GG-DFF: GEOMETRY-GUIDED DECISION-FOCUSED FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Decision-Focused Learning (DFL) trains predictive models for downstream optimization using decision quality as the learning signal. Existing methods make DFL tractable by differentiating through optimization layers, smoothing the optimizer with perturbations, or replacing the true decision loss with tractable surrogates. However, in linear and mixed-integer linear programs with linear objectives, the convex hull of feasible decisions induces a partition of the objective-parameter space into regions associated with different optimal decisions. The decision consequences of a prediction correction depend on the region it reaches, while equal reductions in surrogate loss can lead to decisions of different quality. To address this issue, we propose Geometry-Guided Decision-Focused Fine-Tuning (GG-DFF), which uses feasible-solution differences to guide prediction corrections. We compare the decision under the original prediction with collected feasible solutions. We then use the resulting difference vectors to construct a convex correction region (CCR) around the original prediction, which guides which decision regions prediction corrections can reach. Specifically, we freeze the pretrained predictor and learn corrections through an almost-everywhere differentiable parameterization of the CCR. This parameterization keeps corrected predictions inside the region and allows SPO+ gradients to propagate through its comparison geometry. Our theoretical analysis connects correction boundaries to decision-region reachability, establishes sufficient conditions for excluding decision regions, and derives conditional regret bounds relative to the pretrained decision. Experimental results show that GG-DFF reduces mean normalized decision regret relative to the reference fine-tuning baseline by up to 17.7%, 1.0%, 2.9%, and 2.2% on shortest path, knapsack, Energy-1, and Energy-2, respectively.

1 INTRODUCTION

Decision-Focused Learning (DFL) extends standard predict-then-optimize pipelines by training predictive models with downstream decision quality rather than prediction error alone (Elmachtoub & Grigas, 2022; Mandi et al., 2022; Sadana et al., 2025; Mandi et al., 2020). This perspective is part of a broader literature connecting prediction, contextual information, and optimization (Bertsimas & Kallus, 2020; Ban & Rudin, 2019; Donti et al., 2017; Rychener et al., 2023; Estes & Richard, 2023; Elmachtoub et al., 2025). In standard predict-then-optimize training, prediction and optimization are typically treated as separate stages; in DFL, the downstream optimization problem shapes the learning signal used to update the predictor.

A central difficulty in DFL is that decision losses are generally non-smooth in the predicted parameters. Existing methods make training tractable by differentiating through optimization layers, smoothing or perturbing the optimizer, or replacing the true decision loss with surrogate losses from Smart Predict-then-Optimize (SPO) (Amos & Kolter, 2017; Agrawal et al., 2019; Blondel et al., 2022; Berthet et al., 2020; Dalle et al., 2022; Butler & Kwon, 2023). However, equal reductions in a training loss can lead to decisions of different quality.

This issue can arise in linear programs (LPs) and mixed-integer linear programs (MILPs) with linear objectives. In these problems, the predicted objective coefficients determine which feasible decision attains the optimal objective value. A decision's minimization normal cone consists of all objective coefficients for which that decision is optimal (Farina, 2025). Figure 1(a) illustrates the feasible deci-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

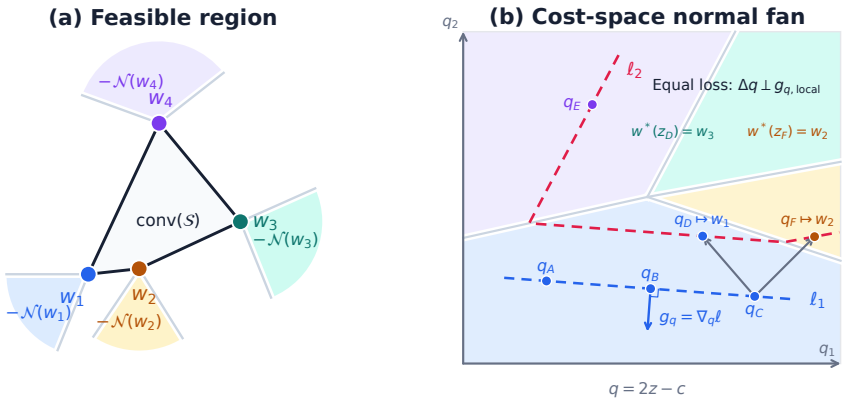


Figure 1: (a) Feasible solutions with outward cones $-\mathcal{N}_{\mathcal{P}}(w_i)$. (b) SPO+ level sets in $q = 2z - c$: the two arrows yield equal loss reductions and induce different final decisions at z.

sions and their normal-cone geometry. When a correction path is orthogonal to the local training-loss gradient on each smooth segment, the loss remains constant along the path. Thus, crossing decision boundaries along such a path can yield decisions of different quality at the same loss (Appendix A.9 for SPO+; Appendix A.10 for KKT differentiation). Figure 1(b) shows SPO+ level sets in reflected-cost coordinates: each dashed line connects points with the same SPO+ loss. The two arrows start from the same point and end on the red contour. After converting the endpoints to predictions, they select different final decisions despite equal loss reductions.

To address this gap, we propose Geometry-Guided Decision-Focused Fine-Tuning (GG-DFF), which uses feasible-solution comparisons to guide which decision regions prediction corrections can reach. Before fine-tuning, we solve downstream problems on training examples and collect the feasible solutions into a fixed bank. Comparisons between the pretrained decision and these solutions determine the boundary directions; uniform slacks or direction-dependent training-error information determine the boundary positions, forming a convex correction region (CCR). These boundaries can exclude some decision regions, so predictions with the same training loss need not all remain reachable. GG-DFF freezes the pretrained predictor and introduces a network that learns prediction corrections. The network's output passes through an almost-everywhere differentiable CCR parameterization to produce a corrected prediction inside the region. SPO+ gradients backpropagate through this map, incorporating solution-comparison geometry into network training. We further establish sufficient conditions for the boundaries to exclude specified decision regions and derive conditional performance bounds relative to the pretrained decision.

Our contributions are as follows:

- We characterize the relationship between solution-comparison constraints and normal-cone reachability, and derive sufficient conditions under which a comparison slack excludes a retained decision's normal cone.
- We propose an almost-everywhere differentiable CCR parameterization with uniform and direction-dependent slacks, incorporating comparison geometry into output constraints while retaining the standard SPO+ training objective.
- We establish conditional performance bounds relative to the pretrained decision and derive sufficient conditions for a strictly better worst-case performance bound than the box used by decision-focused fine-tuning (DFF).

2 Related Work

Decision-Focused Learning and Fine-Tuning. DFL methods obtain downstream training signals through differentiation, perturbation, or surrogate losses. OptNet and differentiable convex optimization layers differentiate through optimality systems (Amos & Kolter, 2017; Agrawal et al., 2019), including interior-point formulations for LP-based prediction and optimization (Mandi & Guns, 2020).

Within this approach, PEAR uses active constraints identified in the forward solve to compute gradients through a reduced linear system (Lee et al., 2026). For combinatorial optimization, blackbox differentiation constructs a backward pass from solver outputs (Vlastelica Pogančić et al., 2020), while perturbation-based methods smooth optimizer outputs or define probabilistic optimization layers (Berthet et al., 2020; Dalle et al., 2022). Surrogate losses provide another way to train predictors using downstream decisions. These include SPO+ (Elmachtoub & Grigas, 2022), learned decision losses such as LODL, and task-dependent metrics such as TaskMet (Shah et al., 2022; Bansal et al., 2023). LAVA constructs its loss by comparing the ground-truth optimum with precomputed adjacent vertices, avoiding repeated optimization solves during training (Berden et al., 2025). Training with decision losses can substantially shift predictions, particularly when data are limited (Yang et al., 2025). DFF addresses this issue by freezing the pretrained predictor and learning output corrections within a trust region. Its multiplicative parameterization limits coordinatewise changes while allowing adaptation to the downstream objective. In smart energy systems, Beichter et al. (2024) also study decision-focused retraining of pretrained forecasting models. GG-DFF adopts the frozen-predictor framework of DFF and retains the SPO+ objective. It uses feasible-solution comparisons to construct the correction region, linking the permitted prediction changes to the downstream decisions they can reach.

Solution Ranking and Optimality Margins. Solution comparisons provide a way to express which decisions predicted objectives should favor. Structured prediction encodes output preferences through margin constraints and uses cutting-plane algorithms to retain informative comparisons (Tsochantaridis et al., 2005; Joachims et al., 2009). Related uses of optimality conditions include input correction through inverse optimization for structured prediction (Daumé et al., 2015) and objective-parameter learning through Maximum Optimality Margin in contextual and inverse linear programming (Sun et al., 2023). In predict-and-optimize learning, Mulamba et al. (2021) use non-optimal feasible solutions as negative examples and cache solutions to reduce repeated oracle calls. Building on this approach, Mandi et al. (2022) develop pointwise, pairwise, and listwise ranking losses; their pairwise formulation compares the observed-cost optimum with cached alternatives. These methods use comparisons and optimality conditions to construct learning objectives or determine objective parameters. GG-DFF uses comparisons between the pretrained decision and solutions collected before fine-tuning to define the boundaries of a convex correction region. Its almost-everywhere differentiable parameterization carries this comparison geometry into both the corrected predictions and backpropagation under the SPO+ objective.

3 Background

3.1 Decision-Focused Learning and the SPO+ Surrogate

For a context x, let $c \in \mathbb{R}^d$ denote the true objective coefficients and $\hat{c} \in \mathbb{R}^d$ their prediction. We state the formulation for minimization. Let $\mathcal{S} \subseteq \mathbb{R}^d$ be the nonempty, compact feasible decision set, and assume that $\mathcal{P} := \text{conv}(\mathcal{S})$ is a polytope with finite vertex set $\mathcal{V}$. A decision $w \in \mathcal{S}$ is expressed in the coordinates paired with the cost vector; Appendix C.1.3 gives this representation for energy scheduling. The optimization oracle uses a fixed tie-breaking rule and returns $w^*(\hat{c}) \in \arg\min_{w \in \mathcal{S}} \hat{c}^\top w$. DFL trains the predictor using downstream decision quality, measured by the regret

$$\text{Regret}(\hat{c}, c) = c^\top w^*(\hat{c}) - c^\top w^*(c). \tag{1}$$

The SPO+ framework provides a convex surrogate for this regret (Elmachtoub & Grigas, 2022):

$$\ell_{\text{SPO+}}(\hat{c}, c) = \max_{w \in \mathcal{S}} \{(c - 2\hat{c})^\top w\} + 2\hat{c}^\top w^*(c) - z^*(c), \tag{2}$$

where $z^*(c) = c^\top w^*(c)$. The maximization term is evaluated using $v_{\text{SPO+}}(\hat{c}, c) \in \arg\min_{w \in \mathcal{S}} (2\hat{c} - c)^\top w$. One valid subgradient with respect to $\hat{c}$ is $2(w^*(c) - v_{\text{SPO+}}(\hat{c}, c))$. For fixed c and a corrected prediction z, write $q = 2z - c$. With the same oracle choice, denote the subgradient with respect to q by $g_q = w^*(c) - v_{\text{SPO+}}(z, c)$ and the subgradient with respect to z by $g_z = 2g_q$.

Bounded fine-tuning. DFF freezes a pretrained predictor $c_0(x)$ and learns an input-dependent output correction (Yang et al., 2025). Its multiplicative parameterization confines the corrected prediction z to the box

$$\mathcal{T}_\epsilon(c_0) = \{z \in \mathbb{R}^d : |z_j - c_{0,j}| \le \epsilon |c_{0,j}|, \quad j = 1, \dots, d\}, \qquad \epsilon \ge 0. \tag{3}$$

Here and below, the dependence of c_0 on the context is suppressed when discussing a fixed instance. The parameter ϵ controls the allowed displacement from the pretrained prediction, while the decision-focused objective trains the correction within this region.

3.2 The Polyhedral Geometry of Decisions

For both LPs and MILPs, a linear objective has the same optimal value over $\mathcal{S}$ and $\mathcal{P}$, and an extreme optimal solution can be selected from a vertex of $\mathcal{P}$. The objective-parameter space is partitioned by the normal fan of $\mathcal{P}$. For any $v \in \mathcal{P}$, its minimization normal cone is

$$\mathcal{N}_{\mathcal{P}}(v) = \{z \in \mathbb{R}^d : z^\top v \leq z^\top w, \quad \forall w \in \mathcal{P}\}. \tag{4}$$

In the interior of a vertex normal cone, the unique optimal decision is that vertex. At a cone boundary, the oracle selects a decision on the exposed optimal face. Thus the optimizer map is locally constant in vertex-cone interiors, with $\nabla_z w^*(z) = 0$ almost everywhere. A prediction can move within one such region and change its prediction error while leaving the induced decision unchanged. The SPO+ surrogate supplies the subgradient defined above for training through this decision geometry.

For a reference decision $w_0 \in \mathcal{S}$, linear optimality over $\mathcal{P}$ can be checked at its vertices, giving the halfspace representation

$$\mathcal{N}_{\mathcal{P}}(w_0) = \bigcap_{v \in \mathcal{V}} \{z \in \mathbb{R}^d : z^\top (v - w_0) \geq 0\}. \tag{5}$$

For distinct feasible decisions v and w_0, the difference $v - w_0$ is normal to their cost-equality hyperplane $z^\top(v - w_0) = 0$; the corresponding inequality means that w_0 is no more costly than v. A shared boundary of vertex normal cones additionally requires both compared vertices to be optimal. Available solution differences therefore supply comparison directions without requiring an explicit construction of the full normal fan. Section 5 uses these directions to parameterize a convex correction region around the pretrained prediction, through which SPO+ trains the correction network.

4 Motivation: Surrogate Progress and Decision Quality

A useful correction must change the predicted costs in a way that improves the selected decision. The SPO+ surrogate supplies a tractable training signal. Corrections with equal surrogate descent can reach decisions of different quality. For a fixed instance, let $v_{\text{SPO+}} = w^*(2c_0 - c)$ and $g_z = 2(w^*(c) - v_{\text{SPO+}})$. Whenever a corrected prediction z retains this SPO+ oracle solution,

$$\ell_{\text{SPO+}}(z, c) - \ell_{\text{SPO+}}(c_0, c) = g_z^\top (z - c_0). \tag{6}$$

Consequently, corrections with the same inner product with g_z have identical surrogate progress. Their components orthogonal to g_z can nevertheless change the relative costs of other feasible decisions. This matters in shared-network training, where the network Jacobian, minibatch aggregation, and optimizer jointly determine how each prediction moves.

We use a training instance from the shortest-path benchmark described in Section 6.1. Its anchor prediction c_0 is produced by a predictor pretrained for 50 epochs using mean squared error (MSE) and then frozen. We construct two corrections to this same prediction. Both corrections lie strictly inside the existing DFF box and move by the same Euclidean distance, 0.25% of the anchor norm. They also retain the same nonzero prediction-space SPO+ subgradient along their entire interpolation from the anchor. Figure 2(a) shows that both corrections reduce SPO+ loss to the same value, a decrease of 10.79% from the anchor. Yet their induced decisions have opposite consequences: correction A reaches the true optimal path, whereas correction B raises normalized decision regret (NDR) from 5.34% to 17.84% (Figure 2(b)). Complete enumeration of all 20 paths verifies these quantities; Appendix A.9 gives the derivation, and Appendix D.1 gives the experimental construction and numerical checks.

The distinction is geometric: the SPO+ queries $2z_A - c$ and $2z_B - c$ remain in the same normal cone, while the predictions z_A and z_B enter different normal cones and select paths of different quality. Thus the same local surrogate signal and the same reduced loss value can correspond to different decision outcomes. Solution comparisons make the affected boundaries explicit through $z^\top(v - w_0) = 0$. This motivates defining admissible corrections in terms of changes in these comparison

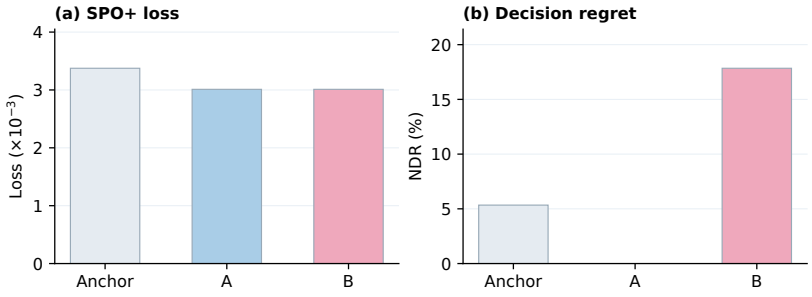


Figure 2: A training-instance example from the shortest-path benchmark. Equal-norm corrections A and B reduce SPO+ loss to the same value but enter different decision regions, yielding different regrets.

gaps. GG-DFF uses the resulting halfspaces to parameterize a convex correction region, and trains within it using SPO+. The construction gives comparison geometry an explicit role in determining which prediction changes are available; the subsequent theory and experiments examine the resulting decision reachability and quality.

5 Decision-Focused Fine-Tuning with a Convex Correction Region

We use feasible-solution comparisons to parameterize the region in which a pretrained prediction can be corrected. Solution differences determine the boundary directions, and positive comparison slacks determine their offsets. The resulting *convex correction region* (CCR) is part of the predictor's forward map, and SPO+ trains the correction network through this map. We consider two slack settings within the same method: uniform (CCR-U) and direction-dependent (CCR-D). Both retain the comparison geometry and the same end-to-end training procedure. We report these settings as GG-DFF-U and GG-DFF-D in the experimental comparison.

5.1 From Solution Comparisons to Correction Boundaries

Let $\sigma = +1$ for minimization and $\sigma = -1$ for maximization, and define $\mathcal{O}_\sigma(c) \in \arg\min_{w \in \mathcal{S}} \sigma c^\top w$. We retain the original cost or profit coordinates. For a frozen MSE-pretrained predictor $c_0(x)$, let $w_0(x) = \mathcal{O}_\sigma(c_0(x))$. A retained feasible decision v defines the signed comparison $d_v(x) = \sigma(v - w_0(x))$. The hyperplane $z^\top d_v(x) = 0$ compares v with the anchor, and these normals encode the corresponding decision geometry.

For a fixed feasible set, we form a bank $\mathcal{H}$ of distinct training solutions obtained at $c_0(x_i)$, c_i, and $2c_0(x_i) - c_i$. When all feasible extreme-point decisions can be enumerated, we use that complete bank. Given positive slacks $s_v(x)$, define

$$\mathcal{C}_s(x) = \bigcap_{v \in \mathcal{H}} \left\{ z : (z - c_0(x))^\top d_v(x) \geq -s_v(x) \right\}. \tag{7}$$

The region is convex and contains the anchor in its interior. For $d_v \neq 0$, the distance from the anchor to this defining hyperplane is $s_v / \|d_v\|_2$. Thus even a common slack produces a geometry-dependent region, with orientations and Euclidean distances determined by the decision differences.

5.2 Uniform and Direction-Dependent Slacks

We express the correction budget in objective units using the training scale

$$L = \left(\frac{1}{|I_{\text{tr}}|} \sum_{i \in I_{\text{tr}}} [c_0(x_i)^\top w_0(x_i)]^2 \right)^{1/2}, \qquad b = \rho L, \tag{8}$$

where $L > 0$ and $\rho > 0$ is selected using validation NDR. The uniform setting uses $s_v = b$ for every comparison.

For the direction-dependent setting, the training residuals define

$$\mathcal{E} = \mathrm{conv}\big(\{0\} \cup \{c_i - c_0(x_i) : i \in I_{\mathrm{tr}}\}\big), \quad e_v(x) = h_{\mathcal{E}}(d_v(x)), \quad E_{\max}(x) = \max_{v \in \mathcal{H}} e_v(x). \tag{9}$$

The support $h_{\mathcal{E}}(d) = \max_{e \in \mathcal{E}} e^\top d$ measures the largest residual contribution to a comparison and is computed from training-residual inner products. The two settings are

$$s_v(x) = \begin{cases} b, & \text{CCR-U}, \\ b + E_{\max}(x) - e_v(x), & \text{CCR-D}. \end{cases} \tag{10}$$

Both satisfy $s_v \geq b > 0$. The bank and the chosen slack rule are fixed during fine-tuning. CCR-U uses a common comparison budget; CCR-D allows additional movement along comparisons with smaller directional error support. The uniform construction can omit residual-support evaluation.

Following the set-valued advice perspective of Golrezaei et al. (2026), the error set also provides a common performance certificate. At the same b, both settings satisfy $e_v + s_v \leq E_{\max} + b$, and CCR-U is contained in CCR-D. This relation is proved in Appendix A.3. They are alternative boundary settings of the same CCR method; experiments select ρ by validation NDR for each setting and report their results separately.

5.3 A Boundary Parameterization for End-to-End Training

For either slack setting, define the nonnegative gauge

$$p_x(u) = \max\left\{0, \max_{v \in \mathcal{H}} \frac{-u^\top d_v(x)}{s_v(x)}\right\}. \tag{11}$$

For a correction network f_θ, parameterize the prediction by

$$u_\theta(x) = a\, c_0(x) \odot f_\theta(x), \qquad z_\theta(x) = c_0(x) + \frac{u_\theta(x)}{1 + p_x(u_\theta(x))}, \tag{12}$$

where $a > 0$ is a fixed coordinate scale. The final layer of f_θ is initialized at zero, so fine-tuning starts at $z_\theta = c_0$. Positive homogeneity gives

$$p_x\left(\frac{u}{1 + p_x(u)}\right) = \frac{p_x(u)}{1 + p_x(u)} < 1. \tag{13}$$

Every finite forward output therefore satisfies the CCR constraints. The map covers the region's interior in unrestricted correction coordinates; its inverse and Jacobian are proved in Appendix A.5.

We minimize the signed SPO+ objective defined in Appendix A.8,

$$\min_\theta \frac{1}{|I_{\mathrm{tr}}|} \sum_{i \in I_{\mathrm{tr}}} \ell^\sigma_{\mathrm{SPO+}}(z_\theta(x_i), c_i), \qquad g_{z,i} = 2\sigma\big(\mathcal{O}_\sigma(c_i) - \mathcal{O}_\sigma(2z_\theta(x_i) - c_i)\big). \tag{14}$$

Both settings backpropagate through the full map in (12); Appendix A.5 specifies branch derivatives and tie handling. The map enforces comparison constraints using inner products, a maximum, and scalar division, with unbounded movement along permitted directions. Algorithm 1 gives the procedure.

5.4 Geometric and Performance Interpretation

For a retained vertex v, (7) implies $z^\top d_v \geq c_0^\top d_v - s_v$. If $c_0^\top d_v > s_v$, the region excludes the decision cone $\{z : z^\top \sigma(w - v) \geq 0,\ \forall w \in \mathcal{S}\}$ and enforces a positive comparison margin of $(c_0^\top d_v - s_v)/\|d_v\|_2$. This connects both boundary settings to downstream decision geometry.

Write $\mathrm{Regret}_\sigma(z, c) = \sigma c^\top(\mathcal{O}_\sigma(z) - \mathcal{O}_\sigma(c))$. Under comparison coverage for the final decision and the directional-error bound $h_{\mathcal{E}}$, the common certificate gives

$$\mathrm{Regret}_\sigma(z, c) \leq \mathrm{Regret}_\sigma(c_0, c) + \tau_s, \qquad \tau_s = \max_{v \in \mathcal{H}}(e_v + s_v) = E_{\max} + \rho L. \tag{15}$$

Proposition A.2 states this guarantee for general positive slacks. Restricting a region improves its exact worst-case lower bound when it excludes every reachable vertex attaining the previous worst case (Proposition A.11). Appendix A.6 gives sufficient conditions for a strictly higher lower bound than DFF's box, applicable to either setting under full comparison coverage. The separate effect of optimizer normalization on auxiliary margin gradients is analyzed in Appendix A.12. For KP, $\sigma = -1$: the comparisons are $d_v = w_0 - v$ and the SPO+ subgradient is $2(\mathcal{O}_{-1}(2z - c) - \mathcal{O}_{-1}(c))$ in profit coordinates. The bound controls profit loss relative to the anchor.

6 EXPERIMENTS

We evaluate GG-DFF on shortest path (SP), 0–1 knapsack (KP), and two energy-cost aware scheduling benchmarks, Energy-1 and Energy-2. These tasks cover path selection, resource allocation, and scheduling, with linear objectives whose coefficients are predicted from contextual features.

6.1 BENCHMARKS

Shortest path. We use the grid shortest-path benchmark studied in predict-and-optimize learning (Elmachtoub & Grigas, 2022; Mandi et al., 2022). The predictor maps five contextual features to 24 edge costs on a directed 4×4 grid. The downstream task selects a minimum-cost path from the source to the destination, moving right or down. The grid admits 20 feasible paths. Its flow formulation is an LP with integral optimal extreme points, allowing exact path comparisons.

Knapsack. We use the published weighted-knapsack data of Mandi et al. (2020). The predictor estimates the profits of 48 items from nine features per item. The downstream task selects a subset maximizing total profit subject to a weight-capacity constraint, giving a binary integer program. We evaluate capacities $60, 90, 120, 150, 180, 210$ and average them equally; each capacity defines a separate feasible decision set.

Energy-cost aware scheduling. We follow the predict-and-optimize scheduling benchmark of Mandi et al. (2020), based on CSPLib Problem 059 (Simonis et al., n.d.). The predictor estimates 48 half-hour electricity prices from the Single Electricity Market Operator (SEMO) data used in energy-price forecasting (Ifrim et al., 2012). Given these prices, the downstream MILP assigns tasks to machines and start times to minimize electricity cost while respecting task durations, time windows, and machine resource capacities. Following the benchmark naming convention of Mandi et al. (2022), we report the two scheduling specifications separately: Energy-1 has 10 tasks on 3 machines, and Energy-2 has 15 tasks on 3 machines. Both use the same price data, but their scheduling constraints define different feasible sets. Appendix C.1 gives the formulations, and Appendix C.2 gives the data splits.

6.2 EVALUATION PROTOCOL

Methods and training. We compare MSE, SPO+, DFF, LODL (Shah et al., 2022), perturbed Fenchel–Young loss (PFYL) (Berthet et al., 2020), 2-Stage (Hu et al., 2023; Mandi & Guns, 2020), and pairwise learning to rank (prLTR) (Mandi et al., 2022). DFF and GG-DFF share a fixed MSE predictor trained for 50 epochs and train their correction networks for 20 epochs. All benchmark comparisons and ablation controls use eight independent random seeds for each method and benchmark. The motivation figures additionally present constructed single-instance examples. DFF and both GG-DFF variants use the same eight seeds, with correction initialization and minibatch randomness paired across methods within each seed. We report the uniform and direction-dependent settings as GG-DFF-U and GG-DFF-D, respectively. All methods, including the standalone baselines, select checkpoints using validation NDR. CCR budgets are also selected using validation NDR. Baseline protocols are specified in Appendix C.4, and seed pairing and aggregation in Appendix C.5.

Metric and aggregation. For minimization, the instance regret is $R_i = [c_i^\top w^*(z_i) - c_i^\top w^*(c_i)]_+$; for maximization, the objective difference is reversed. We report

$$\text{NDR} = \frac{1}{N} \sum_{i=1}^{N} \frac{R_i}{|c_i^\top w^*(c_i)| + \varepsilon_{\text{num}}}, \qquad \varepsilon_{\text{num}} = 10^{-8}. \tag{16}$$

Each seed gives one test-set mean. KP first averages these means equally across the six capacities within a seed. Table 1 reports the mean NDR and MSE across the eight independent random seeds. Each energy benchmark is evaluated over 111 test days. Appendix D.7 shows test-instance regret distributions for the same runs.

Table 1: Normalized regret and MSE averaged over eight independent random seeds (Appendix C.5). Lower is better. Best and second-best NDR values are shown in **bold** and underlined, respectively.

Method	Shortest Path		Knapsack		Energy-1		Energy-2	
	NDR ↓ ($\times 10^{-2}$)	MSE ↓ ($\times 10^{-3}$)	NDR ↓ ($\times 10^{-2}$)	MSE ↓ ($\times 10^{4}$)	NDR ↓ ($\times 10^{-2}$)	MSE ↓ ($\times 10^{3}$)	NDR ↓ ($\times 10^{-2}$)	MSE ↓ ($\times 10^{3}$)
MSE	0.969	0.545	4.864	2.179	1.991	1.034	2.222	1.025
SPO+	1.767	415.981	4.842	7.017	2.229	3.638	2.285	3.563
DFF	0.278	0.830	4.703	2.239	1.835	1.020	1.931	1.010
LODL	0.386	0.411	5.902	2.456	2.048	1.035	2.294	1.045
PFYL	4.881	371.039	5.351	14.641	2.542	6.573	2.464	6.570
2-Stage	2.370	362.407	11.629	14.650	2.122	6.473	2.101	6.582
prLTR	3.728	495.283	7.149	2.844	2.836	6.583	3.076	6.583
GG-DFF-U	<u>0.246</u>	0.821	**4.656**	2.783	**1.783**	1.019	<u>1.900</u>	1.017
GG-DFF-D	**0.229**	0.854	<u>4.665</u>	2.740	<u>1.789</u>	1.019	**1.890**	1.017

6.3 Results

Overall decision quality. GG-DFF-U and GG-DFF-D achieve the two lowest mean NDR values on all four benchmarks in Table 1, with DFF being the strongest remaining baseline. On SP, GG-DFF-D reduces NDR from DFF's 0.278 to 0.229, a relative reduction of 17.7%, while GG-DFF-U achieves 0.246, an 11.5% reduction. On KP, GG-DFF-U and GG-DFF-D attain 4.656 and 4.665, respectively, compared with 4.703 for DFF, corresponding to reductions of 1.0% and 0.8%. On Energy-1, they reduce NDR from 1.835 to 1.783 and 1.789, yielding reductions of 2.9% and 2.5%. On Energy-2, GG-DFF-D performs better, reducing NDR from 1.931 to 1.890 (2.2%), while GG-DFF-U obtains 1.900 (1.6%). Both settings improve mean decision quality over the DFF box, although their ordering varies across benchmarks. These improvements do not consistently coincide with lower prediction error: on KP, both GG-DFF settings achieve lower NDR despite higher MSE than DFF.

Surrogate-loss reduction and regret. We conduct a post-hoc analysis on the SP test set after fixing the validation-selected checkpoints. Since all 20 feasible paths can be enumerated, we directly compute the SPO+ loss, selected path, and regret for each saved prediction. Figure 3 compares these outcomes with those of the frozen pretrained predictor. Among lower-loss comparisons in panel (c), regret remains unchanged in 71.05% of cases for DFF, 72.73% for GG-DFF-U, and 71.05% for GG-DFF-D. The three methods also have similar mean counts of regret-improving comparisons; Appendix D.7.1 provides the detailed statistics and comparison protocol. In SP, the optimizer selects the path with the smallest sum of predicted edge costs. If a correction stays within the same normal-cone interior, the selected path, its true cost, and its regret remain unchanged, even though SPO+ loss may decrease. Thus, normal-cone geometry affects the outcome of DFL training: a lower training loss may leave decision quality unchanged. Loss reduction can also accompany worse decisions: DFF exhibits increased regret in 2.63% of its lower-loss comparisons, while neither GG-DFF variant exhibits such a mismatch in this evaluation. Panels (a,b) illustrate the difference: DFF lowers SPO+ loss from 16.972×10^{-3} to 4.619×10^{-3} but increases NDR from 4.886% to 6.367%. GG-DFF-U preserves the pretrained decision and its NDR despite a higher loss of 5.196×10^{-3}, whereas GG-DFF-D reaches the true optimal path with loss 2.699×10^{-3} and zero regret. In this evaluation, both CCR settings therefore reduce loss with unchanged or improved regret.

Boundary ablation study. Table 2 compares unbounded SPO+ correction, the DFF box, and both CCR settings on SP, using the same frozen pretrained predictor, correction architecture, SPO+ objective, and paired training randomness. All checkpoints are selected by validation NDR. GG-DFF-U and GG-DFF-D reduce mean NDR from the unbounded correction's 0.263 to 0.246 and 0.229, respectively, while the DFF box yields the highest NDR of 0.278. The relatively narrow NDR range of 0.229–0.278 is consistent with Figure 3(c): approximately 71%–73% of lower-loss comparisons for DFF and the two GG-DFF settings leave regret unchanged, so many surrogate-loss reductions do not translate into decision gains. DFF also exhibits a 2.63% loss–regret mismatch rate among its lower-loss comparisons, while neither GG-DFF setting exhibits a mismatch in that analysis. This difference is consistent with DFF's higher NDR in the ablation and connects the choice of correc-



Figure 3: Post-hoc analysis on the SP test set. (a,b) Loss and regret for one observed mismatch. (c) Regret outcomes among comparisons with lower loss than the pretrained reference.

tion region to decision deterioration accompanying loss reduction. The MSE ordering differs from the NDR ordering: GG-DFF-U has the lowest MSE, whereas GG-DFF-D has the lowest NDR. Appendix D.3 reports the KP and Energy-1 ablations, and Appendix D.4 compares slack allocation across benchmarks.

Table 2: SP boundary ablation, averaged over eight independent random seeds. Lower NDR is better. Best and second-best NDR values are shown in **bold** and underlined, respectively.

Correction region	NDR ↓ ($\times 10^{-2}$)	MSE ↓ ($\times 10^{-3}$)
Unbounded SPO+ correction	0.263	0.867
DFF box	0.278	0.830
GG-DFF-U	<u>0.246</u>	0.821
GG-DFF-D	**0.229**	0.854

7 Conclusion

We presented GG-DFF, which uses feasible-solution comparisons to guide prediction corrections around a frozen pretrained predictor. Solution differences and comparison slacks define a convex correction region, and an almost-everywhere differentiable parameterization incorporates this geometry into correction-network training with SPO+. Our analysis relates correction boundaries to normal-cone reachability and establishes conditional performance bounds relative to the pretrained decision. Across four benchmarks, both GG-DFF settings achieve lower mean NDR than DFF. The SP post-hoc analysis shows that lower surrogate loss often leaves regret unchanged and can accompany worse decisions, while neither GG-DFF setting exhibits the latter behavior in this evaluation. Boundary ablations further show that the choice of correction region affects attained decision quality.

AI USE STATEMENT

In this work, we used generative AI tools to assist with manuscript organization, language translation and polishing, checking the consistency and presentation of the text and mathematical statements, reference verification and formatting, and LaTeX compilation. We have reviewed all AI-assisted work. The authors take responsibility for the final content of this work, including all text, claims, and materials prepared with the assistance of generative AI tools.

REFERENCES

Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable convex optimization layers. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Brandon Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 136–145. PMLR, 2017.

Gah-Yi Ban and Cynthia Rudin. The big data newsvendor: Practical insights from machine learning. *Operations Research*, 67(1):90–108, 2019.

Dishank Bansal, Ricky T. Q. Chen, Mustafa Mukadam, and Brandon Amos. TaskMet: Task-driven metric learning for model learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46505–46519. Curran Associates, Inc., 2023.

Maximilian Beichter et al. Decision-focused retraining of forecast models for optimization problems in smart energy systems. In *Proceedings of the 15th ACM International Conference on Future and Sustainable Energy Systems*, 2024.

Senne Berden, Ali Mahmutoğulları, Dimos Tsouros, and Tias Guns. Solver-free decision-focused learning for linear optimization problems. In *Advances in Neural Information Processing Systems*, volume 38, pp. 158127–158145. Curran Associates, Inc., 2025.

Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, and Francis Bach. Learning with differentiable perturbed optimizers. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9508–9519. Curran Associates, Inc., 2020.

Dimitris Bertsimas and Nathan Kallus. From predictive to prescriptive analytics. *Management Science*, 66(3):1025–1044, 2020.

Mathieu Blondel, Quentin Berthet, Marco Cuturi, Roy Frostig, Stephan Hoyer, Felipe Llinares-López, Fabian Pedregosa, and Jean-Philippe Vert. Efficient and modular implicit differentiation. In *Advances in Neural Information Processing Systems*, volume 35, pp. 5230–5242. Curran Associates, Inc., 2022.

Andrew Butler and Roy H. Kwon. Gradient boosting for convex cone predict and optimize problems. *Operations Research Letters*, 51(1):79–83, 2023.

Guillaume Dalle, Léo Baty, Louis Bouvier, and Axel Parmentier. Learning with combinatorial optimization layers: a probabilistic approach. *arXiv preprint arXiv:2207.13513*, 2022.

Hal Daumé, III, Samir Khuller, Manish Purohit, and Gregory Sanders. On correcting inputs: Inverse optimization for online structured prediction. In *35th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2015)*, volume 45 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 38–51. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2015.

Priya Donti, Brandon Amos, and J. Zico Kolter. Task-based end-to-end model learning in stochastic optimization. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". *Management Science*, 68(1): 9–26, 2022.

Adam N. Elmachtoub, Henry Lam, Haofeng Zhang, and Yunfan Zhao. Estimate-then-optimize versus integrated-estimation-optimization versus sample average approximation: A stochastic dominance perspective. *arXiv preprint arXiv:2304.06833*, 2025.

Alexander S. Estes and Jean-Philippe P. Richard. Smart predict-then-optimize for two-stage linear programs with side information. *INFORMS Journal on Optimization*, 5(3):295–320, 2023.

Gabriele Farina. Lecture 2: First-order optimality conditions. Lecture notes for MIT 6.7220/15.084, Nonlinear Optimization, Massachusetts Institute of Technology, 2025.

Negin Golrezaei, Patrick Jaillet, and Zijie Zhou. Online resource allocation with convex-set machine-learned advice. *Operations Research*, 2026. Articles in Advance.

Xinyi Hu, Jasper Lee, and Jimmy Lee. Two-stage predict+optimize for MILPs with unknown parameters in constraints. In *Advances in Neural Information Processing Systems*, volume 36, pp. 14247–14272. Curran Associates, Inc., 2023.

Georgiana Ifrim, Barry O'Sullivan, and Helmut Simonis. Properties of energy-price forecasts for scheduling. In *Principles and Practice of Constraint Programming*, volume 7514 of *Lecture Notes in Computer Science*, pp. 957–972. Springer Berlin Heidelberg, 2012.

Thorsten Joachims, Thomas Finley, and Chun-Nam John Yu. Cutting-plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Junhyeong Lee, Sangjin Jin, and Yongjae Lee. Decision-focused learning via tangent-space projection of prediction error. *arXiv preprint arXiv:2605.01361*, 2026.

Jayanta Mandi and Tias Guns. Interior point solving for LP-based prediction+optimisation. In *Advances in Neural Information Processing Systems*, volume 33, pp. 7272–7282. Curran Associates, Inc., 2020.

Jayanta Mandi, Emir Demirović, Peter J. Stuckey, and Tias Guns. Smart predict-and-optimize for hard combinatorial optimization problems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(2):1603–1610, 2020.

Jayanta Mandi, Víctor Bucarey, Maxime Mulamba Ke Tchomba, and Tias Guns. Decision-focused learning: Through the lens of learning to rank. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 14935–14947. PMLR, 2022.

Maxime Mulamba, Jayanta Mandi, Michelangelo Diligenti, Michele Lombardi, Victor Bucarey, and Tias Guns. Contrastive losses and solution caching for predict-and-optimize. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 2833–2840, 2021.

Yves Rychener, Daniel Kuhn, and Tobias Sutter. End-to-end learning for stochastic optimization: A Bayesian perspective. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 29455–29472. PMLR, 2023.

Utsav Sadana, Abhilash Chenreddy, Erick Delage, Alexandre Forel, Emma Frejinger, and Thibaut Vidal. A survey of contextual optimization methods for decision-making under uncertainty. *European Journal of Operational Research*, 320(2):271–289, 2025.

Sanket Shah, Kai Wang, Bryan Wilder, Andrew Perrault, and Milind Tambe. Decision-focused learning without decision-making: Learning locally optimized decision losses. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1320–1332. Curran Associates, Inc., 2022.

Helmut Simonis, Barry O'Sullivan, Deepak Mehta, Barry Hurley, and Milan De Cauwer. CSPLib problem 059: Energy-cost aware scheduling, n.d. URL `https://www.csplib.org/Problems/prob059/`.

Chunlin Sun, Shang Liu, and Xiaocheng Li. Maximum optimality margin: A unified approach for contextual linear programming and inverse linear programming. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32886–32912. PMLR, 2023.

Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(50):1453–1484, 2005.

Marin Vlastelica Pogančić, Anselm Paulus, Vít Musil, Georg Martius, and Michal Rolínek. Differentiation of blackbox combinatorial solvers. In *International Conference on Learning Representations*, 2020.

Jiaqi Yang, Enming Liang, Zicheng Su, Zhichao Zou, Peng Zhen, Jiecheng Guo, Wanjing Ma, and Kun An. DFF: Decision-focused fine-tuning for smarter predict-then-optimize with limited data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(25):26868–26876, 2025.

Appendices A and B provide the theoretical analysis and implementation details. Appendix C specifies the benchmarks and training protocols, and Appendix D contains supplementary experiments.

A Theoretical Analysis

This section analyzes both slack settings of CCR through a shared positive-slack formulation. We relate the construction to normal-fan geometry, give the common parameterization and conditional performance guarantee, and compare the resulting regions with DFF. We also analyze SPO+ level sets (Appendix A.9), KKT differentiation (Appendix A.10), controlled reachability thresholds (Appendix A.11), and auxiliary-gradient propagation (Appendix A.12).

A.1 Setup and comparison geometry

Fix an input and its pretrained prediction $c_0 \in \mathbb{R}^d$. Let the effective feasible decision set $\mathcal{S}$ be nonempty and compact, and assume that $\mathcal{P} = \mathrm{conv}(\mathcal{S})$ is a polytope with finite vertex set $\mathcal{V}$. We use the minimization convention and write

$$w_0 \in \arg\min_{w \in \mathcal{S}} c_0^\top w, \qquad w_z \in \arg\min_{w \in \mathcal{S}} z^\top w. \tag{17}$$

For $v \in \mathcal{V}$, define the minimization normal cone

$$\mathcal{K}(v) = \{z : z^\top (w - v) \geq 0,\ \forall w \in \mathcal{P}\}. \tag{18}$$

Effective decisions have the same dimension as the predicted parameters. In energy scheduling, they are the induced load profiles in the objective coordinates.

Proposition A.1 (Local decision invariance)**.** *Inside the interior of a vertex normal cone, the optimal decision is unique and constant, and true decision regret is locally constant in the prediction. A pair of feasible decisions defines a comparison hyperplane with normal equal to their difference. The selected SPO+ subgradient is constant whenever its reflected query $2z - c$ selects the same solution, and is nonzero when the true-cost and reflected-query solutions differ.*

Proof. For $z \in \mathrm{int}\,\mathcal{K}(v)$ and $\mathcal{V} \neq \{v\}$, finiteness of the vertex set gives

$$\gamma = \min_{u \in \mathcal{V} \setminus \{v\}} z^\top (u - v) > 0, \qquad D_v = \max_{u \in \mathcal{V} \setminus \{v\}} \|u - v\|_2.$$

For $\|\delta\|_2 < \gamma / D_v$, every comparison remains strictly positive under $z + \delta$, so v remains uniquely optimal. The regret $c^\top v - \min_w c^\top w$ is therefore constant on this neighborhood. The singleton case is immediate. Cost equality between u and v is $z^\top (u - v) = 0$. Finally, the SPO+ subgradient is $2(w^*(c) - w^*(2z - c))$, which is constant for a fixed reflected-query solution and nonzero when the two solutions differ. □

Each solution difference defines an objective-equality hyperplane. Comparisons between adjacent optimal vertices describe shared cone boundaries; retained solution pairs provide a finite representation of available optimality comparisons.

A.2 A correction region from directional error support

Let $\mathcal{E} \subset \mathbb{R}^d$ be a nonempty compact convex error set containing zero, and let $\mathcal{H} \subseteq \mathcal{S}$ be a nonempty finite comparison bank. For each $v \in \mathcal{H}$, set

$$d_v = v - w_0, \qquad e_v = h_{\mathcal{E}}(d_v) = \max_{e \in \mathcal{E}} e^\top d_v. \tag{19}$$

The support value is the largest error contribution to the cost difference between v and the anchor decision. For arbitrary positive slacks $s = (s_v)_{v \in \mathcal{H}}$, define

$$\mathcal{C}_s^{\mathcal{H}} = \bigcap_{v \in \mathcal{H}} \{z : (z - c_0)^\top d_v \geq -s_v\}, \qquad \tau_s = \max_{v \in \mathcal{H}} (e_v + s_v). \tag{20}$$

Proposition A.2 (Common performance certificate for positive slacks)**.** *The region $\mathcal{C}_s^{\mathcal{H}}$ is closed and convex and contains c_0 in its interior. Suppose $z \in \mathcal{C}_s^{\mathcal{H}}$, $w_z \in \mathcal{H}$, and $(c - c_0)^\top (w_z - w_0) \le e_{w_z}$. Then*

$$c^\top (w_z - w_0) \le \tau_s. \tag{21}$$

If $\mathcal{H}$ contains all vertices of $\mathcal{P}$, the bound holds for every $c \in c_0 + \mathcal{E}$ and every optimal decision at z. A retained vertex cone is excluded whenever $c_0^\top d_v > s_v$.

Proof. The region is a finite intersection of closed halfspaces, all strict at the anchor. For $d = w_z - w_0$, optimality gives $z^\top d \le 0$, while the constraint gives $(c_0 - z)^\top d \le s_{w_z}$. Hence

$$c^\top d \le z^\top d + s_{w_z} + e_{w_z} \le \tau_s.$$

With all vertices retained, this argument applies to every optimal vertex for $c \in c_0 + \mathcal{E}$. Linearity extends the same bound to any convex combination in the optimal face. Finally, the constraint for v implies $z^\top d_v \ge c_0^\top d_v - s_v > 0$ under the exclusion condition, which is incompatible with $z \in \mathcal{K}(v)$. □

Writing $E_{\max} = \max_{v \in \mathcal{H}} e_v$, for CCR-U we have $s_v = b$ and $\tau_s = E_{\max} + b$. For CCR-D, $s_v = b + E_{\max} - e_v$ and the same identity holds. The error set is needed to evaluate the conditional certificate for either setting; only CCR-D needs its support values to construct the predictor's boundary. We next develop the performance-budget form used by CCR-D, which also contains CCR-U at the same b.

For a budget τ, define

$$\mathcal{C}_\tau^{\mathcal{H}} = \bigcap_{v \in \mathcal{H}} \{z : (z - c_0)^\top d_v \ge e_v - \tau\}. \tag{22}$$

This construction uses the set-conditional performance perspective of convex-set machine-learned advice (Golrezaei et al., 2026): the error set specifies the advice, and its directional worst cases determine the correction constraints. The error set $\mathcal{E}$ and the predictor's output set $\mathcal{C}_\tau^{\mathcal{H}}$ have different roles.

Proposition A.3 (Convexity, anchor inclusion, and boundary location)**.** *The region $\mathcal{C}_\tau^{\mathcal{H}}$ is closed and convex, and increases by set inclusion as τ increases. If $\tau > \max_{v \in \mathcal{H}} e_v$, then c_0 is an interior point. For every nonzero comparison d_v, the distance from c_0 to its defining hyperplane is*

$$\frac{|\tau - e_v|}{\|d_v\|_2}. \tag{23}$$

Proof. Each constraint is a closed halfspace. Increasing τ relaxes its right-hand side. At $z = c_0$, the left-hand side is zero, so all finitely many constraints are strict under the stated condition. The distance to the hyperplane $(z - c_0)^\top d_v = e_v - \tau$ is the absolute offset divided by $\|d_v\|_2$. Under $\tau > \max_v e_v$, it equals $(\tau - e_v)/\|d_v\|_2$. □

Thus the comparison supplies the boundary orientation, while e_v and τ determine its offset. Directions permitted by all retained comparisons can remain unbounded. Convexification of an empirical error set is exact for these linear comparisons: for any nonempty compact set A and direction d, $h_{\mathrm{conv}(A)}(d) = h_A(d)$. Indeed, a convex combination cannot exceed the largest constituent inner product, and $A \subseteq \mathrm{conv}(A)$ gives the reverse inequality.

A training-only construction consistent with the current parameterization is

$$\mathcal{E} = \mathrm{conv}(\{0\} \cup \{c_i - c_0(x_i) : i \in I_{\mathrm{tr}}\}), \tag{24}$$

$$L = \left(\frac{1}{|I_{\mathrm{tr}}|} \sum_{i \in I_{\mathrm{tr}}} [c_0(x_i)^\top w_{0i}]^2 \right)^{1/2}, \qquad \tau_i = \max_{v \in \mathcal{H}} e_{iv} + \rho L, \quad \rho > 0,\ L > 0. \tag{25}$$

For an empirical residual hull, the certificate applies to directional errors satisfying the coverage inequality in Theorem A.5. This is a deterministic condition evaluated for the decision and input under consideration.

A.3 Uniform and direction-dependent slacks

This subsection relates CCR-U and CCR-D within the common method. Fix a finite bank, write $E_{\max} = \max_v e_v$, and let $b = \rho L > 0$ and $\tau = E_{\max} + b$. Their regions are

$$\mathcal{U}_b = \{z : (z - c_0)^\top d_v \geq -b,\ \forall v \in \mathcal{H}\}, \tag{26}$$

$$\mathcal{C}_\tau^{\mathcal{H}} = \{z : (z - c_0)^\top d_v \geq -[b + E_{\max} - e_v],\ \forall v \in \mathcal{H}\}. \tag{27}$$

Proposition A.4 (Directional allocation under a common certificate)**.** *At the same b, $\mathcal{U}_b \subseteq \mathcal{C}_\tau^{\mathcal{H}}$. Both regions satisfy*

$$\Psi(z) := \max_{v \in \mathcal{H}} \{e_v - (z - c_0)^\top d_v\} \leq \tau. \tag{28}$$

Moreover, $\mathcal{C}_\tau^{\mathcal{H}}$ is exactly the sublevel set $\{z : \Psi(z) \leq \tau\}$. Among choices of a common slack s satisfying the separate sufficient conditions $e_v + s \leq \tau$ for every retained comparison, the largest is $s = b$. CCR-D uses each comparison's largest individual slack $s_v = \tau - e_v$ under these same conditions.

Proof. Since $e_v \leq E_{\max}$, $b + E_{\max} - e_v \geq b$, giving inclusion. A uniform constraint implies $e_v - (z - c_0)^\top d_v \leq e_v + b \leq \tau$. In CCR-D this inequality is its defining constraint, so the sublevel representation is exact. The common-slack conditions give $s \leq \min_v(\tau - e_v) = b$; the individual conditions give $s_v \leq \tau - e_v$. □

For a covered final decision and directional error, (28) implies the performance bound in Theorem A.5. Thus directional allocation can admit additional prediction positions while preserving this common certificate. Its maximality is with respect to the stated comparison certificate. The exact worst-case value further depends on which decisions are reachable. At a common b, the smaller uniform region can have a stronger *exact* worst-case bound by Proposition A.11. The setting comparison tests this difference in slack allocation. The inclusion statement uses the same ρ for both settings. Separate validation choices determine the two regions independently.

A.4 Anchor-relative performance and exclusion of decision cones

Theorem A.5 (Conditional performance bound)**.** *Let $z \in \mathcal{C}_\tau^{\mathcal{H}}$ and $d = w_z - w_0$. Suppose its final comparison satisfies*

$$(z - c_0)^\top d \geq h_{\mathcal{E}}(d) - \tau, \qquad (c - c_0)^\top d \leq h_{\mathcal{E}}(d). \tag{29}$$

Then

$$c^\top (w_z - w_0) \leq \tau, \qquad \text{Regret}(z, c) \leq \text{Regret}(c_0, c) + \tau. \tag{30}$$

If $\mathcal{H}$ contains every vertex of $\mathcal{P}$, this bound holds for every $c \in c_0 + \mathcal{E}$, every $z \in \mathcal{C}_\tau^{\mathcal{H}}$, and every optimal decision at z.

Proof. Using the two inequalities in (29) and optimality of w_z gives

$$c^\top d \leq c_0^\top d + h_{\mathcal{E}}(d) \leq z^\top d + \tau \leq \tau.$$

Subtracting the same true optimal objective on both sides gives the regret inequality. With all vertices represented, the argument applies to every optimal vertex. Any nonvertex optimal decision is a convex combination of vertices in the optimal face, so the same bound follows by linearity. Membership of $c - c_0$ in $\mathcal{E}$ implies the directional error inequality for each vertex. □

Corollary A.6 (Exclusion of cones exceeding the budget)**.** *For a retained vertex $v \in \mathcal{H} \cap \mathcal{V}$, define*

$$\beta_v = c_0^\top d_v + h_{\mathcal{E}}(d_v). \tag{31}$$

If $\beta_v > \tau$, then

$$\mathcal{C}_\tau^{\mathcal{H}} \cap \mathcal{K}(v) = \varnothing. \tag{32}$$

Proof. The CCR constraint requires $z^\top d_v \geq \beta_v - \tau > 0$. Membership in $\mathcal{K}(v)$ instead requires $z^\top (v - w_0) \leq 0$, since w_0 is feasible. These inequalities are incompatible. □

The excluded cone corresponds to a decision whose worst cost increase relative to the anchor, over $c_0+\mathcal{E}$, exceeds the budget. For a nonzero such comparison, the CCR enforces the positive normalized separation $(\beta_v - \tau)/\|d_v\|_2$ from its cost-equality hyperplane. This is the direct connection between the correction boundary, a comparison margin, and normal-fan geometry.

The benchmark constructions supply two levels of comparison coverage. SP enumerates all 20 paths, establishing full comparison coverage. KP and energy scheduling retain finite training solution banks; their pointwise certificates apply to final decisions satisfying the comparison and directional-error inequalities in (29). Retaining the final decision in the bank guarantees its comparison inequality, while the directional-error inequality is a separate requirement. Whole-region guarantees use full vertex coverage together with $c - c_0 \in \mathcal{E}$.

A.5 Direct parameterization and backpropagation

Assume $s_v > 0$ for every retained comparison, as in CCR-U or CCR-D. Define

$$p(u) = \max\left(0, \max_{v\in\mathcal{H}} \frac{-u^\top d_v}{s_v}\right), \qquad F(u) = \frac{u}{1+p(u)}. \tag{33}$$

For a correction network f_θ and a fixed coordinate scale $a > 0$, the predictor is

$$u_\theta(x) = a\,\mathrm{diag}(c_0(x))f_\theta(x), \qquad z_\theta(x) = c_0(x) + F(u_\theta(x)). \tag{34}$$

The final network layer is initialized at zero. The comparison bank, error support, and budget are held fixed during this fine-tuning parameterization.

Theorem A.7 (Interior coverage and optimizer-independent feasibility)**.** *Let $D = \{\delta : p(\delta) \le 1\}$, so that $\mathcal{C}_s^{\mathcal{H}} = c_0 + D$. The map F is a bijection from $\mathbb{R}^d$ onto* int *D, with inverse*

$$F^{-1}(\delta) = \frac{\delta}{1-p(\delta)}. \tag{35}$$

The map is locally Lipschitz and piecewise smooth, hence differentiable almost everywhere. It is differentiable at zero with $DF(0) = I$. Every finite network output in (34) *strictly satisfies the CCR constraints, independently of the optimizer used to update θ.*

Proof. The function p is nonnegative, continuous, and positively homogeneous. The finite halfspace description and strict inclusion of zero give int $D = \{\delta : p(\delta) < 1\}$. Homogeneity yields

$$p(F(u)) = \frac{p(u)}{1+p(u)} < 1.$$

Conversely, substituting (35) for any $p(\delta) < 1$ recovers $F(u) = \delta$; substituting $F(u)$ in the inverse recovers u, proving bijectivity. A finite maximum of linear functions is Lipschitz and piecewise linear. Since $1 + p(u) \ge 1$, F is locally Lipschitz and smooth on each open branch. Because $p(u) = O(\|u\|_2)$ near zero, $F(u) = u + O(\|u\|_2^2)$, so $DF(0) = I$. Feasibility is a property of the forward map for all finite u, and hence persists under any finite parameter update. □

On a branch with a unique active positive ratio indexed by v, the Jacobian is

$$DF(u) = \frac{I}{1+p(u)} + \frac{ud_v^\top}{s_v(1+p(u))^2}. \tag{36}$$

SPO+ propagates through this Jacobian and the correction network by the ordinary chain rule. At nondifferentiable branch ties, backpropagation uses a combination of active-branch derivatives supplied by automatic differentiation. The composite map remains differentiable at zero with derivative I. Comparison geometry thus determines both the forward prediction and its branchwise derivative.

The full logits-to-output interior coverage additionally requires nonzero coordinates of c_0. A zero anchor coordinate is fixed by (34), and the joint outputs across inputs are determined by the shared network architecture. The fixed scale a controls the coordinate parameterization, whereas ρ controls the output set. These are distinct mathematical roles.

Directions left unconstrained. If $r^\top d_v = 0$ for every retained comparison, then $p(r) = 0$ and $F(r) = r$. More generally, $p(u + r) = p(u)$ and $F(u + r) = F(u) + r/(1 + p(u))$. Thus the construction acts on comparison violations and preserves unconstrained nullspace directions. For any ray direction r with $p(r) > 0$, the closed region allows $c_0 + tr$ exactly for $0 \le t \le 1/p(r)$; if $p(r) = 0$, the ray is unrestricted. This follows directly from positive homogeneity.

A.6 Comparing the performance lower bound with DFF

In this subsection the anchor, feasible set, and error set are identical for the two regions, and $\mathcal{H}$ contains all vertices. Write $\mathcal{C}_\tau = \mathcal{C}_\tau^{\mathcal{H}}$ and

$$\mathcal{T}_\epsilon = \{z : |z_j - c_{0j}| \le \epsilon |c_{0j}|\}, \qquad A_v = \sum_j |c_{0j}|\,|d_{vj}|. \tag{37}$$

Proposition A.8 (A support-function certificate for the DFF box)**.** *Let $B_\epsilon = \max_{v\in\mathcal{H}}(e_v + \epsilon A_v)$. Then $\mathcal{T}_\epsilon \subseteq \mathcal{C}_{B_\epsilon}$. For $c \in c_0 + \mathcal{E}$, every box-constrained prediction satisfies $c^\top(w_z - w_0) \le B_\epsilon$.*

Proof. For every box point, $(z - c_0)^\top d_v \ge -\epsilon A_v \ge e_v - B_\epsilon$. Thus every CCR inequality at budget B_ϵ holds, and Theorem A.5 gives the bound. □

Writing decision quality as $Q(c, w) = -c^\top w$, the two certificates are $Q(c, w_z) - Q(c, w_0) \ge -\tau$ for CCR and at least $-B_\epsilon$ for DFF. Choosing $\tau < B_\epsilon$ improves this certificate. For a nonempty prediction set A, define the exact worst-case quantities

$$W(A; \mathcal{E}) = \sup_{z\in A} \sup_{e\in\mathcal{E}} \sup_{w\in \arg\min_{u\in\mathcal{P}} z^\top u} (c_0 + e)^\top (w - w_0), \qquad \mathrm{LB}(A; \mathcal{E}) = -W(A; \mathcal{E}). \tag{38}$$

This definition includes worst-case selection among tied optimal decisions.

Theorem A.9 (Exact normal-cone characterization and strict improvement)**.** *For any nonempty prediction set A,*

$$W(A; \mathcal{E}) = \max_{v\in\mathcal{V}:\, A\cap\mathcal{K}(v)\neq\varnothing} \beta_v. \tag{39}$$

In particular, $W(\mathcal{C}_\tau; \mathcal{E}) \le \tau$. If $\mathcal{C}_\tau$ is nonempty and

$$\tau < W(\mathcal{T}_\epsilon; \mathcal{E}), \tag{40}$$

then the CCR has a strictly stronger worst-case anchor-relative performance lower bound:

$$\mathrm{LB}(\mathcal{C}_\tau; \mathcal{E}) \ge -\tau > -W(\mathcal{T}_\epsilon; \mathcal{E}) = \mathrm{LB}(\mathcal{T}_\epsilon; \mathcal{E}). \tag{41}$$

Proof. At a fixed z, a linear functional of the decision attains its maximum over the optimal face at an optimal vertex. A vertex v can be optimal at some $z \in A$ if and only if $A \cap \mathcal{K}(v)$ is nonempty. For each such vertex, maximizing over $e \in \mathcal{E}$ gives $c_0^\top d_v + h_{\mathcal{E}}(d_v) = \beta_v$. There are finitely many vertices, so taking their maximum gives (39). The CCR bound follows from Theorem A.5. Under (40), negating the worst-cost-increase bound gives (41). □

Corollary A.10 (A sufficient condition for the independent budget)**.** *Let $E_{\max} = \max_{v\in\mathcal{H}} e_v$ and $\tau = E_{\max} + \rho L$, with $L > 0$. A sufficient condition for* (41) *is*

$$W(\mathcal{T}_\epsilon; \mathcal{E}) > E_{\max}, \qquad 0 < \rho < \frac{W(\mathcal{T}_\epsilon; \mathcal{E}) - E_{\max}}{L}. \tag{42}$$

For $\rho_1 \le \rho_2$, the worst cost increase satisfies $W(\mathcal{C}_{\rho_1}; \mathcal{E}) \le W(\mathcal{C}_{\rho_2}; \mathcal{E})$.

Proof. The first claim substitutes the independent budget into (40); strict anchor inclusion follows from $\rho L > 0$. The second follows from region inclusion in Proposition A.3 and monotonicity of a supremum under enlargement of its domain. □

The same strict-improvement condition applies to CCR-U: at the common $b = \rho L$, its region is contained in the CCR-D region and Proposition A.2 gives $W(\mathcal{C}_s^{\mathcal{H}}; \mathcal{E}) \le E_{\max} + \rho L$ under full comparison coverage. More generally, either positive-slack region has a strictly higher lower bound than DFF whenever $\tau_s < W(\mathcal{T}_\epsilon; \mathcal{E})$.

The characterization in (39) also applies to the open interior reached by finite logits. A closed-region audit includes boundary ties and gives a conservative guarantee for these outputs. A fixed tie-breaking implementation can have a better lower bound than the worst-tie definition. The intersection of a polyhedral region with a vertex cone can be checked by a linear feasibility problem; these checks characterize the region geometrically. A validation-selected ρ yields the strict lower-bound comparison when it also satisfies (42).

A strict-improvement example. Let $\mathcal{S} = \{(1,0),(0,1)\}$, $c_0 = (1, 1.2)$, and $\mathcal{E} = \text{conv}\{0,(0.1,0.1)\}$. The anchor is $w_0 = (1,0)$, and the error support of either decision difference is zero. With $\epsilon = 0.5$, the DFF box admits $(0,1)$ as a strictly optimal decision, whose cost increase is always 0.2; hence $W(\mathcal{T}_\epsilon; \mathcal{E}) = 0.2$. Taking $L = 1$ and $\rho = 0.1$ gives $\tau = 0.1$. CCR requires $z_2 - z_1 \ge 0.1$, so it always selects the anchor and $W(\mathcal{C}_\tau; \mathcal{E}) = 0$. The exact lower bound therefore improves from -0.2 to 0. Here the error set contains a nonzero residual with zero support along the decision-difference directions.

The budget condition is substantive. In the same example, $\epsilon = 0.05$ gives $z_2 - z_1 \ge 0.09$ throughout the DFF box and hence $W(\mathcal{T}_\epsilon; \mathcal{E}) = 0$. A CCR budget $\tau = 1$ allows $z = (1.5, 1)$ and the other decision, giving $W(\mathcal{C}_\tau; \mathcal{E}) = 0.2$. These examples identify the budget condition that gives a stronger exact lower bound.

A.7 Boundary restriction and attainable regret

For any nonempty prediction region A, define its reachable vertices by

$$\mathcal{R}(A) = \{v \in \mathcal{V} : A \cap \mathcal{K}(v) \neq \varnothing\}. \tag{43}$$

This set distinguishes restricting predictions from restricting decisions. The following result applies to convex regions, boxes, and the unrestricted parameter space.

Proposition A.11 (Boundary restriction and worst-case decision quality)**.** *Let $\varnothing \neq A \subseteq B$. Under the setup of Theorem A.9,*

$$\text{LB}(A; \mathcal{E}) \ge \text{LB}(B; \mathcal{E}). \tag{44}$$

The inequality is strict exactly when $\mathcal{R}(A)$ contains none of the vertices attaining $\max_{v \in \mathcal{R}(B)} \beta_v$. In particular, if $\mathcal{R}(A) = \mathcal{R}(B)$, the two bounds are equal even if A is much smaller in parameter space.

Proof. Region inclusion implies $\mathcal{R}(A) \subseteq \mathcal{R}(B)$. Apply (39) and negate. Because the vertex sets are finite and nonempty, their maxima differ strictly exactly when all maximizers in the larger set have been removed. □

Taking $B = \mathbb{R}^d$ compares an output boundary with unrestricted correction coordinates: every vertex cone is reachable in B. More generally, B should be the actual accessible coordinate space if some anchor coordinates are fixed. The proposition concerns worst-case decision quality over the region, independently of the training optimizer. Convexity makes the region compatible with a tractable halfspace description and the gauge parameterization; set inclusion is the property responsible for (44).

Corollary A.12 (Improved worst case with a preserved optimal decision)**.** *Fix a true cost c. Suppose $A \subseteq B$ intersects the normal cone of at least one true-cost optimal vertex. Then the smallest attainable true regret over A equals zero. If A also excludes every vertex attaining the largest true regret reachable in B, its largest attainable true regret is strictly smaller than that over B.*

Proof. Intersection with an optimal vertex cone makes a zero-regret decision available. The maximum attained regret over a region is the maximum of $c^\top v - \min_w c^\top w$ over its reachable vertices. Removing all maximizers strictly decreases this finite maximum. These attainability statements allow selection among tied optimal decisions, consistently with (38). □

The corollary identifies geometric conditions under which a boundary excludes worst-regret decisions while preserving a true optimum. Training and evaluation then determine which of the reachable decisions the shared neural predictor attains.

The anchor-relative control supplied by a DFF box. For any $z \in \mathcal{T}_\epsilon$, let $d = w_z - w_0$. Optimality at c_0 and z yields

$$0 \le c_0^\top d \le (c_0 - z)^\top d \le \epsilon \sum_j |c_{0j}|\,|d_j|. \tag{45}$$

Consequently, for any true cost c,

$$c^\top d \le \epsilon \sum_j |c_{0j}|\,|d_j| + \|c - c_0\|_2 \|d\|_2. \tag{46}$$

The first inequality follows from $c_0^\top d \ge 0$ and $z^\top d \le 0$; the last uses the box support function. Adding $(c - c_0)^\top d$ and applying Cauchy–Schwarz proves the second display. These are pointwise output-constraint properties and apply to unseen inputs as well as training inputs. They explain how bounded fine-tuning controls loss relative to its pretrained decision, with the anchor error appearing explicitly.

A.8 Scope of the set-conditional guarantee

The lower bound in (41) is additive and relative to the same pretrained decision, with the same error set and tie convention on both sides. It characterizes the decisions accessible within a prediction region. The experiments measure the mean regret attained by training within that region. For an empirical error set and finite memory, directional error and comparison coverage are the conditions needed in (29).

Objective sense and the SPO+ subgradient. Let $\sigma = +1$ for minimization and $\sigma = -1$ for maximization, with $\mathcal{O}_\sigma(c) \in \arg\min_{w \in \mathcal{S}} \sigma c^\top w$. Set $w_0 = \mathcal{O}_\sigma(c_0)$ and $w_z = \mathcal{O}_\sigma(z)$. In the original objective coordinates, the comparison direction and subgradient are

$$d_v = \sigma(v - w_0), \qquad g_{z,\sigma}(z, c) = 2\sigma(\mathcal{O}_\sigma(c) - \mathcal{O}_\sigma(2z - c)). \tag{47}$$

The SPO+ loss in the original objective coordinates is

$$\ell^\sigma_{\text{SPO+}}(z, c) = \max_{w \in \mathcal{S}} \{-\sigma(2z - c)^\top w\} + \sigma(2z - c)^\top \mathcal{O}_\sigma(c).$$

For maximization, this becomes

$$\ell^{-1}_{\text{SPO+}}(z, c) = \max_{w \in \mathcal{S}} (2z - c)^\top w - 2z^\top \mathcal{O}_{-1}(c) + c^\top \mathcal{O}_{-1}(c).$$

Differentiating an active affine branch gives $g_{z,-1} = 2(\mathcal{O}_{-1}(2z - c) - \mathcal{O}_{-1}(c))$. This is the loss and sign convention used by the KP implementation. With $d_v = w_0 - v$, the support and region inequalities bound $c^\top(w_0 - w_z)$, the profit loss relative to the anchor. Substituting these signed comparisons into the proofs gives the same anchor-relative certificate.

The connection to Golrezaei et al. (2026) is the use of set-valued advice to derive a conditional performance guarantee. Here the support-function halfspaces, direct neural parameterization, and normal-cone characterization apply that perspective to offline prediction correction.

A.9 SPO+ level sets and matched surrogate descent

SPO+ level sets in Figure 1. For fixed c, write $q = 2z - c$ and define

$$\widetilde{\ell}(q, c) = \ell_{\text{SPO+}}((q + c)/2, c) = \max_{v \in \mathcal{V}} q^\top (w^*(c) - v).$$

On the normal cone $\mathcal{K}(v)$ this loss is linear in q, with $g_q = w^*(c) - v$. Thus two points in the same cone have equal loss exactly when their displacement is orthogonal to g_q. For any level λ, the complete level set is

$$\{q : \widetilde{\ell}(q, c) = \lambda\} = \bigcup_{v \in \mathcal{V}} (\mathcal{K}(v) \cap \{q : q^\top (w^*(c) - v) = \lambda\}). \tag{48}$$

The affine pieces agree on shared cone boundaries by equality of the competing objective values.

An explicit crossing follows from positive homogeneity. Suppose $\bar{q}$ lies on a shared facet where exactly two vertices v_A, v_B are optimal, and $\widetilde{\ell}(\bar{q}, c) = \lambda > 0$. Choose r with $r^\top(v_B - v_A) \neq 0$. For sufficiently small $|t|$, define

$$q(t) = \frac{\lambda(\bar{q} + tr)}{\widetilde{\ell}(\bar{q} + tr, c)}. \tag{49}$$

Continuity keeps the denominator positive, and homogeneity gives $\widetilde{\ell}(q(t), c) = \lambda$. The positive rescaling preserves cone membership, so the two signs of t select opposite sides of the shared facet. In two dimensions, the level-set pieces in (48) are straight segments that join at the boundary, as illustrated in Figure 1(b). A common starting point with higher loss gives equal loss reductions to endpoints on this contour. The colored cones in panel (b) determine $w^*(q)$; final decisions are evaluated at $z = (q + c)/2$. In the plotted construction, $w^*(q_D) = w_1$ and $w^*(q_F) = w_2$, whereas $w^*(z_D) = w_3$ and $w^*(z_F) = w_2$, with $c^\top w_3 < c^\top w_2$. Proposition A.13 gives a matched-descent construction in prediction coordinates; Appendix D.1 instantiates it with different final regrets on SP.

Affine surrogate signal. For a fixed true cost c and feasible solution $v_{\text{SPO+}}$, define the SPO+ oracle cell

$$\mathcal{A}_{v_{\text{SPO+}}}(c) = \{z : (2z - c)^\top (v - v_{\text{SPO+}}) \geq 0, \quad \forall v \in \mathcal{V}\}.$$

This convex set contains predictions at which $v_{\text{SPO+}}$ is an optimal SPO+ oracle solution.

Proposition A.13 (Matched surrogate descent)**.** *Let $c_0 \in \mathcal{A}_{v_{\text{SPO+}}}(c)$, $g_z = 2(w^*(c) - v_{\text{SPO+}}) \neq 0$, and $\hat{g}_z = g_z/\|g_z\|_2$. Fix $R > 0$ and $\alpha \in (0, 1)$. For unit vectors e_A, e_B orthogonal to g_z, define*

$$\delta_k = R(-\alpha \hat{g}_z + \sqrt{1 - \alpha^2}\, e_k), \qquad z_k = c_0 + \delta_k, \quad k \in \{A, B\}.$$

If $z_A, z_B \in \mathcal{A}_{v_{\text{SPO+}}}(c)$, both corrections have norm R and reduce SPO+ by exactly $\alpha R\|g_z\|_2$. The vector g_z is a valid subgradient throughout each segment from c_0 to z_k.

Proof. On $\mathcal{A}_{v_{\text{SPO+}}}(c)$ the surrogate is $\ell_{\text{SPO+}}(z, c) = c^\top v_{\text{SPO+}} + 2z^\top(w^*(c) - v_{\text{SPO+}}) - c^\top w^*(c)$, which is affine with gradient g_z whenever the oracle optimum is unique. The same vector is a valid subgradient at ties. Orthogonality gives $\|\delta_k\|_2 = R$ and $g_z^\top \delta_k = -\alpha R\|g_z\|_2$. Convexity of $\mathcal{A}_{v_{\text{SPO+}}}(c)$ keeps both segments within the cell, establishing the claims. □

A.10 KKT differentiation and decision-region geometry

Proposition A.1 establishes local decision invariance inside vertex normal cones. We now examine the relation between a smooth training loss and these decision regions using exact KKT differentiation. Implicit differentiation of optimality conditions and barrier smoothing provide standard constructions of differentiable optimization layers (Amos & Kolter, 2017; Mandi & Guns, 2020).

Exact sensitivity of a smoothed optimization layer. Represent the feasible polytope as $\mathcal{P} = \{w : Aw = b,\ Gw \leq h\}$, where A has full row rank and a point satisfying $Aw = b$, $Gw < h$ exists. For $\mu > 0$, let

$$w_\mu(z) = \underset{Aw=b,\ Gw<h}{\arg\min} \{z^\top w + \mu\phi(w)\}, \qquad L_\mu(z, c) = c^\top w_\mu(z) - \min_{w \in \mathcal{P}} c^\top w, \tag{50}$$

where ϕ is a twice continuously differentiable convex barrier. Assume the minimizer exists in the relative interior and $H_\mu = \mu\nabla^2\phi(w_\mu(z))$ is positive definite. The exact KKT equations are

$$z + \mu\nabla\phi(w_\mu) + A^\top \nu_\mu = 0, \qquad Aw_\mu = b.$$

Differentiating with respect to z gives

$$\begin{bmatrix} H_\mu & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} D_z w_\mu \\ D_z \nu_\mu \end{bmatrix} = -\begin{bmatrix} I \\ 0 \end{bmatrix}, \qquad \nabla_z L_\mu(z, c) = -P_\mu c, \tag{51}$$

where

$$P_\mu = H_\mu^{-1} - H_\mu^{-1}A^\top(AH_\mu^{-1}A^\top)^{-1}AH_\mu^{-1}.$$

The KKT matrix is nonsingular under the stated assumptions, so the implicit function theorem gives a continuously differentiable solution map. Block elimination yields $D_z w_\mu = -P_\mu$, and the chain rule gives the loss gradient. In contrast, evaluation with the original linear objective selects $w^*(z)$ according to the normal cone containing z. The following result characterizes when a smooth loss level set crosses such a decision boundary.

Proposition A.14 (Equal smooth loss across a decision boundary)**.** *Let v_A, v_B be distinct vertices of $\mathcal{P}$ and suppose*

$$\bar{z}^\top v_A = \bar{z}^\top v_B < \bar{z}^\top v \quad \textit{for every } v \in \mathcal{V} \setminus \{v_A, v_B\}.$$

Write $n = v_B - v_A$. Let L be continuously differentiable near $\bar{z}$, with $g = \nabla L(\bar{z}) \neq 0$ and $g \notin \operatorname{span}\{n\}$. There exist points z_A, z_B arbitrarily close to $\bar{z}$ such that

$$L(z_A) = L(z_B) = L(\bar{z}), \qquad w^*(z_A) = v_A, \quad w^*(z_B) = v_B, \tag{52}$$

and both decisions are unique. Their true regrets differ by $c^\top(v_A - v_B)$, which is nonzero whenever the two vertices have different true objective values. A common starting point with loss greater than $L(\bar{z})$ gives equal positive loss reductions to the two endpoints.

Proof. Set $t = n - g(g^\top n)/\|g\|_2^2$. Then $g^\top t = 0$ and $n^\top t = \|t\|_2^2 > 0$. The implicit function theorem gives a continuously differentiable curve γ on the level set of L with $\gamma(0) = \bar{z}$ and $\gamma'(0) = t$. Consequently, $n^\top \gamma(s) = s\|t\|_2^2 + o(|s|)$. For sufficiently small positive s, $\gamma(s)$ strictly prefers v_A to v_B, while $\gamma(-s)$ strictly prefers v_B to v_A. Finiteness of the vertex set preserves the strict comparisons against every other vertex in a neighborhood of $\bar{z}$. Taking $z_A = \gamma(s)$ and $z_B = \gamma(-s)$ proves uniqueness and equal loss. Subtracting the same true optimal objective gives the regret difference. Finally, $L(\bar{z} + \varepsilon g) = L(\bar{z}) + \varepsilon\|g\|_2^2 + o(\varepsilon)$, so a common starting point with higher loss exists for sufficiently small positive ε. □

The condition compares two geometric objects: the tangent space of the training-loss level set and the normal of the decision boundary. When the loss gradient is not parallel to that normal, a direction tangent to the level set can cross the boundary. Proposition A.14 applies in particular to L_μ in (50), whose gradient is obtained from the exact KKT system.

An exact barrier example with opposite regret changes. Consider $\mathcal{P} = \{w \in \mathbb{R}^3 : w \geq 0,\ \mathbf{1}^\top w = 1\}$, $c = (0, 1, 4)^\top$, and $\phi(w) = -\sum_{j=1}^3 \log w_j$. For any interior point p and any $\mu > 0$, define $z(p)_j = \mu/p_j$. The barrier stationarity equations hold at $w = p$ with equality multiplier zero. Strict convexity therefore gives $w_\mu(z(p)) = p$ exactly. The original linear problem selects the coordinate vertex corresponding to the largest component of p, because that component gives the smallest coefficient of $z(p)$.

Take

$$p_0 = \tfrac{1}{20}(2, 11, 7)^\top, \qquad p_A = \tfrac{1}{20}(10, 1, 9)^\top, \qquad p_B = \tfrac{1}{20}(7, 5, 8)^\top, \qquad z_k = z(p_k).$$

Since the true optimum is e_1 with objective value zero,

$$\begin{gathered} L_\mu(z_0, c) = \tfrac{39}{20}, \qquad L_\mu(z_A, c) = L_\mu(z_B, c) = \tfrac{37}{20}, \\ (\text{Regret}(z_0, c), \text{Regret}(z_A, c), \text{Regret}(z_B, c)) = (1, 0, 4). \end{gathered} \tag{53}$$

Thus both corrections decrease the smooth training loss by exactly $1/10$, while one improves the final decision and the other worsens it. All three linear optima are unique, and the smoothed solutions satisfy the exact KKT equations for every $\mu > 0$.

This construction establishes the distinction between equal training-loss progress and decision quality for an exactly differentiated barrier model. Together with the SPO+ construction in Appendix A.9, it motivates controlling prediction corrections through the solution comparisons that define decision boundaries. The reachable-decision analysis in Appendix A.7 then connects restrictions on these regions to attainable regret.

A.11 Reachability thresholds in a controlled example

The edge order is $(A0, B0, B1, A1)$, with

$$c_0 = (10, 1, 2, 12), \quad c = (10, 2, 1, 14), \qquad \mathcal{V} = \{(1,1,0,0), (1,0,1,0), (0,1,0,1), (0,0,1,1)\}.$$

The four true path costs are $12, 11, 16, 15$. Regret is the true path cost minus 11. This constructed example fixes both the anchor prediction and the true costs explicitly.

Proposition A.15 (Different opening orders)**.** *For this example, the infimum budgets for strict reachability of the useful, harmful, and joint exchanges are* $(1/3, 1/11, 1/3)$ *under DFF and* $(1, 2, 3)$ *under CCR-U. Consequently, any DFF region allowing a uniquely optimal zero-regret decision also allows a uniquely optimal regret-5 decision. For every CCR-U budget* $1 < b < 2$*, the best and worst attainable regrets are 0 and 1.*

Proof. For DFF, each stage's two edge intervals overlap at the stated relative-width threshold. Unique exchange requires strict overlap. The two stages use disjoint coordinates, so simultaneous exchange requires the larger of the two thresholds.

For CCR-U, if a candidate has anchor gap g, its constraint implies $z^\top(v - w_0) \geq g - b$. Thus unique optimality requires $b > g$. Conversely, for a single exchange, choose its new price difference to be a sufficiently small negative number and leave the other stage unchanged. For the joint exchange, choose both new differences small and negative. The required comparison displacements approach $-1, -2, -3$, respectively, establishing the thresholds. Below $b = 2$, only the anchor and the useful path remain possible. Their true costs are 12 and 11. □

Finite network coordinates. For $b = 1.5$, the point $z = (10, 1.6, 1.4, 12)$ lies strictly inside CCR-U and selects the useful path. With $\delta = z - c_0$, its gauge is $p(\delta) = 0.8$. The finite vector $u = \delta/(1 - p(\delta))$ satisfies $z = c_0 + u/(1 + p(u))$. Since all anchor coordinates are nonzero, finite correction-network outputs realize this vector.

A.12 Propagation and normalization of auxiliary margin gradients

An auxiliary margin influences the model through a sequence of gradient transformations. For a minibatch B, let $J_i = \partial z_\theta(x_i)/\partial\theta$ and denote the prediction-space SPO+ and margin gradients by $g_{z,i}$ and h_i. The parameter gradient is

$$G_t = \frac{1}{|B|} \sum_{i \in B} J_i^\top (g_{z,i} + \lambda h_i). \tag{54}$$

Backpropagation transforms each comparison direction through $J_i^\top$, and aggregation combines these transformed directions across examples. Adaptive diagonal normalization then rescales update coordinates using accumulated gradient statistics. The resulting influence of h_i on the parameter update depends on the Jacobians, the batch, and the optimizer state. The CCR parameterization gives comparison geometry an explicit role in determining admissible predictions throughout these updates.

We examine two stages of this process: normalization within the margin loss and adaptive normalization of the parameter gradient. Adam provides a concrete case for the latter analysis. Fix the selected solutions, active comparisons, and positive regret weights in a forward pass. A normalized pairwise hinge has derivative

$$h_v(z; r) = \left[r - \frac{z^\top(v - w^*)}{\|v - w^*\|_2} \right]_+, \qquad \nabla_z h_v = -\frac{v - w^*}{\|v - w^*\|_2} \quad \text{when } h_v > 0. \tag{55}$$

On a fixed active branch, its derivative is constant as violation depth varies. Normalizing regret weights by their active sum additionally removes a common positive scale factor in those weights. Optimizer normalization acts at a subsequent level, on the gradients after network backpropagation and minibatch aggregation.

Proposition A.16 (Adam invariance under fixed positive gradient scaling)**.** *Let* $g_t \in \mathbb{R}^p$ *be the sequence entering Adam's moment estimates, and suppose* $\widetilde{g}_t = D g_t$ *for a fixed diagonal matrix*

$D = \mathrm{diag}(a_1, \ldots, a_p)$ *with* $a_j > 0$*. With identical hyperparameters and zero initial moments, the bias-corrected moments satisfy*

$$\widehat{\widehat{m}}_t = D\widehat{m}_t, \qquad \widehat{\widehat{v}}_t = D^2\widehat{v}_t. \tag{56}$$

For numerical stabilization $\eta_A > 0$*, the scaled coordinate update is*

$$-\alpha \frac{\widehat{m}_{t,j}}{\sqrt{\widehat{v}_{t,j}} + \eta_A/a_j}. \tag{57}$$

For $\eta_A = 0$*, the two updates coincide on coordinates with nonzero second moments; coordinates with all-zero gradient histories are assigned zero updates.*

Proof. Adam uses $m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t$ and $v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t \odot g_t$ (Kingma & Ba, 2015). Induction gives $\widetilde{m}_t = Dm_t$ and $\widetilde{v}_t = D^2 v_t$. Bias correction multiplies the corresponding moments by the same scalar. Substituting into the coordinate update and dividing numerator and denominator by a_j yields (57). The zero-stabilization case follows immediately. □

For nonzero stabilization, the updates are approximately identical when the second-moment scale dominates both stabilization terms. The proposition gives identical complete parameter trajectories under ideal Adam if the gradient-scaling relation holds at corresponding iterates throughout training. The trajectory statement requires the same fixed scaling relation for the gradients entering Adam throughout training, including any preceding clipping.

Corollary A.17 (A scale-only margin correction)**.** *Suppose the sole active positive-weight comparison is the current SPO+ query solution* $v = w^*(2z - c) \neq w^*(c)$*. For* $d = v - w^*(c)$ *and* $\lambda \geq 0$*,*

$$g_z = -2d, \qquad g_{\text{margin}} = -\frac{d}{\|d\|_2}, \qquad g_{\text{total}} = \left(1 + \frac{\lambda}{2\|d\|_2}\right) g_z. \tag{58}$$

For a single sample, this scalar relation also holds after multiplication by the same network Jacobian transpose. If the resulting total parameter-gradient history is a fixed positive scaling of the baseline history, Proposition A.16 applies.

Proof. The sole normalized positive weight equals one. Substituting the selected hinge derivative and SPO+ subgradient gives (58). Linearity of the Jacobian transpose preserves a scalar multiple for that sample. □

Minibatch aggregation combines sample-specific scales, and additional solution comparisons can rotate the parameter gradient. For example, adding $(0, -2)$ to $(1, 1)$ reverses the second coordinate of an ideal first-step Adam update. The propositions identify the fixed-scaling conditions under which gradient magnitude information is attenuated. Theorem A.7 establishes feasibility of the CCR output after every finite parameter update.

B Algorithm and Implementation

B.1 Training and inference procedure

The correction network has a fully connected hidden layer of width 128, a rectified linear unit (ReLU) activation, and a linear output layer. Its final weights and biases are zero at initialization. The backbone is frozen, and the comparison bank, anchor decisions, chosen slack rule, and budgets are constant with respect to the trainable parameters. All derivatives of the gauge map are retained. The coordinate multiplier a changes the parameterization; ρ changes the feasible prediction region.

B.2 Preprocessing and computational cost

For n training examples, bank construction requires at most three oracle queries per example before deduplication; cached solutions may reduce actual solves. New inputs require an anchor decision computed from their pretrained prediction. Training labels and pretrained predictions determine the

Algorithm 1 SPO+ fine-tuning with comparison-defined CCR

Require: Training pairs (x_i, c_i), frozen predictor c_0, $\rho > 0$, $a > 0$, setting CCR-U or CCR-D
Require: Objective sense $\sigma \in \{+1, -1\}$; oracle $\mathcal{O}_\sigma(c) \in \arg\min_{w \in \mathcal{S}} \sigma c^\top w$
1: Cache $w_i^* = \mathcal{O}_\sigma(c_i)$ and $w_{0i} = \mathcal{O}_\sigma(c_0(x_i))$.
2: Form $\mathcal{H}$ from distinct training decisions at c_i, $c_0(x_i)$, and $2c_0(x_i) - c_i$; use full enumeration when available.
3: Compute the training scale L by (8) and signed comparison normals $d_{iv} = \sigma(v - w_{0i})$.
4: **if** CCR-U **then**
5: Set $s_{iv} = \rho L$ for every retained comparison.
6: **else**
7: Compute training-residual supports e_{iv} by (9).
8: Set $s_{iv} = \rho L + \max_{v' \in \mathcal{H}} e_{iv'} - e_{iv}$.
9: **end if**
10: Initialize f_θ with a zero final layer.
11: **for** each training minibatch **do**
12: Compute $u_i = a\, c_0(x_i) \odot f_\theta(x_i)$ and $z_i = c_0(x_i) + u_i/(1 + p_i(u_i))$.
13: Query $v_i = \mathcal{O}_\sigma(2z_i - c_i)$ and select $g_{z,i} = 2\sigma(w_i^* - v_i)$.
14: Backpropagate $g_{z,i}$ through the full parameterization and update θ with Adam.
15: **end for**
16: Select a checkpoint by validation NDR.
17: At inference, compute $c_0(x)$ and $w_0(x) = \mathcal{O}_\sigma(c_0(x))$; construct the selected slacks, using training-residual supports for CCR-D.
18: Return $\mathcal{O}_\sigma(z_\theta(x))$.

bank, residual set, and objective scale. For a changing feasible set, a bank must contain decisions feasible for that input; our benchmark runs use one feasible set per capacity or scheduling specification.

Let $K = |\mathcal{H}|$ and d be the cost dimension. For CCR-D, direct residual support evaluation costs $O(nKd)$ per input and can be precomputed for a fixed dataset. CCR-U constructs its boundary directly from the common slack. The gauge costs $O(Kd)$ per example in both the forward and backward passes. A bank requires $O(Kd)$ storage, while precomputed supports for N inputs require $O(NK)$ storage. With observed-cost optima cached, SPO+ uses one reflected-cost oracle query per training example per epoch. The parameterization itself uses inner products, a maximum, and a scalar division. At inference, the anchor query precedes the ordinary final decision query.

B.3 Differentiation and numerical details

The forward map is almost everywhere differentiable. At a unique active positive comparison, the Jacobian is (36); if $p(u) = 0$ on an open branch, the map is the identity. The maximum over comparison ratios is computed with PyTorch's `amax`, which distributes gradients equally among tied maximizers. At zero, the composite map has derivative I. The positive slack $s_v \geq \rho L$ avoids zero denominators. The implementation requires a finite positive training scale L.

C Experimental Setup

This section specifies the downstream formulations, datasets, training settings, baseline implementations, and reporting protocol used in Section 6.

C.1 Problem formulations

C.1.1 Shortest path

The shortest-path benchmark is a linear program over an s–t grid graph. Let $\mathcal{E}$ denote the directed edge set and let c_e be the predicted cost of edge e. The downstream optimizer solves

$$\min_x \quad \sum_{e\in\mathcal{E}} c_e x_e \tag{59}$$

$$\text{s.t.} \quad Bx = b \tag{60}$$

$$0 \le x_e \le 1, \quad \forall e \in \mathcal{E}, \tag{61}$$

where B is the path-incidence matrix and b sends one unit of path mass from the source to the sink. Although this is an LP relaxation, the incidence matrix is totally unimodular, so optimal extreme-point solutions correspond to shortest paths.

C.1.2 Knapsack

Given item profits v_i, known weights w_i, and capacity c, the binary knapsack problem is

$$\max_x \quad \sum_{i=1}^{N} v_i x_i \tag{62}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} w_i x_i \le c \tag{63}$$

$$x_i \in \{0,1\}, \quad \forall i \in \{1,\dots,N\} \tag{64}$$

where x_i is the binary decision variable indicating whether item i is selected, v_i denotes the predicted profit of the item, w_i denotes the known physical weight, and c is the total capacity.

C.1.3 Energy-cost aware scheduling

Following the predict-and-optimize formulation of Mandi et al. (2020) and the Energy-1/Energy-2 benchmark convention of Mandi et al. (2022), the objective is to assign a set of tasks $\mathcal{J}$ to a set of machines $\mathcal{M}$ over a discretized time horizon $\mathcal{T}$ (with timeslots of q minutes) such that the overall energy cost is minimized. The energy price c_t at each timeslot t is the unknown parameter that the model must predict.

Each task $j \in \mathcal{J}$ has specific physical parameters: processing duration d_j, a vector of resource requirements $r_{j,k}$ across different resource types $k \in \mathcal{K}$ (e.g., CPU, memory), power consumption p_j, and time windows indicating the earliest start time es_j and latest end time le_j. Each machine $m \in \mathcal{M}$ has a maximum resource capacity vector $R_{m,k}$ for each resource type k. Additionally, tasks are non-preemptive, cannot migrate between machines once initiated, and cannot stretch over midnight.

Let $x_{j,m,t} \in \{0,1\}$ be a binary decision variable that equals 1 if task j starts on machine m exactly at timeslot t. The precise optimization formulation is:

$$\min_x \quad \sum_{j\in\mathcal{J}} \sum_{m\in\mathcal{M}} \sum_{t=es_j}^{le_j-d_j} x_{j,m,t} \left(p_j \frac{q}{60} \sum_{\tau=t}^{t+d_j-1} c_\tau \right) \tag{65}$$

$$\text{s.t.} \quad \sum_{m\in\mathcal{M}} \sum_{t=es_j}^{le_j-d_j} x_{j,m,t} = 1, \quad \forall j \in \mathcal{J} \tag{66}$$

$$\sum_{j\in\mathcal{J}} r_{j,k} \sum_{\tau=\max(0,t-d_j+1)}^{t} x_{j,m,\tau} \le R_{m,k}, \quad \forall m \in \mathcal{M}, \forall t \in \mathcal{T}, \forall k \in \mathcal{K} \tag{67}$$

$$x_{j,m,t} \in \{0,1\} \tag{68}$$

Constraint (66) ensures that every task is scheduled exactly once within its valid time window. Constraint (67) guarantees that the aggregate requirements of all concurrently running tasks for every

resource type k do not exceed the specific capacity $R_{m,k}$ of any machine m at any timeslot t. The non-preemptive and non-migration characteristics are inherently satisfied by the definition of the start-time variables $x_{j,m,t}$.

Although the MILP contains hundreds or thousands of binary start variables, its predicted objective parameter is the 48-dimensional price vector. Let $A^\top$ denote the fixed linear map from a start-variable assignment to its aggregate load profile and define the effective decision

$$w(x) = A^\top x \in \mathbb{R}^{48}, \qquad w_\tau(x) = \frac{q}{60} \sum_{j,m,t:\, t \le \tau < t+d_j} p_j x_{j,m,t}. \tag{69}$$

Then the objective in (65) is exactly $c^\top w(x)$. Consequently, the relevant feasible set for the prediction-space normal fan is the load-profile image $\mathcal{S}_{\text{Energy}} = \{A^\top x : x \text{ satisfies (66)–(68)}\}$. For two schedules x and x^*, their pairwise normal in price space is $A^\top(x - x^*) = w(x) - w(x^*)$, and its Euclidean norm determines the normalizing length of the comparison. The energy-scheduling oracle therefore returns the 48-dimensional load profile to SPO+ and GG-DFF; both the DFF box and CCR constraints use the same price coordinates.

Energy-1 uses `load1/day12`, with 10 tasks, 3 machines, and 411 valid binary start variables. Energy-2 uses `load2/day10`, with 15 tasks, 3 machines, and 744 valid binary start variables. Each specification fixes durations, time windows, resource requirements, and machine capacities across prediction days. Their distinct feasible sets define separate decision benchmarks.

C.2 Datasets and splits

Table 3 summarizes the fixed data splits. SP uses the grid shortest-path task of Elmachtoub & Grigas (2022), with 700 examples produced by the data-generation routine. A fixed 80/20 train–test split followed by a 15% validation holdout from the training portion gives the counts below. KP uses the published `Data.npz` file from Mandi et al. (2020), containing mildly correlated instances with 48 items and nine features per item. Its split is shared across the six capacities. Energy-1 and Energy-2 use the downloaded SEMO `prices2013.dat` data (Ifrim et al., 2012; Mandi et al., 2020) and share the same held-out days. The public code release of Mandi et al. (2020) distributes both `Data.npz` and `prices2013.dat`. The scheduling specifications are drawn from the standard-load instance files for CSPLib Problem 059 (Simonis et al., n.d.). Feature scaling is fitted on training data only.

Table 3: Dataset sizes and predicted objective dimensions. Energy-1 and Energy-2 share the price dataset.

Benchmark	Train	Validation	Test	Cost dimension
SP	476	84	140	24
KP	552	60	177	48
Energy-1	389	56	111	48
Energy-2	389	56	111	48

C.3 Architectures and fine-tuning

The predictive multilayer perceptron (MLP) has two hidden layers of width 128, ReLU activations, and a Softplus output. Correction networks have one ReLU hidden layer of width 128 and a linear output initialized at zero. Table 4 gives the fine-tuning settings. DFF and CCR share a frozen 50-epoch MSE backbone within each benchmark and use paired correction initialization and minibatch randomness. The eight runs use independent random seeds. Within each seed, correction initialization and minibatch randomness are paired across DFF and both CCR settings; standalone predictor runs vary predictor initialization as well.

For DFF, we expand the region-width search range based on the settings recommended by Yang et al. (2025).

Both CCR settings use the validation-selected budgets in Table 5.

All correction runs use Adam, batch size 32, zero weight decay, and gradient-norm clipping at 1. Epoch 0 is eligible for checkpoint selection; the earliest checkpoint is retained in a validation tie.

Table 4: Fine-tuning settings for DFF and both CCR settings in Table 1.

Setting	SP	KP	Energy-1	Energy-2
MSE pretraining epochs	50	50	50	50
Fine-tuning epochs	20	20	20	20
Learning rate	10^{-3}	5×10^{-4}	5×10^{-4}	5×10^{-4}
Coordinate scale a	0.375	0.125	0.150	0.150
DFF width ϵ	0.750	0.250	0.300	0.300
Validation interval (epochs)	2	1	2	2

The budget grid is $\rho \in \{0.01, 0.05, 0.2, 0.5, 1, 2\}$, selected by mean validation NDR. SP uses the full bank of 20 paths. KP and both energy benchmarks use distinct training solutions in their respective feasible sets, and all error supports use training residuals.

Table 5: Validation-selected budgets for the setting comparison. KP lists capacities $60, 90, 120, 150, 180, 210$ in order.

Benchmark	CCR-U: ρ	CCR-D: ρ
SP	2	2
KP	$2, 2, 1, 2, 2, 0.050$	$1, 0.500, 2, 2, 2, 0.010$
Energy-1	1	1
Energy-2	2	2

C.4 Baseline implementations and model selection

SP and KP standalone predictors train for 70 epochs at learning rate 10^{-3} and select the minimum-validation-NDR checkpoint every two epochs, including epoch 0. Both energy benchmarks train standalone predictors for 70 epochs at learning rate 10^{-3} and select the checkpoint with the lowest validation NDR. This validation-based checkpoint selection also applies to the Energy-1 2-Stage baseline. All use batch size 32, zero weight decay, and gradient clipping at 1.

Pairwise ranking. We implement the best-versus-rest pairwise hinge and feasible-solution cache of Mandi et al. (2022). The cache is expanded with an oracle query at each training example, using solve probability 1. SP trains an independently initialized predictor for 70 epochs; a single margin of 0.1 is selected from $\{0.1, 1\}$ using mean validation NDR. Both energy benchmarks train an independent predictor for 70 epochs with margin 1. KP transfers pairwise ranking to the same pretrained MSE initialization and updates the predictor for 20 epochs at learning rate 5×10^{-4}, with margin 1 and validation selection every two epochs. This KP initialization differs from the standalone SP and energy ranking runs. The separately implemented DFF adaptation freezes the backbone and trains a bounded correction module; its training protocol is described in Appendix D.6.

Perturbation and local-loss settings. PFYL uses eight perturbations for SP and KP and ten for both energy benchmarks. The Gaussian standard deviation is $10^{-4}s_0$ for SP, 0.1 for KP, and $10^{-3}\max(s_0, 10^{-6})$ for both energy benchmarks, where s_0 is the standard deviation of pretrained predictions on training data. LODL fits nonnegative coordinate weights for squared error and holds them fixed during predictor training. SP and KP use 20 perturbations, relative noise scale 0.1, and ridge coefficient 10^{-4}. KP local loss fitting uses the LP relaxation. Both energy benchmarks use eight perturbations, relative noise scale 0.05, ridge coefficient 10^{-4}, and up to 80 training examples for fitting its local loss.

C.5 Random seeds and aggregation

Each benchmark (SP, KP, Energy-1, and Energy-2) uses eight independent random seeds. DFF, CCR-U, and CCR-D use the same eight seeds, with correction initialization and minibatch randomness paired within each seed. KP uses the same eight seeds at each of its six capacities. All methods,

including standalone baselines, DFF, and both GG-DFF variants, select checkpoints using validation NDR. CCR budgets are also selected using validation NDR.

Every standalone baseline is also evaluated with eight independent random seeds on each benchmark. All reported comparisons and ablation controls therefore use eight runs per method and benchmark (and per capacity for KP).

For each method and run, NDR is the test-instance mean in (16); MSE is averaged over test instances and cost coordinates. KP first averages equally across capacities within each run. The main table then averages equally across the eight independent random seeds. The instance distributions in Appendix D.7 use the same runs and capacities.

C.6 Optimization oracles

SP evaluates paths exactly. KP correction runs use the exact discrete oracle; standalone predictors use MOSEK with binary relative and absolute gaps 10^{-9}. Energy scheduling uses the same Gurobi scheduling formulation throughout. Observed-cost reference decisions are cached for SPO+ fine-tuning. The 2-Stage method differentiates the corresponding barrier-smoothed LP relaxation with penalty 0.25, barrier parameter 10^{-3}, and Karush–Kuhn–Tucker (KKT) damping 10^{-6}. Numerical scaling and residual-checked linear solves implement these sensitivities without changing the relaxation's equations.

D Supplementary Experiments

We first give the motivation diagnostic protocol and complementary reachability calculations, then present boundary ablations, the two CCR settings, capacity-specific results, the ranking adaptation protocol, and test-regret distributions.

D.1 Matched-correction motivation experiment

We instantiate the matched-descent construction of Proposition A.13 in Appendix A.9 on the SP benchmark. This section gives the experimental construction and numerical checks.

Training-instance construction. We use the 476 training instances from the shortest-path benchmark and the predictions of a frozen model pretrained for 50 epochs using mean squared error. All 20 monotone paths on the 4×4 grid are enumerated. For each instance with nonzero g_z, we generate 2,048 Gaussian vectors and project them onto $g_z^{\perp}$, then normalize them to obtain candidate e_k. An exploratory scan uses $R/\|c_0\|_2 \in \{0.001, 0.0025, 0.005, 0.01, 0.025, 0.05, 0.1, 0.2, 0.4\}$ and $\alpha \in \{0.1, 0.01, 0\}$; the $\alpha = 0$ case gives displacements orthogonal to g_z and preserves the anchor loss. We retain candidates with an unchanged SPO+ oracle and, for the displayed experiment, strict feasibility in the existing DFF box with $\epsilon = 0.75$. The complete scan is retained with the experiment records.

Figure 2 uses $R/\|c_0\|_2 = 0.0025$ and $\alpha = 0.1$. We display a training instance with retained candidates that produce both an improvement and a deterioration relative to the same anchor, using one candidate of each sign. These constructed corrections demonstrate the decision consequences of equal surrogate descent on a training instance.

Exact decision and loss checks. Numerical values below are rounded for presentation; all oracle and interpolation checks use the unrounded predictions. The anchor, A, and B have SPO+ losses of approximately 3.375×10^{-3}, 3.011×10^{-3}, and 3.011×10^{-3}, respectively. Their NDRs are approximately 5.34%, 0%, and 17.84%. Their selected paths are distinct, while all three SPO+ queries have the same unique optimum and prediction-space subgradient norm $\sqrt{32}$. The maximum relative coordinate changes for A and B are approximately 0.536 and 0.312, both below the existing width 0.75. Their Euclidean correction norms agree to numerical precision. Enumeration verifies unique path optimality at the three predictions and unique SPO+ oracle optimality at their queries. We additionally check interpolation points along each correction segment, confirming the constant oracle and affine loss identity to absolute tolerance 10^{-14}.

The solution comparisons locate the decision boundaries crossed by the two corrections. Let w_A and w_B be the paths selected by A and B. For the improving path, $z^\top(w_A - w_0)$ is approximately -9.799×10^{-5} at A and 4.773×10^{-4} at B. For the worsening path, $z^\top(w_B - w_0)$ is approximately 9.730×10^{-4} at A and -3.693×10^{-5} at B. Thus the two corrections cross different decision boundaries while producing identical SPO+ progress. The experiment establishes this local distinction; the CCR analysis separately states the coverage and error conditions connecting comparison constraints to performance bounds.

D.2 Correction-region reachability analysis

D.2.1 Numerical verification of reachability thresholds

For the controlled example in Appendix A.11, the numerical audit reconstructs all four reachable DFF paths at $\epsilon = 0.4$ and both reachable CCR-U paths at $b = 1.5$ using the deployed model classes. Forward values, unique optimal decisions, and finite nonzero backpropagation are checked. These explicit network weights establish pointwise realizability of the displayed decisions.

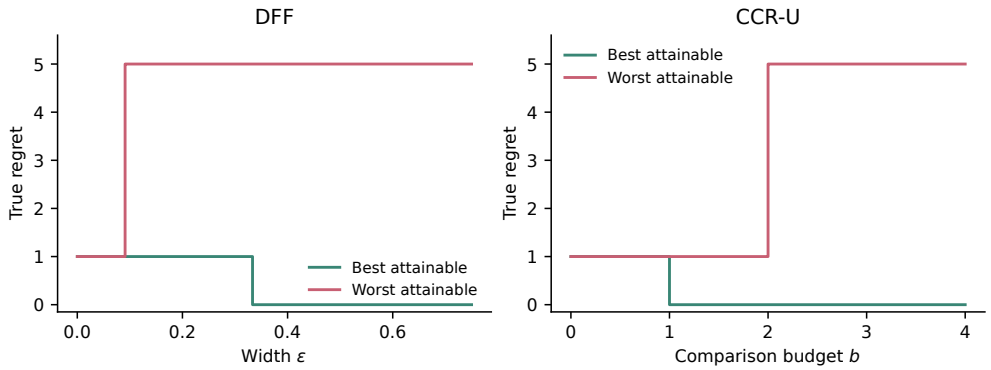


Figure 4: Complete budget scans for the control problem. The axes retain each method's own budget units. Threshold values are infima; strict reachability uses budgets above them. Independent LP checks verify both sides of every nonzero threshold.

D.2.2 Shortest-path validation analysis

We evaluate SP boundary geometry over eight independent random-seed runs, using the frozen 50-epoch MSE predictor, all 84 validation inputs per run, and all 20 paths. For each input, method, and path, an LP minimizes the budget required for optimality: ϵ for DFF and $\rho = b/L$ for CCR. The training solution bank and CCR-D residual-support offsets remain fixed. Validation costs evaluate the reachable decisions.

We enumerate all distinct thresholds to obtain the complete common-budget frontier. Each point uses one budget across the validation set within each run. For each input, we compute extrema over reachable decisions, then average across inputs and the eight runs. Table 6 averages statistics at each run's smallest common budget retaining every strictly improving path, plus 10^{-6} for strict reachability. This diagnostic evaluates boundary geometry using validation costs with the pretrained predictor fixed.

Table 6: SP validation statistics averaged over eight independent random seeds at complete retention of improving paths. Worst mean NDR averages per-input worst attainable regrets across inputs and runs (10^{-2} units).

Method	Budget	Improving kept	Worsening excluded	Worst mean NDR
DFF	0.470	19/19	1153/1577	86.167
CCR-U	0.066	19/19	1450/1577	29.723
CCR-D	1.000×10^{-6}	19/19	1062/1577	247.987

At matched complete retention of the improving paths, CCR-U excludes more worsening paths and has a lower worst attainable mean NDR than DFF. At the same retention target, CCR-D excludes fewer worsening paths and has higher worst attainable mean NDR than DFF. These reachable-decision extrema characterize the boundary tradeoff for each slack setting.

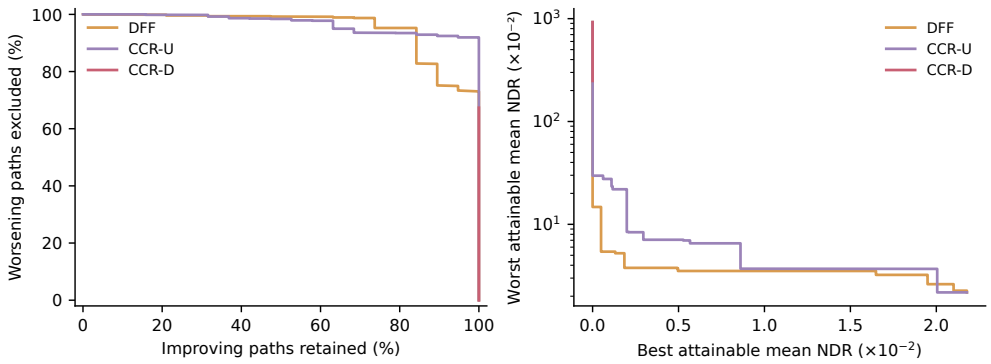


Figure 5: SP validation frontiers averaged over eight independent random-seed runs, with one common budget per point. Left: improving-path retention versus worsening-path exclusion, relative to the anchor. Right: best versus worst attainable mean NDR over reachable decisions. The right panel uses a logarithmic vertical axis. These curves evaluate boundary geometry with a frozen predictor.

D.3 Boundary ablation on KP and Energy-1

The KP and Energy-1 ablations extend the SP comparison in Table 2. Within each benchmark, the unbounded SPO+ correction, DFF, and both CCR settings share the same frozen predictor pretrained for 50 epochs using mean squared error, correction architecture, and eight independent random seeds. All correction checkpoints are selected by validation NDR. Removing the boundary retains the trainable correction and SPO+ objective. KP averages equally across the six capacities within each seed before averaging across seeds.

Table 7: Mean test NDR ($\times 10^{-2}$) for paired boundary controls on KP and Energy-1, averaged over eight independent random seeds. Lower is better.

Correction region	KP	Energy-1
Unbounded SPO+ correction	4.665	1.789
DFF box	4.703	1.835
GG-DFF-U	4.656	1.783
GG-DFF-D	4.665	1.789

Table 7 compares the four correction settings on KP and Energy-1. CCR-U attains the lowest mean NDR on both benchmarks, while CCR-D and the unbounded correction are close at the reported precision. The unbounded correction itself outperforms the DFF box on both benchmarks. The comparison geometry and slack allocation determine which corrections are available, and their effect on attained regret depends on the benchmark.

Table 8: Mean test MSE for the same paired boundary controls as Table 7, averaged over eight independent random seeds. Column headings specify the scale; lower is better.

Correction region	KP ($\times 10^4$)	Energy-1 ($\times 10^3$)
Unbounded SPO+ correction	2.845	1.019
DFF box	2.239	1.020
GG-DFF-U	2.783	1.019
GG-DFF-D	2.740	1.019

Table 8 reports prediction MSE for the same runs. The unbounded correction has the largest MSE on KP; all three bounded corrections reduce MSE relative to it. On Energy-1, DFF has the largest MSE, followed by the unbounded correction, while both CCR settings attain lower values. Prediction accuracy and decision quality can favor different settings: DFF has the lowest MSE on KP but the highest NDR among these controls, whereas CCR-U achieves the lowest NDR. The solution-comparison boundaries guide corrections in objective space, and their effects are evaluated through both prediction error and decision regret.

D.4 Uniform and direction-dependent CCR

CCR-U assigns a common comparison slack; CCR-D adjusts that slack using the training error support. Both retain the feasible-solution comparison directions and the almost-everywhere differentiable forward parameterization. Table 9 uses the same eight seeds per benchmark as the main table.

Table 9: Mean test NDR ($\times 10^{-2}$) for the paired CCR settings, averaged over eight independent random seeds.

Benchmark	DFF	GG-DFF-U	GG-DFF-D
SP	0.278	0.246	0.229
KP	4.703	4.656	4.665
Energy-1	1.835	1.783	1.789
Energy-2	1.931	1.900	1.890

CCR-D has the lower mean on SP and Energy-2, whereas CCR-U has the lower mean on Energy-1 and KP. On SP and both energy benchmarks, validation selects the same base budget for the two settings. Here CCR-D admits a superset of CCR-U's region while preserving their common conditional certificate. Its additional admissible corrections accompany lower attained regret on SP and Energy-2, but the smaller uniform region attains lower mean regret on Energy-1. This ordering distinguishes decision reachability from the result of training within the region: a larger admissible set permits additional decisions, while SPO+ optimization and the learned predictor determine which are reached. Appendix D.5 further separates the KP result by feasible-set capacity.

D.5 Results by knapsack capacity

Table 10 separates the six capacities using the same eight seeds for DFF and both CCR settings. The main-table KP result is the equally weighted average of these capacity means.

Table 10: Mean test NDR ($\times 10^{-2}$) by knapsack capacity, averaged over eight independent random seeds.

Capacity	DFF	GG-DFF-U	GG-DFF-D
60	10.023	9.872	9.861
90	7.332	7.207	7.315
120	4.921	4.835	4.815
150	2.723	2.770	2.771
180	1.875	1.901	1.881
210	1.343	1.351	1.347

The capacity comparison holds the prediction data and eight independent random seeds fixed while changing the feasible subsets. Both CCR settings improve on DFF at capacities 60–120, where DFF's mean NDR ranges from 4.921 to 10.023 on the 10^{-2} scale. At capacities 150–210, DFF's NDR is lower and the method ordering reverses. Thus the aggregate improvement is concentrated in the capacity regimes with higher baseline regret. This pattern is consistent with a role for the downstream feasible set: a change in predicted item profits can alter the best subset under one capacity and leave the optimizer unchanged under another. Each capacity also uses its own solution bank and validation-selected budget, so this comparison evaluates the complete capacity-specific fine-tuning procedure.

Under equal capacity weights, the absolute reductions at capacities 60–120 exceed the increases at capacities 150–210, yielding the mean reduction in Table 1.

D.6 Pairwise-ranking adaptation protocol

The knapsack ranking control starts from the same predictor pretrained for 50 epochs using mean squared error and uses a pairwise hinge with margin 1, solve probability 1, and 20 training epochs at learning rate 5×10^{-4}. The predictor version updates its weights; the DFF adaptation freezes them and trains a bounded correction with $\epsilon = 0.25$. Both select checkpoints by validation NDR every two epochs, including epoch 0. The prLTR row in Table 1 reports the predictor version; the bounded correction is a separate adaptation of the ranking objective. These two implementations differ in both the trained parameters and the output parameterization; the boundary controls in Appendix D.3 hold the correction architecture fixed within each benchmark.

D.7 Test regret distributions

Following the instance-distribution presentation of Mandi et al. (2022), Figure 6 pools test-instance regrets from the eight independent random-seed runs per method used in Table 1. For KP, each seed contributes the same number of instances at each of the six capacities, so the pooled mean agrees with the equally weighted table mean. These distributions describe test-instance variability over the evaluation runs.

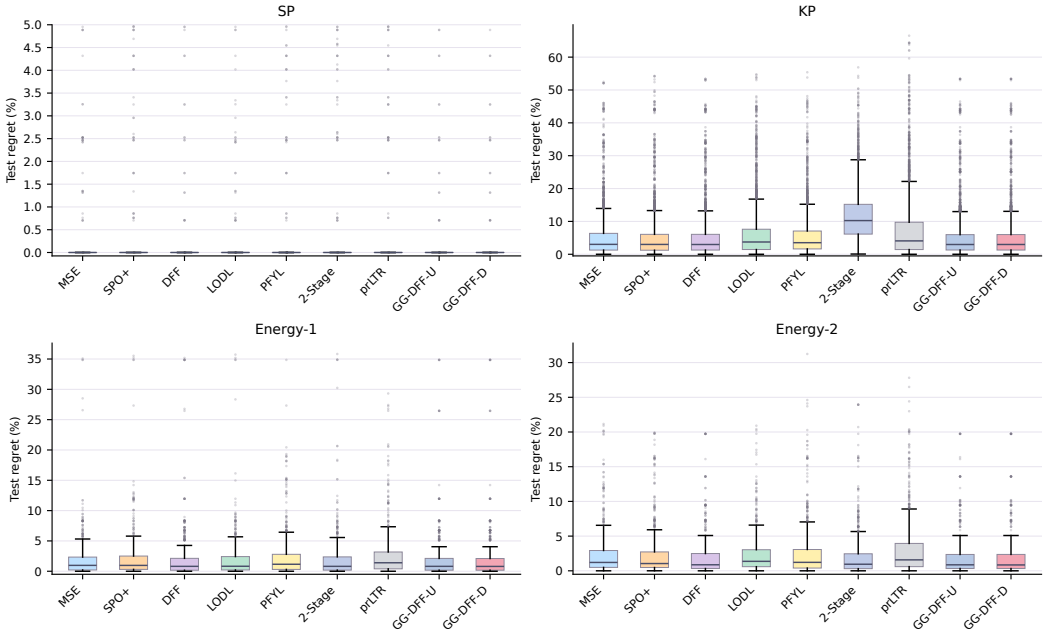


Figure 6: Test-instance normalized regret for all methods. Boxes show the interquartile range and median; whiskers extend to the most extreme observations within 1.5 interquartile ranges, and points show observations beyond the whiskers.

The distributions in Figure 6 complement the mean comparisons in Table 1. Similar medians and interquartile ranges can accompany different means when the positive-regret tails differ. The boxes summarize variability across test instances and runs.

Paired decision changes. To connect the mean differences to actual decisions, we compare the stored test decisions of each CCR run with its paired DFF run. The SP and KP vectors encode paths and selected items; the energy vectors encode aggregate consumption in the 48 cost coordinates. Equality of the latter means equality of the cost-relevant load profile, even if different task assignments realize it. We use coordinatewise tolerances 10^{-7} (absolute) and 10^{-9} (relative). Counts pool

instance–run pairs from the evaluation runs, with equal numbers of test instances per KP capacity. Repeated test instances across seeds remain repeated observations.

For each pair, let Δ_i be DFF's instance NDR minus CCR's instance NDR, in the 10^{-2} reporting units. The mean improvement decomposes as

$$\overline{\Delta} = \frac{1}{N}\sum_i [\Delta_i]_+ - \frac{1}{N}\sum_i [-\Delta_i]_+.$$

Pairs with unchanged cost-relevant decision vectors have zero regret difference. The decomposition separates the magnitudes of improvements and deteriorations, allowing us to assess decision quality beyond the fraction of decisions that change.

For KP, an aggregate regret reduction can combine improving and worsening decisions. Positive and negative contributions can offset, and their net effect depends on objective magnitudes. The capacity comparison in Table 10 makes this distinction concrete: both CCR settings improve on DFF at capacities 60–120, while DFF has lower regret at capacities 150–210.

On the energy benchmarks, the decision vector represents the cost-relevant load profile. Table 9 shows that both CCR settings lower mean NDR relative to DFF. The benefit of a changed profile is measured by its true-cost difference.

Decision improvement and prediction error. KP CCR-U lowers mean NDR despite increasing MSE from 2.239×10^4 to 2.783×10^4. In the normal-fan representation, preserving a decision requires inequalities on solution differences, whereas MSE measures errors in every cost coordinate. Thus lower decision regret can coexist with higher prediction error, as also shown by the main-table KP comparison. We therefore assess the correction region through the quality of reached decisions alongside prediction accuracy.

D.7.1 Post-hoc loss–regret analysis on the shortest-path test set

This is a post-hoc test-set diagnostic performed after the validation-selected checkpoints are fixed. We evaluate all 140 SP test instances in each of the eight runs, retaining their original saved predictions. With eight independent random seeds, the evaluation comprises $8 \times 140 = 1{,}120$ instance–run comparisons per method. Every method uses the same frozen predictor pretrained for 50 epochs with mean squared error as the comparison reference. Standalone SPO+ trains the full predictor from independent initialization for 70 epochs, selecting its checkpoint by validation NDR. Its comparison with the reference measures the difference between the two trained endpoints. All methods follow the seed and aggregation protocol in Appendix C.5.

Let ℓ_{0i} and R_{0i} be the reference SPO+ loss and NDR, and ℓ_{mi}, R_{mi} those of method m, where i indexes instance–run pairs. Define the lower-loss set $D_m = \{i : \ell_{mi} < \ell_{0i} - 10^{-10}\}$. **Only comparisons in D_m are counted:** regret improves if $R_{mi} < R_{0i} - 10^{-6}$, is unchanged if $|R_{mi} - R_{0i}| \le 10^{-6}$, and worsens if $R_{mi} > R_{0i} + 10^{-6}$, with NDR in percentage units. The last category is the loss–regret mismatch. Table 11 gives the counts underlying Figure 3(c), which divides each category by $|D_m|$. Comparisons outside D_m remain in the saved records. Losses, reference costs, and decisions are computed by exact path enumeration in double precision for every method.

Table 11: Mean per-run counts of regret outcomes among SP test comparisons **with lower SPO+ loss than the frozen pretrained reference**. Worsened regret is the mismatch category in Figure 3(c).

Method	Lower loss	Improved	Unchanged	Worsened
SPO+	28.667	10.667	17.667	0.333
DFF	38.000	10.000	27.000	1.000
GG-DFF-U	38.500	10.500	28.000	0.000
GG-DFF-D	38.000	11.000	27.000	0.000