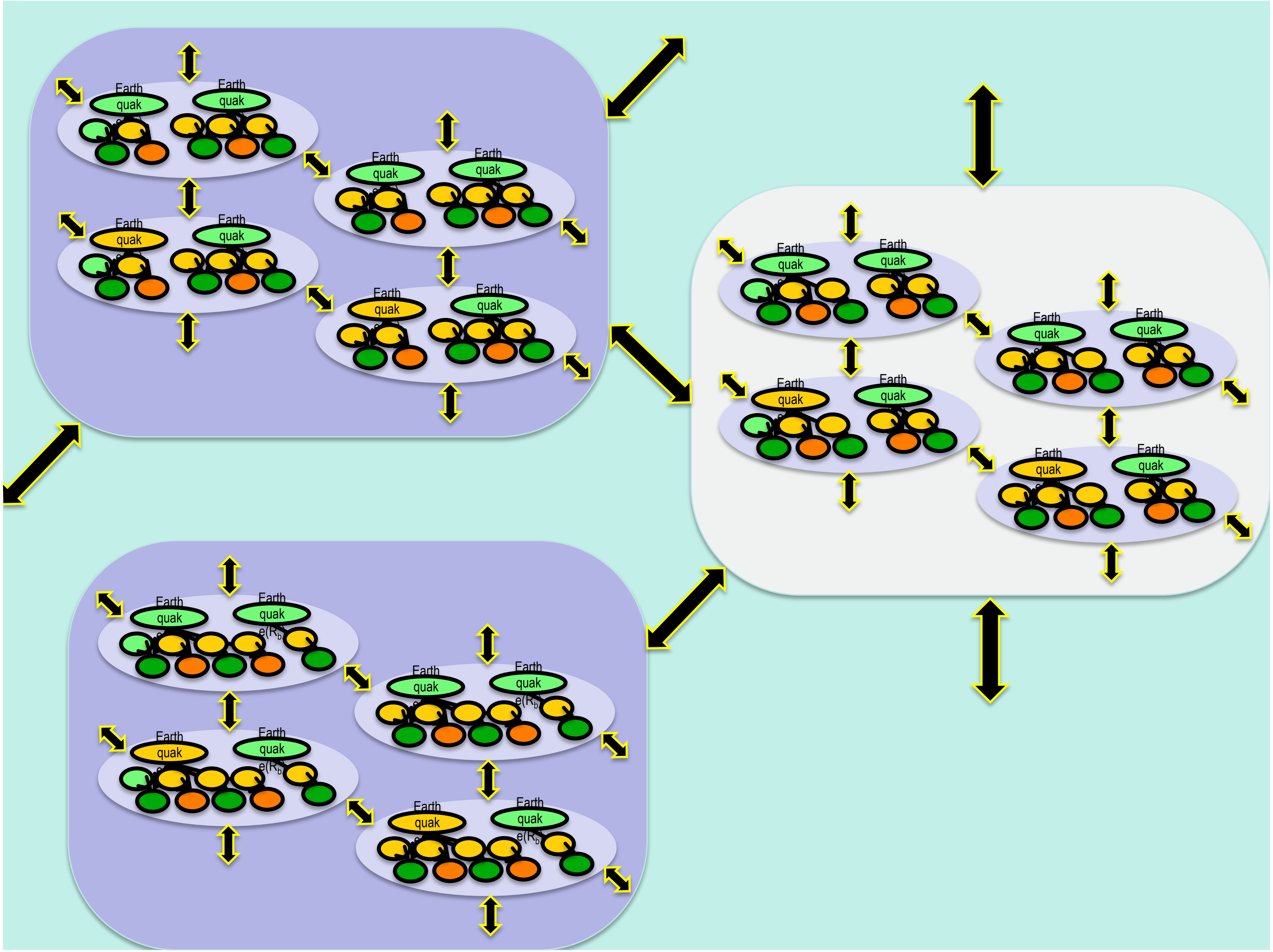
MCMC on values

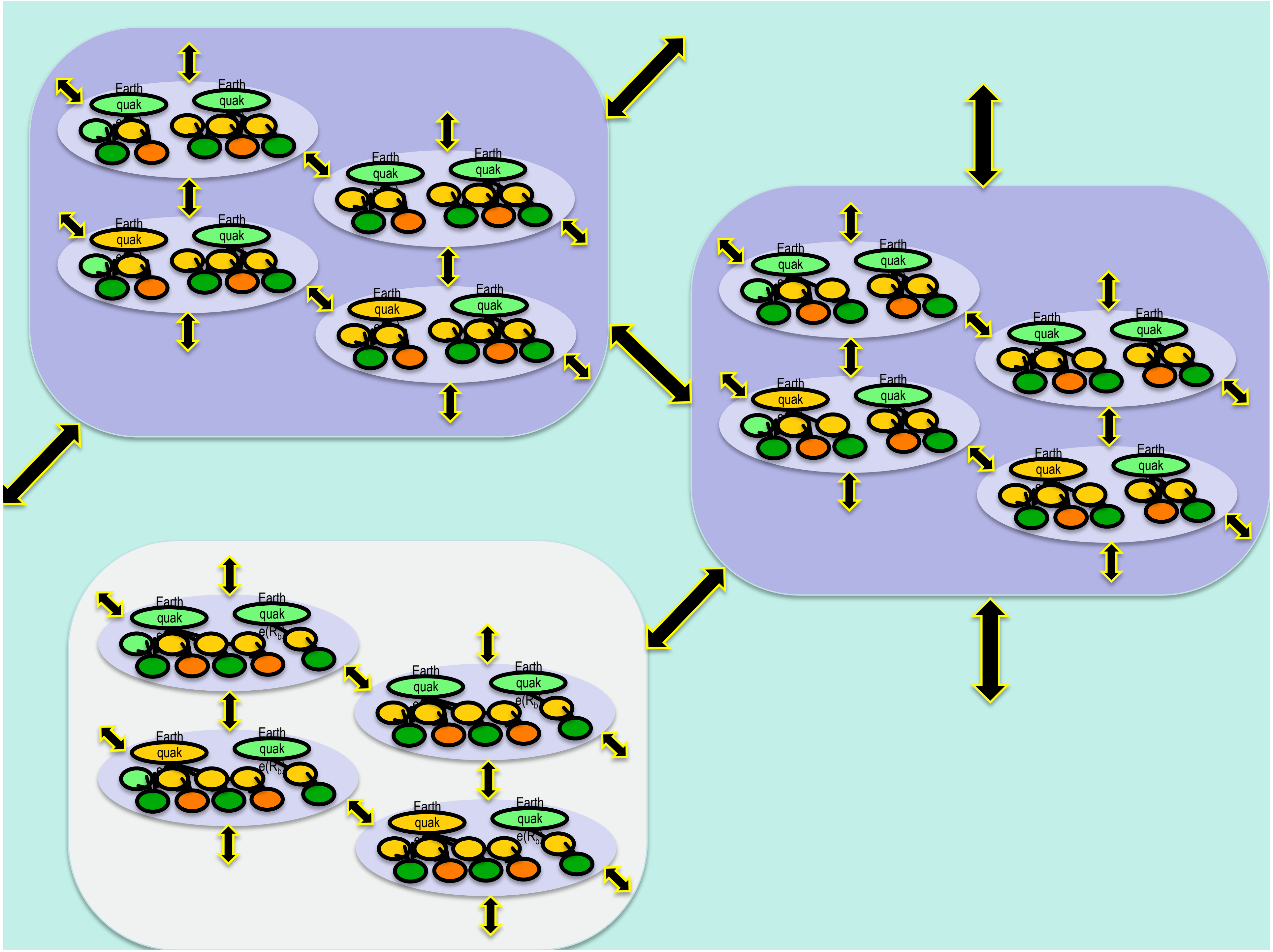


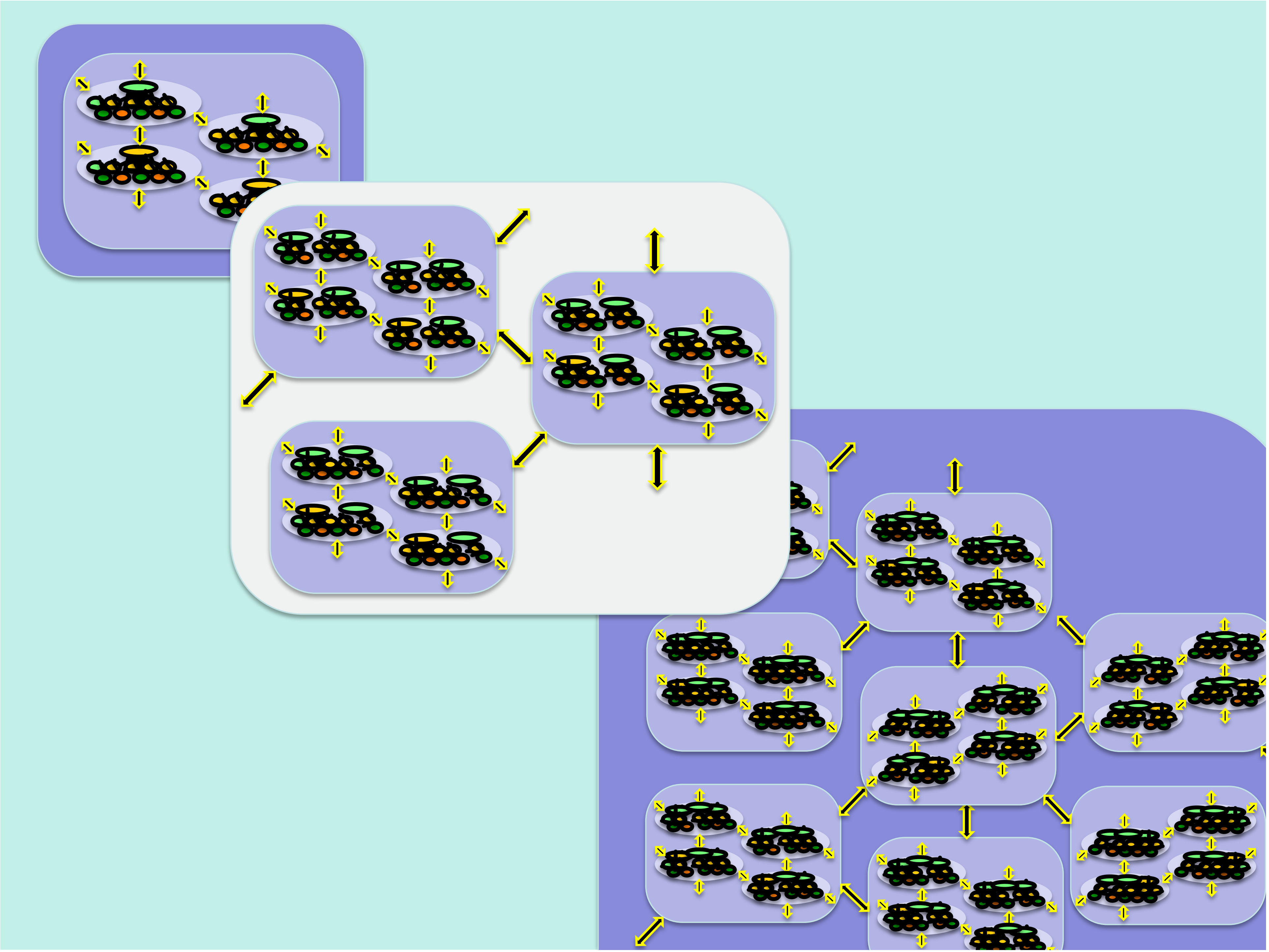
MCMC on values

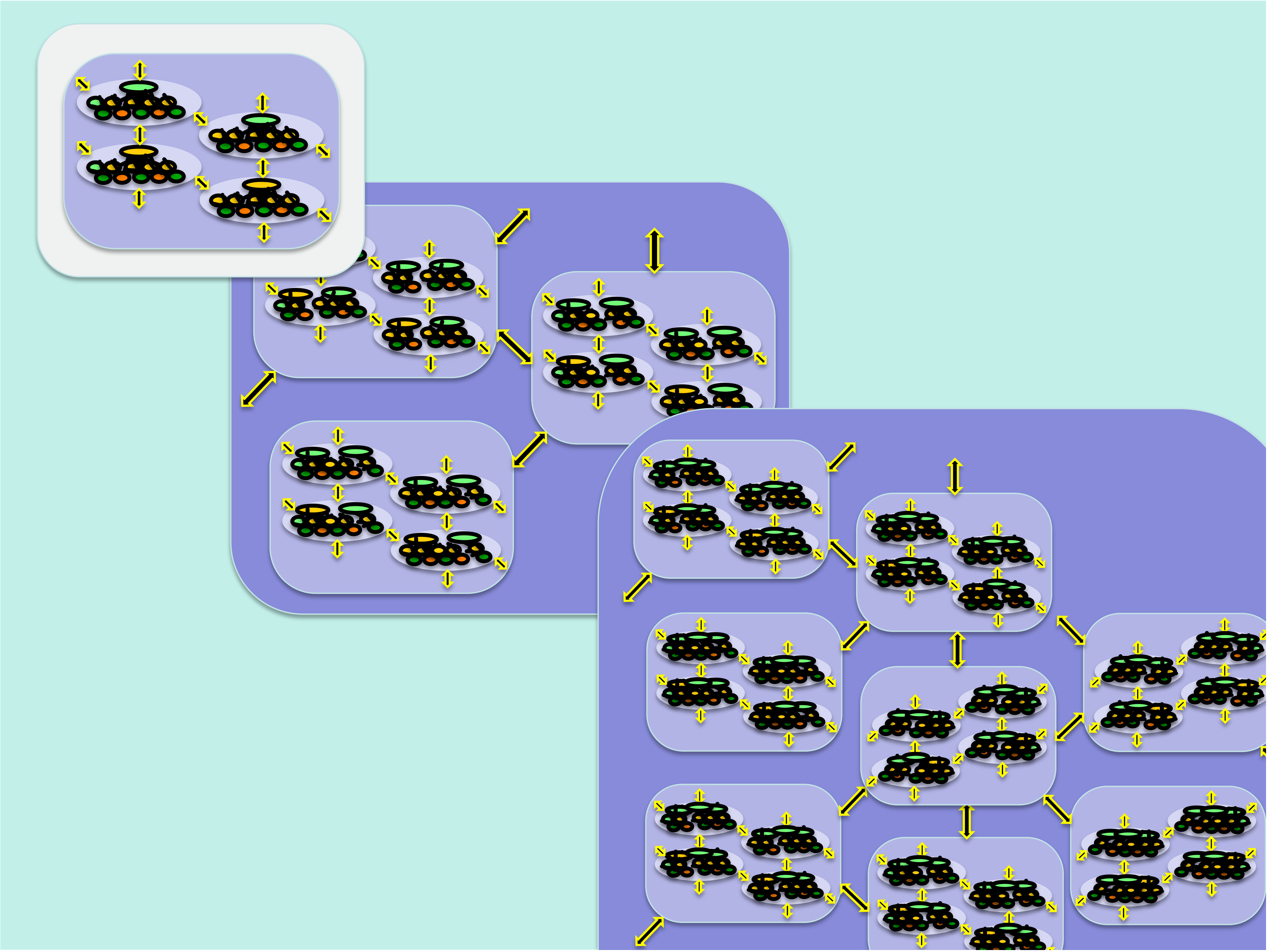


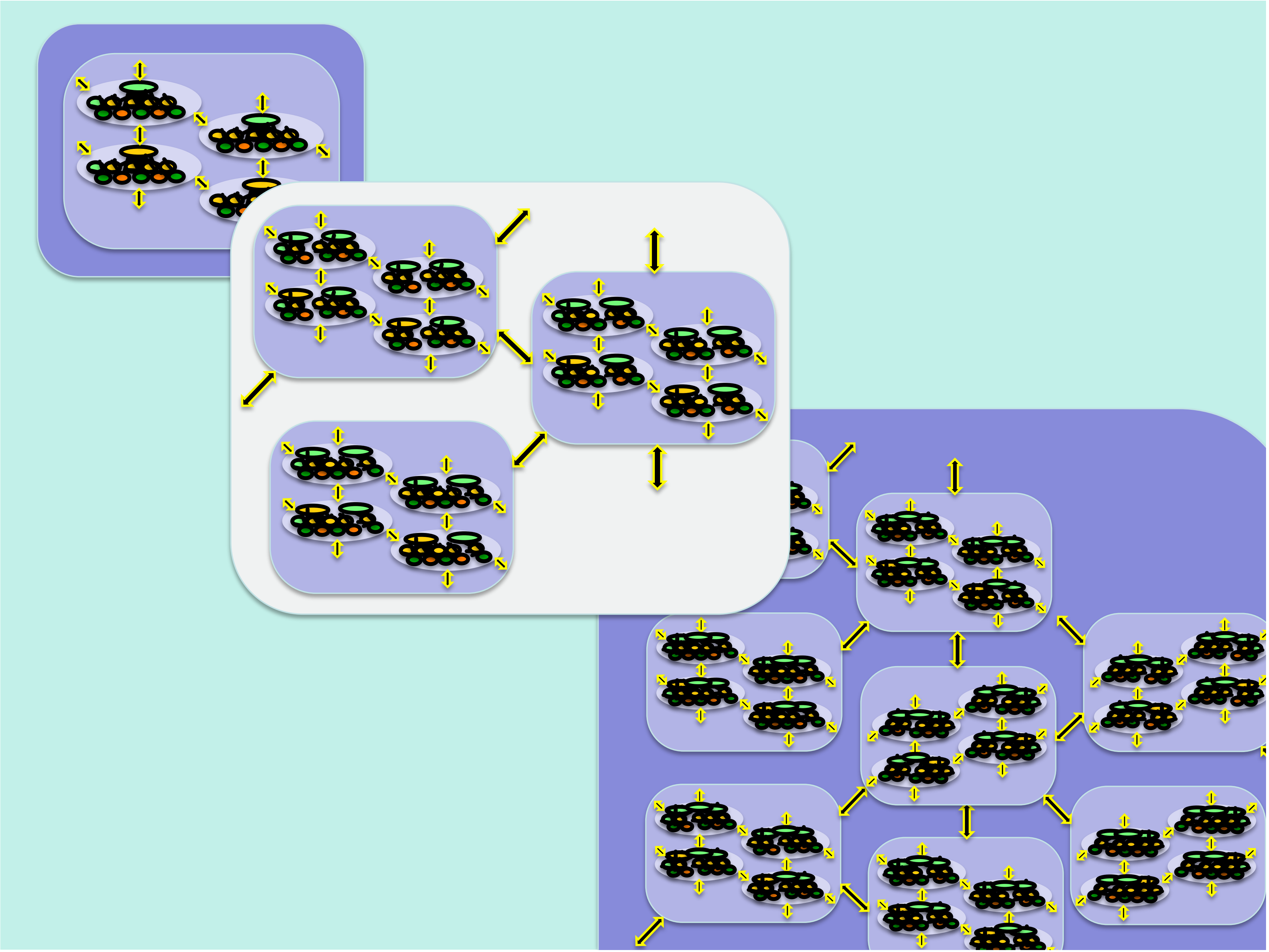

Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
e(R_b)
Earth
quak
Earth
quak
e(R_b)
Earth
quak
Earth
quak
e(R_b)
Earth
quak
Earth
quak
e(R_b)

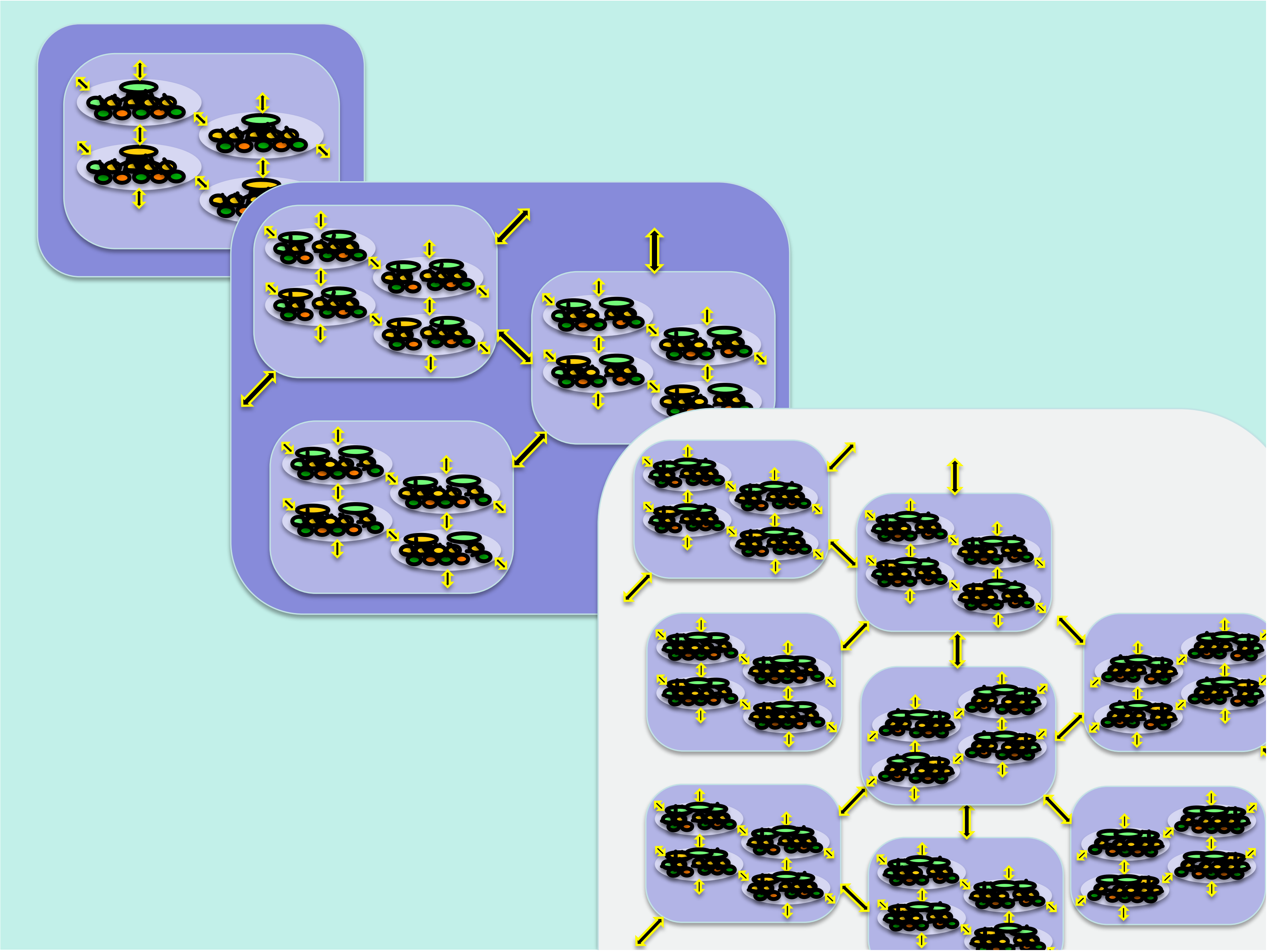

Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak
Earth
quak


Earth
quak
e(R$_b$









Inference

- Efficient inference is the bottleneck
    - Real-world applications use special-purpose inference
- A substantial engineering problem
    - Modular design with “plug-in” expert samplers
    - Optimizing compiler technology to reduce overhead
    - Data and process parallelism
    - Incremental query evaluation (cf database systems)

Application: CTBT

- Bans testing of nuclear weapons on earth
  - Allows outside inspection of 1000km^2 (18km radius)
- 183/195 states have signed
- 158/195 have ratified
- Need 8 more ratifications including US, China
- US Senate refused to ratify in 1998
  - "too hard to monitor"

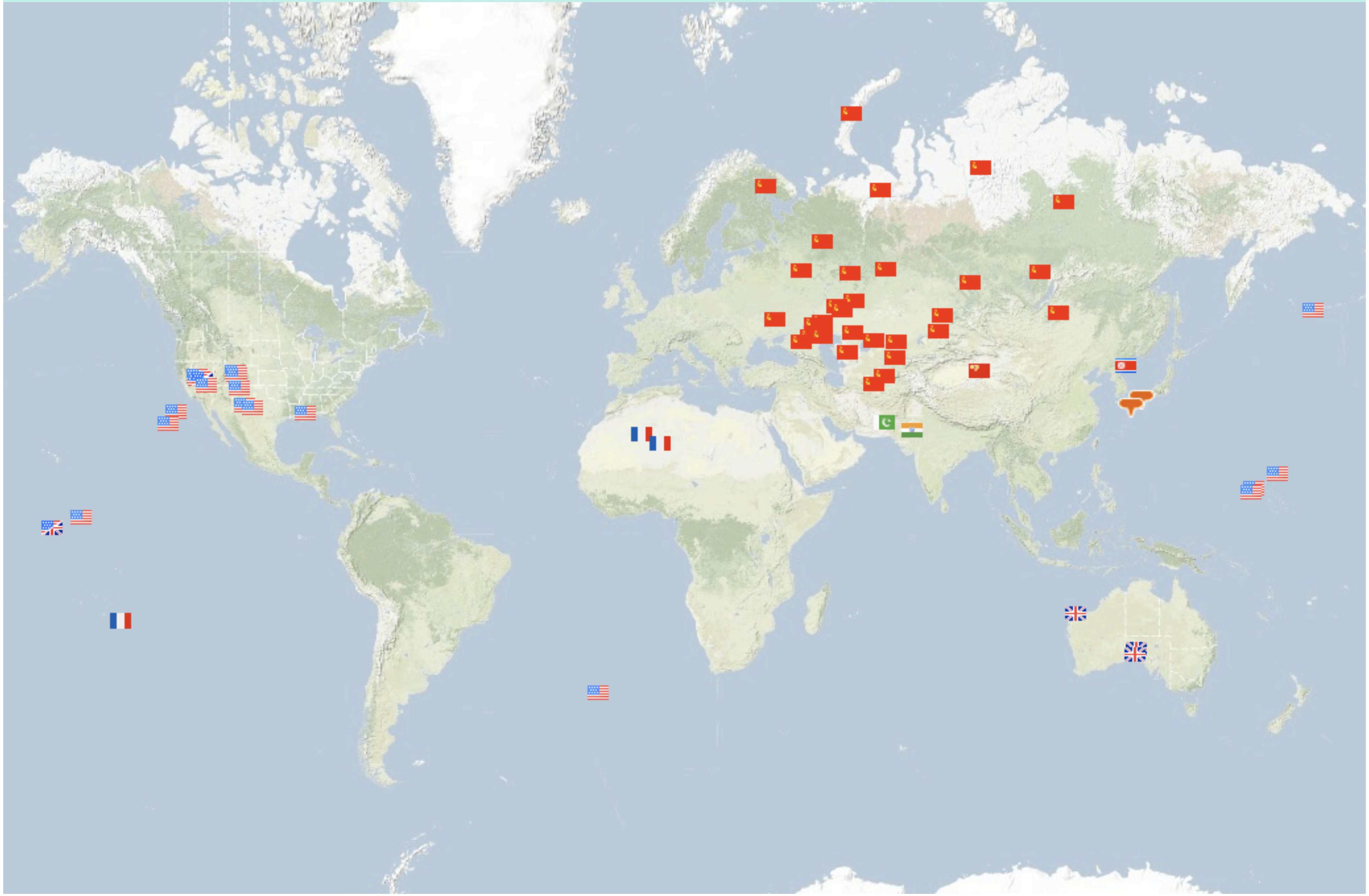
Application: CTBT

- Bans testing of nuclear weapons on earth
  - Allows outside inspection of 1000km^2 (18km radius)
- 183/195 states have signed (including France)
- 158/195 have ratified
- Need 8 more ratifications including US, China
- US Senate refused to ratify in 1998
  - “too hard to monitor”

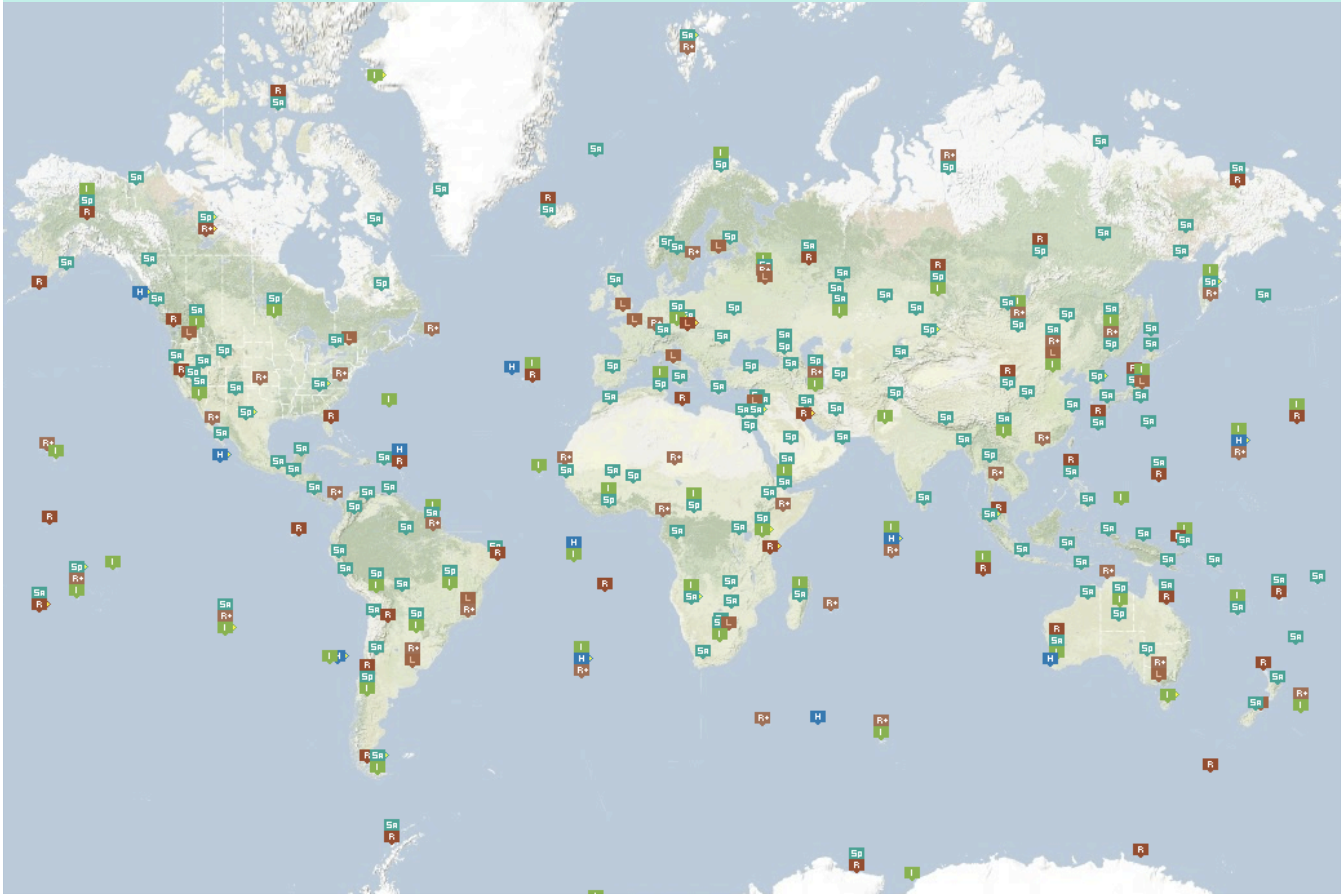
Application: CTBT

- Bans testing of nuclear weapons on earth
  - Allows outside inspection of 1000km^2 (18km radius)
- 183/195 states have signed (including France)
- 158/195 have ratified (including France)
- Need 8 more ratifications including US, China
- US Senate refused to ratify in 1998
  - "too hard to monitor"

2053 nuclear explosions, 300K deaths



254 monitoring stations

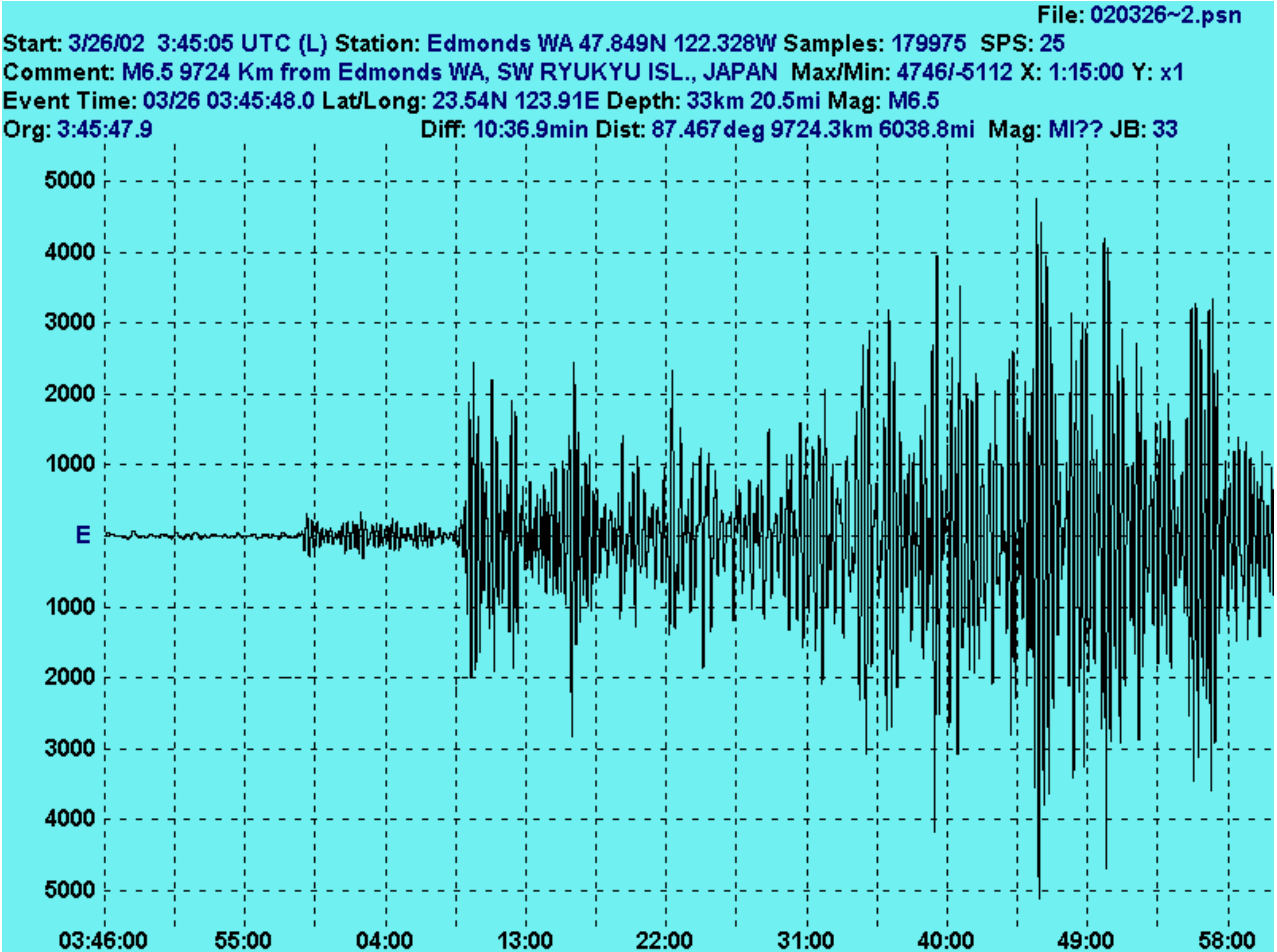


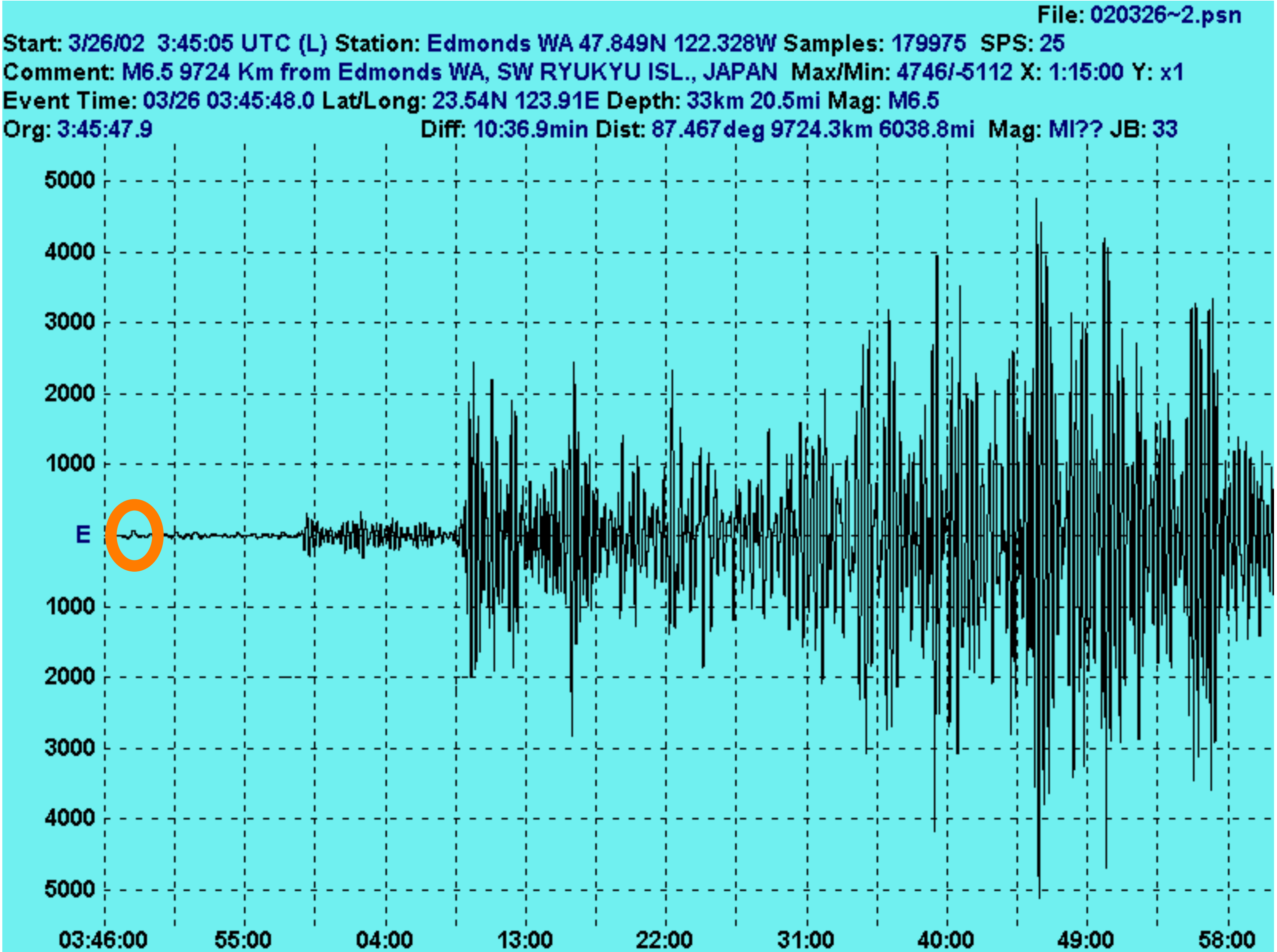
Global seismic monitoring

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

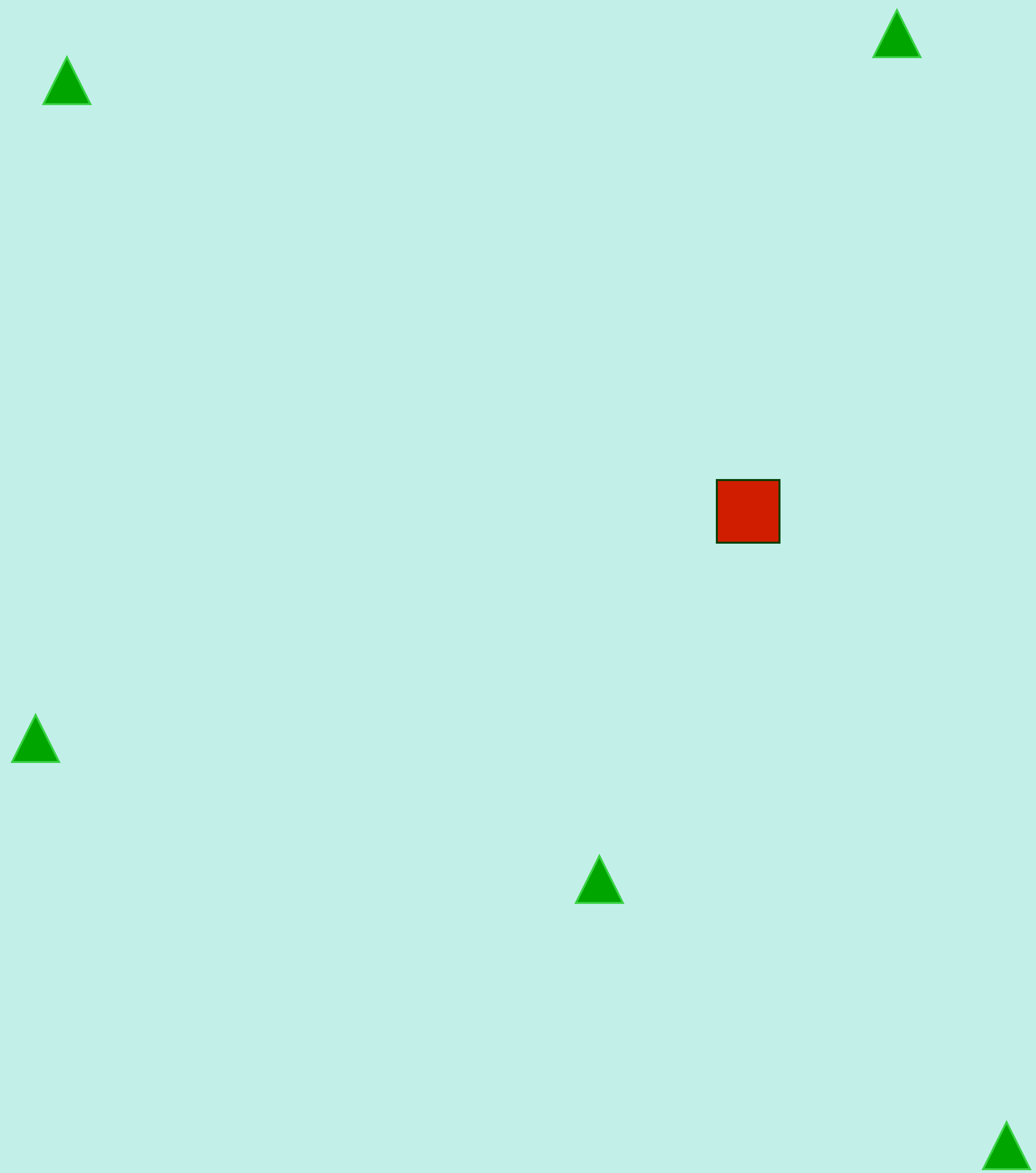
Why is this a hard problem?

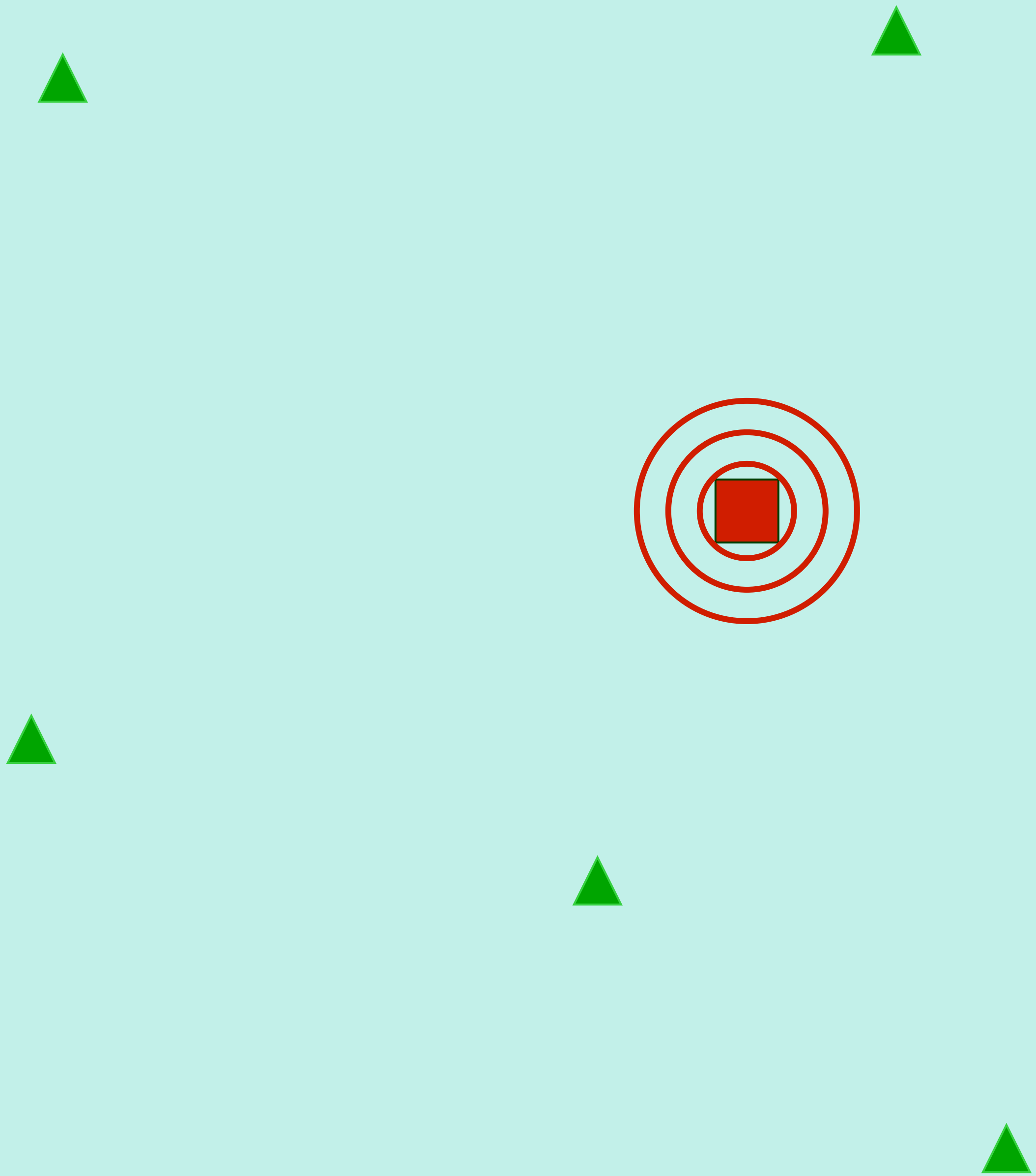
- ~10 000 “detections” per day, *90% false*
- Signals take 15 minutes to several hours to traverse the earth, so they are all mixed up
- CTBTO system (GA→SEL3) developed over 10 years, $100M software plus $1B network
  - Finds 69% of significant events plus about twice as many spurious (nonexistent) events
  - 16 human analysts correct or discard SEL3 events, create new events, generate LEB (“ground truth”)
  - Unreliable below magnitude 4 (1kT)

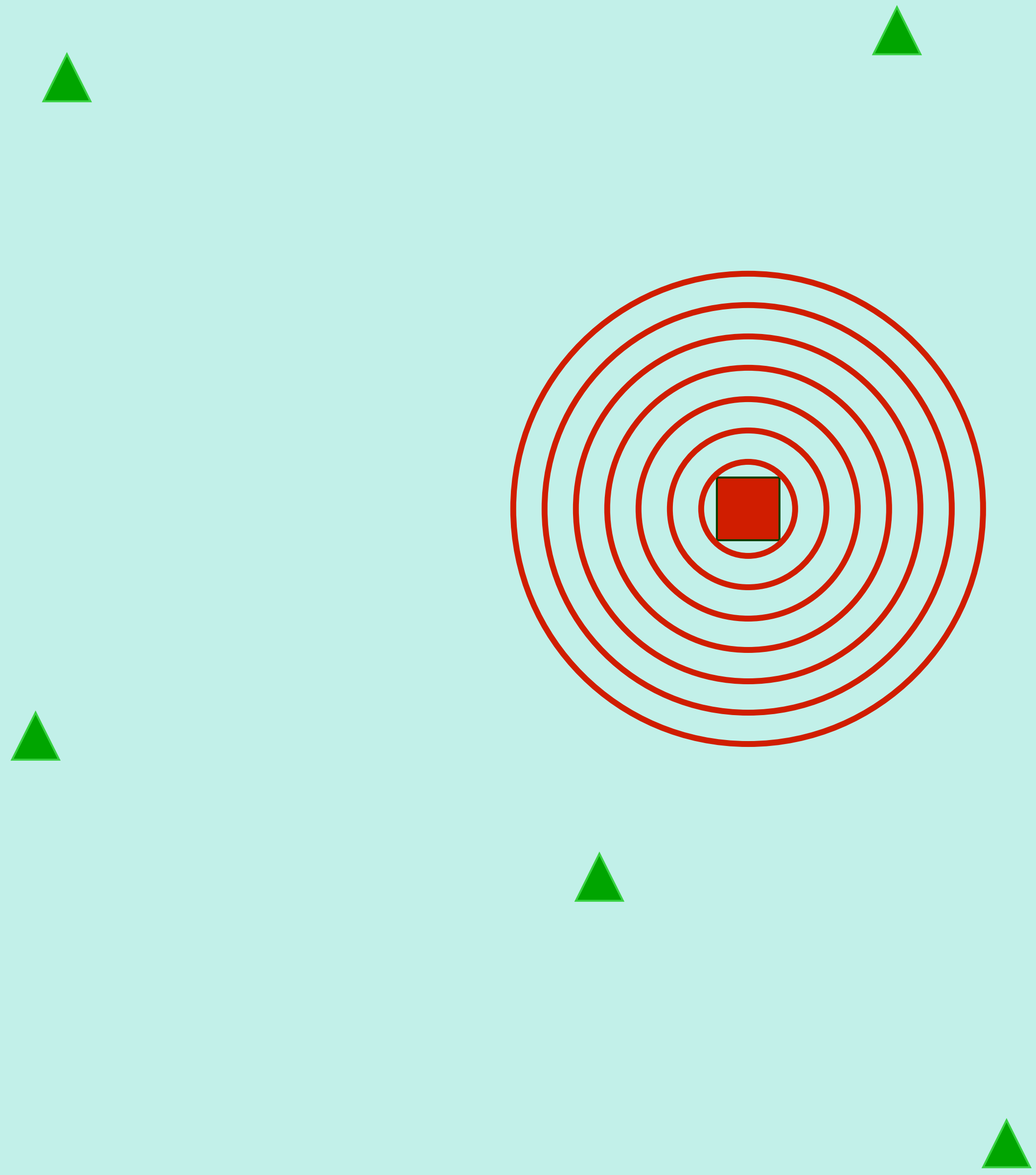

File: 020326~2.psn
Start: 3/26/02 3:45:05 UTC (L) Station: Edmonds WA 47.849N 122.328W Samples: 179975 SPS: 25
Comment: M6.5 9724 Km from Edmonds WA, SW RYUKYU ISL., JAPAN Max/Min: 4746/-5112 X: 1:15:00 Y: x1
Event Time: 03/26 03:45:48.0 Lat/Long: 23.54N 123.91E Depth: 33km 20.5mi Mag: M6.5
Org: 3:45:47.9
Diff: 10:36.9min Dist: 87.467deg 9724.3km 6038.8mi Mag: Ml?? JB: 33
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

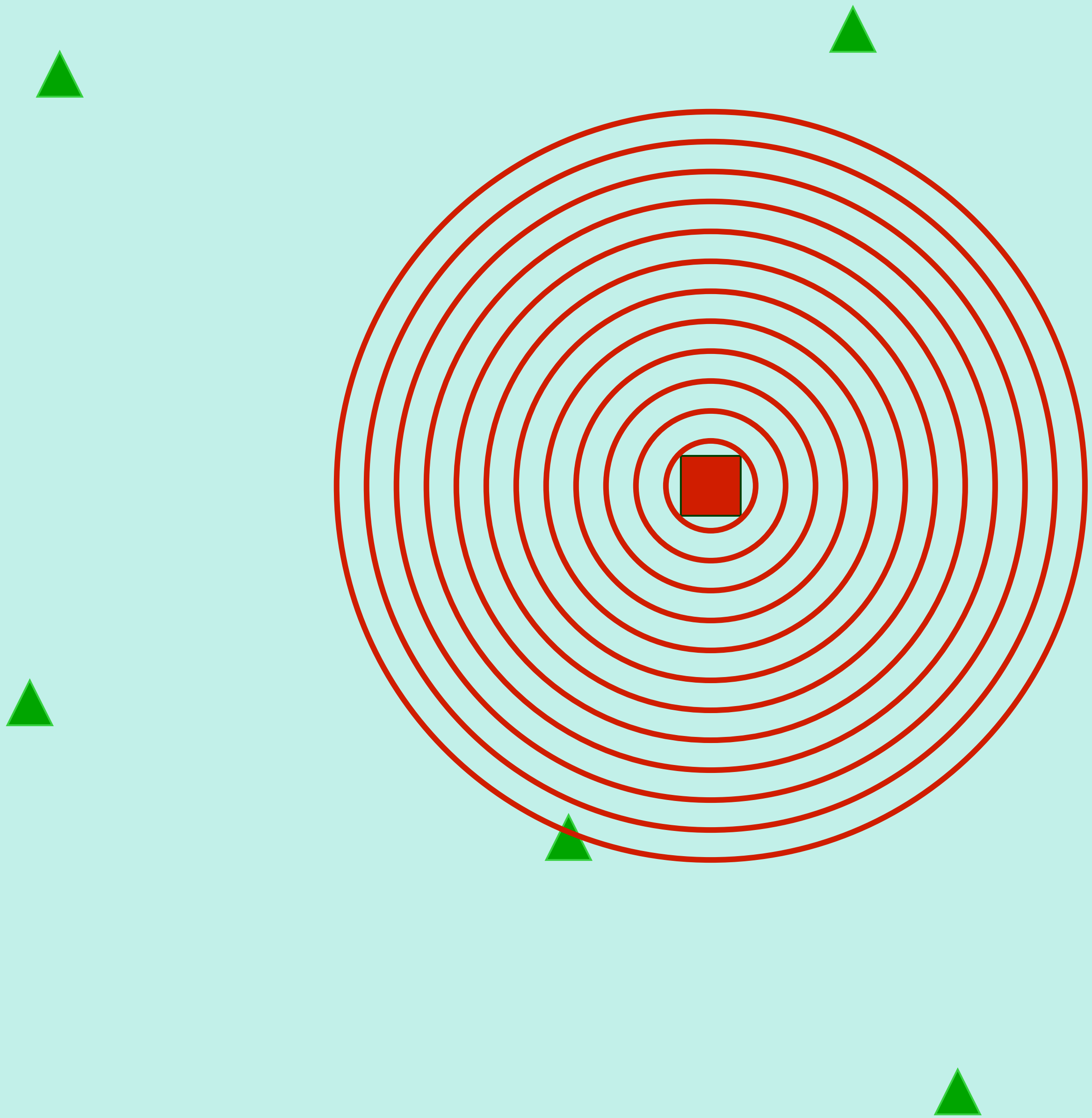

File: 020326~2.psn
Start: 3/26/02 3:45:05 UTC (L) Station: Edmonds WA 47.849N 122.328W Samples: 179975 SPS: 25
Comment: M6.5 9724 Km from Edmonds WA, SW RYUKYU ISL., JAPAN Max/Min: 4746/-5112 X: 1:15:00 Y: x1
Event Time: 03/26 03:45:48.0 Lat/Long: 23.54N 123.91E Depth: 33km 20.5mi Mag: M6.5
Org: 3:45:47.9
Diff: 10:36.9min Dist: 87.467deg 9724.3km 6038.8mi Mag: Ml?? JB: 33
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

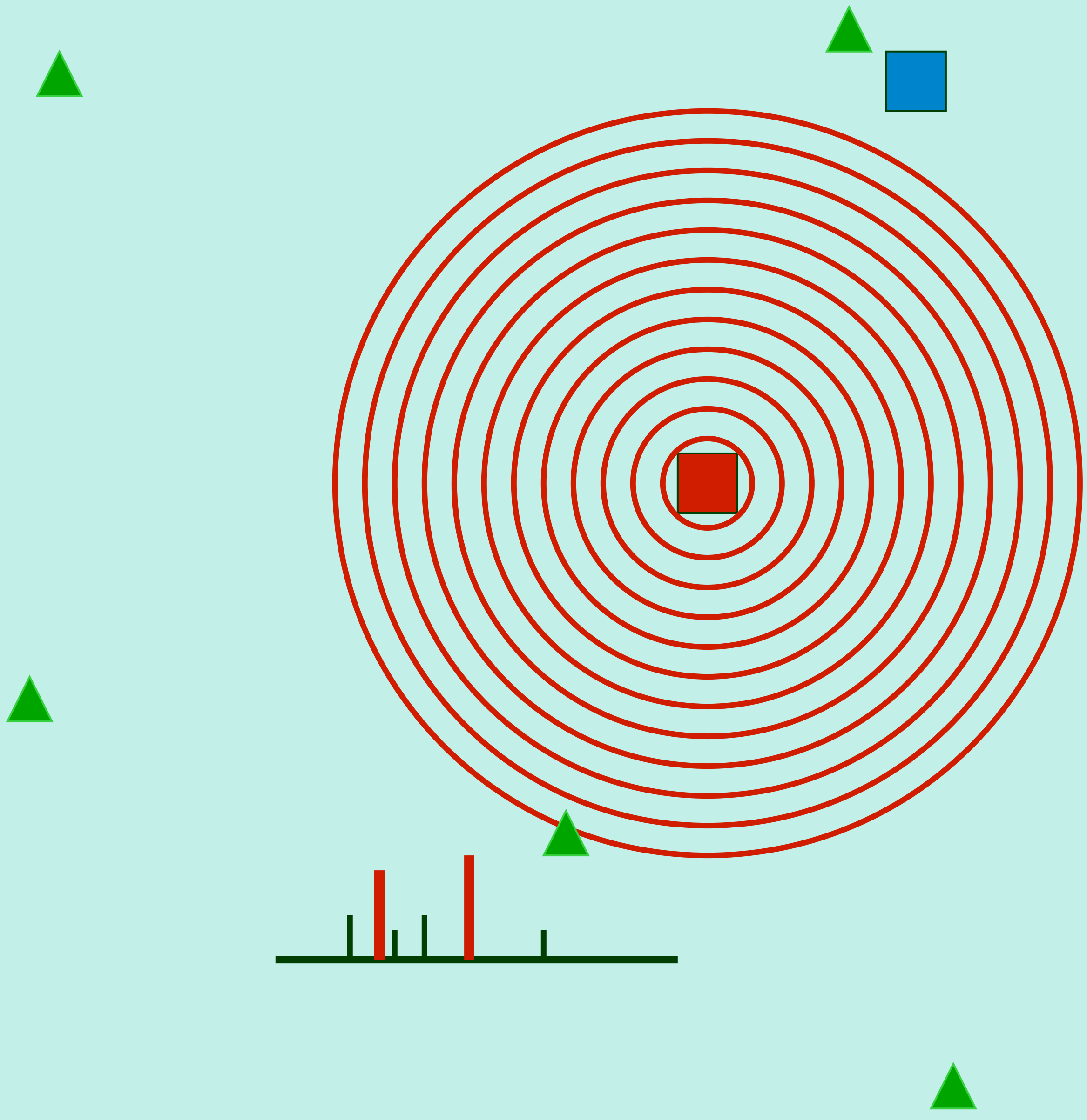
Very short course in seismology

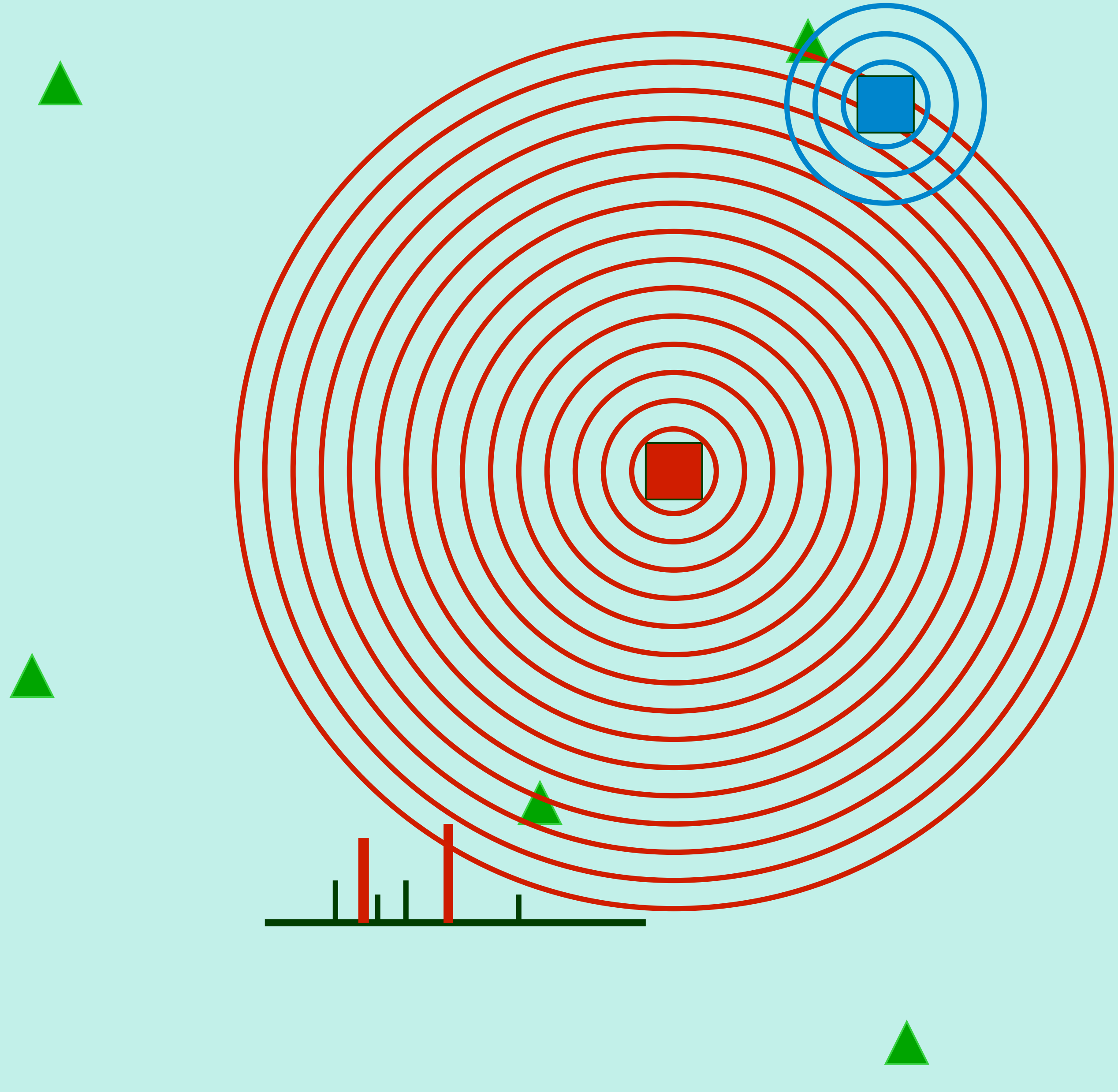


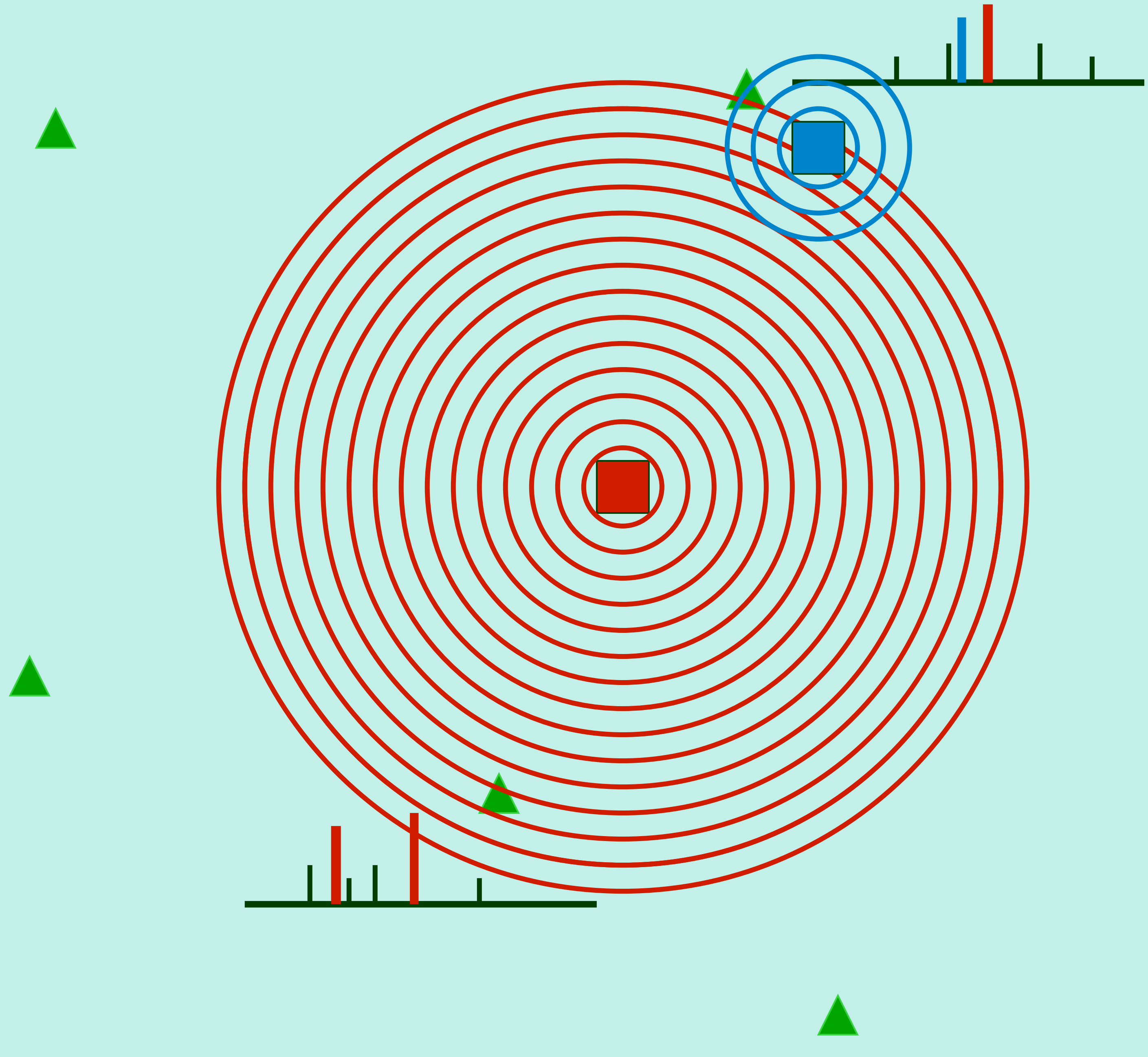


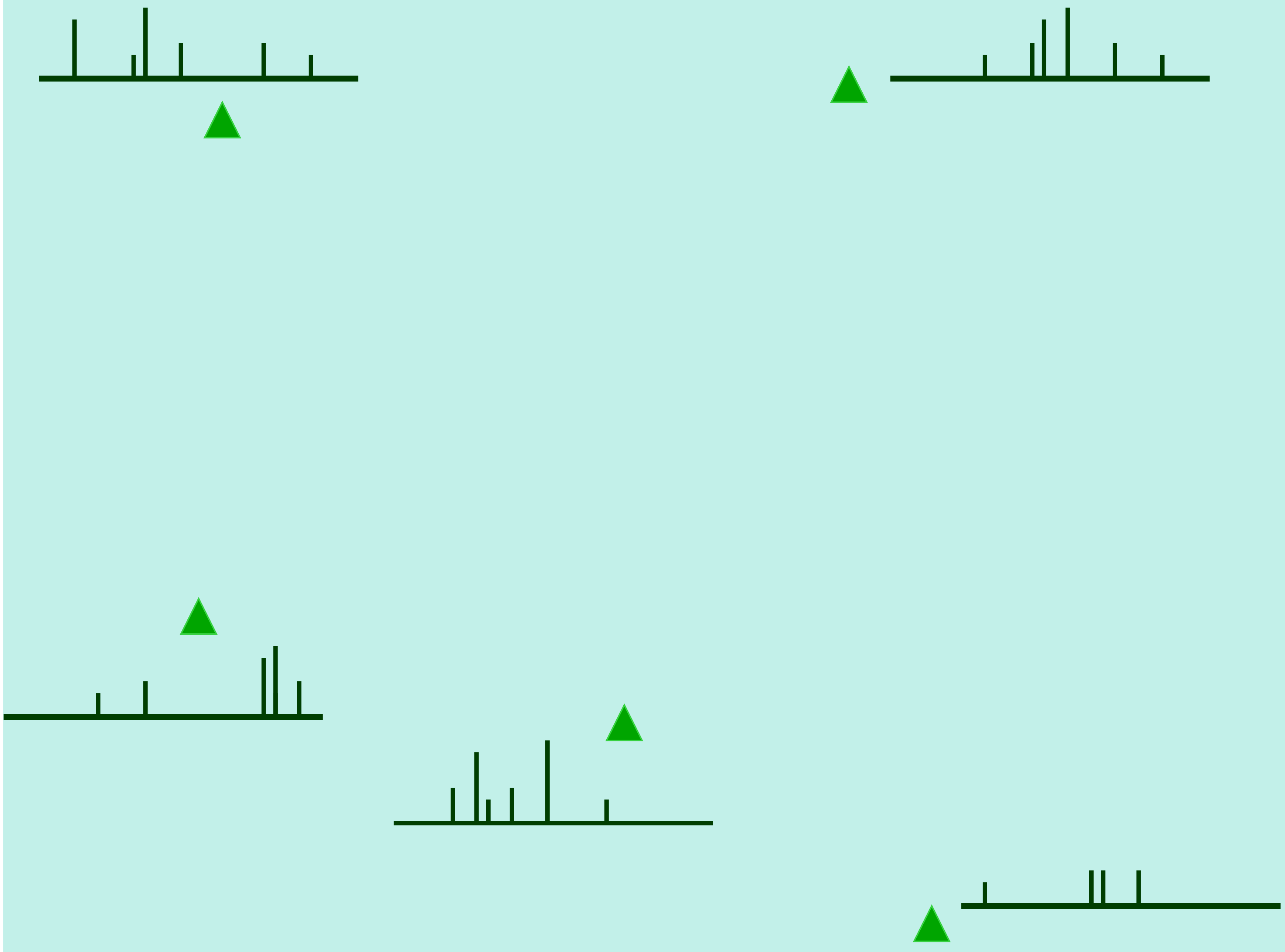




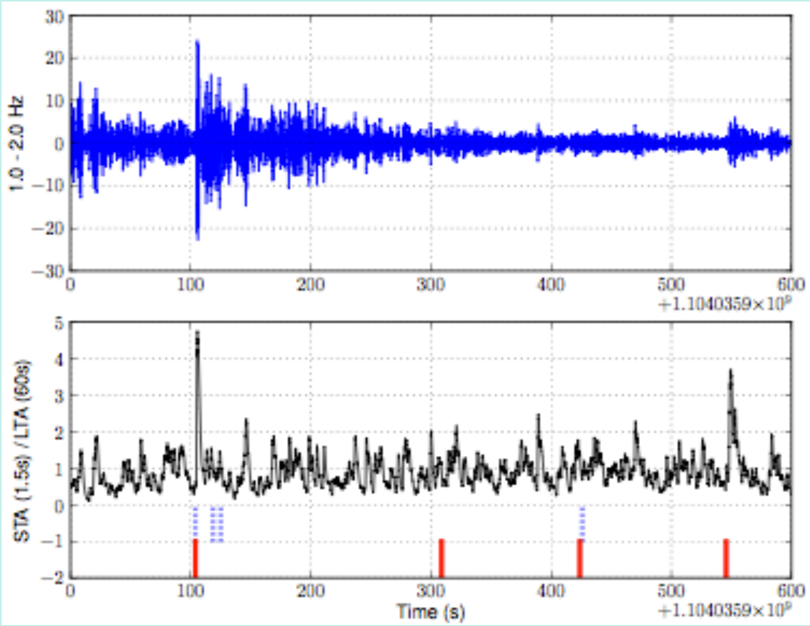








Detections



- Local spike in signal value; attributes are:
  - *Onset time**
  - *Amplitude**
  - *Azimuth* and slowness** (= direction it arrives from)
  - *Phase** (= one of 14 distinct wave types: P, S, etc.)

Open-universe model

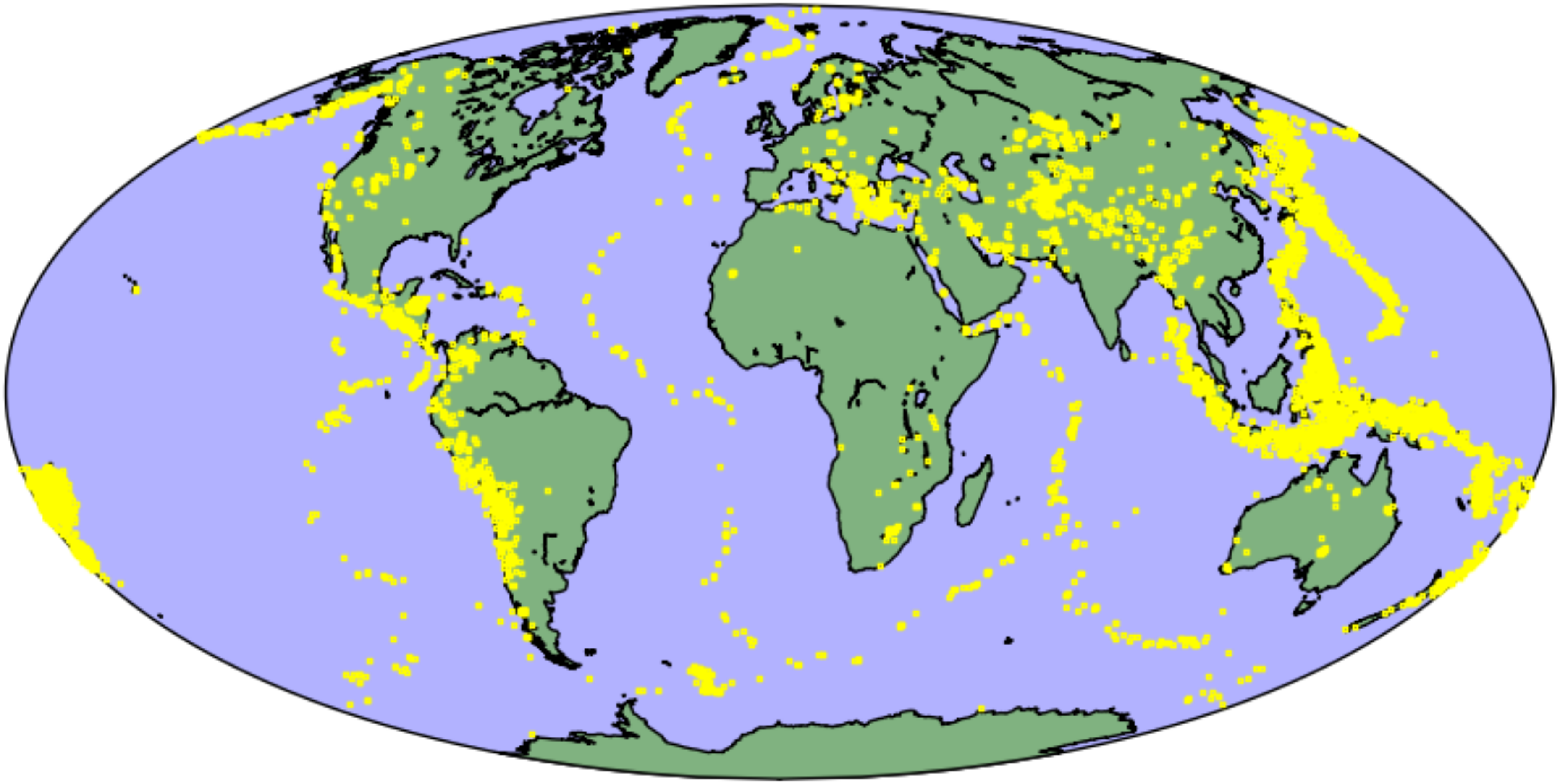
- Created a BLOG model describing
  - Event occurrence
  - Signal propagation
  - Signal detection probability
  - Measurement uncertainty
  - Noise processes producing false detections
- Wrote a fast inference algorithm for this model
- => NET-VISA (network vertically integrated seismic analysis)

#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 Locations



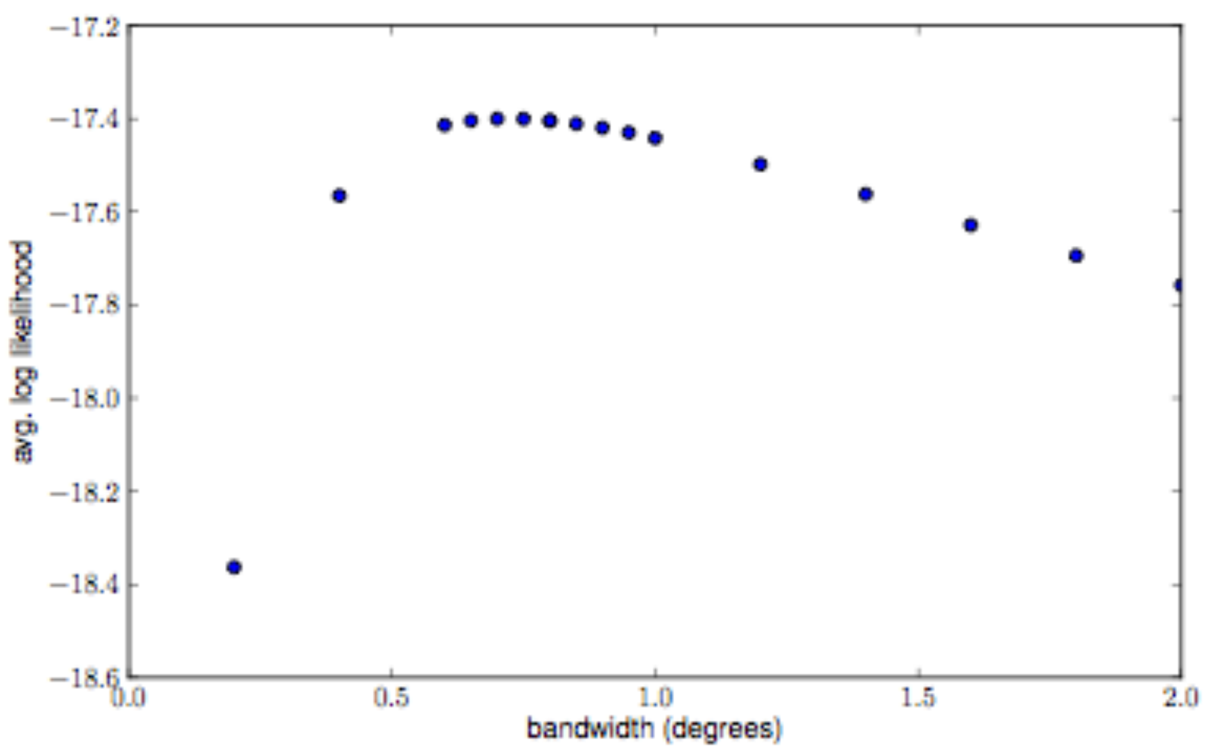
Estimating the location prior

- Kernel density estimate plus uniform component:

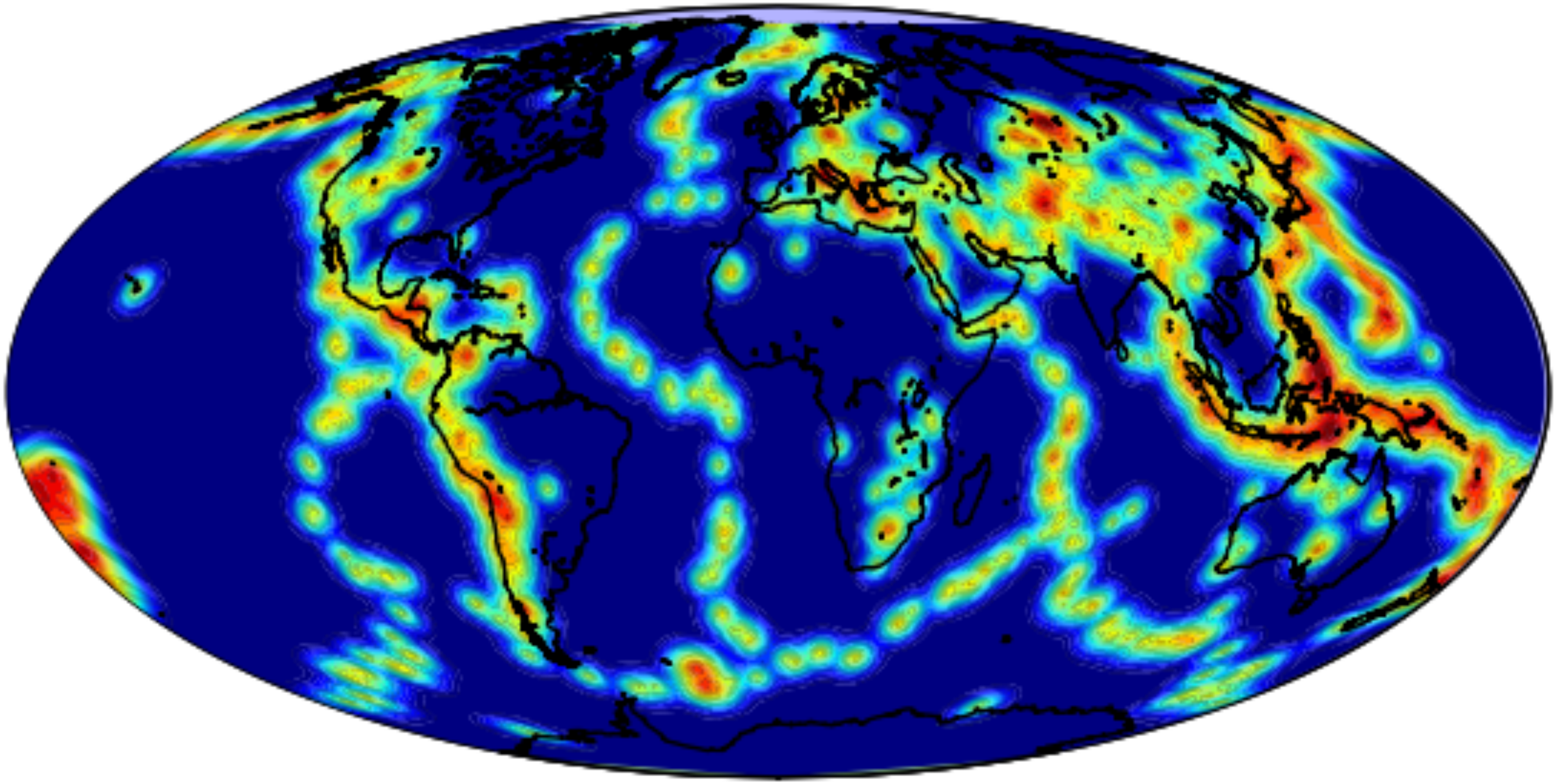
$$P_{\theta,l}(e_l) = .001\frac{1}{4\pi R^2} + .999\frac{1}{H}\sum_{h=1}^{H} K_{b,g_l^h}(e_l)$$

$$K_{b,x}(y) = \frac{1+1/b^2}{2\pi R^2}\frac{\exp\left(-\Delta_{xy}/b\right)}{1+\exp\left(-\pi/b\right)}$$

- Kernel width *b* estimated by LOOCV:



Event location prior

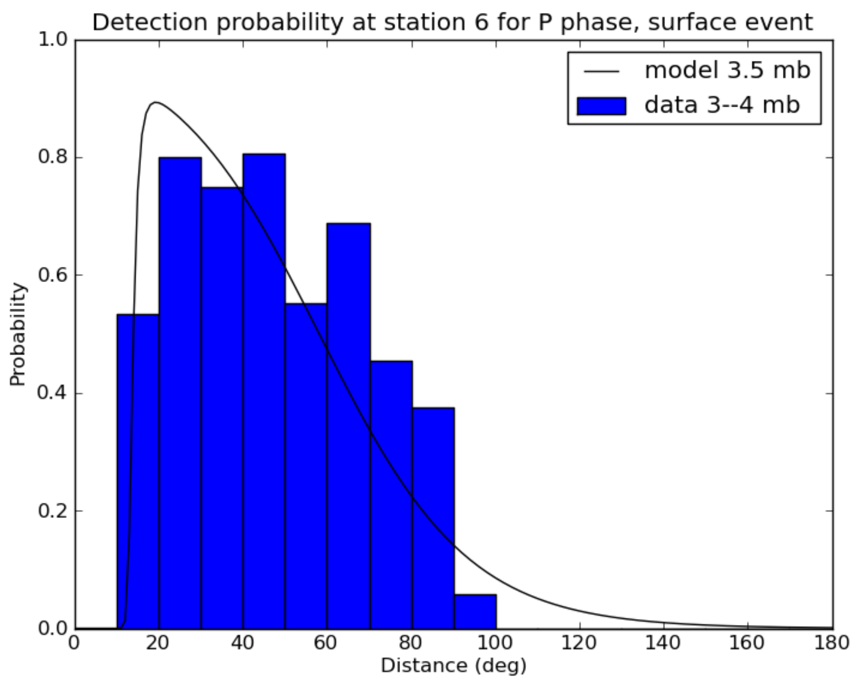


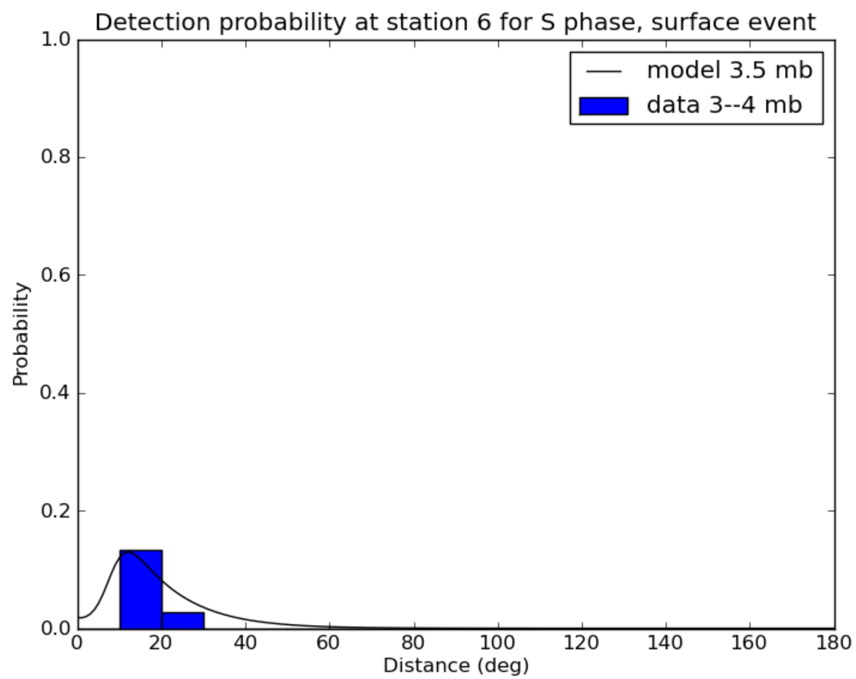
#SeismicEvents ~ Poisson[T*λ_e];
***Time(e)* ~** Uniform(0,T)
IsEarthQuake(e) ~ Bernoulli(.999);
Location(e) ~ if IsEarthQuake(e) then SpatialPrior() else UniformEarthDistribution();
Depth(e) ~ if IsEarthQuake(e) then Uniform[0,700] else 0;
Magnitude(e) ~ Exponential(log(10));
IsDetected(e,p,s) ~ Logistic[weights(s,p)](Magnitude(e), 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

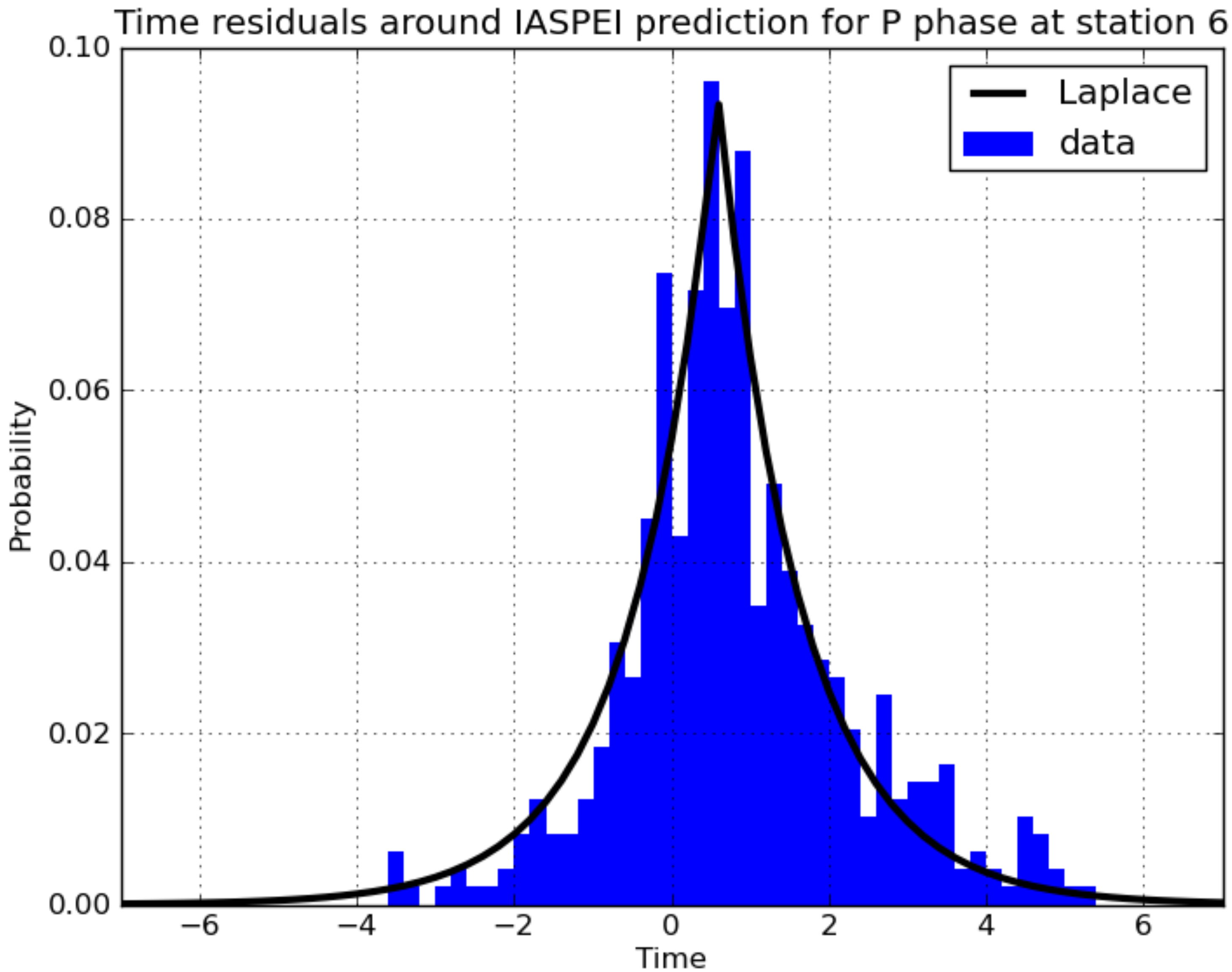
S phase





#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))

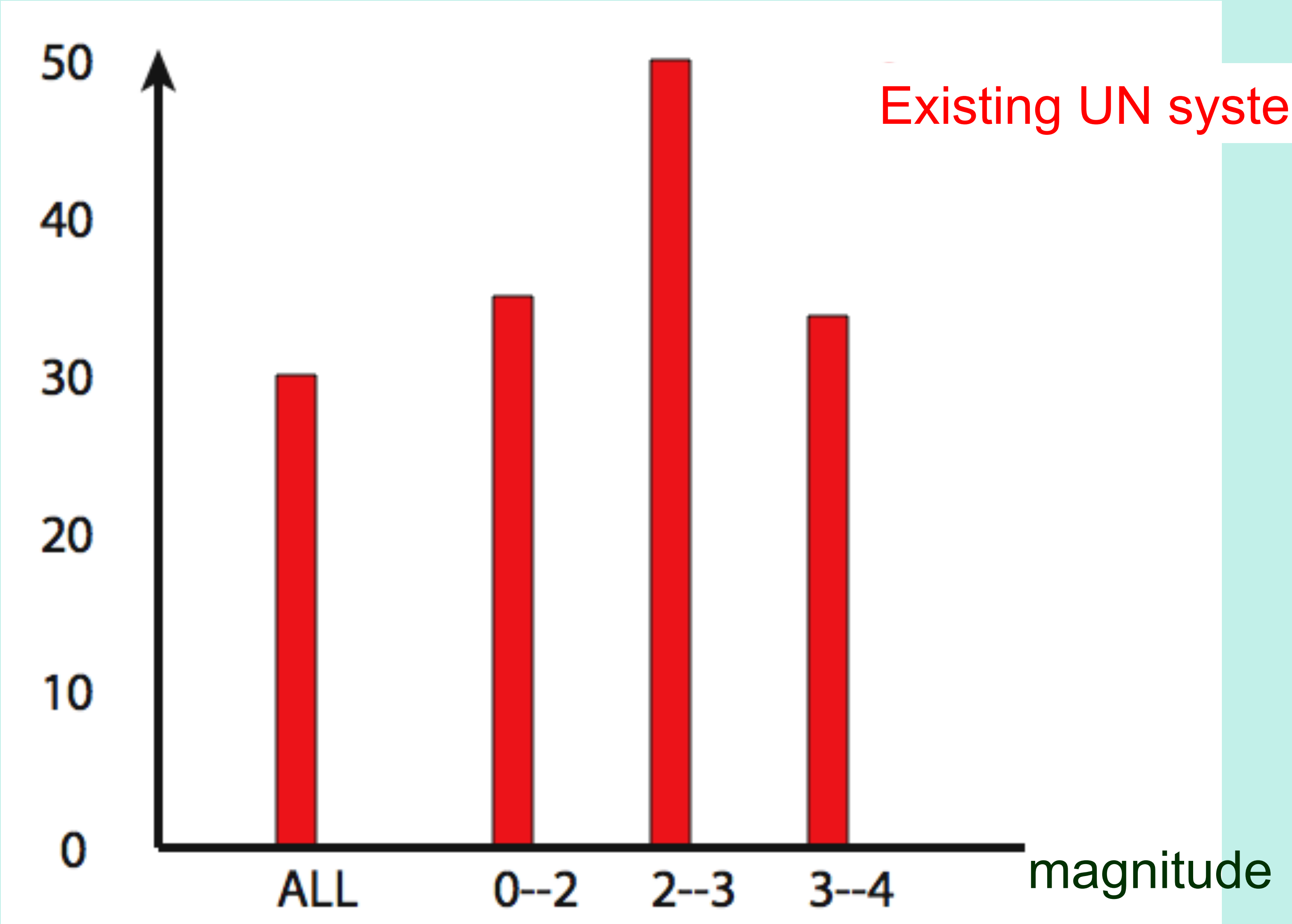
Travel-time residual (station 6)



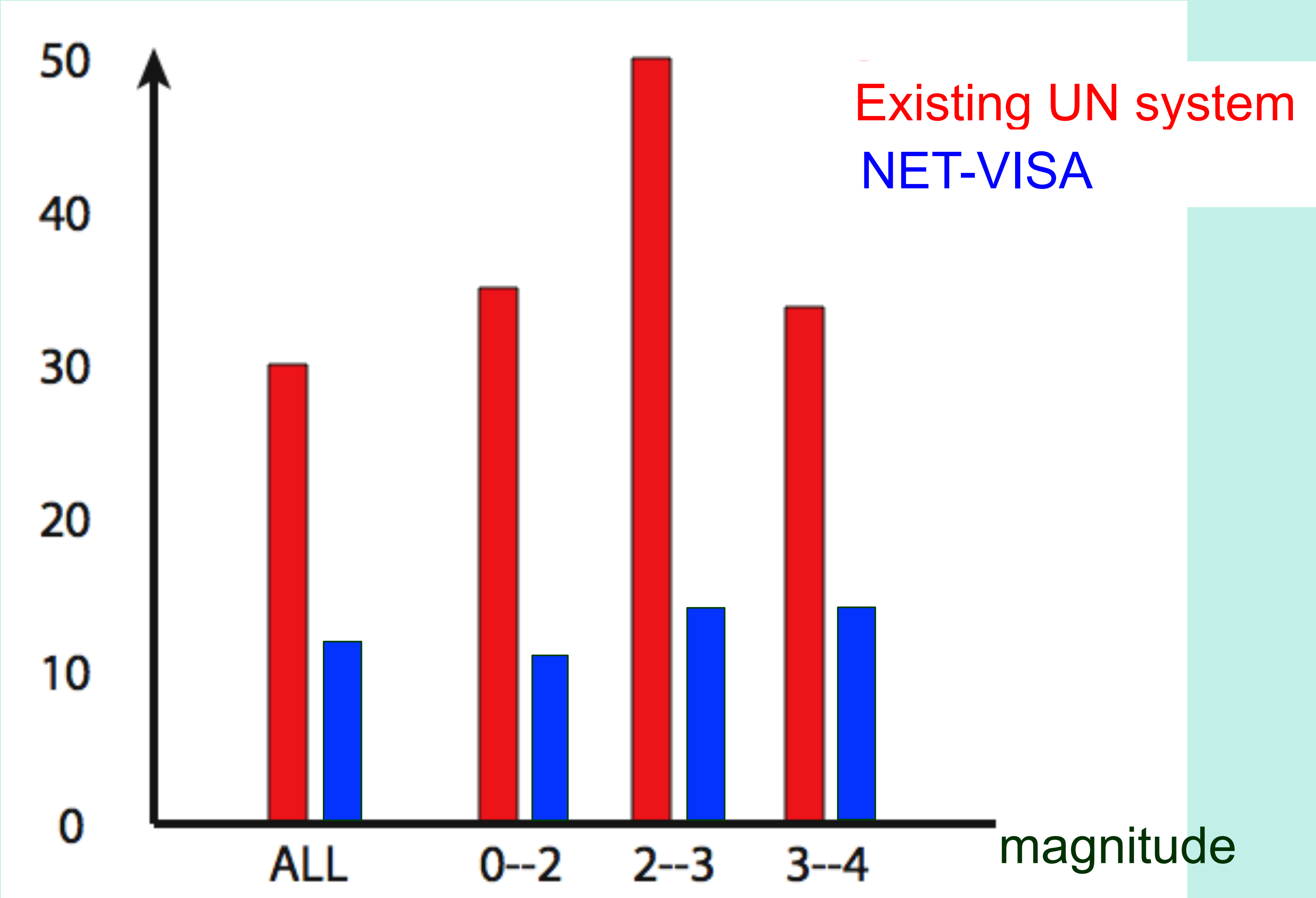
Evaluation

- 11 weeks of training data, April 6 – June 20, 2009
- 1 week of validation data, March 22-29, 2009
  - 832 LEB events
- Evaluated existing UN automated system (GA) and NET-VISA using LEB as “ground truth”

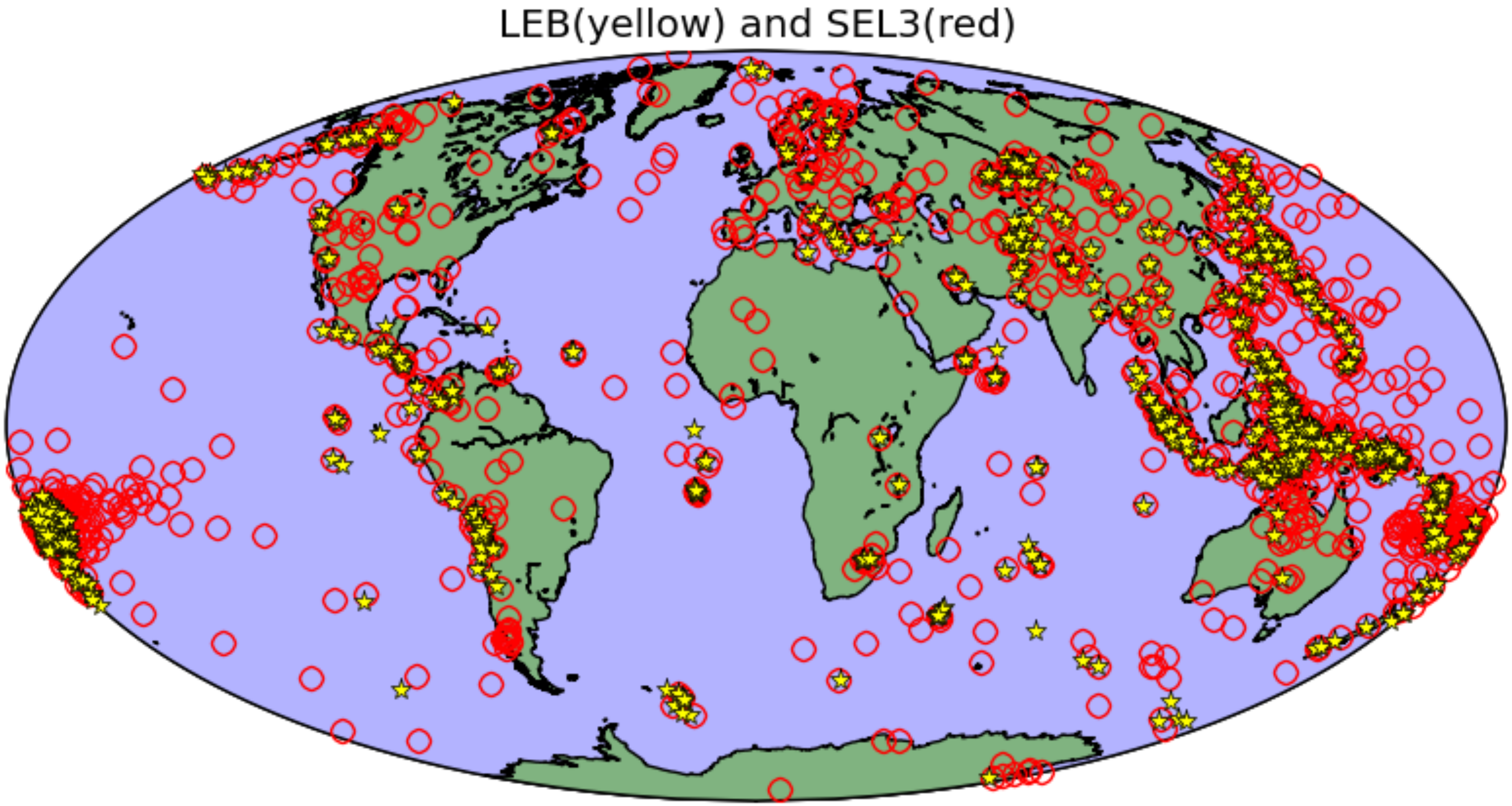
Fraction of events missed



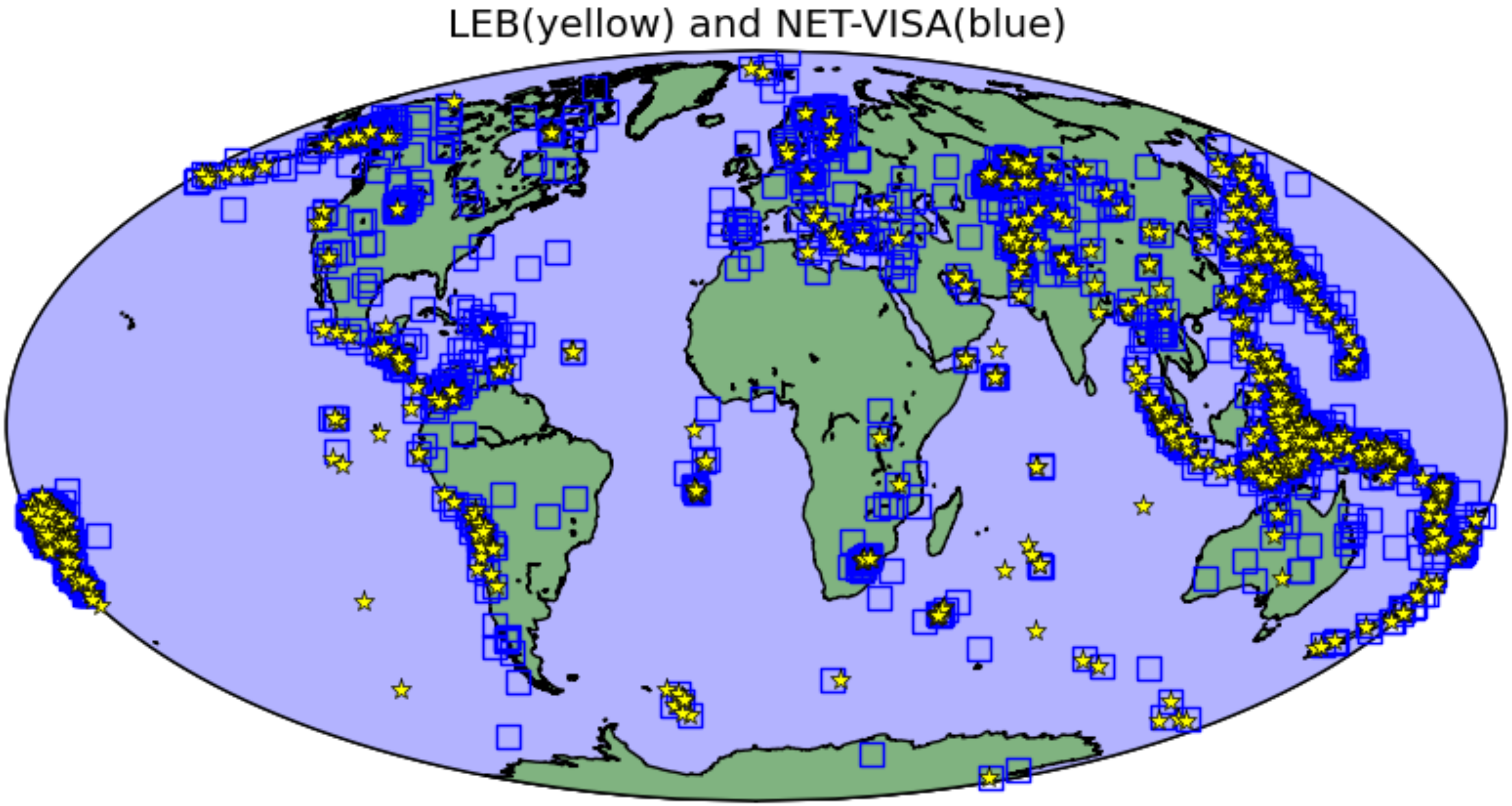
Fraction of events missed



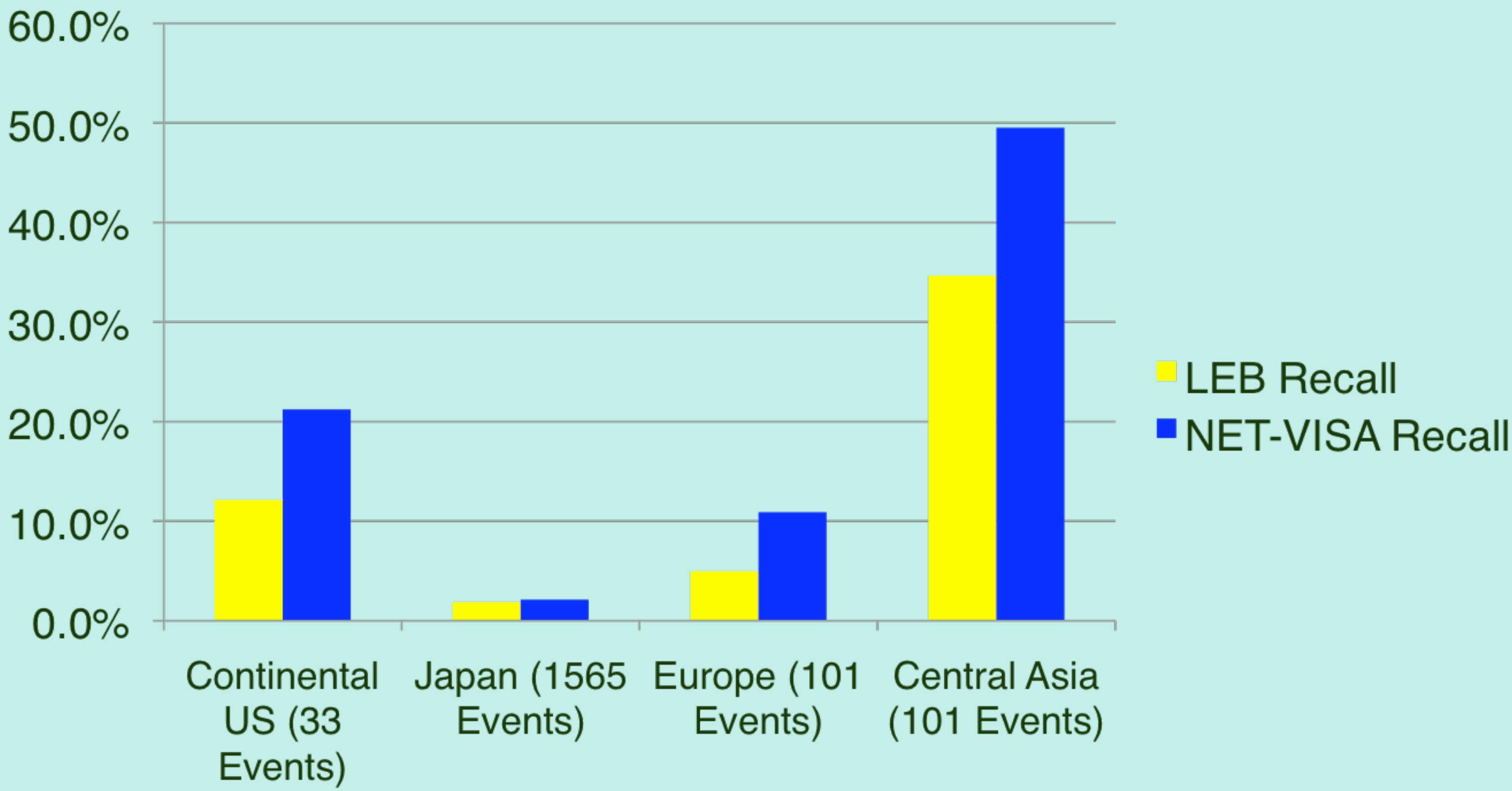
Event distribution: LEB vs SEL3



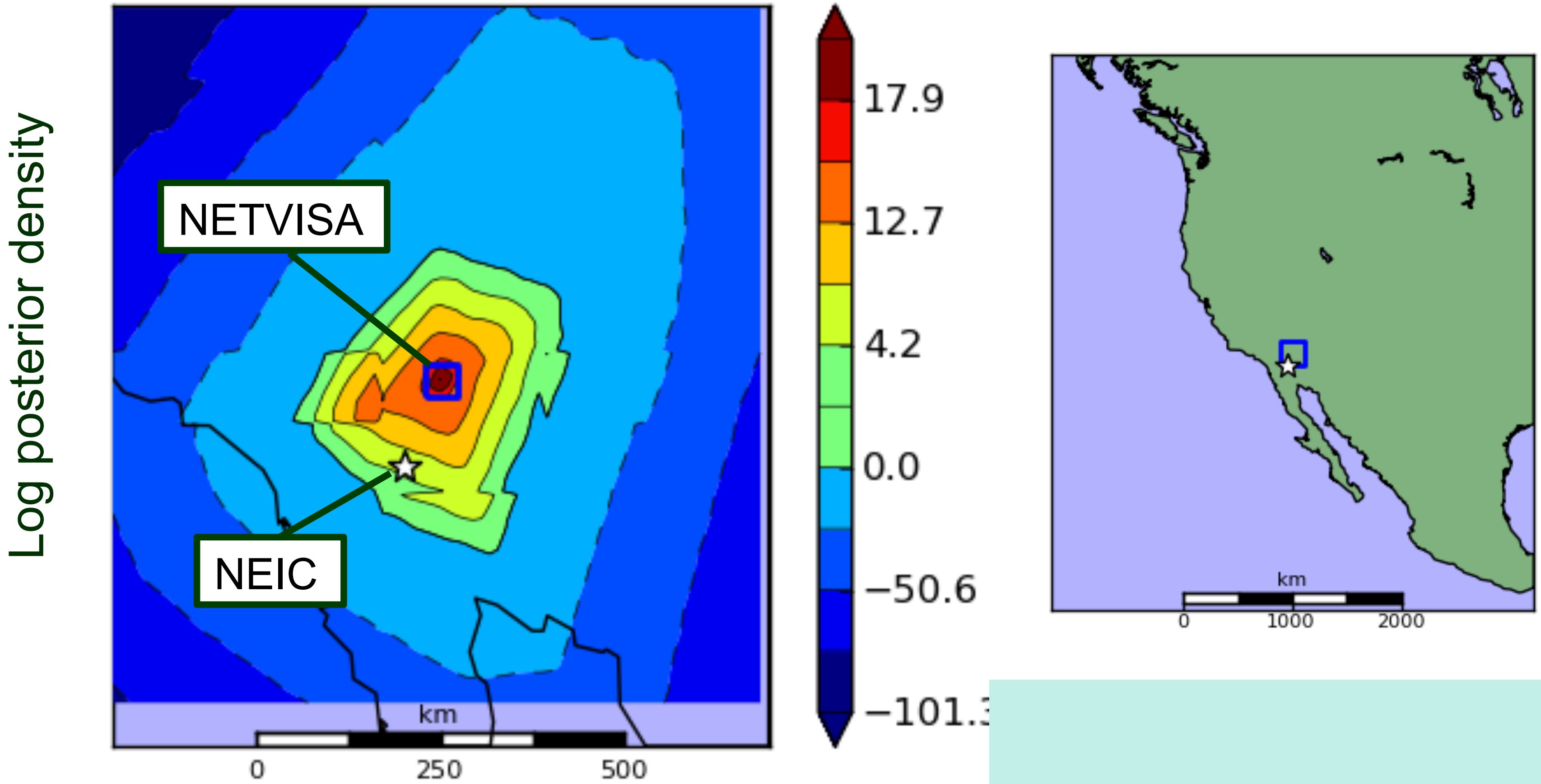
Event distribution: LEB vs NET-VISA



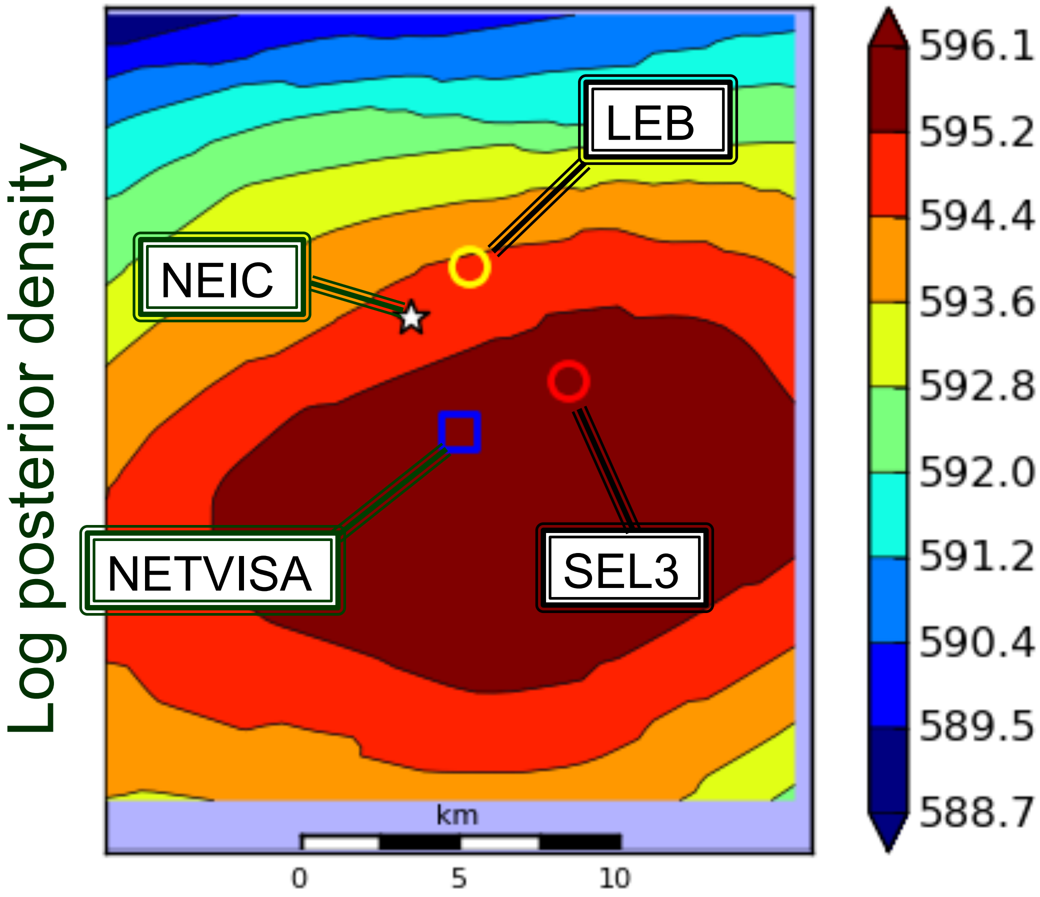
Detecting Events Found by Regional Networks



NEIC Event not in LEB



North Korea event of 5/25/09



Separate training set
1/4/08-1/4/09
Test set 1/5/09-1/26/09

Number of associated detections for event:

SEL3	39
NET-VISA	53
LEB	53

50 of 53 detections in common between LEB and NET-VISA;
LEB added 8 by hand

Concluding remarks

- AI, control theory, operations research, and statistics tried and failed to find common ground in the 1950s

Concluding remarks

- AI, control theory, operations research, and statistics tried and failed to find common ground in the 1950s
  - They had no common mathematical formalism!

Concluding remarks

- AI, control theory, operations research, and statistics tried and failed to find common ground in the 1950s
  - They had no common mathematical formalism!
- Robotics and vision separated from AI in the 1970s and 1980s for similar reasons

Concluding remarks

- AI, control theory, operations research, and statistics tried and failed to find common ground in the 1950s
  - They had no common mathematical formalism!
- Robotics and vision separated from AI in the 1970s and 1980s for similar reasons
- Bayesian networks and statistical machine learning helped to reintegrate some of these fields in the 1990s

Concluding remarks

- AI, control theory, operations research, and statistics tried and failed to find common ground in the 1950s
  - They had no common mathematical formalism!
- Robotics and vision separated from AI in the 1970s and 1980s for similar reasons
- Bayesian networks and statistical machine learning helped to reintegrate some of these fields in the 1990s
- Unifying logic and probability may help us to
  - complete the reintegration of reasoning, planning, perception, and language understanding
  - find the light at the end of the tunnel

Merci pour votre attention!

L'orateur est soutenu par, et cette présentation est donnée sous les auspices de, la Chaire Internationale de Recherche Blaise Pascal financée par l'État et la Région Île de France, gérée par la Fondation de l'École Normale Supérieure.

Joint work with Brian Milch, David Sontag, Andrey Kolo bov, 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

+ help from Ola Dahlman, Ronan LeBras, Lassina Zerbo, Sheila Vaidya, Bob Engdahl, Barbara Romanowicz, Jeff Given, Spilio Spiliopoulos, Elena Tomuta

Research funded by DARPA, CTBTO, and DTRA

Questions?