



# Online Deployment of Dynamic Traffic Assignment

-- architecture and run-time management

Yang Wen  
wenyang@mit.edu

# Outline

- Introduction
- DynaMIT at the Los Angeles TMC
- Interfaces, efficiency, and robustness
- Preliminary evaluation
- Conclusion and on-going research

# Introduction

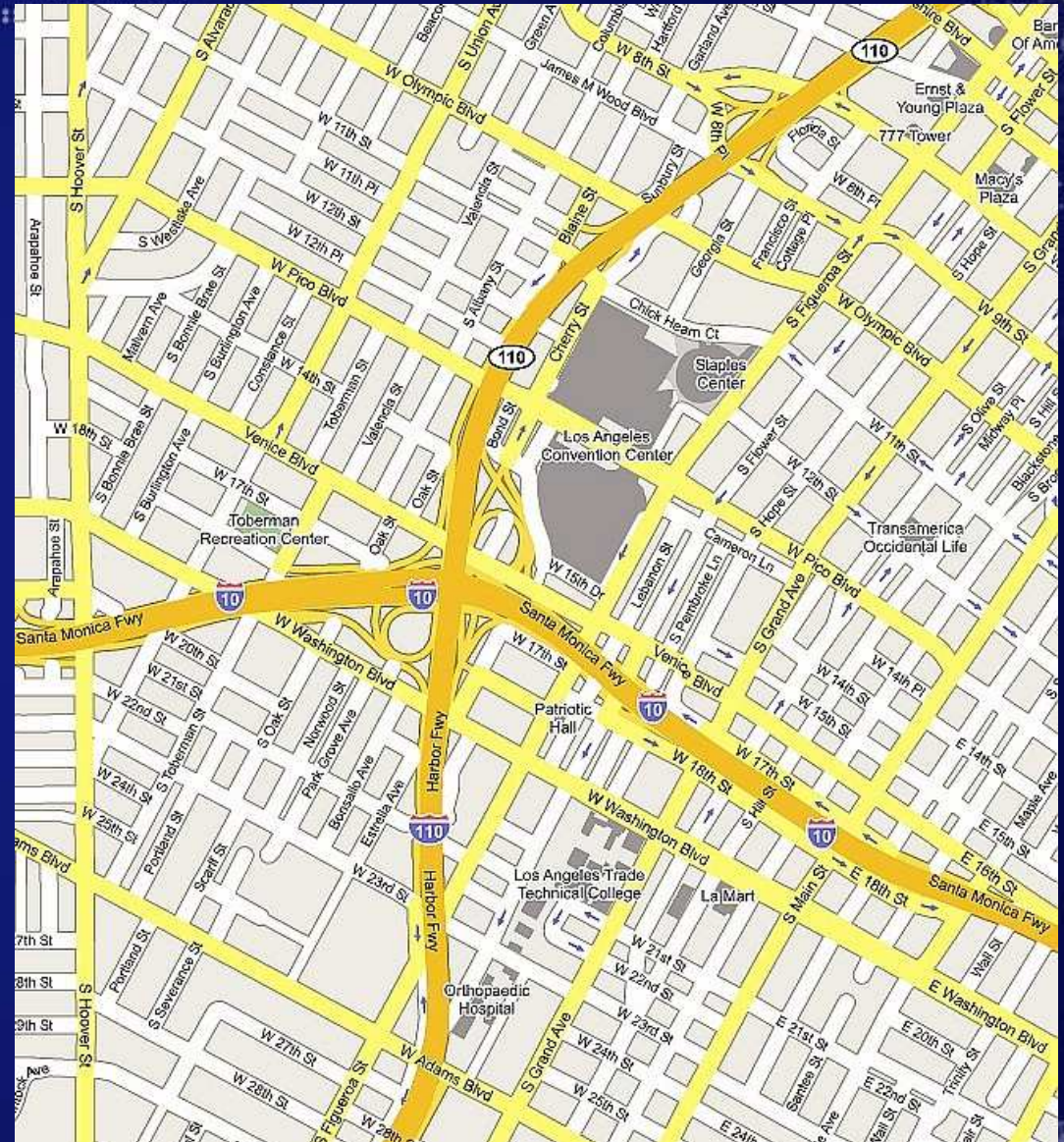
- DTA captures the many interactions between
  - complex travel demand patterns
  - individual driver decisions
  - network supply phenomena
- Estimation of time-varying OD demand
- Predict information for TMC operators or travelers
  - Travel time
  - Especially useful under non-recurrent incidents
- DynaMIT

# DynaMIT at LA

- Project background
  - Integrate with LADOT's Automated Traffic Surveillance and Control system (ATSAC)
  - Implement incident detection and prediction-based traffic management capabilities
- Role of DynaMIT
  - Determine “normal” traffic conditions for other systems
  - Predict short-term traffic conditions

# The network

- Downtown LA
  - South park area
  - Freeways
  - Arterials
  - Staple Center, etc.
- Model
  - 243 nodes
  - 606 links
  - 1129 OD pairs
  - 182 sensors

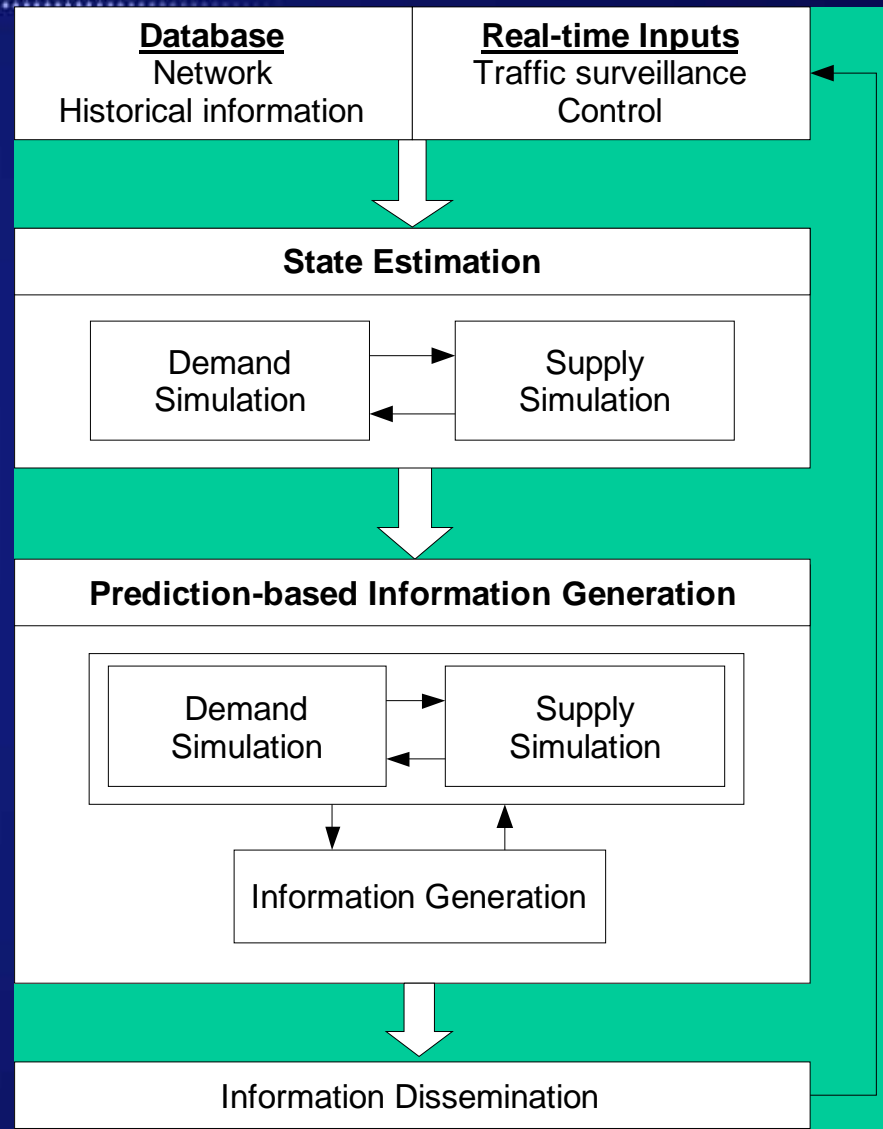


# Focus of the project

- Design, implementation and management of an on-line DynaMIT system that performs satisfactorily in real-time.

# DynaMIT Overview

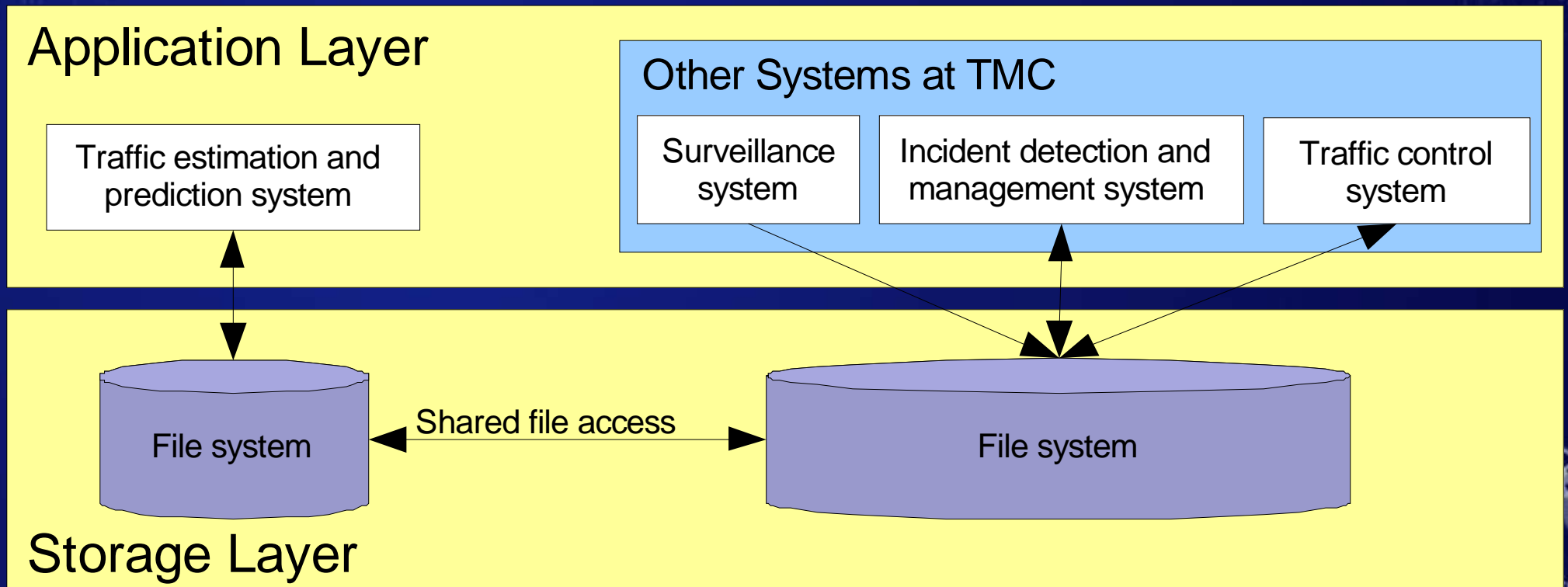
- State Estimation
  - Demand Simulator
  - Supply Simulator
- Prediction-based Guidance Generation
  - Generates predictive guidance
  - iterative framework is used to obtain *consistent* guidance



Source: Ben-Akiva et al. (2000)

# Interfaces

- File-based Input/Output



# Statistics for file sizes

| <b>Content</b>  | <b>Average size<br/>(KB / interval)</b> |
|---|---|
| Traffic counts for all 182 sensors                                  | 3                                       |
| Incidents   | 0.1/incident                            |
| Estimated and predicted flows for all sensors                       | 1.6                                     |
| Travel time guidance (606 links and a 45-minute prediction horizon) | 250                                     |

# Interface (cont.)

- Advantages for file-based I/O
  - Ease of implementation
  - Debugging convenience
  - Compact representation
- Disadvantages
  - Greater storage requirement
  - Need for concurrency management

# Computational Efficiency

- Bottle-neck identified
  - OD estimation:  $O(n^3)$
  - Supply simulation
- Efficient algorithm employed
  - Sparse constrained generalized least squares
- Implementation
  - Overcome major delays in memory sub-system
  - Improve cache performance

# Evaluation of efficiency

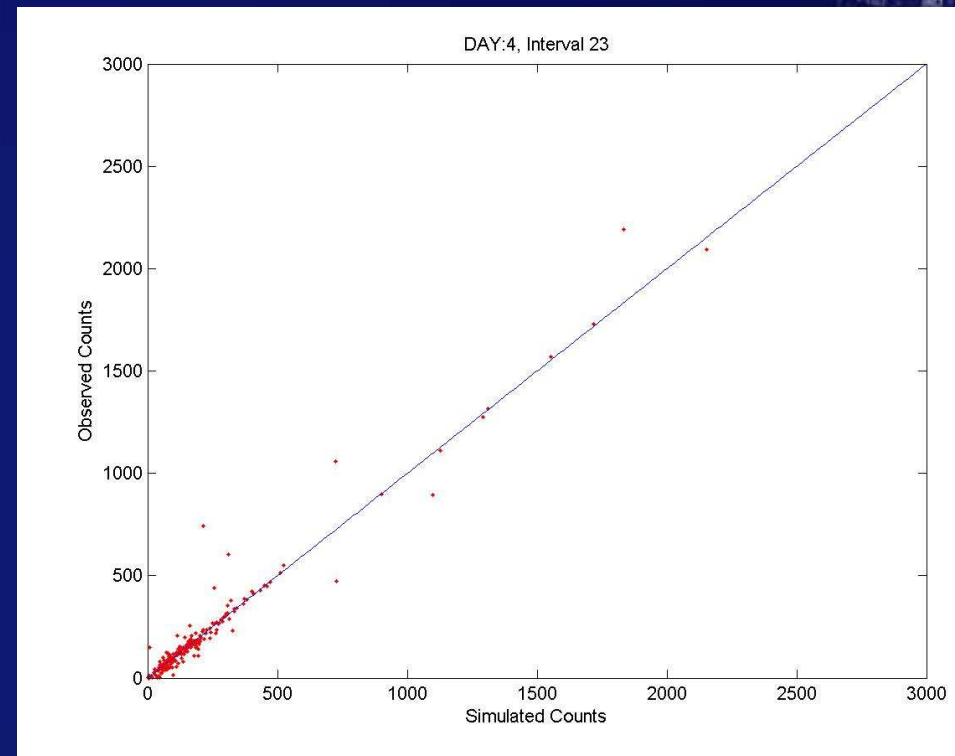
- Deployed on a PC with Redhat Linux 9
  - a 2.5 GHz Pentium IV CPU, 256 KB L2 Cache
  - 512 MB Memory
- Horizons
  - 5-minute estimation intervals
  - 30-minute prediction intervals
- Run time statistics
  - 1~2 minutes for each interval (2 iterations)
  - Additional iteration for estimation requires 10 to 15 seconds

# Robustness

- Input validation at run-time
  - Sensor may be unavailable
  - erroneous input
- Day-to-day management
  - Start DynaMIT everyday
  - Backup useful data at midnight
  - Cleanup temp files

# Preliminary evaluation of accuracy

- Off-line calibration
  - archived data from September 2004
- On-line evaluation
  - Real-time data
  - estimated counts generally stay close to the actual counts
  - predicted counts also follow the same trends



Source: Gupta, 2005

# Conclusion

- Key-points for on-line deployment of DynaMIT
  - Simple and effective interface
  - Efficient implementation and hardware configuration to run in real-time
  - Handle erroneous input gracefully
- Future works
  - Integration of on-line calibration
  - Efficient supply simulation

# Acknowledgement

- Co-workers of paper "On-line Deployment of Dynamic Traffic Assignment: Architecture and Run-time Management"
  - Ramachandran Balakrishna
  - Moshe Ben-Akiva
  - Scott Smith
- DynaMIT project in Los Angeles
  - Federal Highway Administration and the Los Angeles Department of Transportation for their financial and logistical support