Addressing Parameter Uncertainty in SD Models with Fit-to-History and Monte-Carlo Sensitivity Methods

Wayne Wakeland, Portland State University
Jack Homer, Homer Consulting

Link to video of the talk made beforehand

Background: assessing parameter sensitivity

- Conventional MC-based sensitivity testing of uncertain parameters
  - Has been done for decades[1,2]
  - The key idea is to vary parameters systematically over properly established ranges
- More recently Markov Chain Monte Carlo (MCMC) has been preferred[3,4,5]
  - An optimal strategy for sampling from very large parameter spaces
  - And, more efficient than "brute force" MC
- But we had not seen a practical procedure for what we typically need: (1) to narrow MC or MCMC parameter spaces based on *fit-to-history* (data) constraints; and (2) to systematically test the narrowed space for *policy sensitivity*.

[1] Ford 1990, [2]Sterman 2000, [3]Osgood 2015, [4]Fiddaman & Yeager 2015, [5]Sterman et al. 2018

Aim of the paper

- How should useful parameter sets be identified?
  - Why not let model parameters run free over their plausible ranges and report all the results?
  - Because randomly selected sets of parameter values are very unlikely to produce results that resemble the historical data
- But isn't the answer already known? Isn't MCMC the method of choice? Why not just use it?
- Indeed. This is where we started
  - But the samples identified were puzzling, not even close to Gaussian; rather they looked rather like spruce trees.
  - We experimented with the settings that control the MCMC algorithm; we consulted with MCMC/Vensim experts; but to no avail.
  - Either MCMC itself is not optimal for our application (12+ data variables, not just 1 or 2); or, perhaps Vensim/MCMC is somehow sensitive to user settings.
- Our paper offers an overall approach designed to work with MCMC or MC, and demonstrates the latter

The approach

1. Specify parameter uncertainty ranges, define error metrics (MAEMs) for data variables, and find optimal parameter set (OPS)

2. Monte Carlo Testing:
   a. Markov Chain (MCMC), or
   b. Very large (e.g. 1e7 runs) standard MC

3. Qualifying Parameter Sets (QPS):
   a. MCMC gives "statistically valid sample", or
   b. From MC, select QPS based on MAEM criteria

4. Save QPS as text file and produce histograms of each uncertain parameter

5. Do base run for each QPS and produce box & whisker time graphs vs. data

6. Do alternative (non-base) runs for each QPS and calculate deltas vs. base run

7. Produce summary graphs and tables of run results and deltas (histograms, means & uncertainty intervals)

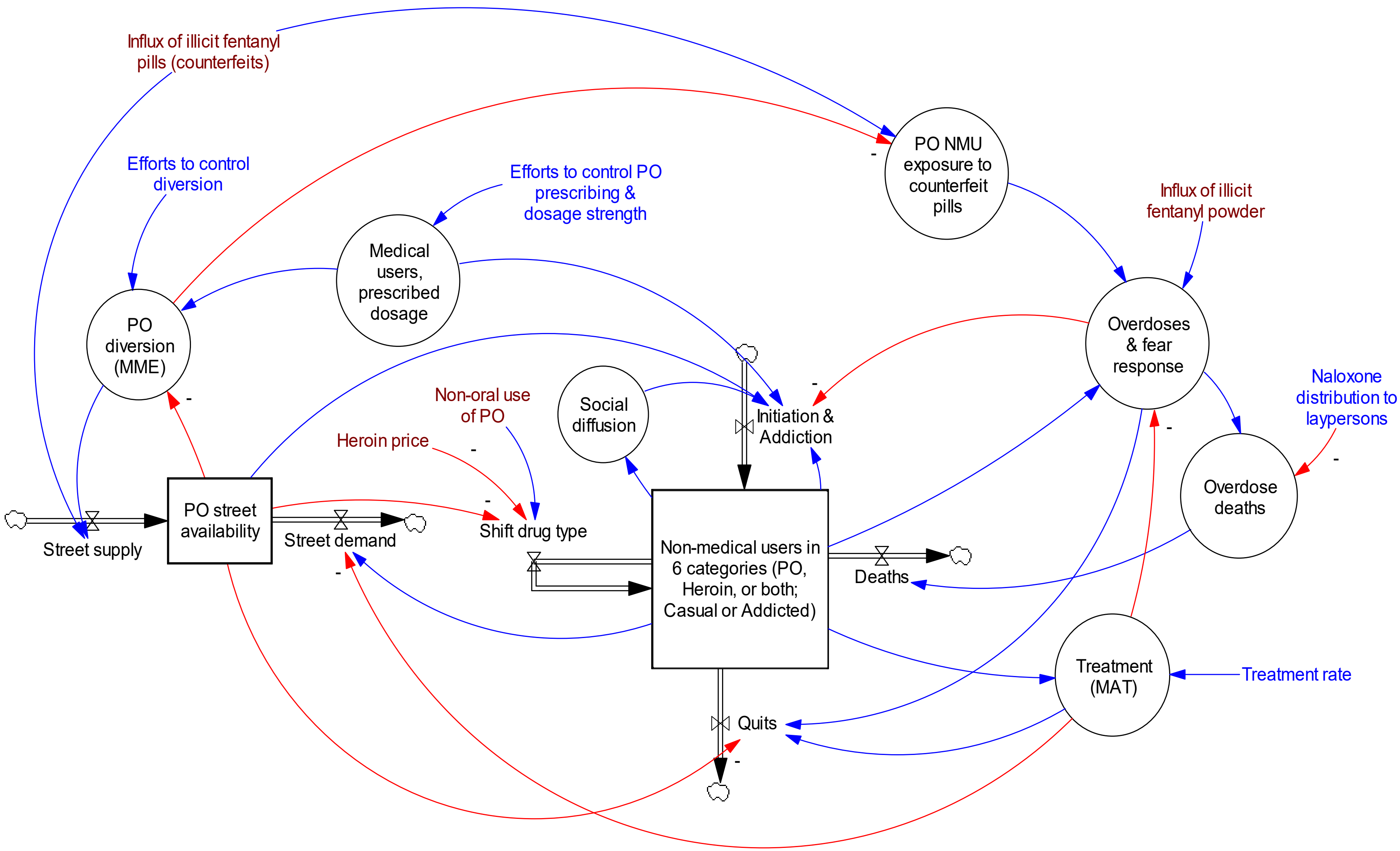
[Link to detailed version]

Illustrative results

- We applied the MC-version of the method to an SD model of significant complexity[1]
  - An SD model of the opioid epidemic in the United States from 1990 to 2030
  - Designed to replicate the history and current status of the opioid "crisis"
  - And then to evaluate policy options for abatement

1. Homer J, Wakeland W. A Dynamic Model of the Opioid Drug Epidemic with Implications for Policy. *Am J Drug Alcohol Abuse* June 2020.

High level model diagram (for context)



1. Add model fit statistics to the model (using the SSTATS[1] macro)

Time (Year)	2020
MAEM PO Abusers	0.0635467
MAEM Addicted PO Abusers	0.0904467
MAEM Addicted frac PO Abusers	0.0565265
MAEM PO abuse initiates	0.108133
MAEM H users	0.121204
MAEM Addicted H users	0.0900014
MAEM Addicted frac H users	0.0569164
MAEM H initiates	0.18316
MAEM frac H users also PO	0.138318
MAEM frac H initiates also PO	0.1005
MAEM street price PO	0.179008
MAEM OD deaths from PO	0.0509725
MAEM OD deaths from illicits	0.0392809
MAEM OD deaths total	0.0363724
Simple avg of all MAEM	0.0938847
Weighted avg of all MAEM	0.0892688

Time (Year)	2020
R2 PO abusers	0.920513
R2 Addicted PO abusers	0.792607
R2 Addicted frac PO abusers	0.797944
R2 PO abuse initiates	0.822349
R2 H users	0.865612
R2 Addicted H users	0.896018
R2 Addicted frac H users	0.293248
R2 H initiates	0.292869
R2 frac H users also PO	0.691761
R2 frac H initiates also PO	0.0539205
R2 street price PO	0.69684
R2 OD deaths from PO	0.956034
R2 OD deaths from illicits	0.998476
R2 OD deaths total	0.997179

[1] Sterman 2000, pg. 875; the code, written by Tom Fiddaman, is available in the appendix

2. Use optimization to help find parameter values that yield best fit

- Balance of modeler judgment & optimization
  - Powell optimization provided with Vensim Pro
- Must select which input parameters to allow to vary, the min/max values & distribution for each
- And specify the objective function
  - E.g. to minimize the mean error (MAEMs) in the model-calculated outcome trajectories vs. data
  - And weights for each term of the objective function
    - Each term is the fitness error for an outcome trajectory
    - Each weight is related to the std. deviation in that trajectory, adjusted possibly for missing data

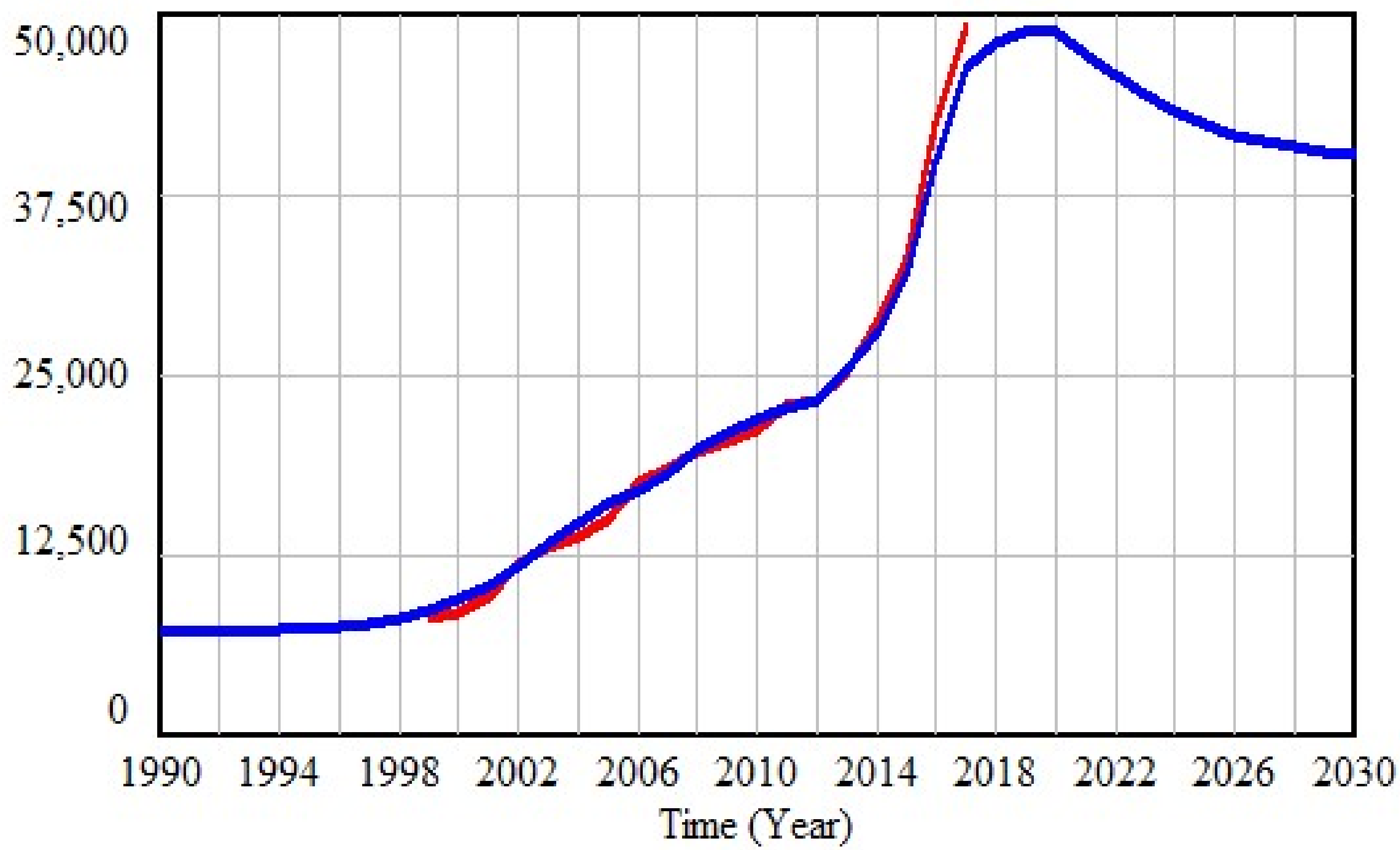
Parameter documentation, incl. min/max

Parameter	units	Value	Sources	min value	max value
Addicted frac of H users initial	Fract	0.65	Optimized; our NSDUH analysis shows 60.8% 2000, 61.1% 2005.	0.6	0.7
Addicted frac of PONHA initial	Fract	0.123	Optimized; our NSDUH analysis shows 11.4% 2000, 14.2% 2005.	0.1	0.15
Addicted H user OD death rate initial	1/year	0.010	Optimized	0.005	0.015
Addicted H user quit rate initial	1/year	0.138	Optimized	0.07	0.21
Addicted opioid abuser misc death rate	1/year	0.0045	Ray et al 2016 gives mortality hazard ratio of 1.94 vs general popn for "high dose users" (>60 mg ME). Multiply by general popn: average of NVSR death rates for [age 25-34, 35-44, 45-54] = .0023 for 2000-2010 x 1.94 = .0045.		
Addicted PONHA move to heroin rate initial	1/year	0.021	Optimized	0.01	0.03
Addicted PONHA OD death rate initial	1/year	0.0059	Optimized	0.004	0.007
Addicted PONHA quit rate initial	1/year	0.149	Optimized	0.08	0.22

3.Model Fitness (w/o uncertainty)

- To establish how well model with the "optimal" parameter values capture the dynamics of the target system

Total Opioid OD Deaths, Model vs. RBP



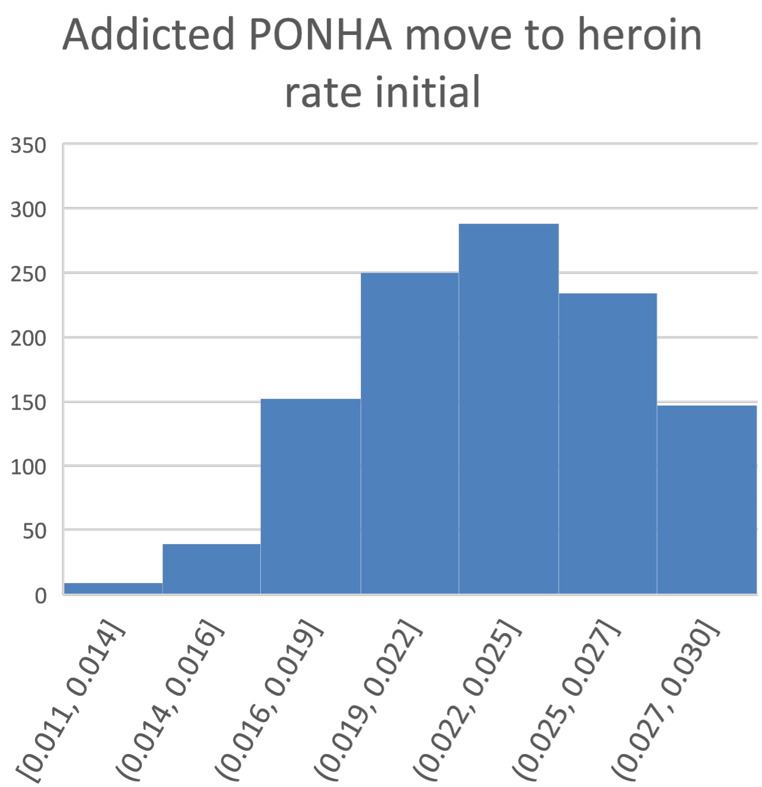
4. Make **large** Monte Carlo run (millions)

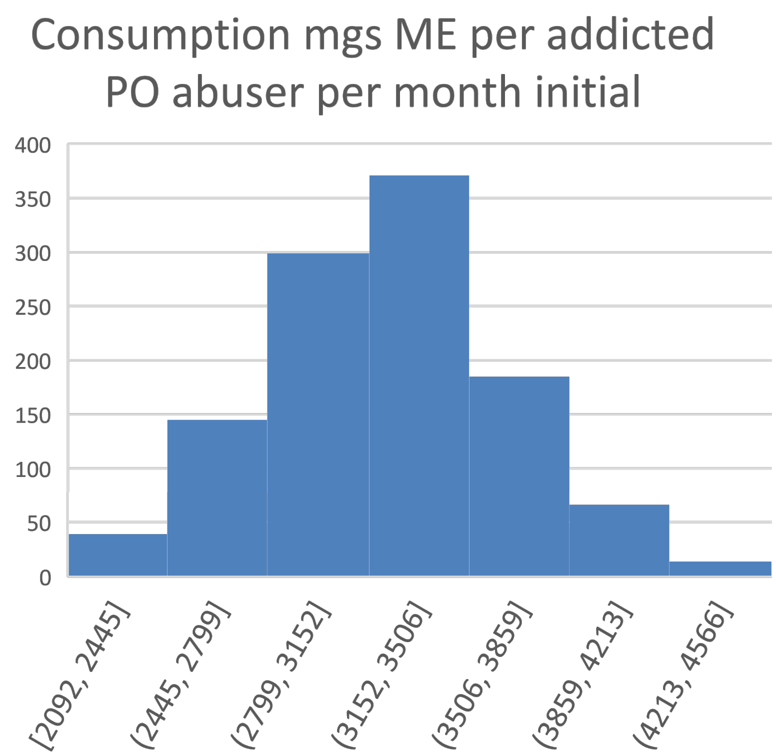
- Used Vensim sensitivity analysis feature in our case
- Could be Unif distributions between appropr. min and max
  - Could be the same min/max used during calibration (or smaller)
  - Could instead use a Triangular dist. w/optimum value as mode
    - Would increase samples near the optima
    - This option was chosen for our illustration
- Must specify a list (.lst file) of variables to be saved for ea. run
  - Sensitivity runs automatically saves the varied parameters
  - It is also useful to know the maxMAEM and avgMAEM
    - Used to select "qualified" runs
    - Since we don't need to know the time trajectories for this step, we set the model SAVEPER to 40 to keep the output file modest in size
- The weighted average MAEM was below .11 for 300-600 runs in each batch of 1M runs
  - These rows were kept. The file was then sorted by max MAEM
  - Yielding 100-130 runs with max MAEM < .20; these were kept
  - Made additional runs of 1M until ~1000 qualifying runs were found

Excel file illustrating MC results

Simulation #	Addicted frac of H users initial	Addicted frac of PONHA initial	Addicted H user OD death rate initial	…	maxofMAEMs	Simple avg of all MAEM	Weighted avg of all MAEM
681526	0.630252	0.126897	0.012121		0.199446	0.100208	0.095847
376905	0.691254	0.118593	0.012554		0.197496	0.101902	0.09691
131761	0.645974	0.118036	0.009821		0.19669	0.105488	0.097978
67350	0.684089	0.117152	0.007821		0.171332	0.101261	0.098184
726864	0.650072	0.124617	0.010844		0.183783	0.101797	0.098296
736791	0.653768	0.123594	0.010931		0.190377	0.115002	0.109998
358518	0.688724	0.122379	0.009984		0.184925	0.114706	0.109999
MIN of sims	0.6012	0.1003	0.005936		0.1612	0.1002	0.0958
MAX of sims	0.6998	0.1488	0.014533		0.2000	0.1191	0.1100
MIN allowed	0.6	0.1	0.005				
MAX allowed	0.7	0.15	0.015				
Optimized value	0.65	0.123	0.010		0.1795	0.0994	0.0935
Sample Mean	0.6487	0.1247	0.010454				
Sample std. dev.	0.0204	0.0100	0.001473				

Check distributions in the qualified parameter set for each input parameter (QPS)

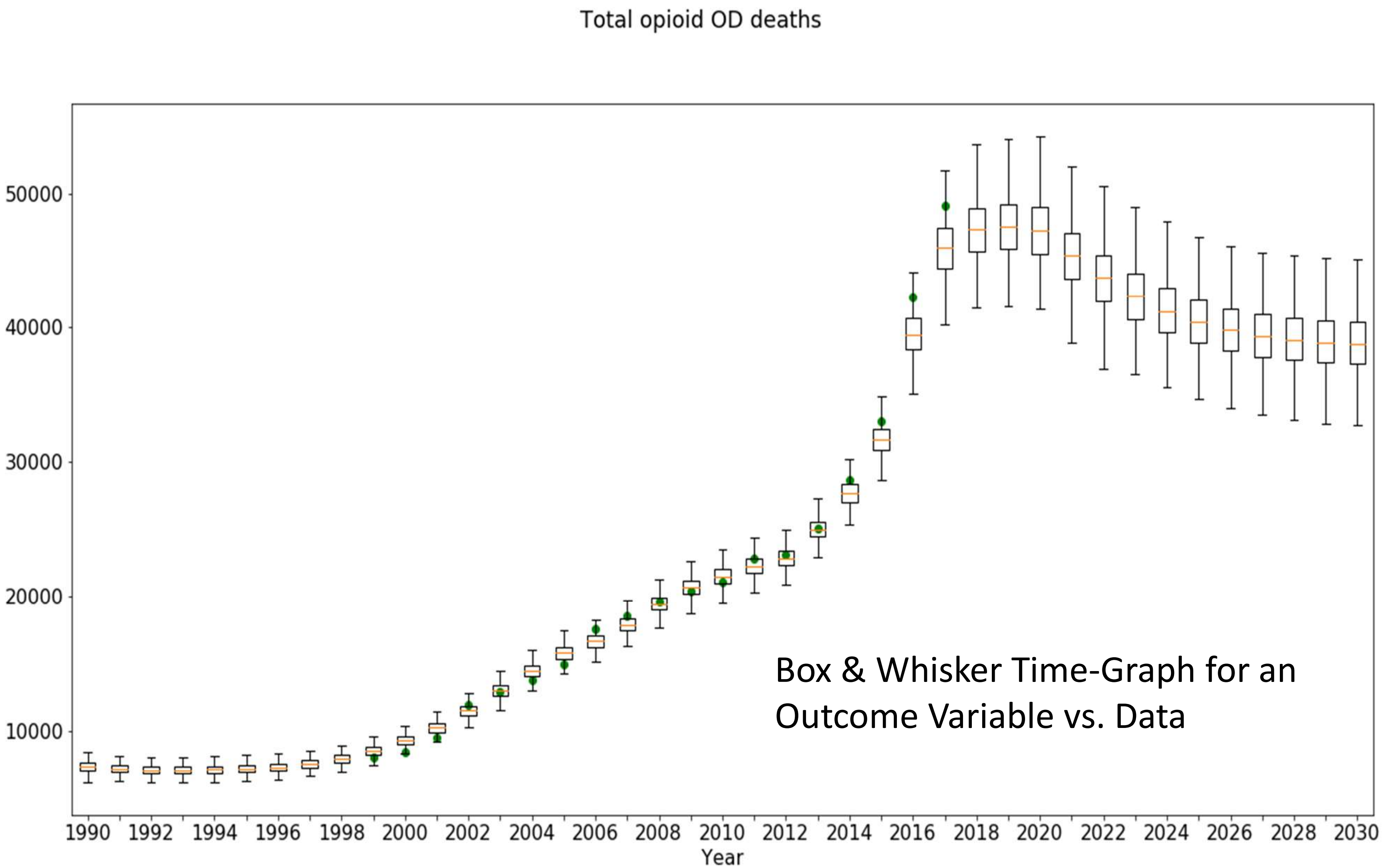




5. Use QPS to run a file driven sensitivity run

- This time, saving all of the outcome trajectories for each qualified parameter set
- Excel can be used to create a visualization, but Python provides more flexibility
- To display the uncertainty interval at time at each time point along the trajectory in a box & whisker format, along with the actual data

Example outcome trajectory uncertainty intervals



Make policy analysis runs and compute uncertainty intervals

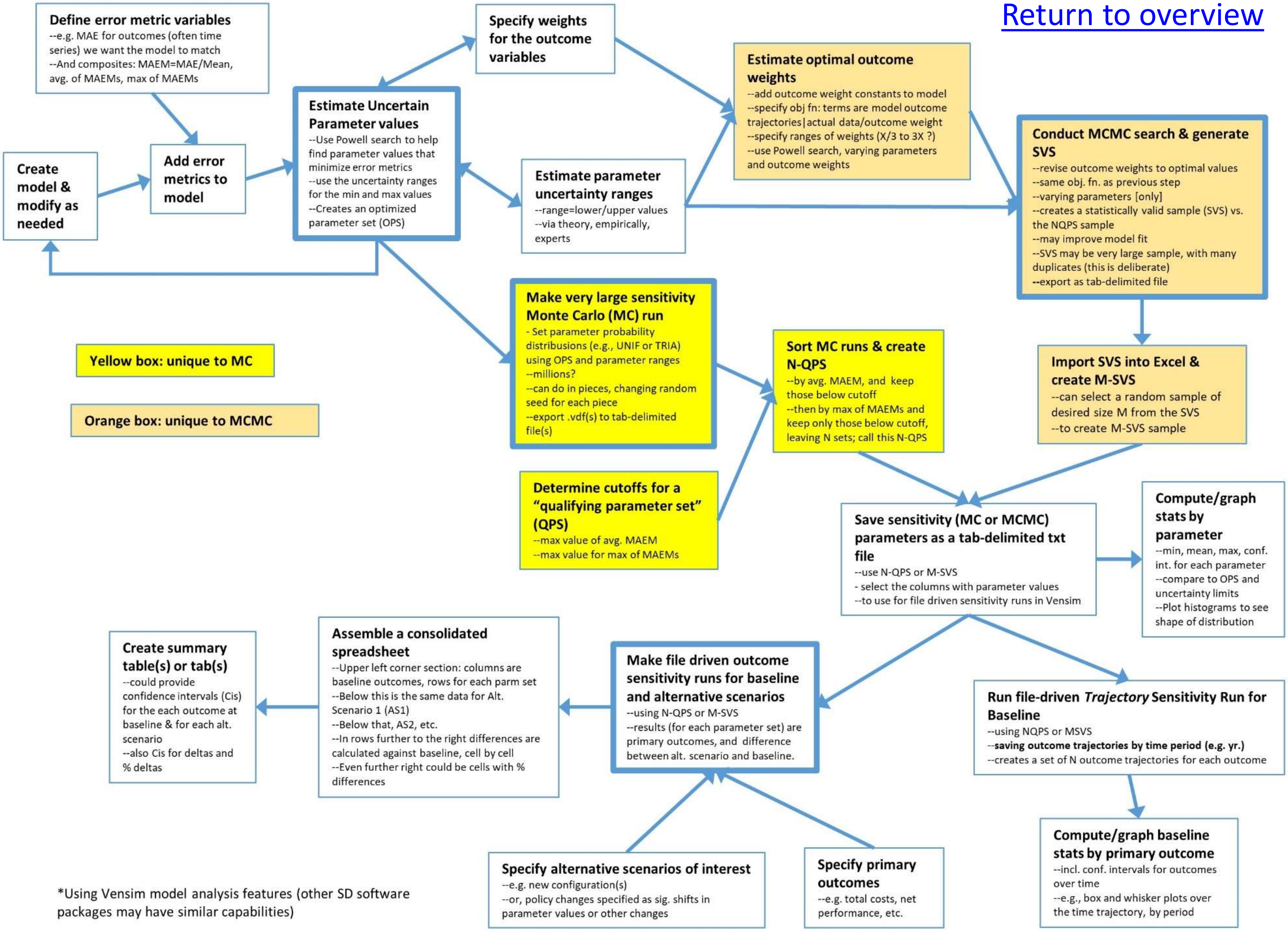
- Use the QPS to run each policy alternative for each qualified set of parameters
- The key here is that the differences at each time point for each outcome, are computed exactly, run by run
- Yielding a meaningful distribution of the effects of parameter uncertainty on the predicted impact of the policy change
  - Perhaps better on average, but could be worse

Example Policy Analysis Results

		Optimized parameter set		QPS 1119 MC Result			QPS 1119 MC, % change vs baseline		
OUTCOME MEASURE	TEST CONDITION	Result	% chg vs Baseline	Mean	Range (min, max)		Mean %Δ	Range (min, max)	
Persons with OUD (thou)	Baseline	1,694		1,593	1,111	2,084			
	Avg MME dose down 20%	1,510	-10.9%	1,416	1,035	1,823	-11.1%	-25.7%	-3.4%
	Diversion Control 30%	1,428	-15.7%	1,339	1,007	1,716	-15.9%	-37.4%	-4.6%
	Treatment rate 65% (from 45%)	1,713	1.1%	1,585	1,054	2,130	-0.5%	-9.0%	5.0%
	Naloxone lay use 20% (from 4%)	1,728	2.0%	1,624	1,150	2,111	1.9%	1.3%	2.3%
	All 4 policies combined	1,285	-24.1%	1,189	905	1,560	-25.4%	-60.2%	-6.5%
Overdoses seen at ED	Baseline	154,710		149,450	124,745	179,297			
	Avg MME dose down 20%	152,686	-1.3%	145,473	118,491	176,363	-2.7%	-8.2%	3.8%

Process for Addressing SD Model Uncertainty*

Return to overview



*Using Vensim model analysis features (other SD software packages may have similar capabilities)