Visual exploration and analysis of time series earthquake data

XSEDE ESRT project
Amit Chourasia\textsuperscript{1}, Keith Richards-Dinger\textsuperscript{2}, James Dietrich\textsuperscript{2} & Yifeng Cui\textsuperscript{1}
\textsuperscript{1}SDSC/UC San Diego, \textsuperscript{2}UC Riverside

XSEDE ECSS Symposium May 16, 2017
Research motivation
Earthquake hazard estimation is an important problem. It requires long term past records of earthquakes at all scales (magnitude, space, time)

Problem: We neither have, nor could gather such data from the field

Solution: Develop synthetic method based on first principals to generate such records that could bridge this critical gap of missing data

RSQSim visualization : Amit Chourasia, SDSC
RSQsim
Generates synthetic quakes at all scales i.e. magnitude, space and time

Computation

• Code runs on $O(1K)$ cores with $O(100K)$ elements
• Future needs: $O(1M)$ elements
• Memory usage scales as $N_{\text{elem}}^2$

Scaling and optimization work by Dmitry Pekurovsky (SDSC)
Output data
Event (12 variables): csv file
Fault geometry (11 variables): csv file
Event action (8 variables + index) : binary files

The data is spread over 11 files (2 ASCII, 9 binary)

Sample data catalog

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Size</td>
<td>2.4 GB</td>
</tr>
<tr>
<td>Files</td>
<td>2 ascii, 9 binary files</td>
</tr>
<tr>
<td>Catalog time duration</td>
<td>From 50k to 90k years (40k years)</td>
</tr>
<tr>
<td>Number of events</td>
<td>5,970,621</td>
</tr>
<tr>
<td>Event variables</td>
<td>12</td>
</tr>
<tr>
<td>Number of fault patches</td>
<td>260,051</td>
</tr>
<tr>
<td>Fault patch variables</td>
<td>11</td>
</tr>
<tr>
<td>Number of event actions</td>
<td>19,127,461</td>
</tr>
<tr>
<td>Event action variables</td>
<td>8</td>
</tr>
</tbody>
</table>

RSQSim visualization : Amit Chourasia, SDSC
Early data exploration

- Number of events per year (frequency)
- Max magnitude for each year (maxM)

~ 40,000 records with year, frequency and maxM
<500 events per year
RSQSim visualization: Amit Chourasia, SDSC
Project initiation

- Scientist has a custom 3D interactive vis application
  Lacks geographic context
  Lacks multiuser capability

- Develop an interactive visualization in 2D with richer geographic context, ideally usable from a web browser
Challenges

Data marshaling

- Normalize index to 1 based
- Normalize variable names
- Translate 3D data to 2D
  (triangle patches stored with vertex + 3D rotations)
  (rectangle patches stored with center + length, width + 3D rotations)
- Add geographic projections from UTM to EPSG 4236
- Consider using raw data vs database

Visualization

- What to display?
- How to display? Which visualization idioms to use?
- Can it work in a web browser?

RSQSim visualization: Amit Chourasia, SDSC
Visualization goal: Support geographic exploration

- Show an event
- Show fault patches affected by the event
- Interactive
Implementation
Developed a web app using the following

- **Python**
  - Flask framework: server/client brokering
  - Folium: map plotting

- **JavaScript**
  - Leaflet: mapping
  - C3JS: time series plots
  - noUISliderJS: mobile friendly sliders

- **HTML**
Implementation Phase 1

• Build an app with raw data
• Data wrangling
  Format switched from ASCII to binary
  Need to also support rectangular patch elements stored with different scheme

Problems

  Slow start up 20 min (pickling brought it to 2min)
  Considerable memory use
  Painful development/debugging
Implementation Phase 2

- Transform data to database
  Which database to use?
  Develop database schema?
  Verify data translation to database
  Verify and validate geographic projection transformations

- Retool application to use database
  Switch search patterns to queries
  Slow query performance
Implementation Phase 3

- Develop user interface
  Select event by ID (~ 6M)
  Select event by time (40,000 years at second granularity)
  Other filters

- Link map display with time series
- UI needs to be mobile friendly
Visualization design
Show selected event with ID: 1419930

Show fault patches affected by selected event  (595 affected)
Show trail before/after events (± 5 events connected by a line)
Show time series adjacent to selected event (± 100 events)
Interaction

- **Selection**
  - Event
  - Time
- **Filters**
  - Magnitude
  - Number of patches
  - Trail events
- **Map**
  - Layers
  - Markers pop-ups
- **Time series**
  - Linked with map
  - Enable constraints

RSQSim visualization: Amit Chourasia, SDSC
Analysis

Calculate mean recurrence interval for all fault sub sections (~2600)

- With nucleation only: 1,785 sec
- With participation: 12,613 sec
- Verify and validate calculations

Benchmark hardware
MacPro workstation
2x 2.26 Ghz Quad Core Intel Xeon processor
16 GB memory

RSQSim visualization: Amit Chourasia, SDSC
Analysis

Calculate mean recurrence interval for all fault sub sections (~2600)

- With nucleation only: 1,785 sec
- With participation: 12,613 sec
- Verify and validate calculations

Benchmark hardware
MacPro workstation
2x 2.26 Ghz Quad Core Intel Xeon processor
16 GB memory
Conclusions

• Transformed data to a database
  Normalized data with mixed index in raw data
  Easier data management/sharing, only one file
  Scalable, as data does not need to be loaded into memory
  Fast and easy querying

• Developed interactive visualization web application
  Enables quake exploration in rich geographic context for a large catalog
  Can support multiple concurrent users

• Provided sample scientific analysis script
  Mean recurrence interval with nucleation and with nucleation and participation

RSQSim visualization : Amit Chourasia, SDSC
ESRT project deliverables

- Translation script to create SQLite database from raw data
- Web based visualization application
- Analysis script and other query examples
- Documentation
Potential for research team

• Use less code for analysis (no parsing needed)
• Maintain less error prone code for analysis (queries simplify manual filtering)
• Write output data to SQLite database from simulation instead of ascii+binary files
• Scientists will need to learn to write SQL queries
Technical lessons learned

Gained familiarity with

- Database schema design
- SQLite database
- Benchmarking and optimizing database queries
- Python modules and framework

Found a bug in Folium with geojson data

Found a bug in popular heatmapjs plugin for Leaflet

- Developing web based graphical user interface
Future possibilities

RSQSim gateway

- Data service
  - Allow web based querying
  - Allow download of filtered data

- Light visualizations
  - Allow spatial filtering
  - Add heatmap
Publication

ACKNOWLEDGMENTS
This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575. The National Science Foundation grant number EAR-1135455 and Keck Foundation grant number 005590-00001 also supported this work.