PTV GROUP

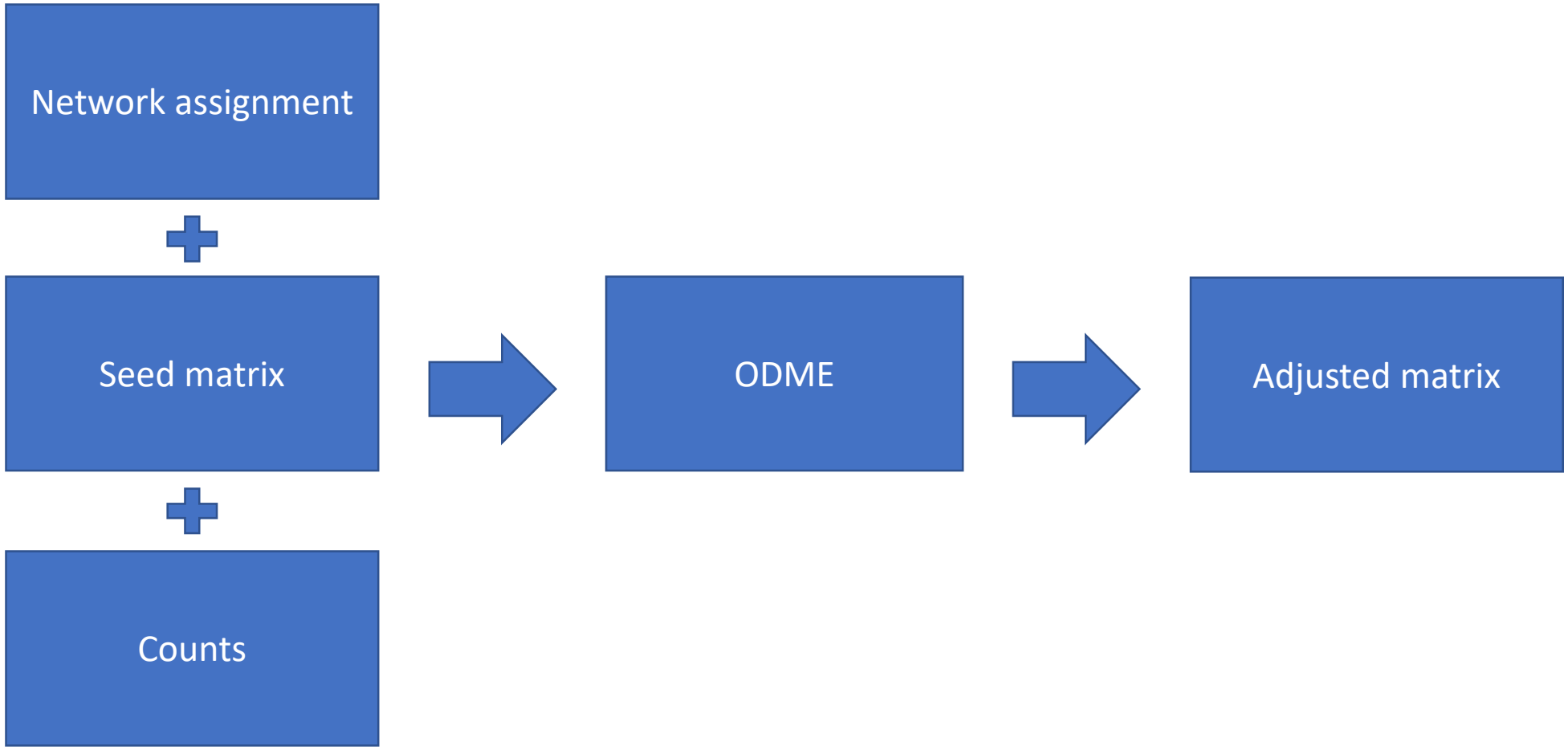
the mind of movement

TRB Innovations – 2018, Atlanta GA

# Open Source Sparse Matrix Implementation of Least Squares Matrix Estimation

Chetan Joshi, Portland OR

# Matrix Estimation Overview



# Motivation

- Started as a prototyping experiment for application on large scale networks with noisy count data and possible use in persistently updated real time models
- Exploring the least squares formulation and the gradient method for solution quality and performance
- Making the open source prototype available in public domain may help demystify the ODME black box and encourage…

"proper and responsible application of ODME" –ref: http://tfresource.org/Destination_Choice:_Data_Sources

# Formulation

Broadly, two well known formulations for ODME problem (problem is underspecified):

- Maximum entropy via ME2 etc. [Willumsen (1978)]

$$\max S(T_{ij}) = -\sum_{ij}(T_{ij}\log T_{ij} - T_{ij})$$

ST:

$$C_a - \sum_{ij} T_{ij} p_{ij}^a = 0$$

$$T_{ij} \geq 0$$

$a = link\ index$
$C_a = count\ at\ link\ a$
$T_{ij} = trips\ between\ ij$
$p_{ij}^a = contribution\ of\ od\ pair\ ij\ to\ flow\ on\ link\ a$

- Least squares [Spiess (1990)]

$$min(Z) = \frac{1}{2} \times \sum_a \left\{ \sum_{ij} T_{ij} \times p_{ij}^a - C_a \right\}^2$$

# Solution Algorithm

*Gradient descent in general…*

$$X_{k+1} = X_k - \lambda_k \nabla f|_{x_k}$$

$$\nabla f \equiv \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n}\right]^T$$

Numerous applications including Machine Learning etc…

*Where,*

$X_k$ = *initial solution or solution at iteration k*

$X_{k+1}$ = *updated solution at iteration k+1*

$\nabla f|_{x_k} = gradient\ (first\ order\ partial\ drivative)\ value\ at\ x_k$

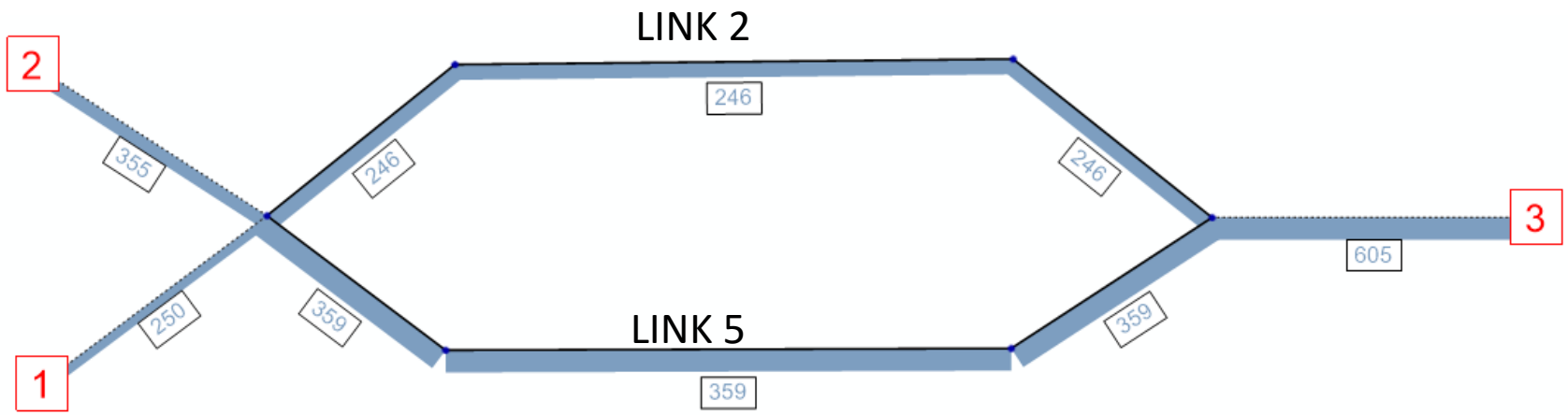
$\lambda_k = suitable\ step\ size$

# Application to ODME

OD Flows

| OD PAIR | TRIPS |
|---|---|
| 1 -> 3 | 250 |
| 2 -> 3 | 355 |

Assignment...



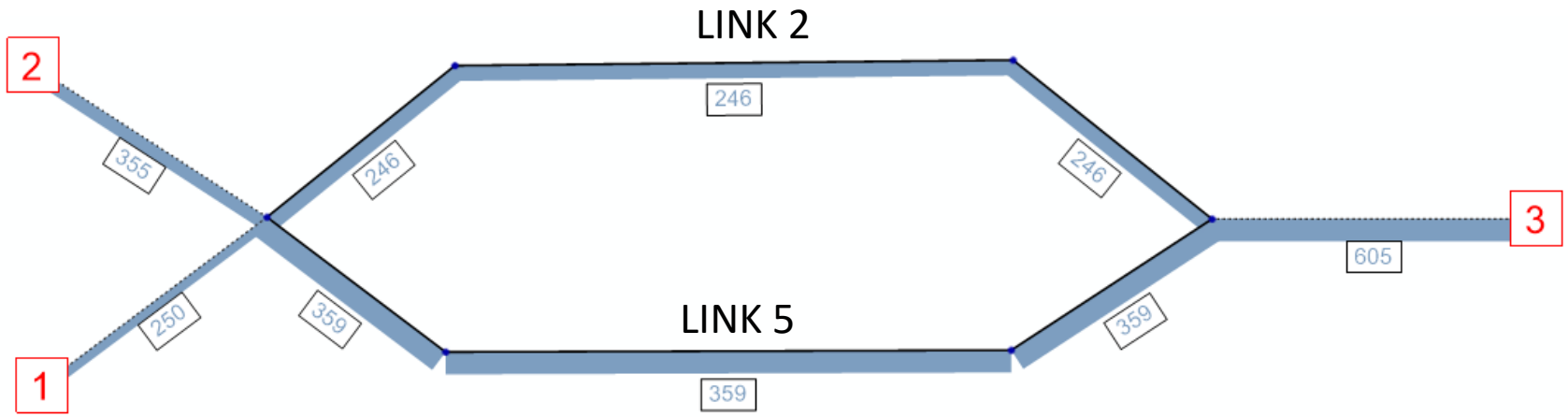

Path Flows

| OZONE | DZONE | PATH INDEX | PATHFLOW |
|---|---|---|---|
| 1 | 3 | 1 | 148 |
| 1 | 3 | 2 | 102 |
| 2 | 3 | 1 | 210 |
| 2 | 3 | 2 | 145 |

Flow Proportion Matrix

| | | OD PAIRS | |
|---|---|---|---|
| | | 1 -> 3 | 2 -> 3 |
| LINKS | 2 | 0.407 | 0.407 |
| | 5 | 0.593 | 0.593 |

# Application to ODME – Link Flow Evaluation




**Dot product**

OD Flows

| OD PAIR | 1 -> 3 | 2 -> 3 |
|---|---|---|
| TRIPS | 250 | 355 |

Flow Proportion Matrix

| | | LINKS | |
|---|---|---|---|
| | | 2 | 5 |
| OD PAIR | 1 -> 3 | 0.407 | 0.593 |
| | 2 -> 3 | 0.407 | 0.593 |

Link Flows

L2 = 250*0.407 + 355*0.407 = 246

L5 = 250*0.593 + 355*0.593 = 359

# Application to ODME – Gradient Evaluation

## Gradient ….

Flow Proportion Matrix

| | | LINKS | |
|---|---|---|---|
| | | 2 | 5 |
| OD PAIR | 1 -> 3 | 0.407 | 0.593 |
| | 2 -> 3 | 0.407 | 0.593 |

**Dot product**

Error (Va – Ca)

| LINKS | 2 | 5 |
|---|---|---|
| ERROR | e1 | e2 |

# Application to ODME – Step Length

## Evaluate Gradient Value at OD flow…

Flow Proportion Matrix

| | | LINKS | |
|---|---|---|---|
| | | 2 | 5 |
| OD PAIR | 1 -> 3 | 0.407 | 0.593 |
| | 2 -> 3 | 0.407 | 0.593 |

**Dot product**

OD Flows

| OD FLOW | 1->3 | 2->3 |
|---|---|---|
| ERROR | 250 | 355 |

## Final Step Length by Projection of Gradient…

{ref: Spiess (1990)}

# Matrix Sparsity

Matrices for implementation of Gradient Method can be huge, including the prior / seed matrix in the objective function can make it even larger…

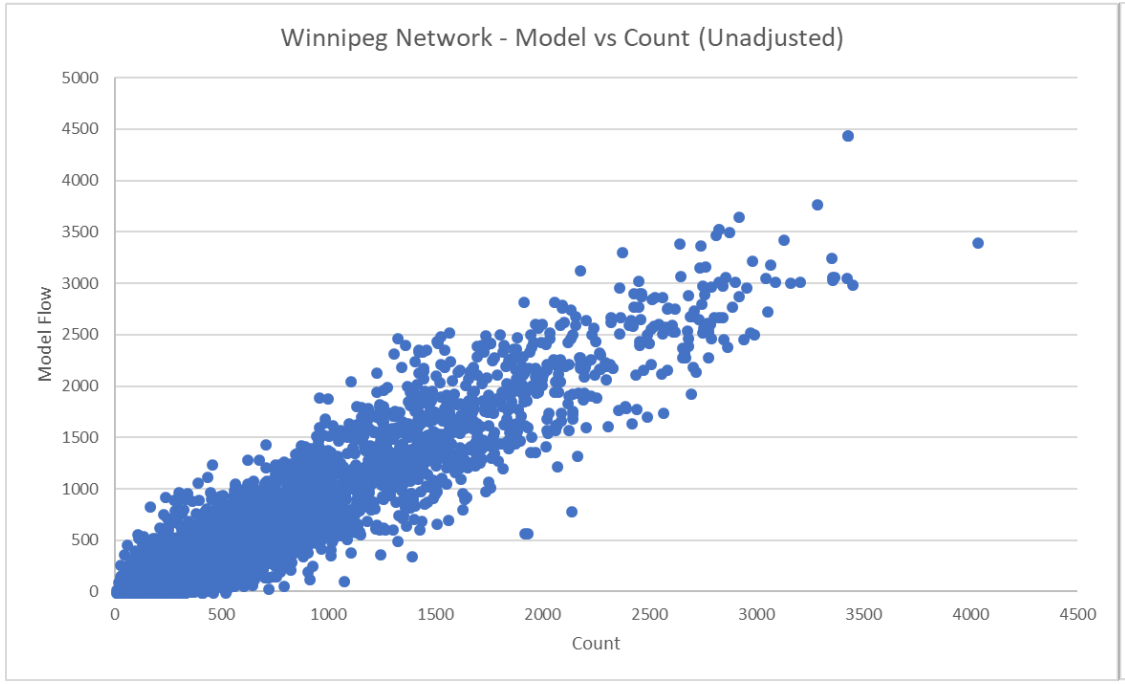
Fortunately, the Flow Matrix is very s…

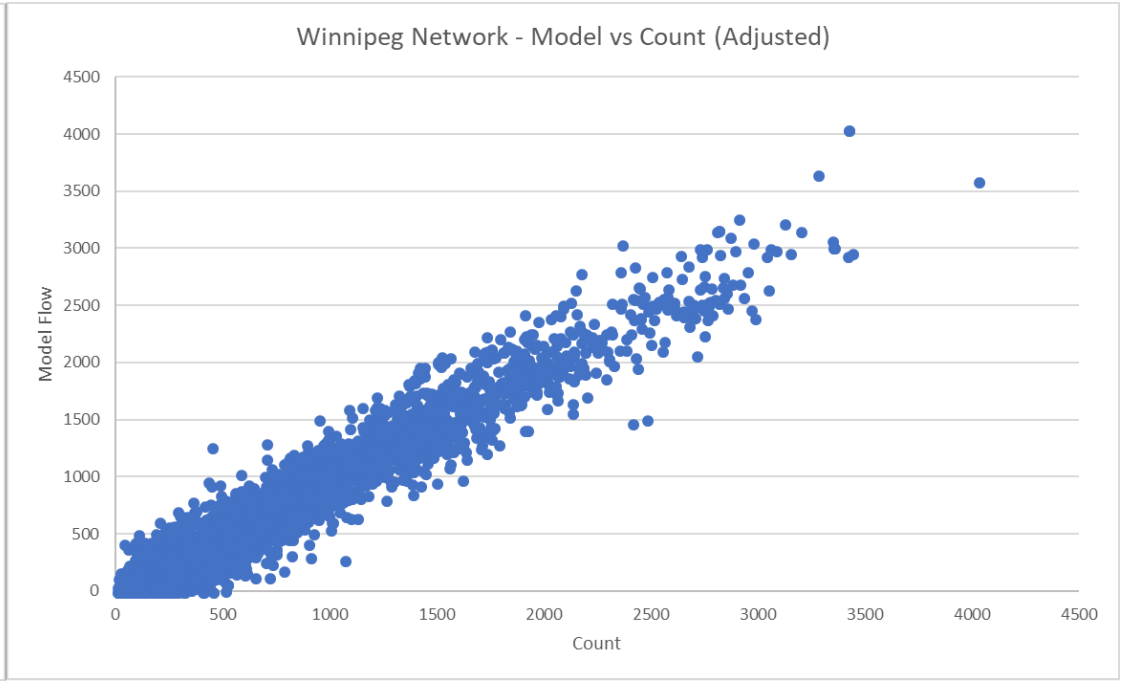
**Sparse dot product with python SciPy (SciPy.org)**

| | | | | |
|---|---|---|---|---|
| | | | [illegible],170,329 | 0.043% |
| Winnipeg | 1136 | 4424 | 1,594,355,889,456 | 0.003% |
| Tokyo | 3406 | 3380 | 3,118,216,303,004 | 0.001% |

# Results from Real World

| | | | Seed | | Adjusted | |
|---|---|---|---|---|---|---|
| **Network** | **Zones** | **Counts** | **R2** | **%RMSE** | **R2** | **%RMSE** |
| Winnipeg | 1136 | 4493 | 0.88 | 38% | 0.94 | 25% |
| Tokyo | 3406 | 3380 | 0.77 | 35% | 0.89 | 24% |



# Results – Quality and Run Time

## Entropy…

| | Entropy | |
|---|---|---|
| **Formulation** | **Newport** | **Winnipeg** |
| Entropy max w OD constraint (Bell) | -12974.58 | -355398.47 |
| Least Squares w OD constraint | -6989.18 | -59786.55 |

## Run Time…

| | Run Time | |
|---|---|---|
| **Formulation and Solution Method** | **Winnipeg** | **Tokyo** |
| Entropy max w OD constraint (Bell) - Newton Raphson | 76min 24s | 52min 11s |
| Least Squares w OD constraint - Gradient Descent | 1min 38s | 3min |

# Finally

- Where is the code:

https://github.com/joshchea/python-tdm/blob/master/scripts/MatEstimateGradient.py

- Credits:

Klaus Noekel – adult supervision

Sergio Grosso – providing test networks and feedback

- References:

[1] Spiess, H., A Gradient Approach for the O-D Matrix Adjustment Problem, Publication 693, CRT, University of Montreal, 1990

[2] Willumsen, L. G., Simplified transport models based on traffic counts, Transportation 10 (3):257-278, 1981

[3] Aerde, Michel, Hesham Rakha, and Harinarayan Paramahamsan, Estimation of Origin-Destination Matrices: Relationship Between Practical and Theoretical Considerations, Transportation Research Record: Journal of the Transportation Research Board, 2003