Parallel Decomposition Strategies for Estimation of a Spatially Distributed, Nonlinear, Childhood Infectious Disease Model

[1]Daniel P. Word, [2]Jean-Paul Watson, [3]David L. Woodruff, [1]Carl D. Laird

1) Artie McFerrin Department of Chemical Engineering, Texas A&M University, College Station, TX 77843
2) Discrete Algorithms and Math Department, Sandia National Laboratories, Albuquerque, NM 87185
3) Graduate School of Management, University of California, Davis, Davis, CA 95616

UC DAVIS GRADUATE SCHOOL OF MANAGEMENT

PREPARING INNOVATIVE
LEADERS FOR GLOBAL IMPACT

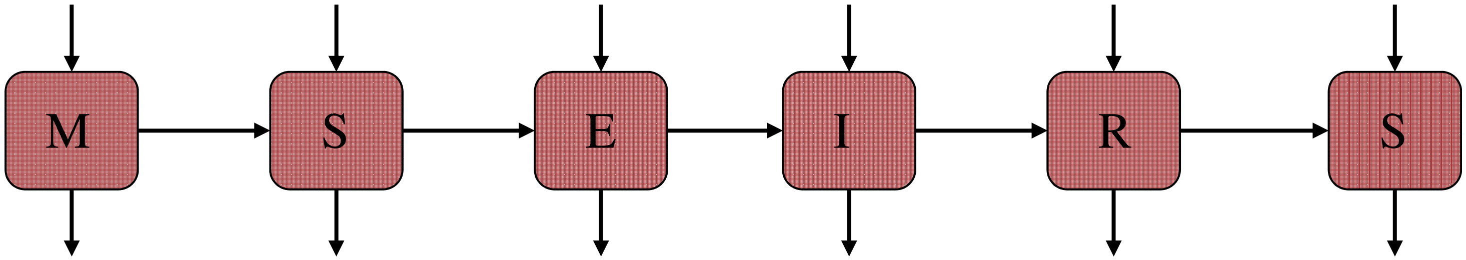
Sandia National Laboratories

COOPR

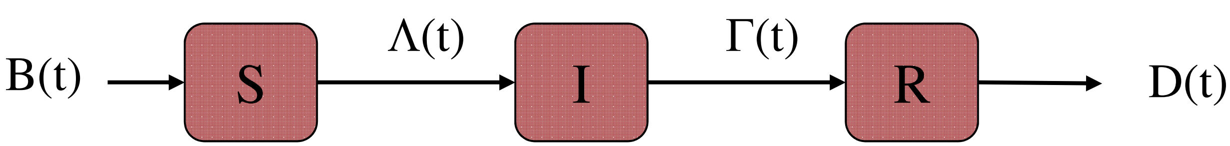
Introduction

- The Study of Infectious Diseases Remains Important
  - Understanding Disease Dynamics
  - Public Health Program Implementation
- Childhood Diseases are Useful for Study
  - Clear Temporal Dynamics
    - Annual and Biennial Drivers Dependent on Birthrate
    - Seasonal Patterns
    - Not Purely Random

Compartment Modeling



- Many Compartment Models Possible [2]
  - Compartments Reflect Disease Progression
- SIR Compartment Model
  - Suitable for Childhood Infectious Diseases



$$\Lambda(t) = \beta I(t) S(t)$$

$$\Gamma(t) = \gamma I(t)$$

Differential Equation Modeling

$$\frac{dS}{dt} = \frac{-\beta(y(t))S(t)I(t)}{N} + \varepsilon_M(t) + \mu(t)N$$

$$\frac{dI}{dt} = \frac{\beta(y(t))S(t)I(t)}{N} + \varepsilon_M(t) - \gamma\, I(t)$$

$$\frac{d\phi}{dt} = \frac{\beta(y(t))S(t)I(t)}{N} + \varepsilon_M(t)$$

$$\Phi_t = \eta_t(\phi_t - \phi_{t-1}) + \varepsilon_{\phi t}$$

Differential Equation Modeling (cont.)

- Transmission parameter, β, with yearly periodicity
- Contains both model and measurement noise
- Discretized using third-order Radau collocation

$$\frac{dx}{dt} = f(x,t)$$

$$x_{i,j} = x_{i-1,3} + h \sum_{k \in CP} a_{k,j} f(x_i, t_{i,k})$$

Measles Data

- Available Data from 60 Cities in England and Wales for the Years 1944-1963
  - Reported Case Counts
  - Reported Population and Birth Records
- Clear Temporal Dynamics
  - Annual and Biennial Drivers Dependent on Birthrate
  - Seasonal Patterns
  - Not Purely Random
- Challenges of Data
  - Underreporting of Case Counts
  - No Susceptible Information
- Other Large Data Sets Exist
  - 900 cities from England and Wales
  - 72 provinces in Thailand
  - 100 cities from the United States

Problem Description

The temporal dynamics observed in the 20 years of data indicate a seasonal driver that is consistent across all cities. By treating each data set as a separate scenario and letting parameters be considered first-stage decision variables and errors be treated as second-stage variables, the parameter estimation problem can be solved using stochastic programming techniques. Here, the problem is formulated in the open-source modeling language Pyomo [6] and solved with two approaches using PySP [8].

Extensive Form Approach

The first approach uses PySP to formulate the extensive form of the problem and standard nonlinear solvers to solve the problem. PySP makes the generation of this extensive form simple since the base model for a single city is formulated in Pyomo, the scenario tree is specified, and PySP automatically generates the modified objective function and coupling constraints to define the extensive form of the problem. This gives:

Extensive Form Approach (cont.)

$$\min \sum_{c\in C}\left(P^c \sum_{t\in\mathcal{T}} \left(\omega_M (\varepsilon^c_{Mt})^2 + \omega_\phi (\varepsilon^c_{\phi t})^2\right)\right)$$

s.t.

$$\frac{dS^c}{dt} = \frac{-\beta^c(y(t))S^c(t)I^c(t)}{N^c} + \varepsilon^c_M(t) + \mu^c(t)N^c$$

$$\frac{dI^c}{dt} = \frac{\beta^c(y(t))S^c(t)I^c(t)}{N^c} + \varepsilon^c_M(t) - \gamma\, I^c(t)$$

$$\frac{d\phi^c}{dt} = \frac{\beta^c(y(t))S^c(t)I^c(t)}{N^c} + \varepsilon^c_M(t)$$

$$\Phi^c_t = \eta_t(\phi^c_t - \phi^c_{t-1}) + \varepsilon^c_{\phi t}$$

$$0 \le I^c(t), S^c(t) \le N^c$$

$$0 \le \phi^c(t)$$

$$0.05 \le \beta^c(y(t)) \le 5.0$$

$$\beta^c(y(t)) = \beta(y(t))$$

Extensive Form Approach (cont.)

This system is for every city, c, for which we have data. The objective function is weighted so that large cities which have more data than smaller cities have a higher weight, P^c. This problem is solved using the interior-point nonlinear solver IPOPT [5].

Progressive Hedging Approach

The second approach uses PySP to solve the problem with an implementation of the progressive hedging (PH) algorithm of Rockafellar and Wets [7]. This uses the same base Pyomo model and scenario tree as the extensive form, but the algorithm solves each city separately allowing for parallelization of the solves. The algorithm is outlined below.

Progressive Hedging Approach (cont.)

For the base model: $f_c(\beta(c))$

1) At the first iteration: $k \leftarrow 0$

2) For all scenarios, $c \in C$:

$$\beta_c^0 \leftarrow \text{argmin } f_c(\beta_c)$$

3) Update the iterate:

$$k \leftarrow k + 1$$

$$\bar{\beta}^{k-1} \leftarrow \sum_{c \in C} P_c \beta_c^{k-1}$$

where $\sum_{c \in C} P_c = 1$

Progressive Hedging Approach (cont.)

4) For all scenarios, $c \in C$:

$$w_c^k \leftarrow w_c^{k-1} + \rho\left(\beta_c^{k-1} - \bar{\beta}^{k-1}\right)$$

$$\beta_c^k \leftarrow \operatorname{argmin}\ f_c(\beta_c) + w_c^k \beta_c + \frac{\rho}{2}||\beta_c - \bar{\beta}^{k-1}||_2^2$$

5) If termination criterion not met, go back to step 3.

Each individual scenario is solved using IPOPT and can be solved in parallel. This allows for the solution of extremely large problems that would overwhelm the available memory for many serial approaches. Additionally, parallel solution can allow for significantly decreased overall run times.

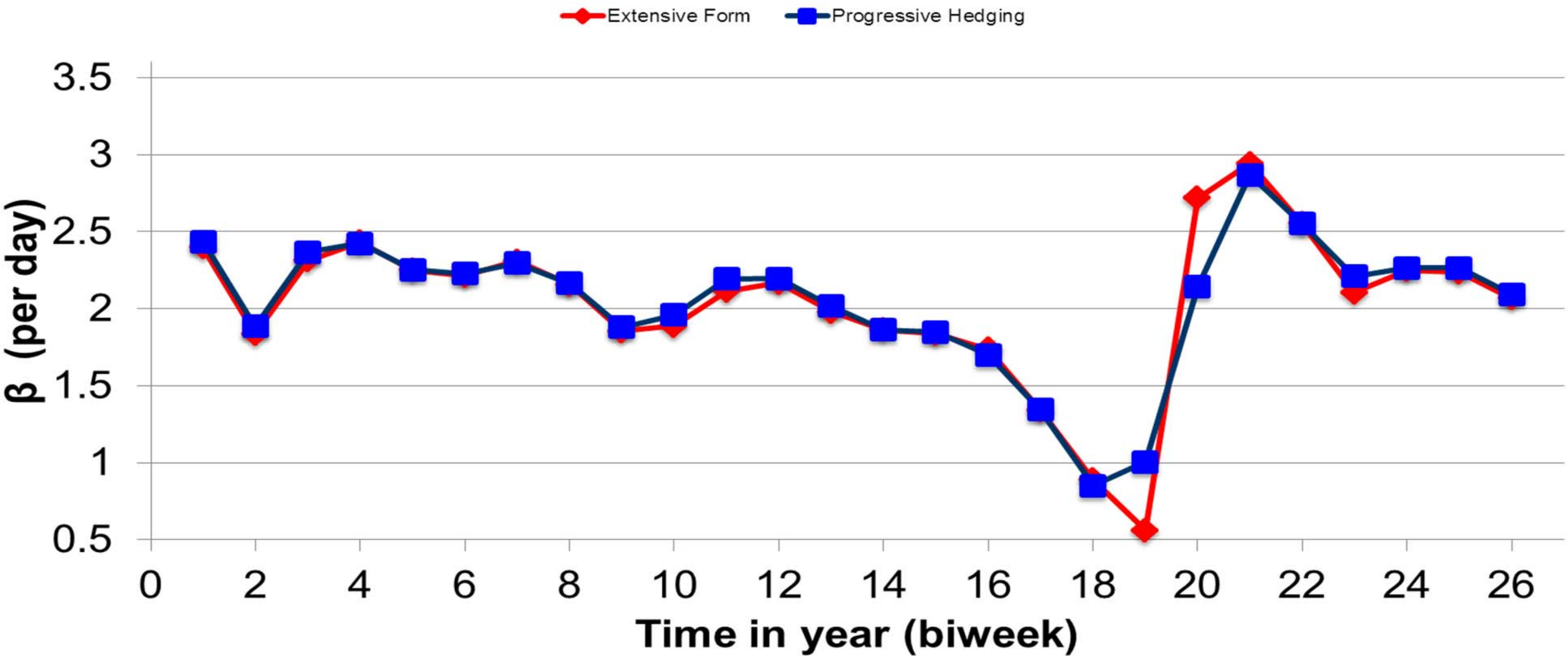
Timing & Memory Results

	Extensive Form	PH (serial)	PH (60 processors)
Jacobian Nonzeros	9987362	166424	166424
Variables	2404348	40070	40070
Constraints	2279461	37963	37963
Solver Time (min)	43.2	371.2	8
PySP Time (min)	46.3	140.1	109
Total Time (min)	89.5	655	117
Solver RAM (GB)	8	0.15	<0.080
PySP RAM (GB)	18	12.6	<0.45

Timing and Memory Results (cont.)

- EF gives very large problem that exceeds memory capacity of most computers
- EF gives optimal solution but requires 26 GB of RAM and requires well over an hour to solve
- In serial, PH gives similar solution in more time than the EF, but using considerably less peak memory
- In parallel, PH gives similar solution in significantly less time than the EF, and no single computing node requires a significant amount of memory

Estimation Results



- The solution from PH is almost exactly the same as from using the EF even though PH is only an approximately optimal solution.
- The results are consistent with literature results [3,4]

PREPARING INNOVATIVE LEADERS FOR GLOBAL IMPACT

Conclusions

- Nearly Identical Solutions Via Different Techniques
  - PySP approach gives approximately optimal solution
  - NLP approach gives optimal solution
- PySP is Useful for Much Larger Problems
  - Approach is highly parallelizable
  - Memory constraints are avoidable
  - Computational costs spread across multiple machines
- Pyomo Provides Convenient Framework for Multiple Solution Approaches
- Future Work
  - Research into appropriate tuning of progressive hedging algorithm parameters
  - Optimization of algorithm code to reduce problem formulation costs

References

1) R.M. Anderson and R.M. May. Infectious Diseases of Humans: Dynamics and Control. Oxford University Press Inc., New York, 1991.
2) H.W. Hethcote. The mathematics of infectious diseases. SIAM Review, 42(4):599–653,October 2000.
3) B.F. Finkenstädt and B.T. Grenfell. Time series modelling of childhood diseases: a dynamical systems approach. Journal of the Royal Statistical Society, Series C, 49:187–205, 2000.
4) S. Cauchemez and N.M. Ferguson. Likelihood-based estimation of continuous-time epidemic models from time-series data: application to measles transmission in London. Journal of the Royal Society Interface, 5(25):885, 2008.
5) A. Wächter and L.T. Biegler. On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming. Mathematical Programming, 106(1):25–57, 2006.
6) W.E. Hart, J-P. Watson, D. Woodruff. Pyomo: modeling and solving mathematical programs in Python. Mathematical Programming Computation, 3(3):219-260, 2011.
7) R.T. Rockafellar and R.J.B. Wets. Scenarios and Policy Aggregation in Optimization Under Uncertainty. Mathematics of Operations Research, 16(1), 1991.
8) J-P. Watson, D.L. Woodruff, W.E. Hart. PySP: Modeling and Solving Stochastic Programs in Python. (under minor revision) Mathematical Programming Computation, 2011.

Acknowledgments

Financial support for this work was provided by Sandia National Laboratories and the Office of Advanced Scientific Computing Research within the DOE Office of Science as part of the Applied Mathematics program. Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the U.S. Department of Energy's National Nuclear Security Administration under contract DEAC04-94AL85000.

UC DAVIS GRADUATE SCHOOL OF MANAGEMENT

PREPARING INNOVATIVE LEADERS FOR GLOBAL IMPACT

ATM

Sandia National Laboratories

COOPR