On Solving Bilevel Programming Programs

Jane J. Ye

Department of Mathematics and Statistics
University of Victoria, Canada

One World Optimization Seminar
May 3, 2021

Outline

- Introduction to bilevel programs (background and applications)
- Reformulations as single level optimization problems
- Optimality conditions: directional optimality conditions for bilevel programs
- Numerical algorithms: difference of convex algorithms for bilevel programs

Bilevel programs

$$(BP) \quad \min_{x,y} \quad F(x,y)$$
$$s.t. \quad y \in S(x)$$

where $S(x)$ denotes the set of solutions of the lower level problem:

$$(P_x) \quad \min_{y \in Y(x)} \quad f(x,y)$$

- For simplicity we have omitted upper level constraints.
- Usually
$$Y(x) := \{y | g(x,y) \leq 0\},$$
and all functions F and g are smooth.

Suppose that for each x, the lower level problem (P_x) has a unique solution $y(x)$. Then by substituting $y(x)$ into the upper level, the bilevel program becomes an one-level optimization problem

$$\min_x F(x, y(x)).$$

If $y(x)$ is a "nice" function of x, then perhaps the above problem can be solved.

But if the lower level problem has multiple solutions, then there are two versions of the bilevel program: optimistic and pessimistic.

- Optimistic: $\min_{x,y}\{F(x,y) : y \in S(x)\}$.
- Pessimistic: $\min_x \max_{y \in S(x)} F(x,y)$.

In this talk we only deal with the optimistic case.

Applications in economics

- The first formulation of a simplier case of the bilevel program was introduced by Stackelberg (1934). Hence it is known as a Stackelberg game in economic game theory.
- The classical principal-agent/moral hazard problem in economics is a bilevel program: This is the situation where the principal can only observe the outcome of the agent's action but not the action itself. How can the principal design a contract in order to maximize the expected utility subject to the optimizing behavior of the agent?
- Nobel prize has been awarded twice for study of the moral hazard problem. Vickrey and Mirrlees shared the 1996 Nobel prize in economics which was awarded for their fundamental contributions to the economic theory of incentives under asymmetric information. Holmström and Hart shared the 2016 Nobel prize in economics which was awarded for their fundamental contributions to contract theory.

Applications in economics

- The first formulation of a simplier case of the bilevel program was introduced by Stackelberg (1934). Hence it is known as a Stackelberg game in economic game theory.
- The classical principal-agent/moral hazard problem in economics is a bilevel program: This is the situation where the principal can only observe the outcome of the agent's action but not the action itself. How can the principal design a contract in order to maximize the expected utility subject to the optimizing behavior of the agent?
- Nobel prize has been awarded twice for study of the moral hazard problem. Vickrey and Mirrlees shared the 1996 Nobel prize in economics which was awarded for their fundamental contributions to the economic theory of incentives under asymmetric information. Holmström and Hart shared the 2016 Nobel prize in economics which was awarded for their fundamental contributions to contract theory.

Applications in economics

- The first formulation of a simplier case of the bilevel program was introduced by Stackelberg (1934). Hence it is known as a Stackelberg game in economic game theory.
- The classical principal-agent/moral hazard problem in economics is a bilevel program: This is the situation where the principal can only observe the outcome of the agent's action but not the action itself. How can the principal design a contract in order to maximize the expected utility subject to the optimizing behavior of the agent?
- Nobel prize has been awarded twice for study of the moral hazard problem. Vickrey and Mirrlees shared the 1996 Nobel prize in economics which was awarded for their fundamental contributions to the economic theory of incentives under asymmetric information. Holmström and Hart shared the 2016 Nobel prize in economics which was awarded for their fundamental contributions to contract theory.

Applications in machine learning

- The bilevel program was first introduced to the optimization community by Bracken and McGill (1973).
- It was first introduced to the model selection in machine learning by Bennett, Hu, Ji, Kunapuli and Pang in 2006.
- Recently there are more and more work on hyper-parameter learning via bilevel optimization:

$$\min_{\theta,\lambda} F(\theta)$$

$$\text{s.t. } \theta \in \arg\underbrace{\min_{\theta'} f(\theta') + \sum_{i=1}^{r} \lambda_i P_i(\theta')}_{\text{lower level training problem}},$$

where $P_i(\theta)$ are penalty functions.

Applications in machine learning

- The bilevel program was first introduced to the optimization community by Bracken and McGill (1973).
- It was first introduced to the model selection in machine learning by Bennett, Hu, Ji, Kunapuli and Pang in 2006.
- Recently there are more and more work on hyper-parameter learning via bilevel optimization:

$$\min_{\theta,\lambda} F(\theta)$$

$$\text{s.t. } \theta \in \arg\underbrace{\min_{\theta'} f(\theta') + \sum_{i=1}^{r} \lambda_i P_i(\theta')}_{\text{lower level training problem}},$$

where $P_i(\theta)$ are penalty functions.

Applications in machine learning

- The bilevel program was first introduced to the optimization community by Bracken and McGill (1973).
- It was first introduced to the model selection in machine learning by Bennett, Hu, Ji, Kunapuli and Pang in 2006.
- Recently there are more and more work on hyper-parameter learning via bilevel optimization:

$$\min_{\theta,\lambda} F(\theta)$$

$$\text{s.t. } \theta \in \arg \underbrace{\min_{\theta'} f(\theta') + \sum_{i=1}^{r} \lambda_i P_i(\theta')}_{\text{lower level training problem}},$$

where $P_i(\theta)$ are penalty functions.

Model selection

- Let $x \in \mathbb{R}^p$ and $y \in \mathbb{R}$ be predictor and the response variables, respectively. Suppose we have a data set containing n observations $\Omega := \{(x_1, y_1), \ldots, (x_n, y_n)\}$. We try to fit a statistical model to study the relationship between x and y.
- If $p \geq n$, i.e., the number of predictor variables are larger than the number of samples, the classical linear regression problem is ill-post. Some irrelevant variables may be included in the fitted model.
- Using lasso (Tibshirani 1996), for given $\lambda > 0$ the regularized problem is solved:

$$\min_{\theta} \sum_{(x_j, y_j) \in \Omega} (x_j^T \theta - y_j)^2 + \lambda \|\theta\|_1.$$

  Bigger λ encourage sparser optimal solution $\hat{\theta}$. But how to select λ so that the model is correct?
- Since $n \leq p$, we can not afford to leave out some observations for testing the results.

Model selection

- Let $x \in \mathbb{R}^p$ and $y \in \mathbb{R}$ be predictor and the response variables, respectively. Suppose we have a data set containing n observations $\Omega := \{(x_1, y_1), \ldots, (x_n, y_n)\}$. We try to fit a statistical model to study the relationship between x and y.
- If $p \geq n$, i.e., the number of predictor variables are larger than the number of samples, the classical linear regression problem is ill-post. Some irrelevant variables may be included in the fitted model.
- Using lasso (Tibshirani 1996), for given $\lambda > 0$ the regularized problem is solved:

$$\min_{\theta} \sum_{(x_j, y_j) \in \Omega} (x_j^T \theta - y_j)^2 + \lambda \|\theta\|_1.$$

Bigger λ encourage sparser optimal solution $\hat{\theta}$. But how to select λ so that the model is correct?

- Since $n \leq p$, we can not afford to leave out some observations for testing the results.

Model selection

- Let $x \in \mathbb{R}^p$ and $y \in \mathbb{R}$ be predictor and the response variables, respectively. Suppose we have a data set containing n observations $\Omega := \{(x_1, y_1), \ldots, (x_n, y_n)\}$. We try to fit a statistical model to study the relationship between x and y.
- If $p \geq n$, i.e., the number of predictor variables are larger than the number of samples, the classical linear regression problem is ill-post. Some irrelevant variables may be included in the fitted model.
- Using lasso (Tibshirani 1996), for given $\lambda > 0$ the regularized problem is solved:

$$\min_{\theta} \sum_{(x_j, y_j) \in \Omega} (x_j^T \theta - y_j)^2 + \lambda \|\theta\|_1.$$

  Bigger λ encourage sparser optimal solution $\hat{\theta}$. But how to select λ so that the model is correct?
- Since $n \leq p$, we can not afford to leave out some observations for testing the results.

Model selection

- Let $x \in \mathbb{R}^p$ and $y \in \mathbb{R}$ be predictor and the response variables, respectively. Suppose we have a data set containing n observations $\Omega := \{(x_1, y_1), \ldots, (x_n, y_n)\}$. We try to fit a statistical model to study the relationship between x and y.
- If $p \geq n$, i.e., the number of predictor variables are larger than the number of samples, the classical linear regression problem is ill-post. Some irrelevant variables may be included in the fitted model.
- Using lasso (Tibshirani 1996), for given $\lambda > 0$ the regularized problem is solved:

$$\min_{\theta} \sum_{(x_j, y_j) \in \Omega} (x_j^T \theta - y_j)^2 + \lambda \|\theta\|_1.$$

  Bigger λ encourage sparser optimal solution $\hat{\theta}$. But how to select λ so that the model is correct?
- Since $n \leq p$, we can not afford to leave out some observations for testing the results.

K-fold cross validation

Given a data set: $\Omega = \{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^{p+1}$.
Step 1: Randomly split the data set into K disjoint blocks with approximately equal size:

$$\Omega = \Omega_1 \cup \cdots \cup \Omega_K.$$

Step 2: For $k = 1, \ldots, K$, use Ω_k as the test set and the rest $(K-1)$ blocks as the training set Ω_{trn}^k, and compute the fitted values θ_k.
Step 3, Compute the mean-squared-error on the observations in Ω_k, i.e., $\mathrm{MSE}(\theta_k) = \sum_{(x_j,y_j)\in\Omega_k}(x_j^T\theta_k - y_j)^2$, and compute the cross validation error

$$CV(\theta_1, \ldots, \theta_K) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{MSE}(\theta_k).$$

Step 4. Repeat Steps 2 and 3 for various values of $\lambda > 0$.
Step 5. Find λ^* that minimize the cross validation error and in the mean time θ^* the best fitted value.

K-fold cross validation

Given a data set: $\Omega = \{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^{p+1}$.
Step 1: Randomly split the data set into K disjoint blocks with approximately equal size:

$$\Omega = \Omega_1 \cup \cdots \cup \Omega_K.$$

Step 2: For $k = 1, \ldots, K$, use Ω_k as the test set and the rest $(K-1)$ blocks as the training set Ω_{trn}^k, and compute the fitted values θ_k.

Step 3, Compute the mean-squared-error on the observations in Ω_k, i.e., $\mathrm{MSE}(\theta_k) = \sum_{(x_j,y_j)\in\Omega_k}(x_j^T\theta_k - y_j)^2$, and compute the cross validation error

$$CV(\theta_1, \ldots, \theta_K) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{MSE}(\theta_k).$$

Step 4. Repeat Steps 2 and 3 for various values of $\lambda > 0$.
Step 5. Find λ^* that minimize the cross validation error and in the mean time θ^* the best fitted value.

K-fold cross validation

Given a data set: $\Omega = \{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^{p+1}$.
Step 1: Randomly split the data set into K disjoint blocks with approximately equal size:

$$\Omega = \Omega_1 \cup \cdots \cup \Omega_K.$$

Step 2: For $k = 1, \ldots, K$, use Ω_k as the test set and the rest $(K-1)$ blocks as the training set Ω_{trn}^k, and compute the fitted values θ_k.
Step 3, Compute the mean-squared-error on the observations in Ω_k, i.e., $\mathrm{MSE}(\theta_k) = \sum_{(x_j,y_j)\in\Omega_k}(x_j^T\theta_k - y_j)^2$, and compute the cross validation error

$$CV(\theta_1, \ldots, \theta_K) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{MSE}(\theta_k).$$

Step 4. Repeat Steps 2 and 3 for various values of $\lambda > 0$.
Step 5. Find λ^* that minimize the cross validation error and in the mean time θ^* the best fitted value.

K-fold cross validation

Given a data set: $\Omega = \{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^{p+1}$.
Step 1: Randomly split the data set into K disjoint blocks with approximately equal size:

$$\Omega = \Omega_1 \cup \cdots \cup \Omega_K.$$

Step 2: For $k = 1, \ldots, K$, use Ω_k as the test set and the rest $(K-1)$ blocks as the training set Ω_{trn}^k, and compute the fitted values θ_k.
Step 3, Compute the mean-squared-error on the observations in Ω_k, i.e., $\mathrm{MSE}(\theta_k) = \sum_{(x_j,y_j)\in\Omega_k}(x_j^T\theta_k - y_j)^2$, and compute the cross validation error

$$CV(\theta_1, \ldots, \theta_K) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{MSE}(\theta_k).$$

Step 4. Repeat Steps 2 and 3 for various values of $\lambda > 0$.
Step 5. Find λ^* that minimize the cross validation error and in the mean time θ^* the best fitted value.

K-fold cross validation

Given a data set: $\Omega = \{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^{p+1}$.
Step 1: Randomly split the data set into K disjoint blocks with approximately equal size:

$$\Omega = \Omega_1 \cup \cdots \cup \Omega_K.$$

Step 2: For $k = 1, \ldots, K$, use Ω_k as the test set and the rest $(K-1)$ blocks as the training set Ω_{trn}^k, and compute the fitted values θ_k.
Step 3, Compute the mean-squared-error on the observations in Ω_k, i.e., $\mathrm{MSE}(\theta_k) = \sum_{(x_j,y_j)\in\Omega_k}(x_j^T\theta_k - y_j)^2$, and compute the cross validation error

$$CV(\theta_1, \ldots, \theta_K) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{MSE}(\theta_k).$$

Step 4. Repeat Steps 2 and 3 for various values of $\lambda > 0$.
Step 5. Find λ^* that minimize the cross validation error and in the mean time θ^* the best fitted value.

Cross validation as a bilevel program

- In statistics, either a grid search or a path following algorithm is performed on λ values to select the value of λ for which the cross-validation error is smallest. But these approaches do not scale well and have a lot of limitations.
- In essence the cross validation in lasso is the following bilevel program:

$$\min_{\lambda,\theta_1,\ldots,\theta_K} \quad CV(\theta_1,\ldots,\theta_K)$$

$$\lambda > 0 \text{ and for each } k = 1,\ldots,K$$

$$\theta_k \in \arg\min_{\theta} \sum_{(x_j,y_j)\in\Omega^k_{trn}} (x_j^T\theta - y_j)^2 + \lambda\|\theta\|_1$$

- If the above bilevel program can be solved, then we can obtain the optimal penalty parameter λ^* and the best fitted value θ^* at once!

Cross validation as a bilevel program

- In statistics, either a grid search or a path following algorithm is performed on λ values to select the value of λ for which the cross-validation error is smallest. But these approaches do not scale well and have a lot of limitations.
- In essence the cross validation in lasso is the following bilevel program:

$$\min_{\lambda,\theta_1,\ldots,\theta_K} \quad CV(\theta_1,\ldots,\theta_K)$$

$$\lambda > 0 \text{ and for each } k = 1,\ldots,K$$

$$\theta_k \in \arg\min_{\theta} \sum_{(x_j,y_j)\in\Omega^k_{trn}} (x_j^T\theta - y_j)^2 + \lambda\|\theta\|_1$$

- If the above bilevel program can be solved, then we can obtain the optimal penalty parameter λ^* and the best fitted value θ^* at once!

Cross validation as a bilevel program

- In statistics, either a grid search or a path following algorithm is performed on λ values to select the value of λ for which the cross-validation error is smallest. But these approaches do not scale well and have a lot of limitations.
- In essence the cross validation in lasso is the following bilevel program:

$$\min_{\lambda,\theta_1,\ldots,\theta_K} \quad CV(\theta_1,\ldots,\theta_K)$$

$$\lambda > 0 \text{ and for each } k = 1,\ldots,K$$

$$\theta_k \in \arg\min_{\theta} \sum_{(x_j,y_j)\in\Omega^k_{trn}} (x_j^T\theta - y_j)^2 + \lambda\|\theta\|_1$$

- If the above bilevel program can be solved, then we can obtain the optimal penalty parameter λ^* and the best fitted value θ^* at once!

The first order approach (FOA, KKT/MPEC Approach)

Basic features of FOA:

- replace the lower level problem by its KKT conditions;
- minimize over the original variables as well as multipliers;
- the resulting problem is the mathematical program with complementarity/equilibrium constraints (MPCC/MPEC)

$$\begin{aligned} \min_{x,y,u} \ & F(x,y) \\ \text{s.t.} \ & \nabla_y f(x,y) + u\nabla_y g(x,y) = 0, \\ & g(x,y) \le 0, \ u \ge 0, \ u^T g(x,y) = 0. \end{aligned}$$

Drawback of FOA:

- The true optimal solution of the bilevel program may not be recovered by solving the corresponding MPEC, cf. Mirrlees (1999).
- Even when the lower level problem is convex, a local optimal solution of MPEC may not be a local optimal solution of BP if the multiplier is non-unique, cf. Dempe and Dutta 2012.

The first order approach (FOA, KKT/MPEC Approach)

Basic features of FOA:

- replace the lower level problem by its KKT conditions;
- minimize over the original variables as well as multipliers;
- the resulting problem is the mathematical program with complementarity/equilibrium constraints (MPCC/MPEC)

$$\begin{aligned} \min_{x,y,u} \ & F(x,y) \\ \text{s.t.} \ & \nabla_y f(x,y) + u\nabla_y g(x,y) = 0, \\ & g(x,y) \le 0, \ u \ge 0, \ u^T g(x,y) = 0. \end{aligned}$$

Drawback of FOA:

- The true optimal solution of the bilevel program may not be recovered by solving the corresponding MPEC, cf. Mirrlees (1999).
- Even when the lower level problem is convex, a local optimal solution of MPEC may not be a local optimal solution of BP if the multiplier is non-unique, cf. Dempe and Dutta 2012.

- If the lower level problem is convex in y, then the bilevel program is equivalent to

$$\begin{array}{ll} \min\limits_{x,y} & F(x,y) \\ s.t. & 0 \in \nabla_y f(x,y) + N_{Y(x)}(y). \end{array}$$

- Some researches have been done for the case $Y(x) = Y$.
- Compared with MPCC reformulation, it is easier for this reformulation to satisfy constraint qualifications; cf. Adam, Henrion and Outrata, 2018. Based on this reformulation, some new sharp necessary optimality conditions have been derived in Gfrerer and JY: "New constraint qualifications for mathematical programs with equilibrium constraints via variational analysis" (2017) and "New sharp necessary optimality conditions for mathematical programs with equilibrium constraints" (2020).

- If the lower level problem is convex in y, then the bilevel program is equivalent to

$$\begin{aligned} \min_{x,y} \quad & F(x,y) \\ s.t. \quad & 0 \in \nabla_y f(x,y) + N_{Y(x)}(y). \end{aligned}$$

- Some researches have been done for the case $Y(x) = Y$.
- Compared with MPCC reformulation, it is easier for this reformulation to satisfy constraint qualifications; cf. Adam, Henrion and Outrata, 2018. Based on this reformulation, some new sharp necessary optimality conditions have been derived in Gfrerer and JY: "New constraint qualifications for mathematical programs with equilibrium constraints via variational analysis" (2017) and "New sharp necessary optimality conditions for mathematical programs with equilibrium constraints" (2020).

- If the lower level problem is convex in y, then the bilevel program is equivalent to

$$\begin{aligned} \min_{x,y} \quad & F(x,y) \\ s.t. \quad & 0 \in \nabla_y f(x,y) + N_{Y(x)}(y). \end{aligned}$$

- Some researches have been done for the case $Y(x) = Y$.
- Compared with MPCC reformulation, it is easier for this reformulation to satisfy constraint qualifications; cf. Adam, Henrion and Outrata, 2018. Based on this reformulation, some new sharp necessary optimality conditions have been derived in Gfrerer and JY: "New constraint qualifications for mathematical programs with equilibrium constraints via variational analysis" (2017) and "New sharp necessary optimality conditions for mathematical programs with equilibrium constraints" (2020).

The value function approach

Basic features of the value function approach:

- replace the original BP by the following equivalent problem (Outrata (1990), JY and Zhu (1995)):

$$(VP) \qquad \min_{x,y} F(x,y)$$
$$\text{s.t. } f(x,y) - v(x) \leq 0, \ g(x,y) \leq 0.$$

  where $v(x) := \inf_y \{f(x,y) : g(x,y) \leq 0\}$ is the value function.

- can deal with BPs without convexity assumption on (P_x).

Drawback of the value function approach:

- The resulting stationary condition based on the value function approach may be too strong.

The value function approach

Basic features of the value function approach:

- replace the original BP by the following equivalent problem (Outrata (1990), JY and Zhu (1995)):

$$(VP) \qquad \min_{x,y} F(x,y)$$
$$\text{s.t. } f(x,y) - v(x) \leq 0, \ g(x,y) \leq 0.$$

  where $v(x) := \inf_y \{f(x,y) : g(x,y) \leq 0\}$ is the value function.

- can deal with BPs without convexity assumption on (P_x).

Drawback of the value function approach:

- The resulting stationary condition based on the value function approach may be too strong.

The combined approach

Basic features of the combined approach:

- replace the lower level problem by both the value function constraint and some necessary optimality condition of the lower level program.
- Combined program with KKT condition (JY-Zhu (2010)):

$$
\begin{aligned}
(CP) \qquad & \min_{x,y,u} F(x,y) \\
& \text{s.t. } f(x,y) - v(x) \le 0, \\
& \quad \nabla_y f(x,y) + u\nabla_y g(x,y) = 0, \\
& \quad g(x,y) \le 0,\ u \ge 0,\ u^T g(x,y) = 0.
\end{aligned}
$$

- Combined program with Fritz John (FJ) condition or Bouligand (B)-condition (Ke, Yao, JY and Zhang, 2021).
- It is easier for the resulting stationary condition to hold than the one based on the value function approach.

The combined approach

Basic features of the combined approach:

- replace the lower level problem by both the value function constraint and some necessary optimality condition of the lower level program.
- Combined program with KKT condition (JY-Zhu (2010)):

$$
\begin{aligned}
(CP) \qquad & \min_{x,y,u} F(x,y) \\
& \text{s.t. } f(x,y) - v(x) \le 0, \\
& \quad \nabla_y f(x,y) + u\nabla_y g(x,y) = 0, \\
& \quad g(x,y) \le 0,\ u \ge 0,\ u^T g(x,y) = 0.
\end{aligned}
$$

- Combined program with Fritz John (FJ) condition or Bouligand (B)-condition (Ke, Yao, JY and Zhang, 2021).
- It is easier for the resulting stationary condition to hold than the one based on the value function approach.

The combined approach

Basic features of the combined approach:

- replace the lower level problem by both the value function constraint and some necessary optimality condition of the lower level program.
- Combined program with KKT condition (JY-Zhu (2010)):

$$
\begin{aligned}
(CP) \qquad & \min_{x,y,u} F(x,y) \\
& \text{s.t. } f(x,y) - v(x) \le 0, \\
& \quad \nabla_y f(x,y) + u \nabla_y g(x,y) = 0, \\
& \quad g(x,y) \le 0, \ u \ge 0, \ u^T g(x,y) = 0.
\end{aligned}
$$

- Combined program with Fritz John (FJ) condition or Bouligand (B)-condition (Ke, Yao, JY and Zhang, 2021).
- It is easier for the resulting stationary condition to hold than the one based on the value function approach.

Due to the existence of the value function constraint $f(x, y) - v(x) \leq 0$,

Mangasarian-Fromovitz constraint qualification (MFCQ) fails for (VP) and (CP)!

Partial calmness condition

Basic features of the partial calmness condition:

- the partial calmness condition allows one to partially penalize the value function constraint $f(x,y) - v(x) \leq 0$ to the objective function. Consequently, the usual constraint qualifications can be applied to the rest of the constraints.
- proposed for the value function reformulation (JY-Zhu (1995), the combined program with KKT condition (JY-Zhu (2010)), and the combined program with FJ condition and B-condition (Ke-Yao-JY-Zhang (2021)).

How stringent is the partial calmness condition?

- Recently in Ke-Yao-JY-Zhang (2021), we have shown that at least for the case where x is one-dimensional, the partial calmness for the combined program is a generic condition while the one for the value function reformulation is not.

Partial calmness condition

Basic features of the partial calmness condition:

- the partial calmness condition allows one to partially penalize the value function constraint $f(x,y) - v(x) \leq 0$ to the objective function. Consequently, the usual constraint qualifications can be applied to the rest of the constraints.
- proposed for the value function reformulation (JY-Zhu (1995), the combined program with KKT condition (JY-Zhu (2010)), and the combined program with FJ condition and B-condition (Ke-Yao-JY-Zhang (2021)).

How stringent is the partial calmness condition?

- Recently in Ke-Yao-JY-Zhang (2021), we have shown that at least for the case where x is one-dimensional, the partial calmness for the combined program is a generic condition while the one for the value function reformulation is not.

Semi-infinite programming reformulation

$$y \in S(x) \Longleftrightarrow g(x,y) \leq 0 \text{ and } f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x)$$

- When all functions are polynomials and KKT condition holds at each $y \in S(x)$, we can find a multiplier of the lower level problem as a polynomial or rational function of (x,y), denoted by $\lambda(x,y)$.
- The bilevel program is equivalent to the generalized SIP:

$$\begin{aligned}(SIP) \qquad \min_{x,y} \; & F(x,y) \\ \text{s.t.} \; & f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x), \\ & \nabla_y f(x,y) + \lambda(x,y) \nabla_y g(x,y) = 0, \\ & g(x,y) \leq 0, \; \lambda(x,y) \geq 0, \; \lambda(x,y)^T g(x,y) = 0.\end{aligned}$$

- Based on this reformulation recently we have proposed a numerical algorithm to globally solve the polynomial bilevel program in Nie, Wang, JY and Zhong, A Lagrange Multiplier Expression Method for Bilevel Polynomial Optimization, arXiv (2007.07933).

Semi-infinite programming reformulation

$$y \in S(x) \Longleftrightarrow g(x,y) \leq 0 \text{ and } f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x)$$

• When all functions are polynomials and KKT condition holds at each $y \in S(x)$, we can find a multiplier of the lower level problem as a polynomial or rational function of (x,y), denoted by $\lambda(x,y)$.

• The bilevel program is equivalent to the generalized SIP:

$$\begin{aligned}(SIP) \qquad \min_{x,y} \; & F(x,y) \\ \text{s.t.} \; & f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x), \\ & \nabla_y f(x,y) + \lambda(x,y) \nabla_y g(x,y) = 0, \\ & g(x,y) \leq 0, \; \lambda(x,y) \geq 0, \; \lambda(x,y)^T g(x,y) = 0.\end{aligned}$$

• Based on this reformulation recently we have proposed a numerical algorithm to globally solve the polynomial bilevel program in Nie, Wang, JY and Zhong, A Lagrange Multiplier Expression Method for Bilevel Polynomial Optimization, arXiv (2007.07933).

Semi-infinite programming reformulation

$$y \in S(x) \Longleftrightarrow g(x,y) \leq 0 \text{ and } f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x)$$

- When all functions are polynomials and KKT condition holds at each $y \in S(x)$, we can find a multiplier of the lower level problem as a polynomial or rational function of (x,y), denoted by $\lambda(x,y)$.
- The bilevel program is equivalent to the generalized SIP:

$$\begin{aligned}(SIP) \qquad \min_{x,y} \; & F(x,y) \\ \text{s.t. } & f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x), \\ & \nabla_y f(x,y) + \lambda(x,y)\nabla_y g(x,y) = 0, \\ & g(x,y) \leq 0, \; \lambda(x,y) \geq 0, \; \lambda(x,y)^T g(x,y) = 0.\end{aligned}$$

- Based on this reformulation recently we have proposed a numerical algorithm to globally solve the polynomial bilevel program in Nie, Wang, JY and Zhong, A Lagrange Multiplier Expression Method for Bilevel Polynomial Optimization, arXiv (2007.07933).

Semi-infinite programming reformulation

$$y \in S(x) \Longleftrightarrow g(x,y) \leq 0 \text{ and } f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x)$$

- When all functions are polynomials and KKT condition holds at each $y \in S(x)$, we can find a multiplier of the lower level problem as a polynomial or rational function of (x,y), denoted by $\lambda(x,y)$.
- The bilevel program is equivalent to the generalized SIP:

$$\begin{aligned}(SIP) \qquad \min_{x,y} \; & F(x,y) \\ \text{s.t. } & f(x,z) - f(x,y) \geq 0 \quad \forall z \in Y(x), \\ & \nabla_y f(x,y) + \lambda(x,y) \nabla_y g(x,y) = 0, \\ & g(x,y) \leq 0, \; \lambda(x,y) \geq 0, \; \lambda(x,y)^T g(x,y) = 0.\end{aligned}$$

- Based on this reformulation recently we have proposed a numerical algorithm to globally solve the polynomial bilevel program in Nie, Wang, JY and Zhong, A Lagrange Multiplier Expression Method for Bilevel Polynomial Optimization, arXiv (2007.07933).

Directional Optimality Conditions for (VP)

• **Motivation:** The usual constraint qualification such as MFCQ does not hold for any of the reformulations of the bilevel program: JY-Zhu-Zhu (1997) for the MPEC approach, JY-Zhu (1995) for the value function reformulation, JY-Zhu (2010) for the combined program.

$$(VP) \quad \min_{x,y} F(x,y)$$
$$\text{s.t. } f(x,y) - v(x) \leq 0, g(x,y) \leq 0.$$

The FJ condition for (VP): $\exists (r, \lambda, \mu) \neq 0$ such that

$$0 \in r\nabla F(\bar{x},\bar{y}) + \lambda(\nabla f(\bar{x},\bar{y}) + \partial(-v)(\bar{x}) \times \{0\}) + \nabla g(\bar{x},\bar{y})^T \mu$$
$$r \geq 0, 0 \leq \mu \perp g(\bar{x},\bar{y}).$$

• If the limiting subdifferential $\partial(-v)(\bar{x})$ can be replaced by a smaller set, then the resulting FJ condition is sharper and the constraint qualification is then weaker.

Directional Optimality Conditions for (VP)

• **Motivation:** The usual constraint qualification such as MFCQ does not hold for any of the reformulations of the bilevel program: JY-Zhu-Zhu (1997) for the MPEC approach, JY-Zhu (1995) for the value function reformulation, JY-Zhu (2010) for the combined program.

$$(VP) \quad \min_{x,y} F(x,y)$$
$$\text{s.t. } f(x,y) - v(x) \leq 0, g(x,y) \leq 0.$$

The FJ condition for (VP): $\exists (r, \lambda, \mu) \neq 0$ such that

$$0 \in r\nabla F(\bar{x},\bar{y}) + \lambda(\nabla f(\bar{x},\bar{y}) + \partial(-v)(\bar{x}) \times \{0\}) + \nabla g(\bar{x},\bar{y})^T \mu$$
$$r \geq 0, 0 \leq \mu \perp g(\bar{x},\bar{y}).$$

• If the limiting subdifferential $\partial(-v)(\bar{x})$ can be replaced by a smaller set, then the resulting FJ condition is sharper and the constraint qualification is then weaker.

By $x^k \xrightarrow{d} \bar{x}$ where d is a vector, we mean there exist $t_k \downarrow 0, d^k \to d$ such that $x^k = \bar{x} + t_k d^k$.

Definition (Directional subdifferentials Ginchev and Mordukhovich 2011)

Let $\varphi : \mathbb{R}^n \to \overline{\mathbb{R}}$ and $\varphi(\bar{x})$ be finite. The (analytic) limiting subdifferential of φ at $\bar{x}$ in direction $d \in \mathbb{R}^n$ is defined as

$$\partial\varphi(\bar{x}; d) := \{\lim_k \xi^k | \exists x^k \xrightarrow{d} \bar{x}, \varphi(x^k) \to \varphi(\bar{x}),\ \xi^k \in \widehat{\partial}\varphi(x^k)\},$$

where $\widehat{\partial}\varphi$ is the regular subdifferential.

- For example, $\varphi(x) = |x|$. Then the limiting subdifferential at 0 is equal to the interval:

$$\partial\varphi(0) = [-1, 1].$$

- But the directional limiting subdifferential at 0 is a singleton:

$$\partial\varphi(0; 1) = \{1\},\ \partial\varphi(0; -1) = \{-1\}.$$

By $x^k \xrightarrow{d} \bar{x}$ where d is a vector, we mean there exist $t_k \downarrow 0, d^k \to d$ such that $x^k = \bar{x} + t_k d^k$.

Definition (Directional subdifferentials Ginchev and Mordukhovich 2011)

Let $\varphi : \mathbb{R}^n \to \overline{\mathbb{R}}$ and $\varphi(\bar{x})$ be finite. The (analytic) limiting subdifferential of φ at $\bar{x}$ in direction $d \in \mathbb{R}^n$ is defined as

$$\partial\varphi(\bar{x}; d) := \{\lim_k \xi^k | \exists x^k \xrightarrow{d} \bar{x}, \varphi(x^k) \to \varphi(\bar{x}),\ \xi^k \in \widehat{\partial}\varphi(x^k)\},$$

where $\widehat{\partial}\varphi$ is the regular subdifferential.

• For example, $\varphi(x) = |x|$. Then the limiting subdifferential at 0 is equal to the interval:

$$\partial\varphi(0) = [-1, 1].$$

• But the directional limiting subdifferential at 0 is a singleton:

$$\partial\varphi(0; 1) = \{1\},\ \partial\varphi(0; -1) = \{-1\}.$$

By $x^k \xrightarrow{d} \bar{x}$ where d is a vector, we mean there exist $t_k \downarrow 0, d^k \to d$ such that $x^k = \bar{x} + t_k d^k$.

Definition (Directional subdifferentials Ginchev and Mordukhovich 2011)

Let $\varphi : \mathbb{R}^n \to \overline{\mathbb{R}}$ and $\varphi(\bar{x})$ be finite. The (analytic) limiting subdifferential of φ at $\bar{x}$ in direction $d \in \mathbb{R}^n$ is defined as

$$\partial\varphi(\bar{x}; d) := \{\lim_k \xi^k | \exists x^k \xrightarrow{d} \bar{x}, \varphi(x^k) \to \varphi(\bar{x}),\ \xi^k \in \widehat{\partial}\varphi(x^k)\},$$

where $\widehat{\partial}\varphi$ is the regular subdifferential.

- For example, $\varphi(x) = |x|$. Then the limiting subdifferential at 0 is equal to the interval:

$$\partial\varphi(0) = [-1, 1].$$

- But the directional limiting subdifferential at 0 is a singleton:

$$\partial\varphi(0; 1) = \{1\},\ \partial\varphi(0; -1) = \{-1\}.$$

- Consider problem (P) : $\min f(x)$ s.t. $g(x) \leq 0$.
- The feasible region: $\mathcal{F} := \{x | g(x) \leq 0\}$. Suppose that f is smooth, $g = (g_1, \ldots, g_m)$ and g_i is directionally differentiable at $\bar{x}$.
- Let $\bar{I}_g := \{i | g_i(\bar{x}) = 0\}$.
- Define $g' := (g_1', \ldots, g_m')$ and $g_i'(\bar{x}; d) = 0$ if $i \notin \bar{I}_g$.

The linearized cone:

$$L(\bar{x}) := \{d | g'(\bar{x}; d) \leq 0\}$$

The critical cone:

$$C(\bar{x}) = L(\bar{x}) \cap \{d | \nabla f(\bar{x})^T d \leq 0\}.$$

- Consider problem (P) : $\min f(x)$ s.t. $g(x) \leq 0$.
- The feasible region: $\mathcal{F} := \{x | g(x) \leq 0\}$. Suppose that f is smooth, $g = (g_1, \ldots, g_m)$ and g_i is directionally differentiable at $\bar{x}$.
- Let $\bar{I}_g := \{i | g_i(\bar{x}) = 0\}$.
- Define $g' := (g_1', \ldots, g_m')$ and $g_i'(\bar{x}; d) = 0$ if $i \notin \bar{I}_g$.

The linearized cone:

$$L(\bar{x}) := \{d | g'(\bar{x}; d) \leq 0\}$$

The critical cone:

$$C(\bar{x}) = L(\bar{x}) \cap \{d | \nabla f(\bar{x})^T d \leq 0\}.$$

- Consider problem $(P) : \min f(x)$ s.t. $g(x) \leq 0$.
- The feasible region: $\mathcal{F} := \{x | g(x) \leq 0\}$. Suppose that f is smooth, $g = (g_1, \ldots, g_m)$ and g_i is directionally differentiable at $\bar{x}$.
- Let $\bar{I}_g := \{i | g_i(\bar{x}) = 0\}$.
- Define $g' := (g_1', \ldots, g_m')$ and $g_i'(\bar{x}; d) = 0$ if $i \notin \bar{I}_g$.

The linearized cone:

$$L(\bar{x}) := \{d | g'(\bar{x}; d) \leq 0\}$$

The critical cone:

$$C(\bar{x}) = L(\bar{x}) \cap \{d | \nabla f(\bar{x})^T d \leq 0\}.$$

Directional optimality conditions proposed by Gfrerer 2013

Suppose $\bar{x}$ is a local optimal solution of (P) and g is Lipschitz continuous at $\bar{x}$. Let $d \in C(\bar{x})$. Suppose g is directional differentiable at $\bar{x}$ in direction d. If the first order sufficient condition for metric subregularity (FOSCMS) holds:

$$\begin{cases} 0 \in \partial g(\bar{x}; d)^T \lambda, \\ 0 \leq \lambda \perp g(\bar{x}), \quad \lambda \perp g'(\bar{x}; d) \end{cases} \implies \lambda = 0,$$

then the directional KKT condition holds:

$$\begin{aligned} &0 \in \nabla f(\bar{x}) + \partial g(\bar{x}; d)^T \lambda, \\ &0 \leq \lambda \perp g(\bar{x}), \quad \lambda \perp g'(\bar{x}; d). \end{aligned}$$

- Since when $d = 0$, FOSCMS is equivalent to MFCQ, for simplicity, I will call FOSCMS the directional MFCQ.

Directional optimality conditions proposed by Gfrerer 2013

Suppose $\bar{x}$ is a local optimal solution of (P) and g is Lipschitz continuous at $\bar{x}$. Let $d \in C(\bar{x})$. Suppose g is directional differentiable at $\bar{x}$ in direction d. If the first order sufficient condition for metric subregularity (FOSCMS) holds:

$$\begin{cases} 0 \in \partial g(\bar{x}; d)^T \lambda, \\ 0 \le \lambda \perp g(\bar{x}), \quad \lambda \perp g'(\bar{x}; d) \end{cases} \Longrightarrow \quad \lambda = 0,$$

then the directional KKT condition holds:

$$\begin{aligned} &0 \in \nabla f(\bar{x}) + \partial g(\bar{x}; d)^T \lambda, \\ &0 \le \lambda \perp g(\bar{x}), \quad \lambda \perp g'(\bar{x}; d). \end{aligned}$$

• Since when $d = 0$, FOSCMS is equivalent to MFCQ, for simplicity, I will call FOSCMS the directional MFCQ.

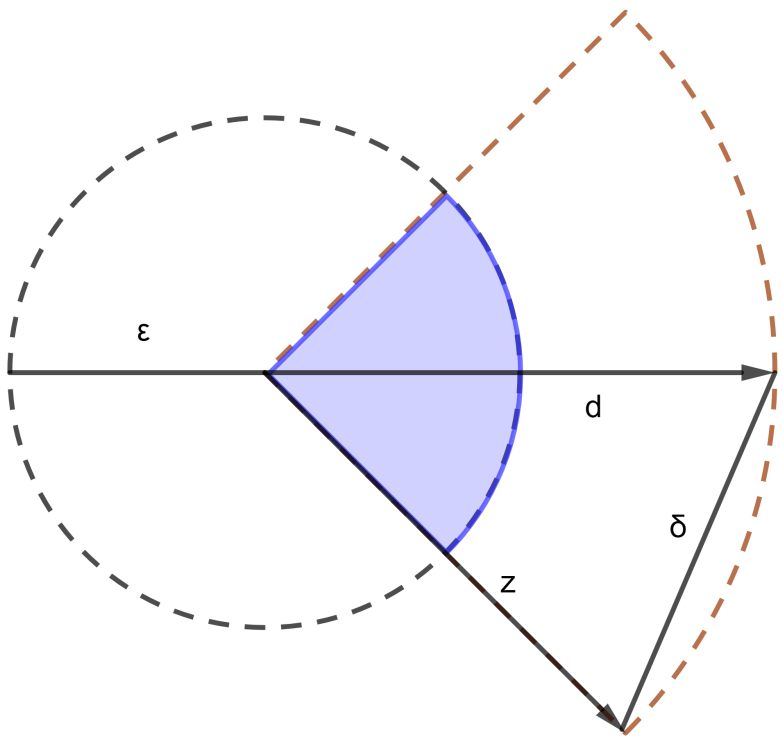
FOSCMS/the directional MFCQ fails for (VP)

Proposition (Bai and JY, 2021)

Assume that $v(x)$ is directionally differentiable at $\bar{x}$. Then FOSCMS/the directional MFCQ fails at any feasible solution of (VP) in any critical direction.

Directional neighborhood of the origin

$$\mathcal{V}_{\varepsilon,\delta}(d) = \begin{cases} \mathbb{B}_\varepsilon(0), & d = 0, \\ \{0\} \cup \{z \in \mathbb{B}_\varepsilon(0) | \left\| \frac{z}{\|z\|} - \frac{d}{\|d\|} \right\| \leq \delta\}, & d \neq 0. \end{cases}$$



Directional Clarke calmness condition

Since the directional MFCQ is too strong to be applicable for bilevel programs, we now introduce a directional version of the calmness condition for (P).

Definition (Directional Clarke calmness; Bai and JY, 2021)

Suppose $\bar{x}$ solves (P). We say that (P) is (Clarke) calm at $\bar{x}$ in direction d if $\bar{x}$ also solves (for some positive ϵ, δ, ρ)

$$\min f(x) + \rho\|g_+(x)\|$$
$$s.t.\ x \in \bar{x} + \mathcal{V}_{\epsilon,\delta}(d).$$

- When $d = 0$, the directional calmness reduces to the calmness introduced by Clarke.
- Directional Clarke calmness with $d \neq 0$ is weaker than the Clarke calmness.

Directional Clarke calmness condition

Since the directional MFCQ is too strong to be applicable for bilevel programs, we now introduce a directional version of the calmness condition for (P).

Definition (Directional Clarke calmness; Bai and JY, 2021)

Suppose $\bar{x}$ solves (P). We say that (P) is (Clarke) calm at $\bar{x}$ in direction d if $\bar{x}$ also solves (for some positive ϵ, δ, ρ)

$$\begin{aligned} &\min f(x) + \rho \|g_+(x)\| \\ &s.t.\ x \in \bar{x} + \mathcal{V}_{\epsilon,\delta}(d). \end{aligned}$$

- When $d = 0$, the directional calmness reduces to the calmness introduced by Clarke.
- Directional Clarke calmness with $d \neq 0$ is weaker than the Clarke calmness.

Directional Clarke calmness condition

Since the directional MFCQ is too strong to be applicable for bilevel programs, we now introduce a directional version of the calmness condition for (P).

Definition (Directional Clarke calmness; Bai and JY, 2021)

Suppose $\bar{x}$ solves (P). We say that (P) is (Clarke) calm at $\bar{x}$ in direction d if $\bar{x}$ also solves (for some positive ϵ, δ, ρ)

$$\begin{aligned} &\min f(x) + \rho \|g_+(x)\| \\ &s.t.\ x \in \bar{x} + \mathcal{V}_{\epsilon,\delta}(d). \end{aligned}$$

- When $d = 0$, the directional calmness reduces to the calmness introduced by Clarke.
- Directional Clarke calmness with $d \neq 0$ is weaker than the Clarke calmness.

Directional KKT condition under directional calmness

Theorem (Bai and JY, 2021)

Let $\bar{x}$ be a local minimizer of (P). Suppose $f(x)$ is continuously differentiable at $\bar{x}$ and $g(x)$ is directionally Lipschitz and directionally differentiable at $\bar{x}$ in direction $d \in C(\bar{x})$. Suppose that the (P) is calm at $\bar{x}$ in direction d. Then there exists a vector $\lambda \in \mathbb{R}^m$ such that the directional KKT condition holds at $\bar{x}$ in direction d:

$$0 \in \nabla f(\bar{x}) + \partial g(\bar{x}; d)^T \lambda$$
$$0 \leq \lambda \perp g(\bar{x}), \ \lambda \perp g'(\bar{x}; d).$$

- When $d = 0$, the directional KKT recovers KKT condition.
- When $d \neq 0$, directional KKT condition is sharper than the (nondirectional) KKT condition under weaker (directional) calmness condition.

Directional KKT condition under directional calmness

Theorem (Bai and JY, 2021)

Let $\bar{x}$ be a local minimizer of (P). Suppose $f(x)$ is continuously differentiable at $\bar{x}$ and $g(x)$ is directionally Lipschitz and directionally differentiable at $\bar{x}$ in direction $d \in C(\bar{x})$. Suppose that the (P) is calm at $\bar{x}$ in direction d. Then there exists a vector $\lambda \in \mathbb{R}^m$ such that the directional KKT condition holds at $\bar{x}$ in direction d:

$$0 \in \nabla f(\bar{x}) + \partial g(\bar{x}; d)^T \lambda$$
$$0 \leq \lambda \perp g(\bar{x}), \ \lambda \perp g'(\bar{x}; d).$$

- When $d = 0$, the directional KKT recovers KKT condition.
- When $d \neq 0$, directional KKT condition is sharper than the (nondirectional) KKT condition under weaker (directional) calmness condition.

Directional KKT condition under directional calmness

Theorem (Bai and JY, 2021)

Let $\bar{x}$ be a local minimizer of (P). Suppose $f(x)$ is continuously differentiable at $\bar{x}$ and $g(x)$ is directionally Lipschitz and directionally differentiable at $\bar{x}$ in direction $d \in C(\bar{x})$. Suppose that the (P) is calm at $\bar{x}$ in direction d. Then there exists a vector $\lambda \in \mathbb{R}^m$ such that the directional KKT condition holds at $\bar{x}$ in direction d:

$$0 \in \nabla f(\bar{x}) + \partial g(\bar{x}; d)^T \lambda$$
$$0 \leq \lambda \perp g(\bar{x}), \ \lambda \perp g'(\bar{x}; d).$$

- When $d = 0$, the directional KKT recovers KKT condition.
- When $d \neq 0$, directional KKT condition is sharper than the (nondirectional) KKT condition under weaker (directional) calmness condition.

Directional KKT condition for bilevel program

Theorem (Bai and JY, 2021)

Let $(\bar{x}, \bar{y})$ be a local minimizer of (BP). Suppose the value function $v(x)$ is Lipschitz continuous and directionally differentiable at $\bar{x}$ in direction d_x and $(d_x, d_y) \in C(\bar{x}, \bar{y})$

$$
\begin{aligned}
C(\bar{x}, \bar{y}) &:= L(\bar{x}, \bar{y}) \cap \{(d_x, d_y) | F(\bar{x}, \bar{y})(d_x, d_y) \leq 0\}, \\
L(\bar{x}, \bar{y}) &:= \left\{ (d_x, d_y) | \begin{array}{l} \nabla f(\bar{x}, \bar{y})(d_x, d_y) \leq v'(\bar{x}; d_x) \\ \nabla g(\bar{x}, \bar{y})(d_x, d_y) \leq 0 \end{array} \right\}.
\end{aligned}
$$

Suppose that (VP) is calm at $(\bar{x}, \bar{y})$ in direction (d_x, d_y). Then there exists $(\lambda, \mu) \geq 0$ such that

$$
\begin{aligned}
0 \in & \nabla F(\bar{x}, \bar{y}) + \lambda \left(\nabla f(\bar{x}, \bar{y}) + \partial(-v)(\bar{x}; d_x) \times \{0\} \right) \\
& + \nabla g(\bar{x}, \bar{y})^T \mu, \\
\mu \perp & \, g(\bar{x}, \bar{y}), \ \mu \perp \nabla g(\bar{x}, \bar{y})(d_x, d_y).
\end{aligned}
$$

Directional Clarke subdifferential

Definition

Suppose $\varphi(x) : \mathbb{R}^n \to \mathbb{R}$ is Lipschitz continuous at $\bar{x}$ in direction d, we define the directional Clarke subdifferential of φ at $\bar{x}$ in direction d as

$$\partial^c \varphi(\bar{x}; d) := co(\partial \varphi(\bar{x}; d)).$$

We have

$$\partial^c(-v)(\bar{x}; d) = -\partial^c v(\bar{x}; d)$$

Under certain conditions, we have derive some upper estimates for the directional Clarke subdifferential of the value function in terms of the problem data. Substitute these upper estimates to the directional KKT condition we obtain the condition in terms of the problem data.

Directional Clarke subdifferential

Definition

Suppose $\varphi(x) : \mathbb{R}^n \to \mathbb{R}$ is Lipschitz continuous at $\bar{x}$ in direction d, we define the directional Clarke subdifferential of φ at $\bar{x}$ in direction d as

$$\partial^c \varphi(\bar{x}; d) := co(\partial \varphi(\bar{x}; d)).$$

We have

$$\partial^c(-v)(\bar{x}; d) = -\partial^c v(\bar{x}; d)$$

Under certain conditions, we have derive some upper estimates for the directional Clarke subdifferential of the value function in terms of the problem data. Substitute these upper estimates to the directional KKT condition we obtain the condition in terms of the problem data.

Difference of Convex Algorithms for Bilevel Programs

• **Motivation:** Many functions can be represented as a difference convex (DC) function: lower C^2 function and C^{1+} function is a difference convex (DC) function, and the class of DC functions is closed under many operations. If the lower level program is completely convex (convex in both variables x and y), then the value function is convex and the value function constraint becomes a DC constraint:

$$f(x, y) - v(x) \leq 0.$$

To use the difference of convex algorithm (DCA) cf. review paper by Horst and Thoai 1999, one needs to study two issues:

- Under what conditions, all functions including the value function are convex and Lipschitz continuous?
- Under what condition, the extended MFCQ holds?

Difference of convex bilevel program

$$\begin{aligned} \min_{x,y} \quad & F(x,y) := F_1(x,y) - F_2(x,y) \\ s.t. \quad & x \in X, y \in S(x) := \arg\min_{y \in Y} \{ f(x,y) \ \text{ s.t.} g(x,y) \leq 0 \}, \end{aligned}$$

where $X \subseteq \mathbb{R}^n$ and $Y \subseteq \mathbb{R}^m$ are nonempty closed convex sets, $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^l$ is convex on an open convex set containing the set $X \times Y$, and the functions $F_1, F_2, f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ are convex on an open convex set containing the set

$$C := \{(x,y) \in X \times Y : g(x,y) \leq 0\}.$$

By Lampariello and Sagratella (2020), if the lower level objective function is in the form of $f(x,y) = f_1(x,y_1) + f_2(y_2)$ where f_2 is convex, $f_1(\cdot, y_1)$ is convex for every y_1 and $f_1(x,\cdot)$ is uniformly strongly convex for every x, then the lower level problem can be reformulated as one with a completely convex objective.

Difference of convex bilevel program

$$\begin{aligned} \min_{x,y} \quad & F(x,y) := F_1(x,y) - F_2(x,y) \\ s.t. \quad & x \in X, y \in S(x) := \arg\min_{y \in Y} \{ f(x,y) \ \text{ s.t.} g(x,y) \leq 0 \}, \end{aligned}$$

where $X \subseteq \mathbb{R}^n$ and $Y \subseteq \mathbb{R}^m$ are nonempty closed convex sets, $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^l$ is convex on an open convex set containing the set $X \times Y$, and the functions $F_1, F_2, f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ are convex on an open convex set containing the set

$$C := \{(x,y) \in X \times Y : g(x,y) \leq 0\}.$$

By Lampariello and Sagratella (2020), if the lower level objective function is in the form of $f(x,y) = f_1(x,y_1) + f_2(y_2)$ where f_2 is convex, $f_1(\cdot, y_1)$ is convex for every y_1 and $f_1(x,\cdot)$ is uniformly strongly convex for every x, then the lower level problem can be reformulated as one with a completely convex objective.

Standing Assumptions

(I) $S(x) \neq \emptyset$ for all $x \in X$. For all x in an open convex set $\mathcal{O} \supseteq X$, the feasible region $\mathcal{F}(x) := \{y \in Y : g(x,y) \leq 0\}$ is nonempty and $f(x,y)$ is bounded below on $\mathcal{F}(x)$.

(II) Assume that the partial derivative formula holds for each of the lower level objective and constraint functions:

$$\partial\phi(x,y) = \partial_x\phi(x,y) \times \partial_y\phi(x,y).$$

Some sufficient conditions for the partial derivative formula:

- $\phi(x,y) = \phi_1(x) + \phi_2(y)$.
- $\phi(x,y)$ is C^1 respect to either x or y.

lasso problem as a bilevel program with a completely convex lower level program

By change of variable $r := \frac{1}{\lambda}$, lasso problem can be equivalently reformulated as:

$$\min_{r,\theta_1,\ldots,\theta_k} \quad CV(\theta_1,\ldots,\theta_k)$$

$$r > 0 \text{ and for each } k = 1,\ldots,K$$

$$\theta_k \in \arg\min_{\theta} \sum_{(x_j,y_j)\in\Omega^k_{trn}} \frac{(\mathrm{x}_j^T\theta - y_j)^2}{r} + \|\theta\|_1.$$

Since a square over linear function

$$\phi(\mathrm{x}, r) = \|\mathrm{x}\|^2/r$$

is completely convex, the lower level is a completely convex bilevel program.

The bilevel model for support vector (SV) classification

Given a training set $\Omega := \{(\mathrm{a}_j, b_j)\}_{j=1}^n$ where $\mathrm{a}_j \in \mathbb{R}^p$, and the labels $b_j = \pm 1$ indicate the class membership. The bilevel model for SV classification (Kunapuli, Bennett, Hu and Pang, 2008):

$$\min_{\lambda,\bar{w},w,\mathrm{c}} \quad \mathrm{CV}(\mathrm{w}^1,\ldots,\mathrm{w}^K,\mathrm{c}^1,\ldots,c^K)$$

$$:= \frac{1}{K}\sum_{k=1}^{K}\sum_{(a_j,b_j)\in\Omega_k} \max(1-b_j(\mathrm{a}_j^T\mathrm{w}^k - c^k),0)$$

$$s.t. \quad \lambda_{lb} \le \lambda \le \lambda_{ub}, \quad \bar{\mathrm{w}}_{lb} \le \bar{\mathrm{w}} \le \bar{\mathrm{w}}_{ub}, \text{and for } k = 1,\ldots,K:$$

$$(\mathrm{w}^k, c^k) \in \underset{\substack{-\bar{\mathrm{w}} \le \mathrm{w} \le \bar{\mathrm{w}} \\ c \in \mathbb{R}}}{\mathrm{argmin}} \left\{ \sum_{(a_j,b_j)\in\Omega^k_{trn}} \max(1-b_j(\mathrm{a}_j^T\mathrm{w} - c),0) + \frac{\lambda}{2}\|\mathrm{w}\|^2 \right\}.$$

- By change of variable $r := \frac{1}{\lambda}$, the bilevel model for SV classification problem can be equivalently reformulated as a bilevel program with completely convex lower level program.

The bilevel model for support vector (SV) classification

Given a training set $\Omega := \{(\mathrm{a}_j, b_j)\}_{j=1}^n$ where $\mathrm{a}_j \in \mathbb{R}^p$, and the labels $b_j = \pm 1$ indicate the class membership. The bilevel model for SV classification (Kunapuli, Bennett, Hu and Pang, 2008):

$$\min_{\lambda,\bar{w},w,\mathrm{c}} \quad \mathrm{CV}(\mathrm{w}^1,\ldots,\mathrm{w}^K,\mathrm{c}^1,\ldots,c^K)$$

$$:= \frac{1}{K}\sum_{k=1}^{K}\sum_{(a_j,b_j)\in\Omega_k} \max(1-b_j(\mathrm{a}_j^T\mathrm{w}^k - c^k),0)$$

$$s.t. \quad \lambda_{lb} \le \lambda \le \lambda_{ub}, \quad \bar{\mathrm{w}}_{lb} \le \bar{\mathrm{w}} \le \bar{\mathrm{w}}_{ub}, \text{and for } k = 1,\ldots,K:$$

$$(\mathrm{w}^k, c^k) \in \underset{\substack{-\bar{\mathrm{w}} \le \mathrm{w} \le \bar{\mathrm{w}} \\ c \in \mathbb{R}}}{\mathrm{argmin}} \left\{ \sum_{(a_j,b_j)\in\Omega_{trn}^k} \max(1-b_j(\mathrm{a}_j^T\mathrm{w} - c),0) + \frac{\lambda}{2}\|\mathrm{w}\|^2 \right\}.$$

- By change of variable $r := \frac{1}{\lambda}$, the bilevel model for SV classification problem can be equivalently reformulated as a bilevel program with completely convex lower level program.

Reformulation of the bilevel program as a DC program

Using the convex analysis in Rockafellar (1970) we can obtain: (1) under the assumptions, all functions F_1, F_2, g are convex and Lipschitz continuous, and the value function $v(x)$ is convex and Lipschitz continuous on X; (2) for any $x \in X$ and $y \in S(x)$,

$$\bigcup_{\gamma \in KT(x,y)} \left(\partial_x f(x,y) + \sum_{i=1}^{l} \gamma_i \partial_x g_i(x,y) \right) \subseteq \partial v(x).$$

For $\epsilon \geq 0$, consider the following difference of convex program:

$$(\mathrm{VP})_\epsilon \qquad \min_{(x,y) \in X \times Y} \quad F_1(x,y) - F_2(x,y)$$
$$s.t. \quad f(x,y) - v(x) \leq \epsilon$$
$$g(x,y) \leq 0.$$

For any $\epsilon > 0$, the extended MFCQ always hold on

$$C := \{(x,y) \in X \times Y | g(x,y) \leq 0\}.$$

Reformulation of the bilevel program as a DC program

Using the convex analysis in Rockafellar (1970) we can obtain: (1) under the assumptions, all functions F_1, F_2, g are convex and Lipschitz continuous, and the value function $v(x)$ is convex and Lipschitz continuous on X; (2) for any $x \in X$ and $y \in S(x)$,

$$\bigcup_{\gamma \in KT(x,y)} \left(\partial_x f(x,y) + \sum_{i=1}^{l} \gamma_i \partial_x g_i(x,y) \right) \subseteq \partial v(x).$$

For $\epsilon \geq 0$, consider the following difference of convex program:

$$(\mathrm{VP})_\epsilon \qquad \min_{(x,y) \in X \times Y} \quad F_1(x,y) - F_2(x,y)$$
$$s.t. \quad f(x,y) - v(x) \leq \epsilon$$
$$g(x,y) \leq 0.$$

For any $\epsilon > 0$, the extended MFCQ always hold on

$$C := \{(x,y) \in X \times Y | g(x,y) \leq 0\}.$$

Reformulation of the bilevel program as a DC program

Using the convex analysis in Rockafellar (1970) we can obtain: (1) under the assumptions, all functions F_1, F_2, g are convex and Lipschitz continuous, and the value function $v(x)$ is convex and Lipschitz continuous on X; (2) for any $x \in X$ and $y \in S(x)$,

$$\bigcup_{\gamma \in KT(x,y)} \left(\partial_x f(x,y) + \sum_{i=1}^{l} \gamma_i \partial_x g_i(x,y) \right) \subseteq \partial v(x).$$

For $\epsilon \geq 0$, consider the following difference of convex program:

$$(\mathrm{VP})_\epsilon \qquad \min_{(x,y) \in X \times Y} \quad F_1(x,y) - F_2(x,y)$$
$$s.t. \quad f(x,y) - v(x) \leq \epsilon$$
$$g(x,y) \leq 0.$$

For any $\epsilon > 0$, the extended MFCQ always hold on

$$C := \{(x,y) \in X \times Y | g(x,y) \leq 0\}.$$

Inexact proximal difference of convex algorithm (iPDCA)

- Given a current iteration point (x^k, y^k), solve the lower level problem (P_{x^k}) with a global minimizer $\tilde{y}^k$ and a corresponding multiplier denoted by γ^k.
- Select

$$\xi_0^k \in \partial F_2(x^k, y^k), \xi_1^k \in \partial_x f(x^k, \tilde{y}^k) + \sum_{i=1}^{l} \gamma_i^k \partial_x g_i(x^k, \tilde{y}^k) \subseteq \partial v(x^k).$$

- Compute (x^{k+1}, y^{k+1}) as an approximate minimizer of the strongly convex subproblem for $(\mathrm{VP})_\epsilon$ given by

$$\min_{(x,y)\in C} \; F_1(x,y) - \underbrace{\langle \xi_0^k, (x,y)\rangle}_{\text{linearization of } F_2 \text{ at } (x^k,y^k)} + \frac{\rho}{2}\|(x,y)-(x^k,y^k)\|^2 + \beta_k \max\{f(x,y) - \underbrace{(f(x^k,\tilde{y}^k) + \langle \xi_1^k, x-x^k\rangle)}_{\text{linearization of } V(x) \text{ at } x^k} - \epsilon, 0\}.$$

Inexact proximal difference of convex algorithm (iPDCA)

- Given a current iteration point (x^k, y^k), solve the lower level problem (P_{x^k}) with a global minimizer $\tilde{y}^k$ and a corresponding multiplier denoted by γ^k.
- Select

$$\xi_0^k \in \partial F_2(x^k, y^k), \xi_1^k \in \partial_x f(x^k, \tilde{y}^k) + \sum_{i=1}^{I} \gamma_i^k \partial_x g_i(x^k, \tilde{y}^k) \subseteq \partial v(x^k).$$

- Compute (x^{k+1}, y^{k+1}) as an approximate minimizer of the strongly convex subproblem for $(\mathrm{VP})_\epsilon$ given by

$$\min_{(x,y)\in C} \quad F_1(x,y) - \underbrace{\langle \xi_0^k, (x,y)\rangle}_{\text{linearization of } F_2 \text{ at } (x^k,y^k)} + \frac{\rho}{2}\|(x,y)-(x^k,y^k)\|^2$$
$$+ \beta_k \max\{f(x,y) - \underbrace{(f(x^k,\tilde{y}^k) + \langle \xi_1^k, x-x^k\rangle)}_{\text{linearization of } V(x) \text{ at } x^k} - \epsilon, 0\}.$$

Inexact proximal difference of convex algorithm (iPDCA)

- Given a current iteration point (x^k, y^k), solve the lower level problem (P_{x^k}) with a global minimizer $\tilde{y}^k$ and a corresponding multiplier denoted by γ^k.
- Select

$$\xi_0^k \in \partial F_2(x^k, y^k), \xi_1^k \in \partial_x f(x^k, \tilde{y}^k) + \sum_{i=1}^{l} \gamma_i^k \partial_x g_i(x^k, \tilde{y}^k) \subseteq \partial v(x^k).$$

- Compute (x^{k+1}, y^{k+1}) as an approximate minimizer of the strongly convex subproblem for $(\mathrm{VP})_\epsilon$ given by

$$\min_{(x,y)\in C} \quad F_1(x,y) - \underbrace{\langle \xi_0^k, (x,y)\rangle}_{\text{linearization of } F_2 \text{ at } (x^k,y^k)} + \frac{\rho}{2}\|(x,y)-(x^k,y^k)\|^2$$
$$+ \beta_k \max\{f(x,y) - \underbrace{(f(x^k,\tilde{y}^k) + \langle \xi_1^k, x - x^k\rangle)}_{\text{linearization of } V(x) \text{ at } x^k} - \epsilon, 0\}.$$

Convergence theorem

Definition

We say a point $(\bar{x}, \bar{y})$ is a KKT point of problem $(\mathrm{VP})_\epsilon$ with $\epsilon \geq 0$ if there exists $\lambda \geq 0$ such that

$$\begin{cases} 0 \in \partial F_1(\bar{x}, \bar{y}) - \partial F_2(\bar{x}, \bar{y}) + \lambda \partial f(\bar{x}, \bar{y}) - \lambda \partial v(\bar{x}) \times \{0\} + \mathcal{N}_C(\bar{x}, \bar{y}), \\ f(\bar{x}, \bar{y}) - v(\bar{x}) - \epsilon \leq 0, \quad \lambda \left(f(\bar{x}, \bar{y}) - v(\bar{x}) - \epsilon \right) = 0. \end{cases}$$

Theorem (JY, Yuan, Zeng and Zhang 2021)

Assume that the upper level objective F is bounded below on C. Let $\{(x^k, y^k)\}$ be an iteration sequence generated by iPDCA . Moreover assume that $KT(x^k, y) \neq \emptyset$ for all $y \in S(x^k)$. Suppose that either $\epsilon > 0$ or $\epsilon = 0$ and the penalty sequence $\{\beta_k\}$ is bounded. Than any accumulation point of $\{(x^k, y^k)\}$ is an KKT point of problem $(\mathrm{VP})_\epsilon$.

Convergence theorem

Definition

We say a point $(\bar{x}, \bar{y})$ is a KKT point of problem $(\mathrm{VP})_\epsilon$ with $\epsilon \geq 0$ if there exists $\lambda \geq 0$ such that

$$\begin{cases} 0 \in \partial F_1(\bar{x}, \bar{y}) - \partial F_2(\bar{x}, \bar{y}) + \lambda \partial f(\bar{x}, \bar{y}) - \lambda \partial v(\bar{x}) \times \{0\} + \mathcal{N}_C(\bar{x}, \bar{y}), \\ f(\bar{x}, \bar{y}) - v(\bar{x}) - \epsilon \leq 0, \quad \lambda \left(f(\bar{x}, \bar{y}) - v(\bar{x}) - \epsilon \right) = 0. \end{cases}$$

Theorem (JY, Yuan, Zeng and Zhang 2021)

Assume that the upper level objective F is bounded below on C. Let $\{(x^k, y^k)\}$ be an iteration sequence generated by iPDCA . Moreover assume that $KT(x^k, y) \neq \emptyset$ for all $y \in S(x^k)$. Suppose that either $\epsilon > 0$ or $\epsilon = 0$ and the penalty sequence $\{\beta_k\}$ is bounded. Than any accumulation point of $\{(x^k, y^k)\}$ is an KKT point of problem $(\mathrm{VP})_\epsilon$.

Table: Numerical results comparing iP-DCA and MPEC approach

Dataset	Method	CV error	Test error	Time(sec)
australian_scale	iP-DCA($\epsilon = 0$, $tol = 10^{-2}$)	0.28 ± 0.03	0.15 ± 0.01	73.7 ± 106.6
	iP-DCA($\epsilon = 0$, $tol = 10^{-3}$)	0.28 ± 0.03	0.15 ± 0.01	81.2 ± 110.8
	iP-DCA($\epsilon = 10^{-2}$, $tol = 10^{-2}$)	0.28 ± 0.03	0.15 ± 0.01	10.7 ± 6.3
	iP-DCA($\epsilon = 10^{-2}$, $tol = 10^{-3}$)	0.28 ± 0.03	0.15 ± 0.01	128.7 ± 74.4
	iP-DCA($\epsilon = 10^{-4}$, $tol = 10^{-2}$)	0.28 ± 0.03	0.15 ± 0.01	74.2 ± 123.8
	iP-DCA($\epsilon = 10^{-4}$, $tol = 10^{-3}$)	0.28 ± 0.03	0.15 ± 0.01	109.0 ± 141.0
	MPEC approach	0.29 ± 0.04	0.15 ± 0.01	491.2 ± 245.1
breast-cancer_scale	iP-DCA($\epsilon = 0$, $tol = 10^{-2}$)	0.06 ± 0.01	0.04 ± 0.00	53.1 ± 67.2
	iP-DCA($\epsilon = 0$, $tol = 10^{-3}$)	0.06 ± 0.01	0.04 ± 0.00	78.3 ± 73.9
	iP-DCA($\epsilon = 10^{-2}$, $tol = 10^{-2}$)	0.06 ± 0.01	0.04 ± 0.00	15.5 ± 2.1
	iP-DCA($\epsilon = 10^{-2}$, $tol = 10^{-3}$)	0.06 ± 0.01	0.04 ± 0.00	108.9 ± 40.4
	iP-DCA($\epsilon = 10^{-4}$, $tol = 10^{-2}$)	0.06 ± 0.01	0.04 ± 0.01	24.6 ± 17.5
	iP-DCA($\epsilon = 10^{-4}$, $tol = 10^{-3}$)	0.06 ± 0.01	0.04 ± 0.01	86.8 ± 59.3
	MPEC approach	0.08 ± 0.01	0.04 ± 0.01	294.5 ± 98.2
diabetes_scale	iP-DCA($\epsilon = 0$, $tol = 10^{-2}$)	0.56 ± 0.03	0.24 ± 0.02	12.0 ± 13.6
	iP-DCA($\epsilon = 0$, $tol = 10^{-3}$)	0.56 ± 0.03	0.24 ± 0.02	25.9 ± 33.2
	iP-DCA($\epsilon = 10^{-2}$, $tol = 10^{-2}$)	0.57 ± 0.03	0.24 ± 0.02	3.1 ± 0.6
	iP-DCA($\epsilon = 10^{-2}$, $tol = 10^{-3}$)	0.56 ± 0.03	0.24 ± 0.02	62.1 ± 31.7
	iP-DCA($\epsilon = 10^{-4}$, $tol = 10^{-2}$)	0.56 ± 0.03	0.24 ± 0.02	12.7 ± 19.7
	iP-DCA($\epsilon = 10^{-4}$, $tol = 10^{-3}$)	0.56 ± 0.03	0.24 ± 0.02	39.2 ± 45.7
	MPEC approach	0.59 ± 0.03	0.25 ± 0.02	346.7 ± 216.9

Reference

- K. Bai and JY, *Directional necessary optimality conditions for bilevel programs*, to appear in Math. Oper. Res., arXiv (2004.01783).
- JY, X. Yuan, S. Zeng and J. Zhang, *Difference of convex algorithms for bilevel programs with applications in hyperparameter selection*, arXiv (2102.09006).
- R. Ke, W. Yao, JY and J. Zhang, *Generic property of the partial calmness condition for bilevel programming problems*, revised for SIOPT.
- J. Nie, L. Wang, JY and S. Zhong, *A Lagrange multiplier expression method for bilevel polynomial optimization*, arXiv (2007.07933), revised for SIOPT.

- Thank You -