Distributed, vehicular computation for map generation

Jonathan J. Davies

Alastair R. Beresford

{jjd27,arb33}@cam.ac.uk

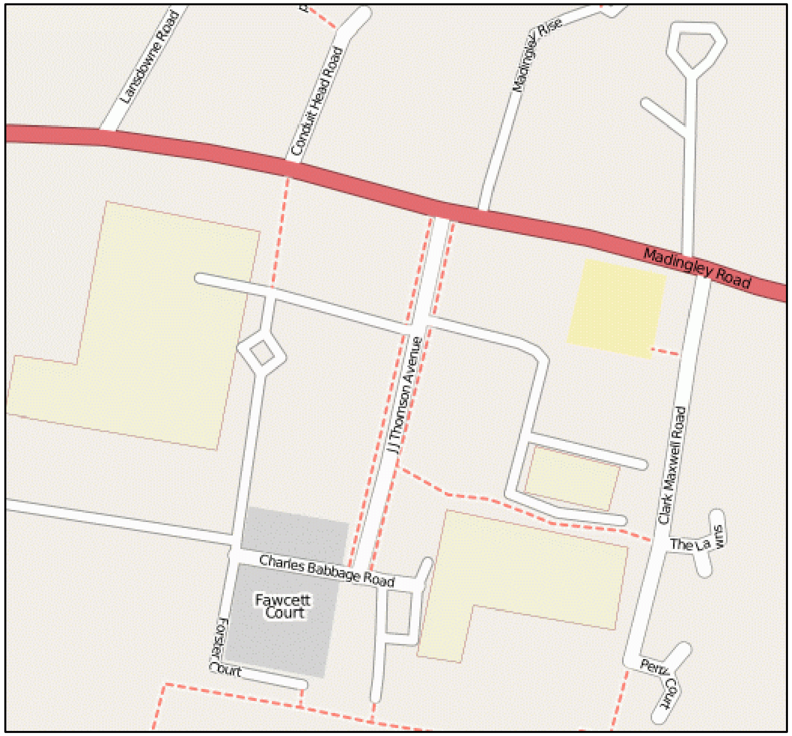
University of Cambridge Computer Laboratory

When SatNav goes wrong

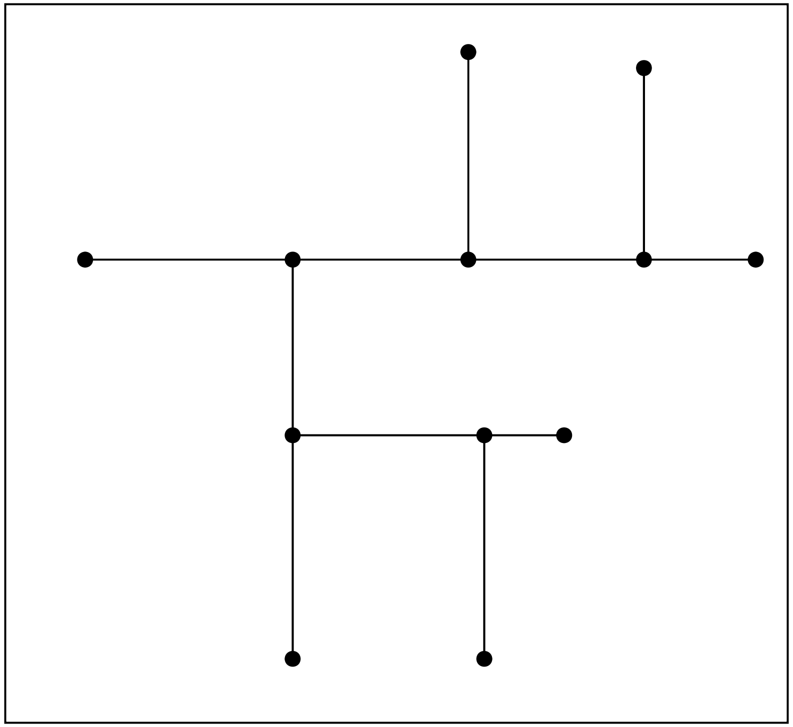
- Satellite Navigation units are becoming increasingly popular
- But their maps don't always match up with reality
  - Errors in the data
  - Recent changes to the road network
  - Temporary changes
- Can be disconcerting… or dangerous

Digital road maps

- Two kinds of road map:



Rendered
For human use



Topological (directed graph)
For computer use
Used for automated route-finding

How are digital road maps produced?

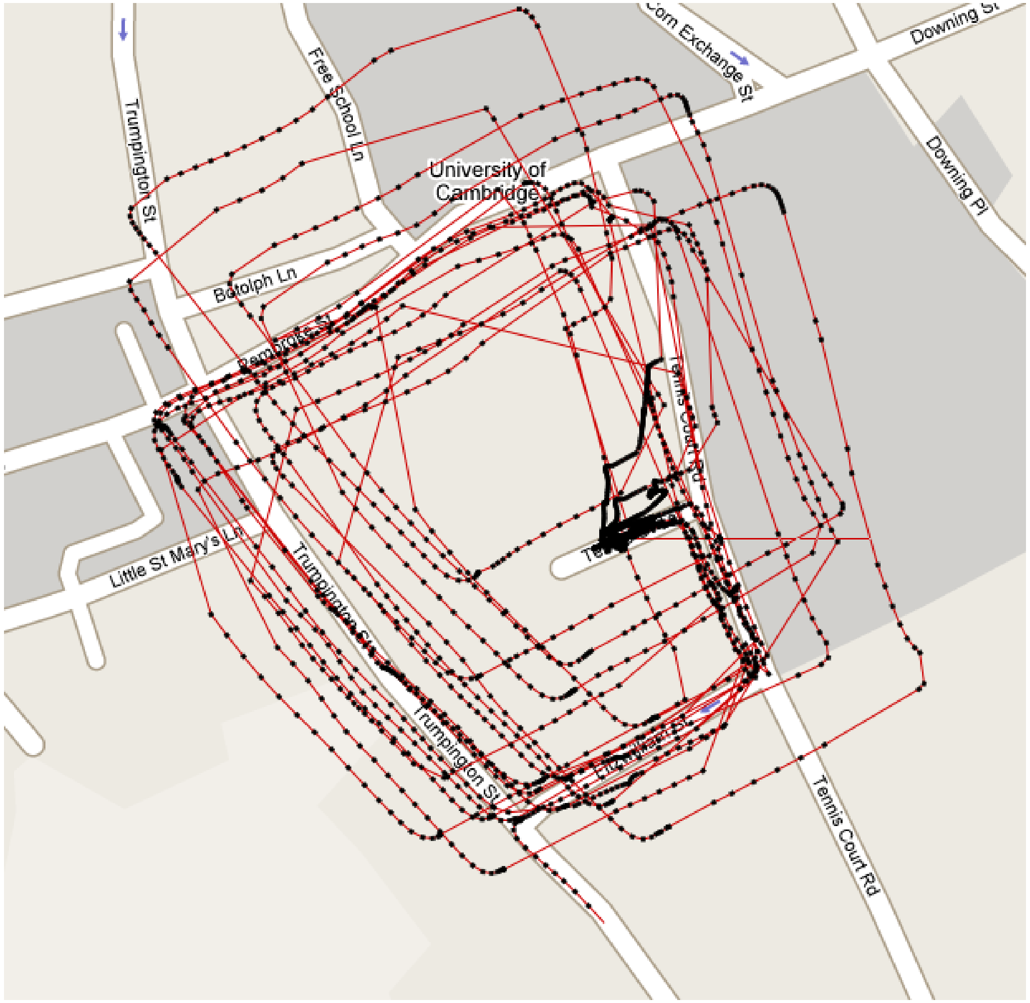
- Two major producers: Navteq, Tele Atlas
- Aerial photographs
  - Expensive to update frequently
  - Some road features not obvious from the sky
- Data from local councils, building contractors
  - Poor spatial and temporal accuracy
- Probe vehicles
  - Expensive
  - Tele Atlas drove 3.5 million miles in 2004
- Navteq spend over $10m per year keeping their databases up to date

A new solution to map generation

- Two observations:
  - Many cars have GPS units
  - Bidirectional communication with vehicles is just around the corner
- Collect GPS traces from vehicles
- Convert these into a directed graph of the road network
- Two questions to be addressed:
  - How can we collect vast quantities of GPS traces?
  - How can we convert GPS traces into graphs?

Map generation: Why is it hard?

- GPS is prone to errors, especially in cities



Producing road topology from GPS traces

1. Deduce where there is road and not road
2. Find the edges of the roads
3. Find the centrelines of the roads
   - Expressed an undirected graph
4. Determine which edges represent one-way roads
   - Makes the graph directed

For full details, see:

Jonathan J. Davies, Alastair R. Beresford, and Andy Hopper. Scalable, distributed, real-time map generation. IEEE Pervasive Computing, 5(4):47–54, Oct–Dec 2006.

Histogram

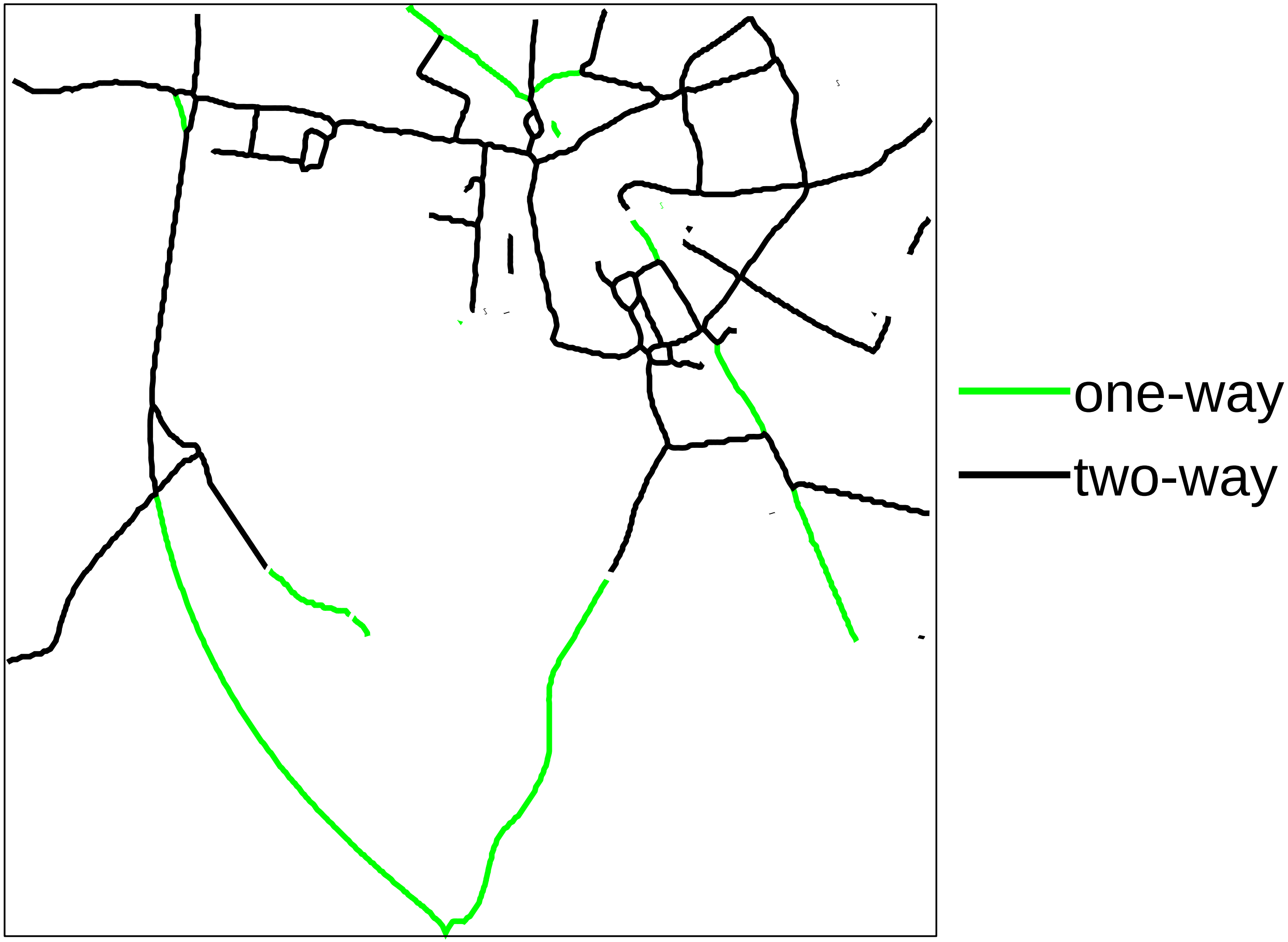
Blur

Threshold

Find Road Edges

Find Road Centrelines

Determine Roads' Directionality



Evaluation

- We can produce a directed graph from GPS data
  - We also get the shape of the roads
  - (Could also derive speed information)
  - The more GPS data, the better
- But it's tricky...
  - Errors in GPS mean that the maps might contain inaccuracies
  - No metadata (e.g. road names) associated with the edges
- However, the technique is good for producing up-to-date maps
  - So it could be useful for updating existing maps

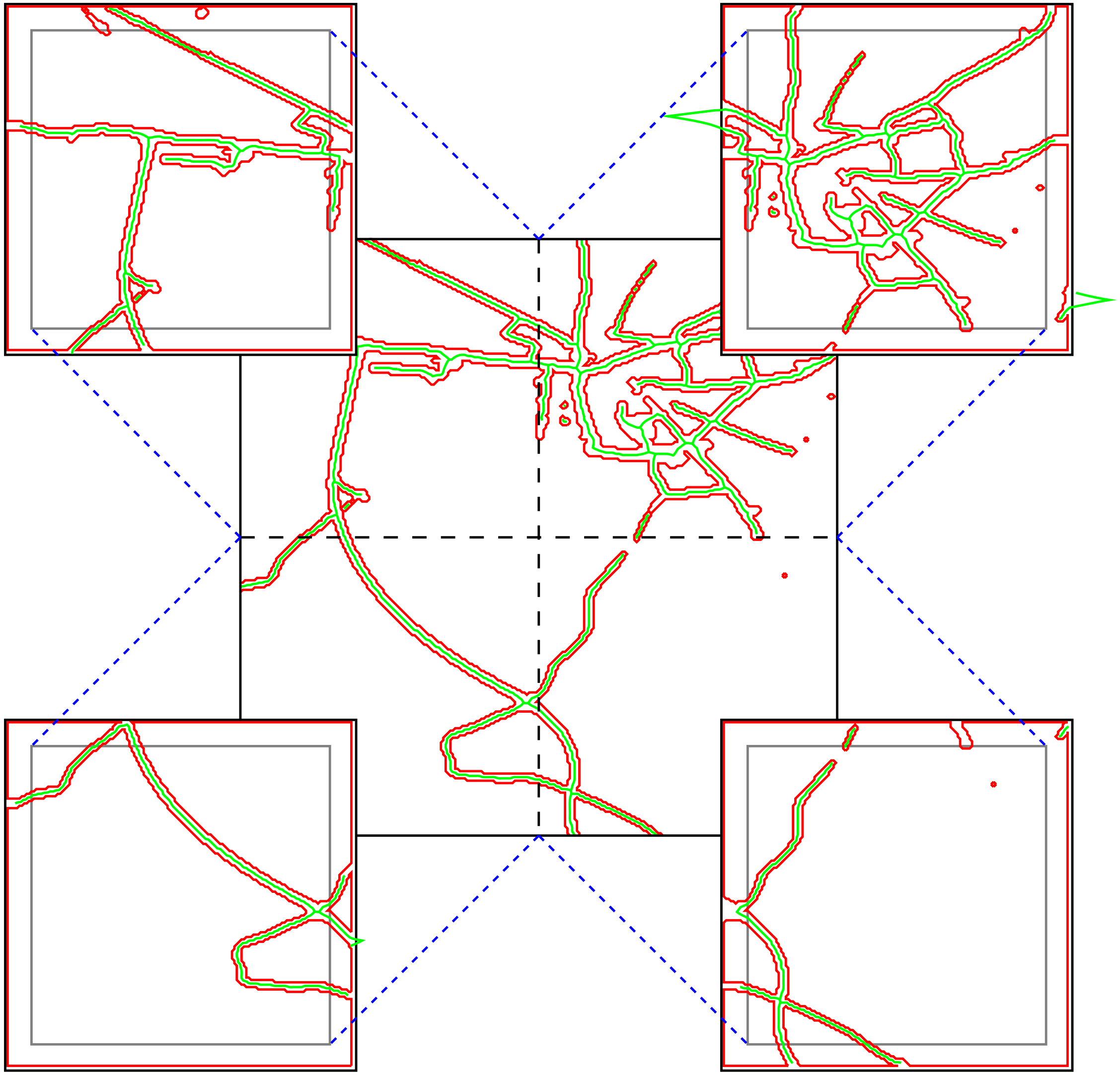
Recall

- Two questions to be addressed:
  - How can we collect vast quantities of GPS traces?
    - Coming up next…
  - How can we convert GPS traces into graphs?
    - Addressed

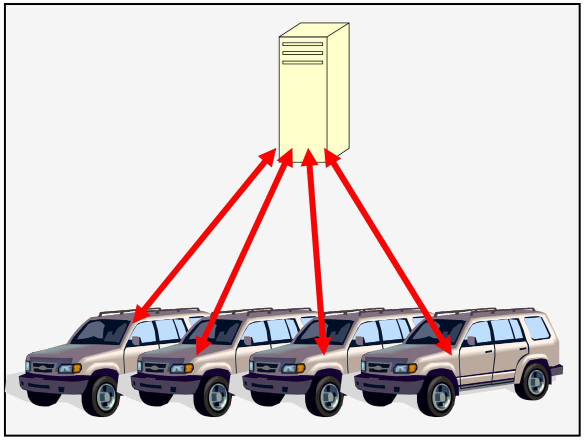
Large-scale, vehicle-centric applications

- Cars are gaining increasing communications capabilities
  - Thus, collection of GPS data will become possible
- Other similar applications involve the sharing, processing and dissemination of sensor data
  - e.g. weather data, traffic conditions, …
- Such applications are challenging to program
  - The "central server" model is not scalable
  - But most applications are parallelisable to some degree

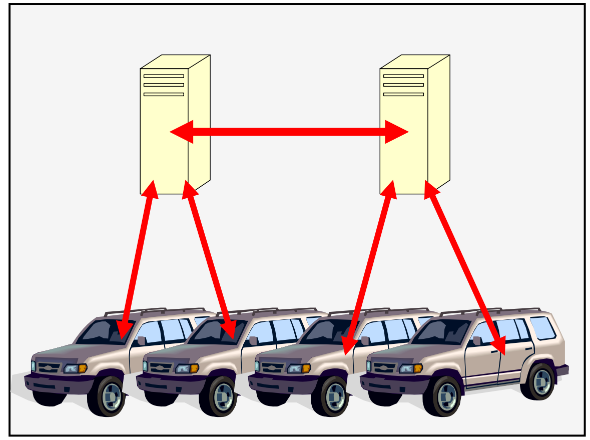
Parallelisation of map generation



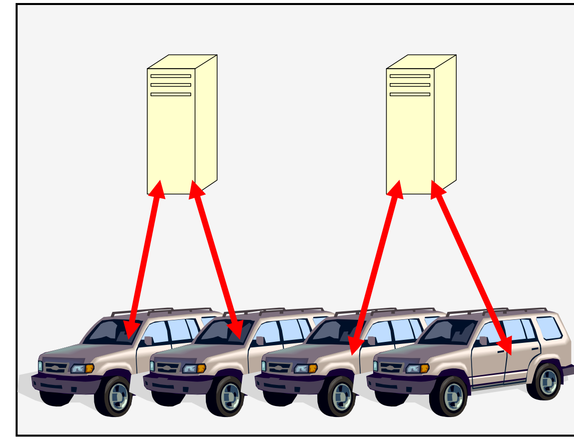
Widespread data collection from vehicles



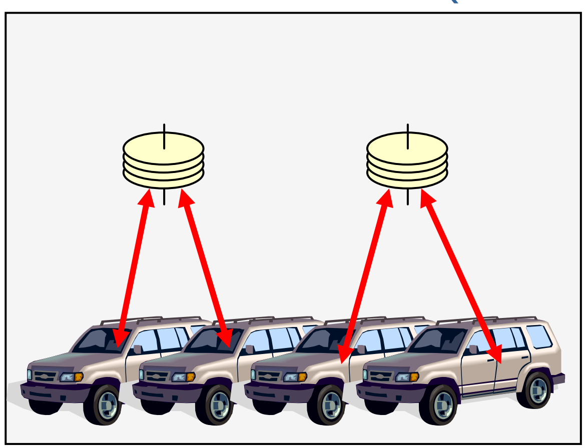
1. Centralised



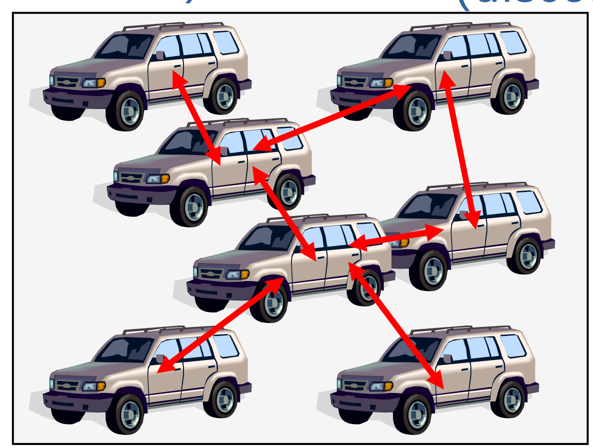
2. Regionalised (interconnected)



3. Regionalised (disconnected)



4. Regionalised (no processing)



5. Peer-to-Peer

Automated task partitioning

- Can we automatically determine the best way for the application to execute?
  - Splitting the application into tasks
  - Determining which processing nodes are best employed for each task
  - Distributing the tasks
  - Dealing with failure and changes in the network
- What metrics should we optimise against?
  - Execution time
  - Privacy
  - Quality of result

Conclusions

- Generating maps from GPS traces is tricky
  - Errors in GPS mean that the maps might contain inaccuracies
  - However, the technique is good for producing up-to-date maps
- Programming parallel applications is hard
  - Currently finalising theory of task partitioning
  - Future work will involve writing a compiler and modifying Java to support such applications

Distributed, vehicular computation for map generation

Jonathan J. Davies

Alastair R. Beresford

{jjd27,arb33}@cam.ac.uk

University of Cambridge Computer Laboratory