Automatic Calibration of Large Traffic Models

Felix Meziere [1]   Gabriel Gomes [2]

California PATH Program

June 26^{st}, 2015

[1] Ecole polytechnique [felix.meziere@polytechnique.edu]
[2] University of California Berkeley [gomes@path.berkeley.edu]

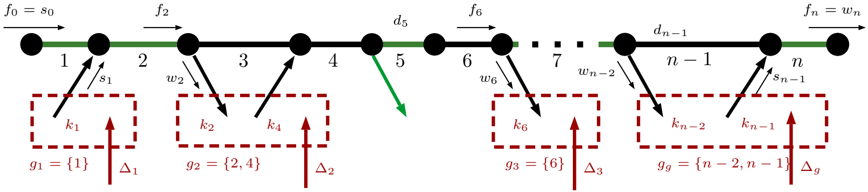
Introduction

and motivation of this work

- Calibrate large traffic models input:
    - Constants of the scenario
    - Source and sink flows depending on day of week & moment of year
- for the output to match:
    - Congestion
    - TVM and TVH
    - Local flows and more...
- Monitored freeway used in this project : 210 East

Setup

Freeway model and notation



$L = [\![1, n]\!]$	Mainline link indexes
$S \subset L$	Source link indexes (index of preceding mainline link)
$W \subset L$	Sink (well) link indexes (index of preceding mainline link)
$M \subset L$	Monitored mainline links indexes
$K = S \cup W \backslash M$	Non-Monitored ramps indexes : the "knob ramps"
$G = (g_i)_{i \in [\![1, g,]\!]}$	Knob group indexes
$k_p = (k_{i_1}^{(p)}, k_{i_2}^{(p)} \ldots, k_{i_K}^{(p)})$	Knobs value vector at iteration p
$\sigma = (\sigma_{i_1}, \sigma_{i_2} \ldots, \sigma_{i_K})$	± 1 : on- or off-ramp indicator for knobs
$(f_i^{(p)})_{i \in L}$	Mainline out flows after evaluation p
$(s_i^{(p)})_{i \in S}$	Source out flows after evaluation p
$(w_i^{(p)})_{i \in W}$	Sink out flows after evaluation p

Setup

Context : PeMS data for 210E

- Average of 5 Tuesdays in fall 2014
- Flow, density and speed on 74 links every 5 minutes for 24 hours
- Sensors (having deleted partial or too biased):
  - 33/135 Mainline links are monitored
  - 26/28 On-ramps are monitored
  - 15/25 Off-ramps are monitored
- Assumptions : uncertainty from sensor sensibility and bias
  - Daily local flow uncertainty on mainline sensors : $I^{local} \approx 10\%.[Average\ of\ mainline\ flow]$
  - Daily global uncertainty (on measures summed over the whole freeway): $I^{global} \approx 5\%.[Global\ measure\ value]$

Setup
BeATS

Macroscopic freeway simulator

- Input:
  - Topography of the freeway
  - Fundamental Diagrams
  - 5-min demands for sources and sinks ("split-ratios as output mode")
    - As real data if the source or sink is monitored
    - As a guessed template if not → need for 12 knobs
- Output: flow, densities and speeds on every link
- Run time: 5-8 seconds

Choice of the algorithm

Requirements

- Continuous search space: here, $\mathcal{S}$ =12-D hypercube
- Function to minimize : mix of correlated and uncorrelated functions with non-differentiable congestion effects

$\rightarrow$ Continuous black-box imputation problem

Well-suited algorithms : evolutionnary or simulated annealing among others

Choice of the algorithm

Covariance Matrix Adaptation - Evolution Strategy (CMA-ES)

- Most powerful evolutionary algorithms for single-objective real-valued optimization (very used)
- "Designed for difficult non-linear non-convex black-box optimisation problems in continuous domain"
- "Typically applied to unconstrained or bounded constraint optimization problems, and search space dimensions between three and a hundred"
- Does not presume existence of approximate gradients : feasible on our non-smooth problem
- *Adaptive* algorithm : almost no parameter tuning $\rightarrow$ suitable to be used on several different freeways and days.
- Time does not matter for this initial study : not the fastest ES but excellent solution quality
- Principle : 12*12 covariance matrix adapts the stochastic sampling direction and step size on-the-go

Knobs

Obvious boundaries

Naive boundaries:

Let $(t_i(t))_{i \in K}$ the templates

$$i \in K,\ \ \mu_i = \frac{[Capacity\ \ of\ \ ramp\ \ associated\ \ to\ \ k_i]}{\max_t t_i(t)}$$

$$[k^{naMin}, k^{naMax}] = [0, \mu] = \begin{bmatrix} 0, \mu_1 \\ 0, \mu_2 \\ \vdots \\ 0, \mu_\kappa \end{bmatrix}$$

Knobs

Perfect values and uncertainty

- Monitored knob groups show that each "knob ramp" is very closely monitored by the mainline sensors:
  - 8 knob groups are single-knob
  - 2 knob groups are composed by 2 knobs

  $\rightarrow$ 8 knobs have a "perfect value" in order to match Δ_j:

  $$\{k_i^*, i \in \{j, card(g_j = 1)\}\}$$

- Keeping the perfect values is impossible:
  - Inexactitude of templates, FDS, simulator
    $\rightarrow$ Let us define α, β, under- and over-evaluation tolerance coefficients for the knobs.
  - Uncertainty on the mainline sensors: I^{local}

Knobs

Refined knob boundaries on single-knob groups

For the single-knob groups, we can refine the knob boundaries using these two uncertainties as a limit:

$$k_i^{min} = max(\{min(\{\alpha.k_i^*; k_i^* - \frac{I^{local}}{\sum_t t_i(t)}\}); 0\})$$

$$k_i^{max} = min(\{max(\{\beta.k_i^*; k_i^* + \frac{I^{local}}{\sum_t t_i(t))}\}); \mu_i\})$$

Knobs

Refined boundaries on multiple-knob groups

- For the multiple-knob groups, use the naive knob boundaries and a "repair and penalize" policy :
  1. Repair the unfeasible input before evaluating the error function
  2. Penalize proportionally to the distance between original input and repaired input

→ This avoids the algorithm from exploring unfeasible solutions and brings it closer to the feasible space

Knobs

Refined boundaries on multiple-knob groups

Repairing : Here, we project the knobs of the group on a space delimited by the two "tolerance hyperplans" for Δ_k:

Let $\overline{k_p}$ the knobs vector to be evaluated at iteration p before projection
Let $J = \{j, card(g_j > 1)\}$ and $\forall i \in K, T_i = \sum_t t_i(t)$

minimize $\left\| k_p - \overline{k_p} \right\|_2$

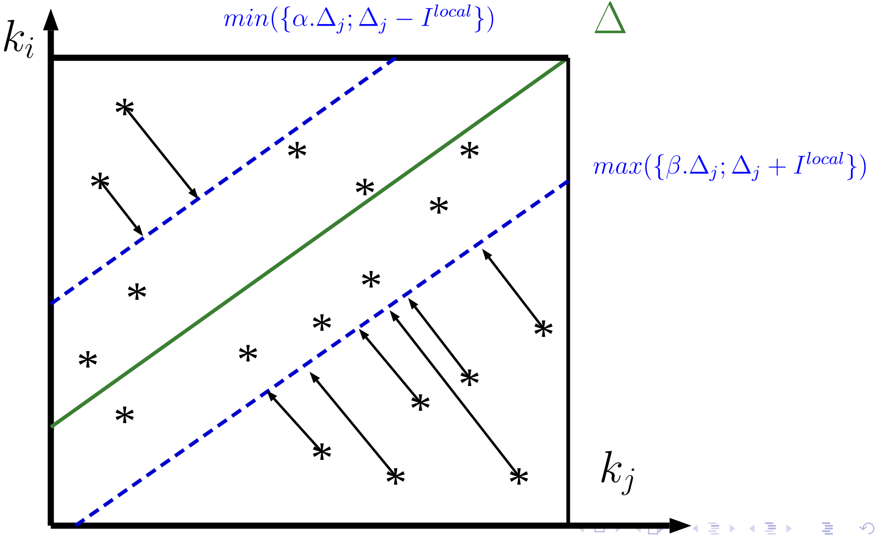
s.t.

$\forall j \in J \quad min(\{\alpha.\Delta_j; \Delta_j - I^{local}\}) < \sum_{i \in g_j} \sigma_i.k_i.T_i < max(\{\beta.\Delta_j; \Delta_j + I^{local}\})$

$k \in [k^{naMin}, k^{naMax}]$

Knobs

Refined boundaries on multiple-knob groups



Fitness function

Generalities

- 4 Performance calculators (PC): global TVM, global TVH, Knobs Distance(KD) and Congestion Pattern (CP)

$$PC : \left| \begin{array}{ccc} \mathcal{S} & \longrightarrow & [0, 100] \\ k_p & \longmapsto & PC(k_p) \end{array} \right.$$

- Normalization: $\tau_p(PC)$ the relative difference for each PC is weighted and summed to get the fitness function f :

$$f(p) = \lambda_1.CP(p) + \lambda_2.TVH(p) + \lambda_3.KD(p) + \lambda_4.TVM(p)$$

- Global uncertainty : $\tau_p(PC) < I^{global} \Rightarrow \tau_p(PC) = 0\%$
- Each PC can be evaluated with any norm (here, only L_1 and L_2 used)

Fitness function

Knobs Distance (KD)

- Penalization for having to project the multiple-knob groups

$\rightarrow$ Forces the algorithm to search close to the feasible space and probably enter in it

- Error computation from BeATS input:

$$KD(k_p) = \left\| k_p - \overline{k_p} \right\|_2$$

- Relative difference : $\tau_p(KD) = 100 * \frac{KD(k_p)}{\|k^{max} - k^{min}\|_2}$

Fitness function

TVM

- Value computation from BeATS output and PeMS:

$$TVM(p) = \sum_{l \in M} \sum_{t} f_l^{(p)}(t) * length(l)$$

- $\forall i \in L, f_i(midnight) \approx 0 \Rightarrow$ TVM can be computed before BeATS:
  Let $TVM^{ref} = TVM((1, ..., 1))$ outputed by BeATS

$$TVM^{a\ priori}(k_p) = TVM^{ref} + \sum_{i \in K} \left[\sigma_i . T_i . k_i . \sum_{j \in M, j > i} length(j) \right]$$

$\rightarrow$ Same project & penalize as KD, within I^{global} :

minimize $\quad \left\| k_p - \underline{k_p} \right\|_2$

s.t.

$TVM^{PeMS}.(1 - I^{global}) < TVM^{a\ priori}(k_p) < TVM^{PeMS}.(1 + I^{global})$

$k_p \in [k^{naMin}, k^{naMax}]$

Fitness function

TVM

- Relative difference : $\tau_p(TVM) = 100 * \frac{TVM^{BeATS}(p) - TVM^{PeMS}(p)}{TVM^{PeMS}(p)}$

Fitness function

TVH

- Value computation from BeATS output and PeMS :

$$TVH(p) = \sum_{l \in M} \sum_{t \in [0, \frac{24h}{dt}]} d_l^{(p)}(t) * \frac{dt}{1\ hour}$$

- Relative difference : $\tau_p(TVH) = 100 * \frac{TVH^{BeATS}(p) - TVH^{PeMS}(p)}{TVH^{PeMS}(p)}$

Fitness function

Congestion Pattern

- Principle : match the congested links and times into a box estimated from PeMS contour plot
- Error computation from BeATS output:
  - A congestion treshold is defined for each mainline link :

$$\frac{Link\ capacity}{Link\ freeflow\ speed} + \delta$$

  - The number of pixels of the contour plot that are in the wrong state is the error:

$$CP(p) = \sum_{t \in \frac{24h}{dt}} \sum_{l \in L} \mathbb{1}_{wrong\ congestion\ state\ links}$$

- Relative difference : $\tau_p(CP) = 100 * \frac{CP(p)}{Total\ area\ of\ the\ boxes}$

Effects of the parameters

- Effect of initial standard deviation
- Effect of changing the boundaries (uncertainties)
- Effect of changing the weights

Results

Issues and things to improve

- Issues:
  - Templates/FDS should be modified
  - Constraints handling could be better

  $\rightarrow$ several knobs end up on their boundaries
  - Uncertainties are symmetric $\rightarrow$ do not take bias into account
  - Solution not even close to unique : find new constraints
- Improvements:
  - Trying to find a "pareto optimum" for each day instead of average fitting
  - Custom boundaries for each knob (depending on each situation/sensor bias)
  - Tuning also monitored ramps within $\pm\epsilon$ %
  - MO-CMAES