Unfinished or Inactive

Pandemic Mitigation Optimization

October 22, 2021

Contents

1 Model

1.1 SIRD Model with Variable-Cost Interventions

Parameters

- Suppose there are N total individuals, of whom I_0 are initially infected. There are m interventions to consider, each of which has (up to) n levels of intensity.

- Let A_{ijt} denote the fixed cost of implementing policy i at level j at time t.
- Let B_{ijt} denote the *switching* cost of implementing policy i at level j at time t (only incurred if policy not implemented in previous period).
- Let C_{ijt} denote the per-susceptible-individual cost of implementing poilcy i at level j at time t.
- Let $C_{infection}$ and C_{death} denote the costs associated with a single individual being infected in a given period, and a single individual losing their life due to disease, respectively.
- Let K_I correspond to the infection rate such that the number of new infections is proportional to K_I multiplied by the number of interactions between susceptible and infected individuals, modeled as the product of the sizes of those populations.
- Let K_R and K_D denote the proportion of infected individuals in each period who recover and die, respectively.
- Let P_{ijt} denote the factor by which new infections are decreased in period t as a result of implementing policy i at level j. In this model, these factors are independent of one another should multiple policies be implemented simultaneously.

Decision Variables

- Let
$$y_{ijt} = \begin{cases} 1 & : \text{policy } i \text{ is implemented at level } j \text{ in time period } t \\ 0 & : \text{otherwise} \end{cases} \tag{8a}$$
- Let
$$z_{ijt} = \begin{cases} 1 & : \begin{array}{l}\text{policy } i \text{ is implemented at level } j \text{ in time period } t,\\ \text{but not } t-1\end{array} \\ 0 & : \text{otherwise} \end{cases} \tag{9a}$$

State Variables

- Let S_t, I_t, R_t, d_t, and D_t denote the population of individuals at time t who are Susceptible, Infected, Recovered, dying (in the current period), and Dead (cumulatively), respectively. These values depend on the interventions applied.
- Let P_t denote the cumulative factor by which new infections are decreased between periods $t-1$ and t. That is,
$$P_t = \prod_{\substack{i,j \text{ s.t.} \\ \text{policy } i \text{ used} \\ \text{at level } j \\ \text{in period } t}} P_{ijt} \tag{6a}$$

Model Formulation: Disease Mitigation Optimization (DMO)

$$
\begin{aligned}
&\underset{y,P,S,I,R,D,d}{\text{Minimize}} && \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{T} A_{ijt}y_{ijt} + B_{ijt}z_{ijt} + C_{ijt}S_t y_{ijt} + \sum_{t=1}^{T} C_{infection}I_t + C_{death}d_t && && (0)\\
&\text{s.t.} \quad S_t && = S_{t-1} - K_I \cdot P_t \cdot S_{t-1} \cdot I_{t-1} && \forall\ t \in \{2,\dots,T\} && (1)\\
&I_t && = I_{t-1} + K_I \cdot P_t \cdot S_{t-1} \cdot I_{t-1} - K_R \cdot I_{t-1} - K_D \cdot I_{t-1} && \forall\ t \in \{2,\dots,T\} && (2)\\
&R_t && = R_{t-1} + K_R \cdot I_{t-1} && \forall\ t \in \{2,\dots,T\} && (3)\\
&d_t && = K_D \cdot I_{t-1} && \forall\ t \in \{2,\dots,T\} && (4)\\
&D_t && = D_{t-1} + d_t && \forall\ t \in \{2,\dots,T\} && (5)\\
&P_t && = \prod_{i=1}^{m}\prod_{j=1}^{n}(1 - y_{ijt} + P_{ijt} \cdot y_{ijt}) && \forall\ t \in \{1,\dots,T\} && (6)\\
&\sum_{j=1}^{n} y_{ijt} && \le 1 && \forall\ i,t && (7)\\
&y_{ijt} && \in \{0,1\} && \forall\ i,j,t && (8)\\
&z_{ijt} && \ge y_{ij(t)} - y_{ij(t-1)} \qquad (\text{let } y_{ij0} = 0\ \forall\ i,j) && \forall\ i,j,t && (9)\\
&0 \le z_{ijt} && \le 1 && \forall\ i,j,\ \forall\ t \ge 1 && (10)\\
&I_1 && = I_0\\
&S_1 && = N - I_0\\
&D_1 && = 0\\
&R_1 && = 0\\
&d_1 && = 0
\end{aligned}
$$

The objective function (0) is the sum of the costs of implementing the policy interventions in all periods and the costs associated with the resulting disease and death in all periods (due to lost productivity and resources).

The constraints (1),(2),(3),(4), and (5) model the SIRD compartment subpopulations as the disease progresses alongside the infection-reduction factors P_t at each period $t = 1, \dots, T$. The constraint (6) models the multiplicative effect of multiple interventions being applied in the same period, as described by (6a). Equation (7) enforces the logical constraint that at most one level from each policy be used in each period, and (8) ensures that the y_{ijt} variables correspond to the binary definition in (8a).

1.2 SIRD Model with Non-Variable-Cost Interventions

Replace the objective (0) in the mathematical program formulation of the **(DMO)** model with

$$\underset{y,P,S,I,R,D,d}{\text{Minimize}} \quad \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{T} A_{ijt}y_{ijt} + B_{ijt}z_{ijt} + C_{ijt}\cdot S_t \cdot y_{ijt} + \sum_{t=1}^{T} C_{infection}\cdot I_t + C_{death}\cdot d_t \tag{0}$$

$$\Downarrow$$

$$\underset{y,P,S,I,R,D,d}{\text{Minimize}} \quad \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{T} A_{ijt}y_{ijt} + B_{ijt}z_{ijt} + C_{ijt}\cdot N \cdot y_{ijt} + \sum_{t=1}^{T} C_{infection}\cdot I_t + C_{death}\cdot d_t \tag{11}$$

2 Heuristics

2.1 Lagrangian Heuristic and Lower Bound

We can utilize a Lagrangian relaxation to the full problem **(DMO)** by relaxing constraints (6) and instead penalizing the objective function using dual multipliers. The relaxed problem decomposes into two computationally less expensive subproblems; by iteratively updating the multipliers using gradient ascent, the lower bound tightens. Furthermore, we can transform a solution to the relaxed problem into a feasible solution for the full problem, yielding a heuristic solution in its own right.

First, we focus on the variant of the model in which interventions do not have a variable cost depending on the size of the susceptible population (S_t), and instead have a "variable" cost proportional to the size of the entire population (N), as in (11) (i.e., the costs do not "vary" between time periods). This allows the Lagrangian minimization problem to be split into two subproblems.

Then, we transform constraint (6) using a logarithm:

$$\ln P_t = \sum_{i=1}^{m}\sum_{j=1}^{n} \ln(1 - y_{ijt} + P_{ijt}y_{ijt}), \ \forall\ t = 1,\ldots,T. \tag{6-log}$$

Next, we remove this constraint from **(DMO)** and augment the objective (11) via multipliers $\lambda_t, t = 1,\ldots,T$ to obtain a relaxed minimization problem:

$$\underset{y,P,S,I,R,D,d}{\text{Minimize}} \quad [\text{Objective (11)}] + \sum_{t=1}^{T}\lambda_t\left(\sum_{i=1}^{m}\sum_{j=1}^{n}\ln(1 - y_{ijt} + P_{ijt}y_{ijt}) - \ln P_t\right) \tag{12}$$

$$\text{s.t.} \quad 0 \le P_t \le 1 \tag{13}$$

Constraints from **(DMO)** except for (6).

An optimal value to the augmented problem (12) is a lower bound to the optimal value of the full problem **(DMO)** . We add an extra constraint (13) to enforce the logical constraints that the policy effectiveness factors are between 0 and 1; note that in a numerical implementation, it may actually be preferable to precompute a reasonable lower bound on $\underline{P}_t \in (0,1)$ and constrain $\underline{P}_t \le P_t \le 1$ because the logarithm in (12) is undefined for $P_t = 0$.

By iteratively solving the augmented problem (12) and then using subgradient ascent to update λ_t for all $t = 1, \ldots, T$, we obtain increasingly tighter lower bounds on the optimal value for the full problem **(DMO)** .

Note that the augmented problem (12) can be decomposed into two minimization problems with optimal values $L_1(\boldsymbol{\lambda})$ and $L_2(\boldsymbol{\lambda})$:
$L_1(\boldsymbol{\lambda})$ is the solution to

$$\underset{y}{\text{Minimize}} \quad \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{T}[A_{ijt} \cdot y_{ijt} + B_{ijt} z_{ijt} + C_{ijt} \cdot N \cdot y_{ijt} + \lambda_t \ln(1 - y_{ijt} + P_{ijt} y_{ijt})]$$

$$\sum_{j=1}^{n} y_{ijt} \leq 1 \qquad \forall\, i, t \tag{7}$$

$$y_{ijt} \in \{0, 1\} \qquad \forall\, i, j, t \tag{8}$$

$$z_{ijt} \geq y_{ij(t)} - y_{ij(t-1)} \quad \forall\, i, j, t \tag{9}$$

$$0 \leq z_{ijt} \leq 1 \qquad \forall\, i, j, \ \forall\, t \geq 1 \tag{10}$$

This integer program can be solved by standard off-the-shelf software such as Gurobi.
Note: Nonlinear integer programs have only been solvable by Gurobi since November 2019. I wonder whether there is a more complete explanation of why this subproblem is in fact "easier" than the full problem.

$L_2(\boldsymbol{\lambda})$ is the solution to

$$\underset{P,S,I,R,D,d}{\text{Minimize}} \quad \sum_{t=1}^{T}[C_{infection} \cdot I_t + C_{death} \cdot d_t - \lambda_t \ln P_t]$$

$$\text{s.t.} \quad S_t = S_{t-1} - K_I \cdot P_t \cdot S_{t-1} \cdot I_{t-1} \qquad \forall\, t \in \{2, \ldots, T\} \tag{1}$$

$$I_t = I_{t-1} + K_I \cdot P_t \cdot S_{t-1} \cdot I_{t-1} - K_R \cdot I_{t-1} - K_D \cdot I_{t-1} \quad \forall\, t \in \{2, \ldots, T\} \tag{2}$$

$$R_t = R_{t-1} + K_R \cdot I_{t-1} \qquad \forall\, t \in \{2, \ldots, T\} \tag{3}$$

$$d_t = K_D \cdot I_{t-1} \qquad \forall\, t \in \{2, \ldots, T\} \tag{4}$$

$$D_t = D_{t-1} + d_t \qquad \forall\, t \in \{2, \ldots, T\} \tag{5}$$

$$P_t \leq 1 \tag{13}$$

$$I_1 = I_0$$

$$S_1 = N - I_0$$

$$D_1 = 0$$

$$R_1 = 0$$

$$d_1 = 0$$

This problem has no integer constraints and can be solved by any nonlinear programming software. To increase the tightness of the bound in the gradient-ascent step for the multipliers $\boldsymbol{\lambda}$, where

$\boldsymbol{\lambda}^+$ represents the vector of multipliers at a subsequent iteration, we use the updating rule:

$$\boldsymbol{\lambda}^+ = \boldsymbol{\lambda} + \gamma \left(\nabla L_1(\boldsymbol{\lambda}) + \nabla L_2(\boldsymbol{\lambda}) \right),$$

i.e.

$$\lambda_t^+ = \lambda_t + \gamma \cdot \left(\sum_{i=1}^{m} \sum_{j=1}^{n} \ln(1 - y_{ijt} + P_{ijt} y_{ijt}) - \ln P_t \right). \tag{14}$$

To obtain a feasible solution to the full problem **(DMO)** after any iteration of this procedure, one can fix the values of $y_{ijt}, i = 1, \ldots, m, j = 1, \ldots, n, t = 1, \ldots, T$ in **(DMO)** to those obtained in the decomposed minimizations, which immediately yields values of $P_t, t = 1, \ldots, T$, which in turn gives values of the compartment subpopulations (S, I, R, d, and D) following basic bookkeeping.

This procedure can be iteratively performed indefinitely, and will yield a sequence of nondecreasing lower bounds to the full problem **(DMO)** . As a stopping criterion, one can terminate when the relative improvement between two iterations is less than a threshold ("improvement"), or when the relative optimality gap between the incumbent feasible solution and the greatest lower bound is less than a threshold ("optimality gap"). More formally, the heuristic corresponding to this procedure can be described as follows:

Initialize $\lambda_t \leftarrow 0$ for all t.
Do
    Minimize to obtain $L_1(\boldsymbol{\lambda})$ and $L_2(\boldsymbol{\lambda})$
    Update $\boldsymbol{\lambda}$ via gradient ascent as in (14)
    The value $L_1(\boldsymbol{\lambda}) + L_2(\boldsymbol{\lambda})$ gives a lower bound.
    The optimal values of y_{ijt} from $L_1(\boldsymbol{\lambda})$ substituted in the original problem **(DMO)**
    yield a feasible solution and thus an upper bound for that problem's optimal solution.
Repeat while stopping condition (improvement or optimality gap) is not met.

2.2 w-Period *Time-Greedy* Heuristic

The computational resources required to solve **(DMO)** to optimality will likely exceed what is available to decisionmakers when the number of policy options and the number of time periods are large. However, even with a large number of policy options, a small number of time periods may make the decision space small enough to solve to optimality even with an unsophisticated exhaustive search. Decisions made when considering a small number of time periods may be reasonable to use over a longer time horizon.

The 1-period *time-greedy* algorithm is the greedy policy in which decisions are made only considering one period at a time. In the w-period greedy heuristic, decisions are made only considering w periods at a time. After decisions have been made optimally over the first w periods, the decisions for period 1 are fixed, and the problem is solved for periods $2, \ldots, w+1$; on the l'th iteration, the horizon of optimization is $l, \ldots, l + w - 1$. This continues until the horizon is $T - (w - 1), \ldots, T$, for a total of $T - (w - 1)$ iterations.

For the following algorithm, we consider the $T_{horizon}$-period objective to be the objective obtained after $T_{horizon}$ periods, rather than the full T. We replace the objective of **(DMO)** (0) with the following:

$$\underset{y,P,S,I,R,D,d}{\text{Minimize}} \quad \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T} A_{ijt}y_{ijt} + B_{ijt}z_{ijt} + C_{ijt}\cdot S_t \cdot y_{ijt} + \sum_{t=1}^{T} C_{infection}\cdot I_t \qquad + C_{death}\cdot d_t \tag{0}$$

$$\Downarrow$$

$$\underset{y,P,S,I,R,D,d}{\text{Minimize}} \quad \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T_{horizon}} A_{ijt}y_{ijt} + B_{ijt}z_{ijt} + C_{ijt}\cdot S_t \cdot y_{ijt} + \sum_{t=1}^{T_{horizon}} C_{infection}\cdot I_t + C_{death}\cdot d_t \tag{15}$$

We refer to the variant of **(DMO)** with only $T_{horizon}$ time periods as $(\mathbf{DMO})_{T_{horizon}}$.

We also use the term "fix" to mean that a variable's value is set, and the variable is no longer treated as a decision variable but a parameter of the problem.

```
Initialize y_ijt ← 0 for all i,j,t.
For T_0 in {1,...,T − w + 1}
    For all t<T_0
        Fix y_ijt to whatever value it currently holds for all i,j.
        Fix P_t, S_t, I_t, R_t, D_t, d_t to whatever values they currently hold.
    For all t∈ {T_0,...,T_0 + w − 1}
        Unfix y_ijt for all i,j. Unfix P_t, S_t, I_t, R_t, D_t, d_t.
    Solve (DMO)_{T−w+1} with the un-fixed variables.
```

2.3 w-Period *Time-Greedy* Heuristic with T^+-period Lookahead

In the execution of the w-period time-greedy solution, and indeed any variant of **(DMO)** , the primary difficulty is optimally finding values of integer-constrained variables. It is trivial, in fact, to consider the disease progression over time periods subsequent to the w periods during which optimal interventions are being considered in the context of the w-period time-greedy heuristic. This motivates a lookahead heuristic, in which the quality of a decision is assessed not just on the disease-related and policy-related costs within a w-period interval, but additionally on the disease-related costs during T^+ subsequent time-periods, during which the decision-maker does not make *any* policy decisions (and so no interventions are chosen).

This should yield more aggressive policy decisions than the w-period time-greedy heuristic, as the disease-related costs associated with any policy intervention menu are higher, and thus it will be desirable to further decrease infections during the w-period window.

The difference between these two heuristics can be summarized as follows:

- In the w-period time-greedy heuristic, only w periods are considered at a time in terms of decisionmaking and disease progression.

- In the T^+-period lookahead variant, the decisionmaker's "hands are tied" (they are forced to use no intervention) after the w periods of decisionmaking, but they calculate and make decisions based on the costs associated with disease progression during an additional T^+ time periods.

The algorithm is as follows:

```
Initialize $y_{ijt} \leftarrow 0$ for all i,j,t.
For $T_0$ in $\{1, \ldots, T - w + 1\}$
    For all t<$T_0$
        Fix $y_{ijt}$ to whatever value it currently holds for all i,j.
        Fix $P_t, S_t, I_t, R_t, D_t, d_t$ to whatever values they currently hold.
    For all t∈ $\{T_0, \ldots, T_0 + w - 1\}$
        Unfix $y_{ijt}$ for all i,j. Unfix $P_t, S_t, I_t, R_t, D_t, d_t$.
    Solve $(\mathbf{DMO})_{T-w+1+T^+}$ with the un-fixed variables.
```

2.4 B-Policy ***Policy-Greedy*** Solution (implemented)

Much of the difficulty of solving **(DMO)** largely stems from the highly nonlinear constraint (6), which involves the product of $m \times n$ integer-constrained variables. On the other hand, if only a single policy with a single level is considered, the problem can be solved to optimality over a long time horizon quickly, with modest computational resources.

The following *policy-greedy* algorithm leverages this fact to make optimal decisions for only one policy at a time, fixing the plan for that policy while considering adding another, until either B policies are chosen or there is no improvement from adding any additional policy (at any level).

To articulate this, we introduce parameters P_t^0 for $t = 1, \ldots, T$, and modify constraint (6) to

$$P_t = \prod_{i=1}^{m} \prod_{j=1}^{n} (1 - y_{ijt} + P_{ijt} \cdot y_{ijt}) \;\; \forall\, t \in \{1, \ldots, T\} \tag{6}$$

$$\Downarrow$$

$$P_t = P_t^0 \prod_{i=1}^{m} \prod_{j=1}^{n} (1 - y_{ijt} + P_{ijt} \cdot y_{ijt}) \;\; \forall\, t \in \{1, \ldots, T\} \tag{16}$$

where P_t^0 (instead of simply "1") represents the factor by which the infection rate is decreased by decisionmaking *if no policies are implemented.*

```
Set USED← {}, OBJECTIVE← ∞
For b in {1,...,B}
    Set ITERATION_OBJECTIVE← ∞
    For each policy i
        If (i,j) ∉USED for any of j∈ {1,...,n}  # Policy i has not been used at any level
            For each level j
                Set m←1, n←1, and solve (DMO) with only policy i at level j.
                Solve this problem
                Set SOLUTION_OBJECTIVE←objective value of this problem
                If SOLUTION_OBJECTIVE<ITERATION_OBJECTIVE
                    Set ITERATION_OBJECTIVE←SOLUTION_OBJECTIVE
    If ITERATION_OBJECTIVE<OBJECTIVE
        Set OBJECTIVE←ITERATION_OBJECTIVE
        Set USED←USED∪{(i,j)}
        Set P_t^0 ← P_t^0 × P_ijt for all t where policy i at level j is used in the solution
        that produced ITERATION_OBJECTIVE
    Else
        Terminate without adding any new policy
```

2.5 Index Policy - basic - questions

- Which periods should the policy be used in?
- What index should be used?

2.6 Index and Assortment Index Policy

2.7 Local Search

2.8 F-Factor Early Stopping Using BARON/Gurobi/DICOPT/BONMIN

The following heuristic requires a solution strategy (a "solver") for the **(DMO)** model formulated in Section 1.1 that can iteratively generate the following two quantities:

1. A sequence of feasible solutions with improving objective function values, referred to as "incumbent solutions" whose objective values serve as upper bounds for the problem, and
2. a sequence of increasing lower bounds for the problem, generated from any of the following:
   - continuous relaxation,
   - Lagrangian relaxation,
   - any other dualization or constraint relaxation.

This is, in fact, what most mathematical programming solvers aim to iteratively produce while solving a problem. We refer to a "solver" as a tool that achieves the two goals above. As the solvers compute, the percent difference between the lower and upper bounds - the "relative optimality gap" - shrinks. Mixed-integer programming tools typically do not prove optimality, but stop when this relative optimality gap falls below an acceptable threshold.

With an upper bound u and a lower bound l to the objective function, solving the problem to desired optimality factor F requires that

$$\frac{u-l}{u} < F.$$

Selecting a large value of F would amount to an "early-stopping" heuristic, and the solution may still be useful even though there is no reason to suspect that the generated solution is globally optimal.

```
Begin solving the (DMO) problem using a solver. For each iteration, do
    If $\frac{u-l}{u} < F$
        Stop
    Else
        Continue
```

3 Results

3.1 Lagrangian Heuristic Lower Bound Improvement

The lower bound on the objective value of **(DMO)** obtained by optimizing the (decomposed) Lagrangian relaxation described in Section 2.1 seems relatively tight on several problem instances. The corresponding heuristic also performs quite well. On the other hand, the BARON solver is able to generate solutions within a few minutes to the full **(DMO)** problem that are extremely high quality, but it does not guarantee a reasonable level of optimality even after running for hours. In particular, the BARON solver generates lower bounds as part of its numerical optimization procedure, but these lower bounds are nowhere near the values it obtains. The bounds generated by the Lagrangian procedure prove that these solutions are nearly optimal. This yields a stopping condition for the BARON solver that guarantees a desired level of optimality.

The following trials were considered to evaluate the performance of the Lagrangian method:

Trial	T	cost multiplier	effect multiplier	m	n	nConstraints	nPolicies	nVariables
0	20.0	0.5	0.50	9	4	1015	51840	1560
1			0.75	9	4	1015	51840	1560
2			1.00	9	4	1015	51840	1560
3			1.25	9	4	1015	51840	1560
4			1.50	9	4	1015	51840	1560
5		0.75	0.50	9	4	1015	51840	1560
6			0.75	9	4	1015	51840	1560
7			1.00	9	4	1015	51840	1560
8			1.25	9	4	1015	51840	1560
9			1.50	9	4	1015	51840	1560
10		1.0	0.50	9	4	1015	51840	1560
11			0.75	9	4	1015	51840	1560
12			1.00	9	4	1015	51840	1560
13			1.25	9	4	1015	51840	1560
14			1.50	9	4	1015	51840	1560
15		1.25	0.50	9	4	1015	51840	1560
16			0.75	9	4	1015	51840	1560
17			1.00	9	4	1015	51840	1560
18			1.25	9	4	1015	51840	1560
19			1.50	9	4	1015	51840	1560
20		1.5	0.50	9	4	1015	51840	1560
21			0.75	9	4	1015	51840	1560
22			1.00	9	4	1015	51840	1560
23			1.25	9	4	1015	51840	1560
24			1.50	9	4	1015	51840	1560
25	50.0	0.5	0.50	9	4	2545	51840	3900
26			0.75	9	4	2545	51840	3900
27			1.00	9	4	2545	51840	3900
28			1.25	9	4	2545	51840	3900
29			1.50	9	4	2545	51840	3900
30		0.75	0.50	9	4	2545	51840	3900
31			0.75	9	4	2545	51840	3900
32			1.00	9	4	2545	51840	3900
33			1.25	9	4	2545	51840	3900
34			1.50	9	4	2545	51840	3900
35		1.0	0.50	9	4	2545	51840	3900
36			0.75	9	4	2545	51840	3900
37			1.00	9	4	2545	51840	3900
38			1.25	9	4	2545	51840	3900
39			1.50	9	4	2545	51840	3900
40		1.25	0.50	9	4	2545	51840	3900
41			0.75	9	4	2545	51840	3900
42			1.00	9	4	2545	51840	3900
43			1.25	9	4	2545	51840	3900
44			1.50	9	4	2545	51840	3900
45		1.5	0.50	9	4	2545	51840	3900
46			0.75	9	4	2545	51840	3900
47			1.00	9	4	2545	51840	3900
48			1.25	9	4	2545	51840	3900
49			1.50	9	4	2545	51840	3900
50	150.0	0.5	0.50	9	4	7645	51840	11700
51			0.75	9	4	7645	51840	11700
52			1.00	9	4	7645	51840	11700
53			1.25	9	4	7645	51840	11700
54			1.50	9	4	7645	51840	11700
55		0.75	0.50	9	4	7645	51840	11700
56			0.75	9	4	7645	51840	11700
57			1.00	9	4	7645	51840	11700
58			1.25	9	4	7645	51840	11700
59			1.50	9	4	7645	51840	11700
60		1.0	0.50	9	4	7645	51840	11700
61			0.75	9	4	7645	51840	11700
62			1.00	9	4	7645	51840	11700
63			1.25	9	4	7645	51840	11700
64			1.50	9	4	7645	51840	11700
65		1.25	0.50	9	4	7645	51840	11700
66			0.75	9	4	7645	51840	11700
67			1.00	9	4	7645	51840	11700
68			1.25	9	4	7645	51840	11700
69			1.50	9	4	7645	51840	11700
70		1.5	0.50	9	4	7645	51840	11700
71			0.75	9	4	7645	51840	11700
72			1.00	9	4	7645	51840	11700
73			1.25	9	4	7645	51840	11700
74			1.50	9	4	7645	51840	11700

(a) *Trials. The* `cost multiplier` *column multiplies all costs in the matrices A, B, and C (*setup, switching*, and* per-individual *costs) by the same factor. An entry μ in the* `effect multiplier` *column alters the intervention effectiveness as $P_{ijt} \mapsto 1 - \mu \cdot (1 - P_{ijt})$ for all i, j, t.*

Trial	no policy obj	solver obj	lagr heuristic obj	lagr LB	solver LB	solver vs lagr lb gap
0	981337	880937	883034	749911	396034	0.174721
1	981337	683476	685243	354328	212886	0.928932
2	981337	404767	406435	-203286	122252	-2.991114
3	981337	197901	199603	182952	79407	0.081711
4	981337	103317	103317	93830	22500	0.101117
5	981337	896531	900446	767323	408269	0.168388
6	981337	699971	702655	371740	225014	0.882957
7	981337	421262	423847	414216	133759	0.017012
8	981337	214397	217015	200364	89022	0.070038
9	981337	119751	119813	110579	66528	0.082944
10	981337	911998	917859	784736	419913	0.162171
11	981337	716468	720068	389153	236813	0.841093
12	981337	437759	441260	431629	144598	0.014203
13	981337	230894	234428	217777	98699	0.060230
14	981337	136143	136143	122793	75635	0.108722
15	981337	926988	935273	802150	431318	0.155629
16	981337	732966	737482	712208	247795	0.029146
17	981337	454257	458674	413021	155100	0.099840
18	981337	247392	251843	235192	106457	0.051875
19	981337	152385	152536	138263	82463	0.102133
20	981337	941625	960275	935378	442283	0.006679
21	981337	749406	754898	729624	258471	0.027113
22	981337	470757	476090	430437	164567	0.093672
23	981337	263891	269258	252607	113650	0.044671
24	981337	168391	168931	154500	85632	0.089912
25	1003570	991946	1003570	961654	419123	0.031501
26	1003570	959398	1003570	912680	225481	0.051187
27	1003570	901317	923735	746862	187482	0.206804
28	1003570	793435	797659	740396	176725	0.071636
29	1003570	562478	564325	544127	119910	0.033726
30	1003570	999683	1003570	961654	431788	0.039545
31	1003570	974478	1003570	912680	238041	0.067711
32	1003570	925739	949305	918538	201353	0.007840
33	1003570	834430	842558	785295	185795	0.062569
34	1003570	606317	609224	589027	128222	0.029355
35	1003570	1003408	1003570	961654	443884	0.043419
36	1003570	986478	1003570	912680	249943	0.080858
37	1003570	946180	964433	870732	216481	0.086649
38	1003570	870132	887459	830196	179929	0.048105
39	1003570	649849	654125	633927	138739	0.025116
40	1003570	1003555	1003570	961654	455450	0.043572
41	1003570	995926	1003570	912680	261257	0.091210
42	1003570	963377	973483	899050	227560	0.071551
43	1003570	901650	959964	871771	180417	0.034274
44	1003570	693126	699027	678829	152044	0.021061
45	1003570	1003570	1003570	961654	466744	0.043588
46	1003570	1002698	1003570	912680	272268	0.098630
47	1003570	977635	982747	948020	239849	0.031239
48	1003570	929328	999723	911529	190799	0.019526
49	1003570	736196	743930	723732	154872	0.017222
50	1003571	991951	1003571	961662	419124	0.031496
51	1003571	–	1003571	912758	–	–
52	1003571	903823	917858	882905	187467	0.023692
53	1003571	–	854682	807463	–	–
54	1003571	–	806410	739964	–	–
55	1003571	999685	1003571	961662	431789	0.039538
56	1003571	–	1003571	912758	–	–
57	1003571	–	930037	816498	–	–
58	1003571	–	879764	861812	–	–
59	1003571	–	882722	799925	–	–
60	1003571	1003960	1003571	961662	446091	0.043984
61	1003571	986510	1003571	912758	249943	0.080802
62	1003571	947445	1003571	917137	216473	0.033046
63	1003571	–	910508	874939	–	–
64	1003571	–	876923	858490	–	–
65	1003571	1003556	1003571	961662	457771	0.043563
66	1003571	995938	1003571	912758	261257	0.091131
67	1003571	–	1003571	917137	–	–
68	1003571	922263	933694	785152	194649	0.174629
69	1003571	–	904439	886800	–	–
70	1003571	–	1003571	961662	–	–
71	1003571	1002701	1003571	912758	272269	0.098540
72	1003571	–	1003571	917137	–	–
73	1003571	–	1003571	907170	–	–
74	1003571	–	924331	899316	–	–

(b) *Accuracy results.* "lagr des L1 optGap*" : 0.03;* "lagr des L2 optGap*" : 0.10;* "lagr des optGap*" : 0.10;* "solver des optGap*" : 0.80;* "solver optGap*" : 0.80;* "lagr optGap*" : 0.00.*

Trial	solver timeToSolve	lagr timeToSolve	lagr timeToSolve L1 total	lagr timeToSolve L2 total
0	1.593573	2	0	0
1	1.539040	2	0	0
2	1.551474	3	0	1
3	1.467471	12	1	8
4	0.995703	57	7	41
5	1.754394	2	0	0
6	1.522290	2	0	0
7	1.781522	8	1	3
8	1.463810	12	1	7
9	1.484372	43	5	31
10	1.582561	2	0	0
11	1.551972	2	0	0
12	1.599526	7	1	3
13	1.476663	11	1	7
14	1.568145	39	3	30
15	1.574137	2	0	0
16	1.572989	5	1	1
17	1.576758	6	1	3
18	1.447450	14	1	9
19	1.555228	34	3	25
20	1.588227	4	1	1
21	1.583232	5	1	1
22	1.618965	5	1	2
23	1.574056	13	1	8
24	1.591432	43	4	33
25	10.727455	3	0	0
26	12.245515	4	0	1
27	13.948242	33	2	24
28	69.767561	59	5	45
29	23.468103	61	5	48
30	9.591028	4	0	0
31	12.168838	4	0	1
32	16.277850	58	5	44
33	27.403951	59	5	46
34	24.339477	61	5	48
35	12.780697	4	0	1
36	10.640248	4	0	1
37	19.338374	50	3	37
38	33.382241	59	5	46
39	24.998904	64	5	48
40	9.646505	5	0	0
41	9.329642	6	0	1
42	25.098967	57	6	41
43	27.010101	61	5	45
44	25.717360	63	5	48
45	9.913437	4	0	0
46	11.827932	4	0	1
47	33.631016	60	5	46
48	29.679266	60	5	45
49	22.322686	62	5	48
50	103.832007	30	0	6
51	519.026021	39	0	15
52	109.478602	169	35	60
53	519.126232	179	48	60
54	519.103991	376	122	120
55	99.923649	28	0	6
56	519.056369	40	0	19
57	519.117285	172	47	60
58	519.096644	233	62	80
59	519.174518	185	56	60
60	87.990617	29	0	6
61	131.486867	35	0	13
62	127.877393	102	21	40
63	519.046002	185	51	60
64	519.212181	302	91	100
65	72.644119	27	0	6
66	112.422904	34	0	13
67	519.054457	101	21	40
68	156.656894	170	47	60
69	519.631569	331	112	100
70	519.943242	29	0	6
71	137.136791	36	0	13
72	519.864556	104	21	40
73	519.824205	104	21	40
74	520.325183	186	48	60

(c) *Timing results*

Trial	no policy cost	no policy deaths	solver cost	solver policy cost	solver disease cost	solver deaths	lagr cost	lagr policy cost	lagr disease cost	lagr deaths
0	9.81e11	9.72e04	8.81e11	3.13e10	8.50e11	8.41e04	8.83e11	3.48e10	8.48e11	8.40e04
1	9.81e11	9.72e04	6.83e11	3.30e10	6.50e11	6.44e04	6.85e11	3.48e10	6.50e11	6.44e04
2	9.81e11	9.72e04	4.05e11	3.30e10	3.72e11	3.68e04	4.06e11	3.48e10	3.72e11	3.68e04
3	9.81e11	9.72e04	1.98e11	3.30e10	1.65e11	1.63e04	2.00e11	3.48e10	1.65e11	1.63e04
4	9.81e11	9.72e04	1.03e11	3.30e10	7.03e10	6.95e03	1.03e11	3.30e10	7.03e10	6.95e03
5	9.81e11	9.72e04	8.97e11	4.64e10	8.50e11	8.42e04	9.00e11	5.22e10	8.48e11	8.40e04
6	9.81e11	9.72e04	7.00e11	4.95e10	6.50e11	6.44e04	7.03e11	5.22e10	6.50e11	6.44e04
7	9.81e11	9.72e04	4.21e11	4.95e10	3.72e11	3.68e04	4.24e11	5.22e10	3.72e11	3.68e04
8	9.81e11	9.72e04	2.14e11	4.95e10	1.65e11	1.63e04	2.17e11	5.22e10	1.65e11	1.63e04
9	9.81e11	9.72e04	1.20e11	4.92e10	7.06e10	6.98e03	1.20e11	4.95e10	7.03e10	6.95e03
10	9.81e11	9.72e04	9.12e11	6.15e10	8.51e11	8.42e04	9.18e11	6.96e10	8.48e11	8.40e04
11	9.81e11	9.72e04	7.16e11	6.60e10	6.50e11	6.44e04	7.20e11	6.96e10	6.50e11	6.44e04
12	9.81e11	9.72e04	4.38e11	6.60e10	3.72e11	3.68e04	4.41e11	6.96e10	3.72e11	3.68e04
13	9.81e11	9.72e04	2.31e11	6.60e10	1.65e11	1.63e04	2.34e11	6.96e10	1.65e11	1.63e04
14	9.81e11	9.72e04	1.36e11	6.56e10	7.06e10	6.98e03	1.36e11	6.56e10	7.06e10	6.98e03
15	9.81e11	9.72e04	9.27e11	7.32e10	8.54e11	8.45e04	9.35e11	8.71e10	8.48e11	8.40e04
16	9.81e11	9.72e04	7.33e11	8.25e10	6.50e11	6.44e04	7.37e11	8.71e10	6.50e11	6.44e04
17	9.81e11	9.72e04	4.54e11	8.25e10	3.72e11	3.68e04	4.59e11	8.71e10	3.72e11	3.68e04
18	9.81e11	9.72e04	2.47e11	8.25e10	1.65e11	1.63e04	2.52e11	8.71e10	1.65e11	1.63e04
19	9.81e11	9.72e04	1.52e11	8.04e10	7.20e10	7.12e03	1.53e11	8.20e10	7.06e10	6.98e03
20	9.81e11	9.72e04	9.42e11	8.78e10	8.54e11	8.45e04	9.60e11	9.25e10	8.68e11	8.59e04
21	9.81e11	9.72e04	7.49e11	9.83e10	6.51e11	6.44e04	7.55e11	1.04e11	6.50e11	6.44e04
22	9.81e11	9.72e04	4.71e11	9.90e10	3.72e11	3.68e04	4.76e11	1.04e11	3.72e11	3.68e04
23	9.81e11	9.72e04	2.64e11	9.90e10	1.65e11	1.63e04	2.69e11	1.04e11	1.65e11	1.63e04
24	9.81e11	9.72e04	1.68e11	9.52e10	7.32e10	7.23e03	1.69e11	9.83e10	7.06e10	6.98e03
25	1.00e12	9.94e04	9.92e11	1.86e10	9.73e11	9.64e04	1.00e12	4.52e02	1.00e12	9.94e04
26	1.00e12	9.94e04	9.59e11	3.32e10	9.26e11	9.17e04	1.00e12	4.52e02	1.00e12	9.94e04
27	1.00e12	9.94e04	9.01e11	5.35e10	8.48e11	8.40e04	9.24e11	8.98e10	8.34e11	8.26e04
28	1.00e12	9.94e04	7.93e11	8.32e10	7.10e11	7.04e04	7.98e11	8.98e10	7.08e11	7.01e04
29	1.00e12	9.94e04	5.62e11	8.78e10	4.75e11	4.70e04	5.64e11	8.98e10	4.75e11	4.70e04
30	1.00e12	9.94e04	1.00e12	1.90e10	9.81e11	9.71e04	1.00e12	1.64e01	1.00e12	9.94e04
31	1.00e12	9.94e04	9.74e11	4.13e10	9.33e11	9.24e04	1.00e12	1.64e01	1.00e12	9.94e04
32	1.00e12	9.94e04	9.26e11	6.71e10	8.59e11	8.51e04	9.49e11	4.10e10	9.08e11	9.00e04
33	1.00e12	9.94e04	8.34e11	1.20e11	7.14e11	7.07e04	8.43e11	1.35e11	7.08e11	7.01e04
34	1.00e12	9.94e04	6.06e11	1.31e11	4.75e11	4.71e04	6.09e11	1.35e11	4.75e11	4.70e04
35	1.00e12	9.94e04	1.00e12	9.00e08	1.00e12	9.93e04	1.00e12	5.70e02	1.00e12	9.94e04
36	1.00e12	9.94e04	9.86e11	4.52e10	9.41e11	9.32e04	1.00e12	5.70e02	1.00e12	9.94e04
37	1.00e12	9.94e04	9.46e11	7.72e10	8.69e11	8.61e04	9.64e11	3.62e10	9.28e11	9.19e04
38	1.00e12	9.94e04	8.70e11	1.34e11	7.37e11	7.30e04	8.87e11	1.80e11	7.08e11	7.01e04
39	1.00e12	9.94e04	6.50e11	1.73e11	4.76e11	4.72e04	6.54e11	1.80e11	4.75e11	4.70e04
40	1.00e12	9.94e04	1.00e12	3.75e08	1.00e12	9.94e04	1.00e12	8.26e02	1.00e12	9.94e04
41	1.00e12	9.94e04	9.96e11	3.87e10	9.57e11	9.48e04	1.00e12	8.26e02	1.00e12	9.94e04
42	1.00e12	9.94e04	9.63e11	7.61e10	8.87e11	8.79e04	9.73e11	4.48e10	9.29e11	9.20e04
43	1.00e12	9.94e04	9.02e11	1.47e11	7.55e11	7.47e04	9.60e11	1.99e11	7.61e11	7.54e04
44	1.00e12	9.94e04	6.93e11	2.16e11	4.77e11	4.73e04	6.99e11	2.24e11	4.75e11	4.70e04
45	1.00e12	9.94e04	1.00e12	0.00e00	1.00e12	9.94e04	1.00e12	0.00e00	1.00e12	9.94e04
46	1.00e12	9.94e04	1.00e12	3.12e10	9.71e11	9.62e04	1.00e12	0.00e00	1.00e12	9.94e04
47	1.00e12	9.94e04	9.78e11	7.69e10	9.01e11	8.92e04	9.83e11	4.91e10	9.34e11	9.25e04
48	1.00e12	9.94e04	9.29e11	1.53e11	7.77e11	7.69e04	1.00e12	2.39e11	7.61e11	7.54e04
49	1.00e12	9.94e04	7.36e11	2.58e11	4.79e11	4.74e04	7.44e11	2.69e11	4.75e11	4.70e04
50	1.00e12	9.94e04	9.92e11	1.86e10	9.73e11	9.64e04	1.00e12	2.97e01	1.00e12	9.94e04
51	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	1.00e12	2.97e01	1.00e12	9.94e04
52	1.00e12	9.94e04	9.04e11	5.18e10	8.52e11	8.44e04	9.18e11	4.09e10	8.77e11	8.69e04
53	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	8.55e11	4.91e10	8.06e11	7.98e04
54	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	8.06e11	1.92e11	6.14e11	6.08e04
55	1.00e12	9.94e04	1.00e12	1.90e10	9.81e11	9.71e04	1.00e12	0.00e00	1.00e12	9.94e04
56	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	1.00e12	0.00e00	1.00e12	9.94e04
57	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	9.30e11	7.67e10	8.53e11	8.45e04
58	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	8.80e11	6.99e10	8.10e11	8.02e04
59	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	8.83e11	9.22e10	7.91e11	7.83e04
60	1.00e12	9.94e04	1.00e12	1.23e10	9.92e11	9.82e04	1.00e12	2.91e00	1.00e12	9.94e04
61	1.00e12	9.94e04	9.87e11	4.33e10	9.43e11	9.34e04	1.00e12	2.91e00	1.00e12	9.94e04
62	1.00e12	9.94e04	9.47e11	7.27e10	8.75e11	8.66e04	1.00e12	2.91e00	1.00e12	9.94e04
63	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	9.11e11	6.97e10	8.41e11	8.33e04
64	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	8.77e11	8.55e10	7.91e11	7.84e04
65	1.00e12	9.94e04	1.00e12	3.75e08	1.00e12	9.94e04	1.00e12	0.00e00	1.00e12	9.94e04
66	1.00e12	9.94e04	9.96e11	3.87e10	9.57e11	9.48e04	1.00e12	0.00e00	1.00e12	9.94e04
67	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	1.00e12	0.00e00	1.00e12	9.94e04
68	1.00e12	9.94e04	9.22e11	1.03e11	8.19e11	8.11e04	9.34e11	1.37e11	7.97e11	7.89e04
69	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	9.04e11	1.11e11	7.93e11	7.86e04
70	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	1.00e12	3.07e01	1.00e12	9.94e04
71	1.00e12	9.94e04	1.00e12	2.99e10	9.73e11	9.64e04	1.00e12	3.07e01	1.00e12	9.94e04
72	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	1.00e12	3.07e01	1.00e12	9.94e04
73	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	1.00e12	3.07e01	1.00e12	9.94e04
74	1.00e12	9.94e04	1.00e07	0.00e00	1.00e07	0.00e00	9.24e11	7.18e10	8.53e11	8.45e04

(d) *System outcome results*

3.2 Lagrangian Subproblem Quasiconvexity

The subproblem $L2(\lambda)$ defined in Section 2.1 has no integer constraints, but it still takes considerable time to solve with the BARON numerical solver. If the problem has suitable structure, such as quasiconvexity, a basic gradient descent algorithm might suffice for part of its solution. In particular, we examined whether $L2(\lambda)$ is quasiconvex in $P_t, t = 1, \ldots, T$.

A function $f(\boldsymbol{x})$ is quasiconvex if and only if for any two values $\boldsymbol{x}_1$, and $\boldsymbol{x}_2$ in its domain, the function of one variable $V(\theta) = f((1-\theta)\boldsymbol{x}_1 + \theta\boldsymbol{x}_2)$ is quasiconvex for values of $\theta \in [0,1]$.

To probe this necessary and sufficient condition for quasiconvexity in P, we define the function $V_{P^1,P^2} : [0,1] \to \mathbb{R}$ for any values P^1, P^2 where P^1_t and P^2_t are fixed for all $t = 1, \ldots, T$. This function is such that $V_{P^1,P^2}(\theta)$ is the optimal value of $L2(\lambda)$ where $P_t = (1-\theta)P^1_t + \theta P^2_t$ for all $t = 1, \ldots, T$.

The following plots illustrate the value of $V_{P^1,P^2}(\theta)$ for $\theta \in [0,1]$. If the curves appear quasiconvex, then a necessary condition is met for $L2(\lambda)$ being quasiconvex in P. In each trial, the values $P^1_t, P^2_t, t = 1, \ldots, T$ were generated in the interval $[0,1]$ uniformly randomly[1].

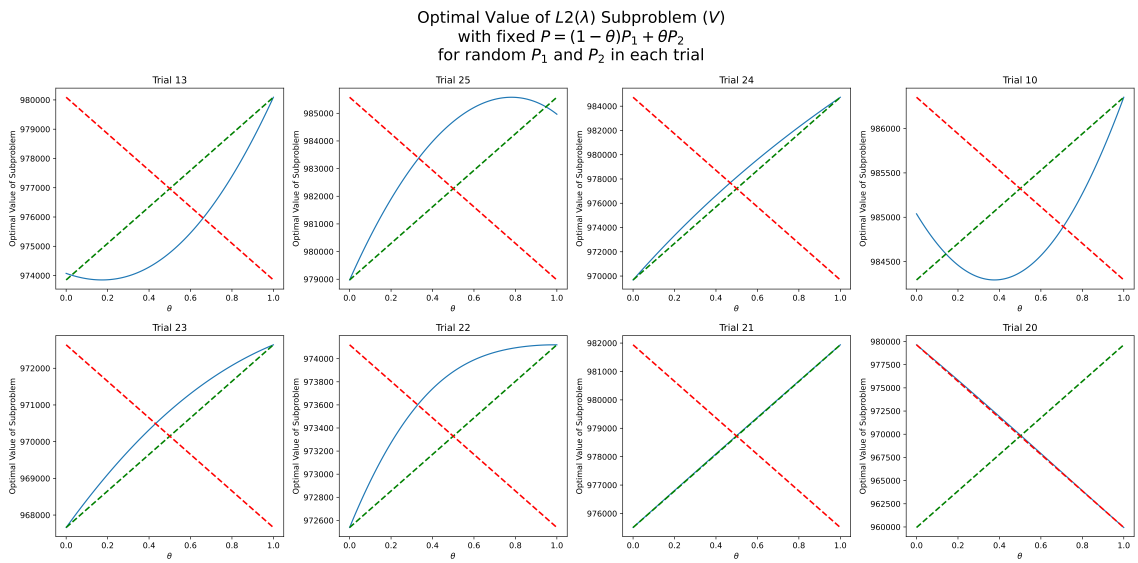

Figure 2: *Optimal value of $L2(\lambda)$ with P fixed to values that vary along a line; that is, $V_{P^1,P^2}(\theta)$ vs θ for several randomly-selected values of P^1 and P^2.*

As can be seen in Figure 2, the function V appears to be neither quasiconvex nor quasiconcave.

To probe the possibility that the function V is not jointly quasiconvex in the P_t variables, but is componentwise quasiconvex in each $P_t, t = 1, \ldots, P$, we can investigate whether the function $V_{P^1,P^2}(\theta)$ is quasiconvex for any P^1, P^2 such that the line segment connecting P^1 and P^2 is parallel to an axis P_t (for some t), i.e. by repeating the same experiment but only varying one P_t at a time.

[1] In fact, for this model P_t cannot be equal to 0, because of its definition in the full model as a product of "intervention effectiveness" factors. So, the lowest value it can possibly take on is the product of the effectiveness factors of *all* possible interventions. Both in the full model and in this investigation of quasiconvexity, values of P_t are constrained to be in the interval $[P_{lb}, 1]$, where P_{lb} is this small value. Since the logarithm of P_t is part of the model, the domain becomes effectively open (the objective is undefined at 0), this adjustment improves performance of the solver by giving a closed domain without loss of generality of solutions. The restriction of the domain for this quasiconvexity test is also without loss of generality, because for the purposes of this model, only values of P_t in the constrained interval are relevant.

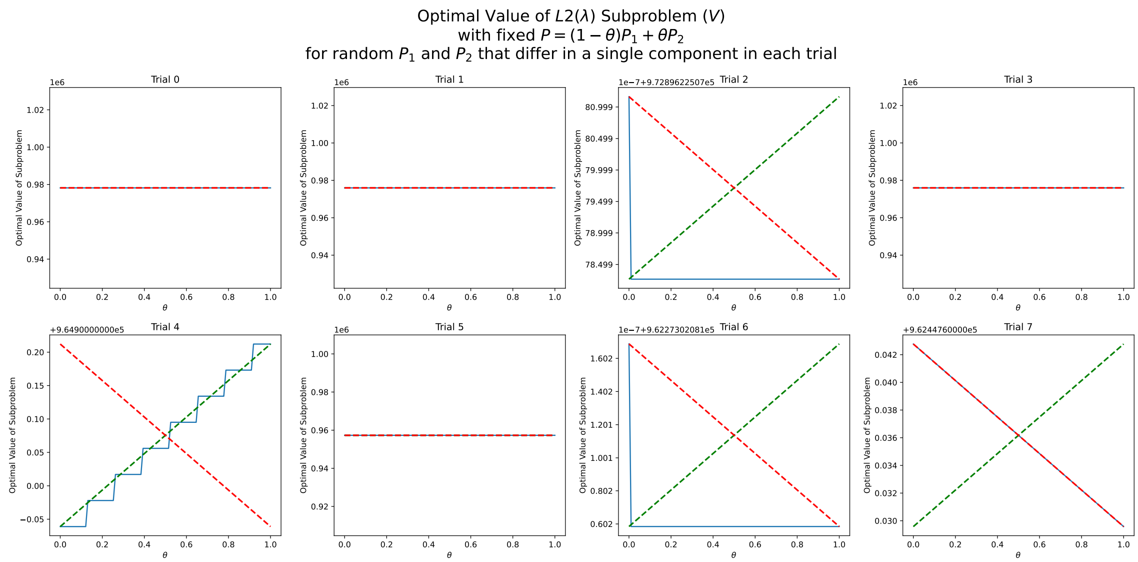

Figure 3: *The same experiment was repeated as is illustrated in Figure 2, but by only varying a single P_t component at a time.*

Based on the plots in Figure 3, it is inconclusive whether V is componentwise quasiconcave or quasiconvex. The variation in the objective of the $L2(\lambda)$ subproblem may be small with respect to any one component P_t.

4 Conclusion

References