

Research Proposal

Yuchen Ge

September 2022

Contents

1	Project Objections	2
1.1	Prerequisite on Gröbner Basis and Graver Basis	3
1.2	Objectives	4
2	Background	4
2.1	Distribution-driven Scenario Generation	4
2.2	Problem-driven Scenario Generation	5
3	Research Plan and Methodology	5
3.1	Generating sub-gradients/ Lagrangian Cuts with Gröbner basis and Graver Basis	5
3.2	Statistic Question of Scenario Clustering Method	6

Abstract

Traditionally machine learning and optimization are two different branches of computer science. They need to accomplish two different types of tasks, and they are studied by two different sets of domain experts. However, Stochastic Programming, which applies a stochastic optimization approach to maximizing the expected value of a random quantity, is a typical example of data-driven optimization in the intersection of both fields. In many applications, this requires a large number of scenarios, which is often computationally very expensive, sometimes prohibitively so. To deal with the mentioned difficulty in Stochastic Integer Programming, We have two routines.

To reduce the complexity of large-scale stochastic optimization problems, we have developed an unsupervised machine learning approach (cost-space scenario clustering) that clusters these scenarios into similarity groups. In the present case, the similarity between the scenarios is measured based on solutions characterizing them. Specifically, two scenarios are considered close to one another if the solution induced by one scenario is good with respect to the other scenario. We are now looking to improve our methodology by employing algebraic methods. The research project consists of the application of the idea of Gröbner bases and similar algebraic objects to reduce the computational complexity of our machine-learning approach.

Furthermore, we can apply the Gröbner basis, Graver Basis, and similar algebraic objects to reduce the computational cost of existing combinatorial methods for Stochastic Integer Programming, for example, the Lagrangian relaxation method and Lagrangian cuts method.

1 Project Objections

Many models are special cases of Stochastic Mixed Integer Program, for example, applications of stochastic integer programming to vehicle routing ([21]), capacity expansion ([22]), inventory management ([23]), nurse staffing and scheduling ([24]). To be precise, a Stochastic Mixed Integer Program (SMIP) is to solve

$$\begin{aligned} \min \quad & c^T x + \mathbb{E}[Q(x, \xi)] \\ \text{s.t.} \quad & Ax \geq b \\ & x \in \mathbb{R}_+^{n_1} \times \mathbb{Z}_+^{p_1} \end{aligned} \tag{1}$$

where $\xi = (q, h, T, W)$ and

$$\begin{aligned} Q(x, \xi) = \min \quad & q^T y \\ \text{s.t.} \quad & Wy = h - Tx \\ & y \in \mathbb{R}_+^{n_2} \times \mathbb{Z}_+^{p_2} \end{aligned}$$

where x denotes the first-stage decision variables and y denotes the second-stage decision variables and sometimes we assume n_1, p_1, n_2 , or p_2 are zero, which depends.

We assume the random data ξ is represented by a finite set of scenarios: $(q^i, h^i, T^i, W^i), i = 1, \dots, N$, where scenario i occurs with probability p_i . So we have the following extensive form (deterministic equivalent) of an SMIP.

Definition 1.1. Deterministic equivalent of an SMIP is

$$\begin{aligned} \min \quad & c^T x + \sum_{i=1}^N p_i q_i^T y_i \\ \text{s.t.} \quad & Ax \geq b \\ & T_i x + W_i y_i = h_i \\ & x \in \mathbb{R}_+^{n_1} \times \mathbb{Z}_+^{p_1} \\ & y_i \in \mathbb{R}_+^{n_2} \times \mathbb{Z}_+^{p_2}, \quad i = 1, \dots, N \end{aligned} \tag{2}$$

When $n_1 = 0$ and $n_2 = 0$, we call such SMIP to be purely combinatorial, or simply SIP. For example, we can apply the Lagrangian relaxation method [1] or generate Lagrangian cuts [41] to cope with the situation. In the framework of these typical algorithms, we need to solve objects of similar structures many times, which follows that **we can extract some common features/objects needed without the dissipate of computational cost**. Algebraic objects are suitable for the specific purpose and every IP or SIP problem corresponds to a unique Gröbner basis/Graver Basis.

This project will carry out systematic research on the challenging topics – purely combinatorial SMIP, to study behaviors of existing algorithms and to apply algebraic methods to accelerate the algorithm by combining existing combinatorial methods with manipulating objects from algebraic geometry. For instance, based on the Lagrangian relaxation method [1] or generate Lagrangian cuts [41], I’ve designed a new variant of Lagrangian relaxation algorithm by applying Graver Basis in the algorithm, extracting relationships between algebraic objects needed in different steps of the algorithm (For example, we need many stochastic integer programming problems in the algorithm and thus we research on the relationships between these algebraic objects corresponding to the problems.) to calculate the sub-gradients of sub-problems more efficiently and save repetitive computation. Moreover, I’m managing to develop new algorithms to use the Gröbner basis and Graver Basis to generate Lagrangian cuts efficiently and automatically. It is worth mentioning that in practice with the application of computational algebraic geometry, we could greatly reduce the computational cost as is already being checked by numerical experiments. However, for large scenarios, it’s not practical to exactly solve the optimization problem anymore. Interested readers may refer to [40].

Therefore, this proposal will also develop the theory of cost-space scenario clustering method [36], a special case of scenario clustering, which is relatively young, and statistic questions of general theory are far from completely investigated. Scenario clustering is an approximation by finding a smaller subset of scenarios, which reduces the numerical complexity while keeping the error at an acceptable level. Return to our topic. The central idea of the cost-space scenario clustering method is to cluster these scenarios into similarity groups with the similarity between the scenarios being measured on the basis of solutions characterizing them. Specifically, two scenarios are considered close to one another if the solution induced by one scenario is good with respect to the other scenario. We are now looking to improve our methodology by utilizing algebraic methods. I’ve developed faster IP/SIP algorithms, checked by numerical experiments [38], with Gröbner basis and Graver Basis to extract some common steps needed (e.g. computation of shared generating sets corresponding to Gröbner basis) without the dissipate of computational cost. Interested readers may refer to [38].

In detail, for the second routine, we consider

$$\min_{x \in X \subseteq \mathbb{R}^n} \left\{ f(x) := \frac{1}{N} \sum_{i=1}^N F(x, \xi_i) \right\}, \quad (3)$$

we consider $D(C_k)$, the discrepancy of the cluster C_k , which measures how much the cost function $F(x, \tilde{\xi}_k)$ of its representative scenario $\tilde{\xi}_k$ matches the average cost values of the whole cluster C_k over the feasible set X . we then approximate $D(C_k)$ by:

$$D(C_k) \simeq \left| \frac{1}{|C_k|} \sum_{i \in C_k} F(x_k^*, \xi_i) - F(x_k^*, \tilde{\xi}_k) \right|$$

Therefore, the central ingredient of the CSSC algorithm is the opportunity-cost matrix $\mathbf{V} = (V_{i,j})$ where $V_{i,j} = F(x_i^*, \xi_j)$ and $x_i^* \in \underset{x \in X}{\operatorname{argmin}} F(x, \xi_i)$.

1.1 Prerequisite on Gröbner Basis and Graver Basis

For an integer programming problem:

$$(IP)_{c,b} : \min \{cx : Ax = b \text{ and } x \in N^n\},$$

we have Grobner Basis $\mathcal{G}_{c,A}$ and Graver Basis \mathcal{GR}_A , where the subscript represents what necessarily determines the basis.

First we recall that the reduced Gröbner Basis and test set are related as follows.

$$(x^{\alpha_i} - x^{\beta_i}) \leftrightarrow (\alpha_i - \beta_i)$$

And by choosing appropriate cost c and monomial order $>$, we can get all monomial orders in the form of the composite order $>_c$, for example, lexicographic order. See for the details in the proof of theorem 10 in [37].

When ξ_i varies,

$$F(x, \xi_i)$$

also varies. So to best diminish the computational cost, we study the relationships between the Gröbner Basis of $(IP)_{c,b} : \min \{cx : Ax = b \text{ and } x \in N^n\}$ as b, c varies. Generally, there’s no relationship between them.

Actually, no relationships can be found when b, c are varying since the change can be random. However, we can significantly reduce the time of computation by means of the following route.

Algorithm 1.2. (Compute the Gröbner Basis fixing each ξ_j)

Input: matrix A and c of $(IP)_{c,b}$

Output: a test set \mathcal{T}_c for $(IP)_{c,b}$

1. Compute a small generating set of $\text{Ker}(A)$. (**Algorithm 6.13 in [37]**)
2. Apply the bunchberger’s algorithm to compute the the Gröbner Basis of ξ_j

So from the algorithm, only A and the required ingredient of the scenario ξ_j is needed as an input. And the computational cost is greatly reduced since we don’t need to compute the Gröbner Basis of an ideal spanned by a large set any more.

1.2 Objectives

1. Accelerate the computation of Lagrangian relaxation and Lagrangian cuts with the Gröbner basis, Graver Basis, and similar objects from algebraic geometry.
2. Develop faster IP/SIP algorithms with Gröbner bases and similar algebraic objects, which is especially useful to cost-space scenario clustering (CSSC) [36].

2 Background

[8] and [9] provide literature reviews on SIP. There has been considerable progress for Benders decomposition in stochastic programming with continuous recourse [6] by iterating between a “master problem” and continuous “sub-problems.” However, originally this approach is not specially designed for stochastic integer programs, which generally have non-convex and non-continuous recourse functions [7]. Several decomposition approaches have been proposed, based on enumeration ([2]), branch and bound ([10]), branch and cut ([11]), Gomory cuts ([12]), split cuts ([13]), sample average approximation ([14]), disjunctive programming ([15]) and dynamic programming ([16]).

The ideas of Benders decomposition have been extended to integer recourse functions using Lagrangian relaxation and branch-and-bound algorithms ([3]). One of the most widely applied solution approaches is the integer L-shaped method from [17]. This method generates optimality and feasibility cuts iteratively to the master problem by leveraging the optimal value of the second-stage objective function for any first-stage integer solution. This method was extended with non-linear and Gomory cuts ([18]), with specialized branching ([10]), with thin-direction branching and multi-cut heuristics ([24]), and with alternating cuts and a cut-generating optimization model ([20]). At the same time, the integer L-shaped method is most effective when the first-stage problem has a limited solution space. Many problems, however, admit an exponentially large first-stage solution space—so that the integer L-shaped algorithm did not converge even in small instances.

These challenges motivate new ideas for proposing algorithms for stochastic integer programming. [25] proposes a novel decomposition approach that leverages the dual linear programming relaxation of the second-stage integer problem and its reduced costs. As such, their approach shares similarities with the additive bounding procedure from [26], which uses dual information from a model relaxation to derive a lower bound for NP-hard combinatorial optimization problems—with applications to the traveling salesman and vehicle routing problems ([27],[28], [29]).

2.1 Distribution-driven Scenario Generation

A challenge in stochastic programming involves defining uncertainty scenarios and their probabilities. Most applications rely on distributional assumptions regarding the uncertain parameters to sample scenarios. Scenario reduction methods have been proposed when the number of scenarios is too large to ensure computational tractability ([30], [31]). Scenario reduction, i.e. distribution-driven methods, perform scenario generation based on a criterion linked to the probability distribution that underlies the problem, and not directly the problem itself. The scenario reduction problem is typically solved using heuristic algorithms. It has been applied, for instance, to the unit commitment and capacity expansion problems in energy planning ([32], [33]).

Actually, a distribution-driven method will still operate almost in the same way when applied to two different stochastic problems with the same underlying probability distribution.

2.2 Problem-driven Scenario Generation

A problem-driven method aims at generating scenarios by considering the problem as a whole by considering underlying probability distribution and its objective function and constraints. The problem-driven approaches are logically viewed as being more powerful, since they are made to consider the problem as a whole and not solely its probability distribution. However, research in this direction has been very sparse and essentially limited to theoretical results. Julien Keutchan [36] has given a comprehensive review of existing problem-driven techniques.

Practitioners currently lack practical algorithms that could be used systematically to generate scenarios for their problem. The situation is different for the distribution-driven approaches, which provide plenty of algorithms available off the shelf, such as Monte Carlo sampling or the k-means clustering. [36] has provided one such systematic problem-driven algorithm.

3 Research Plan and Methodology

Motivated by some present results and methods employed in IP with algebraic methods, for example, the test set and universal test set, we believe there are still some interesting and challenging topics in applying Gröbner basis and Graver Basis to SIP. In particular, when the stochastic structure is complicated, or the size of scenarios are too large, we need to develop more advanced techniques combining algebraic ingredients.

3.1 Generating sub-gradients/ Lagrangian Cuts with Gröbner basis and Graver Basis

Recall that In dual decomposition for

$$\begin{aligned} \min_{x, x^s, y^s} \quad & \sum_{s \in S} p_s \left(c^T x^s + (q^s)^T y^s \right) \\ \text{s.t.} \quad & Ax^s \geq b, \\ & T^s x^s + W^s y^s \geq h^s, \quad s \in S, \\ & x^s \in X, y^s \in Y, \quad s \in S, \\ & x^s = x, \quad s \in S \end{aligned}$$

We have that Lagrangian relaxation is applied to the nonanticipativity constraints $x^s = x$ in this formulation with multipliers λ^s for $s \in S$, which gives the following Lagrangian relaxation problem:

$$\begin{aligned} z(\lambda) = \min_{x, x^s, y^s} \quad & \sum_{s \in S} p_s \left(c^T x^s + (q^s)^T y^s \right) + \sum_{s \in S} p_s (\lambda^s)^T (x^s - x) \\ \text{s.t.} \quad & (x^s, y^s) \in K^s, \quad s \in S \end{aligned}$$

where $K^s := \{x \in X, y \in Y : Ax \geq b, T^s x + W^s y \geq h^s\}$ for each $s \in S$. Throughout this paper we assume that K^s is nonempty for each $s \in S$.

The BDD algorithm is a version of Benders decomposition that uses strengthened Benders cuts and Lagrangian cuts to tighten the Benders model. In particular, they solve two types of scenario MIPs to generate optimality and feasibility cuts for E^s :

1. Given any $\lambda \in \mathbb{R}^n$, let $(\bar{x}_\lambda^s, \bar{y}_\lambda^s)$ be an optimal solution of

$$\min_{x, y} \left\{ \lambda^T x + (q^s)^T y : (x, y) \in K^s \right\}.$$

Then the following Lagrangian optimality cut is valid for E^s :

$$\lambda^T (x - \bar{x}_\lambda^s) + \theta_s \geq (q^s)^T \bar{y}_\lambda^s.$$

2. Given any $\lambda \in \mathbb{R}^n$, let $(\hat{x}_\lambda^s, \hat{y}_\lambda^s, \hat{u}_\lambda^s, \hat{v}_\lambda^s)$ be an optimal solution of

$$\min_{x, y, u, v} \{1^T v + 1^T u + \lambda^T x : Ax + u \geq b, T^s x + W^s y + v \geq h^s, x \in X, y \in Y\}.$$

Then the following Lagrangian feasibility cut is valid for E^s :

$$\lambda^T (x - \hat{x}_\lambda^s) \geq 1^T \hat{v}_\lambda^s + 1^T \hat{u}_\lambda^s.$$

The BDD algorithm generates both types of Lagrangian cuts by heuristically solving a Lagrangian cut generation problem. Our goal in this work is to provide new methods for quickly finding strong Lagrangian cuts. Here to compute every cuts we can apply algorithm 1.2 with Gröbner Basis and Graver Basis efficiently.

3.2 Statistic Question of Scenario Clustering Method

Scenario Clustering method is a problem-driven method, which means that it lacks some statistical foundations. Our aim is to prove that under certain types of probability distribution for scenarios, the variance of the difference between the real solution and approximated solution is within the good range as is studied similarly by Robust Optimization.

References

- [1] Hemmecke, R., and R. Schultz. “Decomposition of Test Sets in Stochastic Integer Programming.” *Mathematical Programming*, vol. 94, no. 2-3, 2003, pp. 323–341., doi: <https://doi.org/10.1007/s10107-002-0322-1>.
- [2] Schultz, Rüdiger, et al. “Solving Stochastic Programs with Integer Recourse by Enumeration: A Framework Using Gröbner Basis.” *Mathematical Programming*, vol. 83, no. 1-3, 1998, pp. 229–252., doi: <https://doi.org/10.1007/bf02680560>.
- [3] Carøe, Claus C., and Rüdiger Schultz. “Dual Decomposition in Stochastic Integer Programming.” *Operations Research Letters*, vol. 24, no. 1-2, 1999, pp. 37–45., doi: [https://doi.org/10.1016/s0167-6377\(98\)00050-9](https://doi.org/10.1016/s0167-6377(98)00050-9).
- [4] “Integer Programming: Lagrangian Relaxation.” SpringerReference, doi: <https://doi.org/10.1007/springerreference.72368>.
- [5] Nemhauser, George L. *Integer and Combinatorial Optimization*. John Wiley and Sons, 1999.
- [6] Benders, J. F. “Partitioning Procedures for Solving Mixed-Variables Programming Problems.” *Numerische Mathematik*, vol. 4, no. 1, 1962, pp. 238–252., doi: <https://doi.org/10.1007/bf01386316>.
- [7] Schultz, Rüdiger. “Continuity Properties of Expectation Functions in Stochastic Integer Programming.” *Mathematics of Operations Research*, vol. 18, no. 3, 1993, pp. 578–589., doi: <https://doi.org/10.1287/moor.18.3.578>.
- [8] Louveaux, François V., and Rüdiger Schultz. “Stochastic Integer Programming.” *Handbooks in Operations Research and Management Science*, 2003, pp. 213–266., doi: [https://doi.org/10.1016/s0927-0507\(03\)10004-7](https://doi.org/10.1016/s0927-0507(03)10004-7).
- [9] Sen, Suvrajeet. “Algorithms for Stochastic Mixed-Integer Programming Models.” *Discrete Optimization*, 2005, pp. 515–558., doi: [https://doi.org/10.1016/s0927-0507\(05\)12009-x](https://doi.org/10.1016/s0927-0507(05)12009-x).
- [10] Ahmed, Shabbir, et al. “A Finite Branch-and-Bound Algorithm for Two-Stage Stochastic Integer Programs.” *Mathematical Programming*, vol. 100, no. 2, 2004, pp. 355–377., doi: <https://doi.org/10.1007/s10107-003-0475-6>.
- [11] Sen, Suvrajeet, and Hanif D. Sherali. “Decomposition with Branch-and-Cut Approaches for Two-Stage Stochastic Mixed-Integer Programming.” *Mathematical Programming*, vol. 106, no. 2, 2005, pp. 203–223., doi: <https://doi.org/10.1007/s10107-005-0592-5>.
- [12] Zhang, Minjiao, and Küçükyavuz Simge. “Finitely Convergent Decomposition Algorithms for Two-Stage Stochastic Pure Integer Programs.” *SIAM Journal on Optimization*, vol. 24, no. 4, 2014, pp. 1933–1951., doi: <https://doi.org/10.1137/13092678x>.
- [13] Sen, Suvrajeet, and Julia L. Higle. “The C3 Theorem and a D2 Algorithm for Large Scale Stochastic Mixed-Integer Programming: Set Convexification.” *Mathematical Programming*, vol. 104, no. 1, 2005, pp. 1–20., doi: <https://doi.org/10.1007/s10107-004-0566-z>.
- [14] Kleywegt, Anton J., et al. “The Sample Average Approximation Method for Stochastic Discrete Optimization.” *SIAM Journal on Optimization*, vol. 12, no. 2, 2002, pp. 479–502., doi: <https://doi.org/10.1137/s1052623499363220>.
- [15] Ntaimo, Lewis. “Disjunctive Decomposition for Two-Stage Stochastic Mixed-Binary Programs with Random Recourse.” *Operations Research*, vol. 58, no. 1, 2010, pp. 229–243., doi: <https://doi.org/10.1287/opre.1090.0693>.
- [16] Zou, Jikai, et al. “Stochastic Dual Dynamic Integer Programming.” *Mathematical Programming*, vol. 175, no. 1-2, 2018, pp. 461–502., doi: <https://doi.org/10.1007/s10107-018-1249-5>.
- [17] Laporte, Gilbert, and François V. Louveaux. “The Integer L-Shaped Method for Stochastic Integer Programs with Complete Recourse.” *Operations Research Letters*, vol. 13, no. 3, 1993, pp. 133–142., doi: [https://doi.org/10.1016/0167-6377\(93\)90002-x](https://doi.org/10.1016/0167-6377(93)90002-x).

- [18] Carøe, Claus C., and Jørgen Tind. “L-Shaped Decomposition of Two-Stage Stochastic Programs with Integer Recourse.” *Mathematical Programming*, vol. 83, no. 1-3, 1998, pp. 451–464., doi: <https://doi.org/10.1007/bf02680570>.
- [19] Kim, Kibaek, and Sanjay Mehrotra. “A Two-Stage Stochastic Integer Programming Approach to Integrated Staffing and Scheduling with Application to Nurse Management.” *Operations Research*, vol. 63, no. 6, 2015, pp. 1431–1451., doi: <https://doi.org/10.1287/opre.2015.1421>.
- [20] Angulo, Gustavo, et al. “Improving the Integer L-Shaped Method.” *INFORMS Journal on Computing*, vol. 28, no. 3, 2016, pp. 483–499., doi: <https://doi.org/10.1287/ijoc.2016.0695>.
- [21] Laporte, Gilbert, and François V. Louveaux. “The Integer L-Shaped Method for Stochastic Integer Programs with Complete Recourse.” *Operations Research Letters*, vol. 13, no. 3, 1993, pp. 133–142., doi: [https://doi.org/10.1016/0167-6377\(93\)90002-x](https://doi.org/10.1016/0167-6377(93)90002-x).
- [22] : Ahmed, Shabbir, and Nikolaos V. Sahinidis. “An Approximation Scheme for Stochastic Integer Programs Arising in Capacity Expansion.” *Operations Research*, vol. 51, no. 3, 2003, pp. 461–471., doi: <https://doi.org/10.1287/opre.51.3.461.14960>.
- [23] Gunpinar, Serkan, and Grisselle Centeno. “Stochastic Integer Programming Models for Reducing Wastages and Shortages of Blood Products at Hospitals.” *Computers Operations Research*, vol. 54, 2015, pp. 129–141., doi: <https://doi.org/10.1016/j.cor.2014.08.017>.
- [24] Kim, Kibaek, and Sanjay Mehrotra. “A Two-Stage Stochastic Integer Programming Approach to Integrated Staffing and Scheduling with Application to Nurse Management.” *Operations Research*, vol. 63, no. 6, 2015, pp. 1431–1451., doi: <https://doi.org/10.1287/opre.2015.1421>.
- [25] Wang, Kai, and Alexandre Jacquillat. “A Stochastic Integer Programming Approach to Air Traffic Scheduling and Operations.” *Operations Research*, vol. 68, no. 5, 2020, pp. 1375–1402., doi: <https://doi.org/10.1287/opre.2020.1985>.
- [26] Fischetti, Matteo, and Paolo Toth. “An Additive Bounding Procedure for Combinatorial Optimization Problems.” *Operations Research*, vol. 37, no. 2, 1989, pp. 319–328., doi: <https://doi.org/10.1287/opre.37.2.319>.
- [27] Fischetti, Matteo, and Paolo Toth. “An Additive Bounding Procedure for the Asymmetric Travelling Salesman Problem.” *Mathematical Programming*, vol. 53, no. 1-3, 1992, pp. 173–197., doi: <https://doi.org/10.1007/bf01585701>.
- [28] Baldacci, Roberto, et al. “An Exact Algorithm for the Vehicle Routing Problem Based on the Set Partitioning Formulation with Additional Cuts.” *Mathematical Programming*, vol. 115, no. 2, 2007, pp. 351–385.,doi: <https://doi.org/10.1007/s10107-007-0178-5>.
- [29] Baldacci, Roberto, and Aristide Mingozzi. “A Unified Exact Method for Solving Different Classes of Vehicle Routing Problems.” *Mathematical Programming*, vol. 120, no. 2, 2008, pp. 347–380., doi: <https://doi.org/10.1007/s10107-008-0218-9>.
- [30] Dupacova, J., et al. “Scenario Reduction in Stochastic Programming.” *Mathematical Programming*, vol. 95, no. 3, 2003, pp. 493–511., doi: <https://doi.org/10.1007/s10107-002-0331-0>.
- [31] Römis, Werner. “Scenario Reduction Techniques in Stochastic Programming.” *Stochastic Algorithms: Foundations and Applications*, 2009, pp. 1–14., doi: https://doi.org/10.1007/978-3-642-04944-6_1.
- [32] Carrion, Miguel, et al. “A Stochastic Programming Approach to Electric Energy Procurement for Large Consumers.” *IEEE Transactions on Power Systems*, vol. 22, no. 2, 2007, pp. 744–754., doi: <https://doi.org/10.1109/tpwrs.2007.895164>.
- [33] Morales, J.M., et al. “Scenario Reduction for Futures Market Trading in Electricity Markets.” *IEEE Transactions on Power Systems*, vol. 24, no. 2, 2009, pp. 878–888., doi: <https://doi.org/10.1109/tpwrs.2009.2016072>.
- [34] Kall, Peter, and Stein W. Wallace. *Stochastic Programming*. John Wiley Sons, 1997.

- [35] Biase, Fausto Di, and Rüdiger Urbanke. “An Algorithm to Calculate the Kernel of Certain Polynomial Ring Homomorphisms.” *Experimental Mathematics*, vol. 4, no. 3, 1995, pp. 227–234., doi: <https://doi.org/10.1080/10586458.1995.10504323>.
- [36] Keutchayan, Julien, et al. “Problem-Driven Scenario Clustering in Stochastic Optimization.” *ArXiv.org*, 22 June 2021, <https://arxiv.org/abs/2106.11717>.
- [37] Yuchen Ge. “Algebra for Machine Learning and Stochastic Programming.” August, <https://gycdwwd.github.io/Writing/Algsto1.pdf>.
- [38] Yuchen Ge. “Algebra for Machine Learning and Stochastic Programming II.” August, <https://gycdwwd.github.io/Writing/Algsto2.pdf>.
- [39] Yuchen Ge. “Algebra for Machine Learning and Stochastic Programming III.” August, <https://gycdwwd.github.io/Writing/Algsto3.pdf>.
- [40] Yuchen Ge. “Algebra for Machine Learning and Stochastic Programming IV.” August, <https://gycdwwd.github.io/Writing/Algsto4.pdf>.
- [41] Chen, Rui, and James Luedtke. “On Generating Lagrangian Cuts for Two-Stage Stochastic Integer Programs.” *INFORMS Journal on Computing*, vol. 34, no. 4, 2022, pp. 2332–2349., <https://doi.org/10.1287/ijoc.2022.1185>.