Linear System Co-Design via Lifted Polyhedral Query Solving

Yubo Cai, Yujun Huang, Meshal Alharbi, Gioele Zardini

***Abstract*— Monotone co-design provides a mathematical framework for managing the tightly coupled trade-offs between subsystem functionalities and resource requirements that arise in complex engineered systems. In many practical settings the underlying component models are already linear, yet existing solvers still resort to generic discretization, sacrificing both accuracy and complexity by ignoring the topological structure of the problem. This paper isolates the corresponding exact regime by introducing Linear Design Problems (LDPs), a class of co-design problems whose feasible functionality–resource sets are polyhedra over Euclidean posets. We show that queries on LDPs reduce exactly to Multi-Objective Linear Programs (MOLPs), thereby bridging monotone co-design semantics and polyhedral multiobjective optimization. Furthermore, we prove that the LDP class is closed under series, parallel, intersection, and feedback interconnections, so that any composition of linear components yields a system-level LDP. Exploiting this closure, we derive a monolithic lifted formulation that retains block-angular sparsity and resolves a system-level co-design query through a single MOLP. We validate the approach on a rigid gripper co-design benchmark whose every component is an LDP. The monolithic algorithm recovers the exact Pareto front with zero approximation error and orders-of-magnitude faster runtime, substantially outperforming the state-of-the-art co-design solver MCDPL in both accuracy and speed.**

I. Introduction

Monotone co-design provides a mathematical framework for jointly designing interacting subsystems under coupled functionality and resource constraints [1]–[3]. Each component is modeled as a *design problem (DP)*: a monotone relation between partially ordered sets (posets) of functionalities and resources, and series, parallel, intersection, and feedback interconnections induce a system-level DP compositionally [4]–[6]. This separation of *structure* (interconnection graph) from *semantics* (each component's local feasibility relation) makes the framework broadly applicable to autonomy, robotics, and cyber-physical systems [3], [7], [8]. However, the generality of existing solvers comes at a computational price. In particular, MCDPL [1], [3] handles continuous functionality–resource relations by replacing them with finite-cardinality discretizations parameterized by a resolution R; the approximation refines only as $R \to \infty$. In many practical settings the underlying component models already possess special structure, such as linearity or convexity, yet such structure is entirely ignored by the discretization, sacrificing both solving accuracy and computational efficiency. This raises a natural question: *when component trade-offs are linear, can compositional co-design queries be solved exactly and efficiently, without discretization?*

The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 02139 Cambridge (MA), USA, {yubocai,yujun233,meshal,gzardini}@mit.edu

The work was supported by DARPA YFA (D25AC00373).

This paper answers affirmatively by introducing *Linear Design Problems (LDPs)*, a class of co-design problems whose feasible functionality–resource sets are polyhedra over Euclidean posets. Within this class, the central co-design query, i.e., computing the minimal resources needed to provide a given functionality, reduces exactly to a Multiple-objective Linear Programming (MOLP)/vector linear programming (VLP), thereby bridging monotone co-design semantics and polyhedral multiobjective optimization [9]–[12]. A key structural result underpins the scalability of this reduction: we prove that the LDP class is *closed* under series, parallel, intersection, and feedback interconnections, so that any linear co-design problem (LCDP) assembled from linear components inherits a system-level LDP. Exploiting this closure, we derive a *monolithic lifted formulation* that retains block-angular sparsity and resolves the system-level query through a single MOLP, without explicitly projecting out internal variables. To demonstrate the practical impact, we apply the approach to a rigid gripper co-design benchmark whose every component is an LDP; the monolithic solver recovers the exact Pareto front without the approximation artifacts inherent in discretization, while achieving orders-of-magnitude runtime reductions over MCDPL, confirming that exploiting linear structure at the formulation level yields both accuracy and scalability gains in practice.

II. Linear Co-Design Theory

We formalize linear design problems (LCDPs), i.e., DPs whose feasible sets are polyhedra over Euclidean posets. We first recall the order-theoretic and convex-geometric notation used throughout, then introduce LDPs and their query semantics.

A. Mathematical preliminaries

Definition 1 (Poset)**.** A *poset* is a tuple $\mathcal{P} = \langle P, \preceq_{\mathcal{P}} \rangle$, where P is a set and $\preceq_{\mathcal{P}}$ is a partial order (a reflexive, transitive, and antisymmetric relation). If clear from context, we use $\mathcal{P}$ for a poset, and $\preceq$ for its order.

Definition 2 (Opposite poset)**.** The *opposite* of a poset $\mathcal{P} = \langle P, \preceq_{\mathcal{P}} \rangle$ is the poset $\mathcal{P}^{\mathrm{op}} \stackrel{\text{def}}{=} \langle P, \preceq_{\mathcal{P}}^{\mathrm{op}} \rangle$ with the same elements and reversed ordering: $x_{\mathcal{P}} \preceq_{\mathcal{P}}^{\mathrm{op}} y_{\mathcal{P}} \Leftrightarrow y_{\mathcal{P}} \preceq_{\mathcal{P}} x_{\mathcal{P}}$.

Definition 3 (Product poset)**.** Given posets $\langle P, \preceq_{\mathcal{P}} \rangle$ and $\langle Q, \preceq_{\mathcal{Q}} \rangle$, their *product* $\langle P \times Q, \preceq_{\mathcal{P}\times\mathcal{Q}} \rangle$ is the poset with

$$\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \preceq_{\mathcal{P}\times\mathcal{Q}} \langle y_{\mathcal{P}}, y_{\mathcal{Q}} \rangle \Leftrightarrow (x_{\mathcal{P}} \preceq_{\mathcal{P}} y_{\mathcal{P}}) \wedge (x_{\mathcal{Q}} \preceq_{\mathcal{Q}} y_{\mathcal{Q}}).$$

Definition 4 (Upper closure)**.** Let $\mathcal{P}$ be a poset. The *upper closure* of a subset $X_{\mathcal{P}} \subseteq \mathcal{P}$ contains all elements of $\mathcal{P}$ that are greater or equal to some $y_{\mathcal{P}} \in X_{\mathcal{P}}$:

$$\uparrow X_{\mathcal{P}} \stackrel{\text{def}}{=} \{x_{\mathcal{P}} \in \mathcal{P} \mid \exists y_{\mathcal{P}} \in X_{\mathcal{P}} : y_{\mathcal{P}} \preceq_{\mathcal{P}} x_{\mathcal{P}}\}.$$

Definition 5 (Upper set)**.** A subset $X_{\mathcal{P}} \subseteq \mathcal{P}$ of a poset is called an *upper set* if it is upwards closed: $\uparrow X_{\mathcal{P}} = X_{\mathcal{P}}$. We write $\mathtt{U}(\mathcal{P})$ for the set of upper sets of $\mathcal{P}$. We regard $\mathtt{U}(\mathcal{P})$ as a poset with $U \preceq U' \Leftrightarrow U \supseteq U'$. Thus larger upper sets are considered *smaller* in this order: they will encode weaker constraints.

Definition 6 (Convex-geometric notation)**.** For a set $C \subseteq \mathbb{R}^n$, we write $\operatorname{conv}(C)$, $\operatorname{cone}(C)$, and $\operatorname{cl}(C)$ for the convex hull, conical hull, and topological closure of C, respectively [13, see Sec. 2.1.5]. A point $v \in P$ is a *vertex* of P if it cannot be written as $v = \frac{1}{2}(x + y)$ for distinct $x, y \in P$; we denote the vertex set by $\operatorname{vert}(P)$. A *halfspace* is a set $H^+ \overset{\text{def}}{=} \{x \in \mathbb{R}^n \mid a^\top x \geq b\}$ with $a \in \mathbb{R}^n$, $b \in \mathbb{R}$.

Definition 7 (Polyhedron)**.** A *polyhedron* P is the intersection of a finite number of halfspaces. That is:

$$\begin{aligned} P = \bigcap_{i=1}^{m} H_i^+ &\overset{\text{def}}{=} \bigcap_{i=1}^{m} \{x \in \mathbb{R}^n : a_i^\top x \geq b_i\} \\ &= \{x \in \mathbb{R}^n : Ax \geq b\}, \end{aligned}$$

where $a_i^\top$ form the rows of A and b_i are the components of b. This is called the $\mathbf{H}$*-representation* of a polyhedron. A polyhedron can also be written in $\mathbf{V}$*-representation* as

$$P = \operatorname{conv}\{v^1, \ldots, v^p\} + \operatorname{cone}\{r^1, \ldots, r^q\},$$

where $\{v^1, \ldots, v^p\}$ are vertices and $\{r^1, \ldots, r^q\}$ are extreme rays. A bounded polyhedron (equivalently, $q = 0$) is called a *polytope*.

Remark. By the Minkowski–Weyl theorem, $\mathbf{H}$- and $\mathbf{V}$-representations of polyhedra are equivalent, though conversion can be exponential [14]. We work primarily in $\mathbf{H}$-representation; $\mathbf{V}$-information is recovered via MOLP/VLP solvers when needed [11], [15].

B. Background on co-design and queries

Co-design takes a compositional view of complex system design: each subsystem is characterized by the trade-off between what it can provide and what it must consume, and these local trade-offs are then interconnected into a system-level problem. Consider, for instance, the design of a unmanned aerial vehicle (UAV) platform (see [16, Fig 1(a)]). The airframe, actuators, energetics, and batteries each impose their own functionality–resource relations, i.e., a battery delivers *energy capacity* at the cost of *mass* and *price*; the actuator converts *lift force* from *power* and *weight*, and these relations are coupled through shared variables such as total mass and power budget. Crucially, the relevant quantities often live in spaces that are only *partially* ordered: two battery chemistries may offer incomparable mass-versus-cost profiles, so neither dominates the other. Monotone co-design formalizes this structure by modeling each component as a DP defined over posets of functionalities and resources [1]–[3].

Definition 8 (DP)**.** Given posets F and R of *functionalities* and *resources*, a *DP* is an upper set of $F^{\mathrm{op}} \times R$. We denote the set of such DPs by $\mathsf{DP}\{F, R\}$. Given a DP dp, a pair $\langle x_F, x_R\rangle$ of functionality x_F and resource x_R is *feasible* if $\langle x_F, x_R\rangle \in \mathrm{dp}$. We order $\mathsf{DP}\{F, R\}$ by inclusion: $\mathrm{dp}_a \preceq \mathrm{dp}_b \Leftrightarrow \mathrm{dp}_a \subseteq \mathrm{dp}_b$. Note that this is the opposite of the ordering used for upper sets, so that larger feasible sets correspond to "better" Design Problems (DPs).

The upper set condition captures the following intuition: if a resource x_R suffices to provide functionality x_F, then it also suffices for any worse functionality $x'_F \preceq x_F$. Moreover, any better resource $x'_R \succeq x_R$ should also suffice to provide x_F. Equivalently, dp is a monotone relation: if $\langle x_F, x_R\rangle \in \mathrm{dp}$ and $x'_F \preceq x_F$, $x'_R \succeq x_R$, then $\langle x'_F, x'_R\rangle \in \mathrm{dp}$.

A core tenet of co-design is to compose systems out of simpler sub-systems. Such composites are formalized as multi-graphs of DPs called *Co-Design Problems (CDPs)*. We summarize the main composition operations in Definition 9, with some shown diagrammatically in Fig. 1.

Definition 9 (Composition operations for DPs)**.** The following operations construct new DPs from old:

Series: Given DPs $\mathrm{dp}_a \in \mathsf{DP}\{\mathcal{P}, \mathcal{Q}\}$ and $\mathrm{dp}_b \in \mathsf{DP}\{\mathcal{Q}, \mathcal{R}\}$, their series connection $\mathrm{dp}_a \mathbin{⨟} \mathrm{dp}_b \in \mathsf{DP}\{\mathcal{P}, \mathcal{R}\}$ is defined as

$$\{\langle x_{\mathcal{P}}, x_{\mathcal{R}}\rangle \mid \exists x_{\mathcal{Q}} : \langle x_{\mathcal{P}}, x_{\mathcal{Q}}\rangle \in \mathrm{dp}_a \text{ and } \langle x_{\mathcal{Q}}, x_{\mathcal{R}}\rangle \in \mathrm{dp}_b\}.$$

This models situations where dp_a uses the functionalities provided by dp_b as its resources.

Parallel: For $\mathrm{dp}_a \in \mathsf{DP}\{\mathcal{P}, \mathcal{Q}\}$, $\mathrm{dp}'_a \in \mathsf{DP}\{\mathcal{P}', \mathcal{Q}'\}$, their parallel connection $\mathrm{dp}_a \otimes \mathrm{dp}'_a \in \mathsf{DP}\{\mathcal{P} \times \mathcal{P}', \mathcal{Q} \times \mathcal{Q}'\}$ is

$$\{\langle\langle x_{\mathcal{P}}, x'_{\mathcal{P}}\rangle, \langle x_{\mathcal{Q}}, x'_{\mathcal{Q}}\rangle\rangle \mid \langle x_{\mathcal{P}}, x_{\mathcal{Q}}\rangle \in \mathrm{dp}_a, \langle x'_{\mathcal{P}}, x'_{\mathcal{Q}}\rangle \in \mathrm{dp}'_a\},$$

representing two non-interacting systems.

Intersection: For $\mathrm{dp}_a, \mathrm{dp}_b \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$, their intersection $\mathrm{dp}_a \wedge \mathrm{dp}_b \in \mathsf{DP}\{\mathcal{P}, \mathcal{Q}\}$ is

$$\{\langle x_{\mathcal{P}}, x_{\mathcal{Q}}\rangle \mid \langle x_{\mathcal{P}}, x_{\mathcal{Q}}\rangle \in \mathrm{dp}_a, \langle x_{\mathcal{P}}, x_{\mathcal{Q}}\rangle \in \mathrm{dp}_b\}.$$

Feedback/Trace: For $\mathrm{dp} \in \mathsf{DP}\{\mathcal{P} \times \mathcal{R}, \mathcal{Q} \times \mathcal{R}\}$, its trace $\operatorname{Tr}(\mathrm{dp}) \in \mathsf{DP}\{\mathcal{P}, \mathcal{Q}\}$ is defined as

$$\{\langle x_{\mathcal{P}}, x_{\mathcal{Q}}\rangle \mid \exists x_{\mathcal{R}} : \langle\langle x_{\mathcal{P}}, x_{\mathcal{R}}\rangle, \langle x_{\mathcal{Q}}, x_{\mathcal{R}}\rangle\rangle \in \mathrm{dp}\}.$$

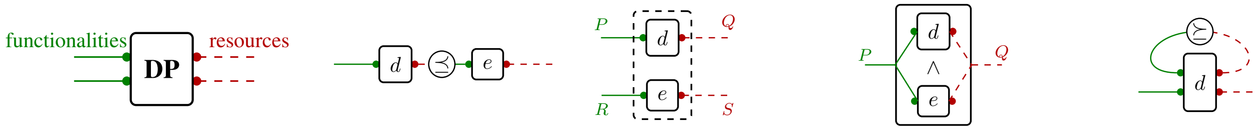
This models the case where functionalities provided by dp are used as its own resources.

These compositions allow the construction of a DP from co-design problems (CDPs) such as Fig. 2 [3].

Definition 10 (CDPs)**.** We call $\langle F, R, \langle \mathcal{V}, \mathcal{E}\rangle\rangle$ a CDP, where $F = \Pi_{i=1}^{n_F} F_i$ and $R = \Pi_{j=1}^{n_R} R_j$ are the system-level functionalities and resources, and $\langle \mathcal{V}, \mathcal{E}\rangle$ is a multi-graph where:

(i) Each node in $\mathcal{V}$ is a DP d with functionality $F_d = \Pi_{i=1}^{n_{d,F}} F_{d,i}$ and resource $R_d = \Pi_{j=1}^{n_{d,R}} R_{d,j}$.
(ii) Each edge is a pair $\langle\langle d_1, i_1\rangle, \langle d_2, j_2\rangle\rangle$ with the corresponding posets R_{d_1,i_1} and F_{d_2,j_2} being the same, representing the inequality $x_{R_{d_1,i_1}} \preceq x_{F d_2,j_2}$ (the i_1-th resource required by d_1 is at most the j_2-th functionality provided by d_2).
(iii) The system-level functionality/resource is the production of some component level functionalities/resources.

With these modeling primitives in hand, we can ask for optimal solutions to DPs, formalized as *queries*.



(a) A DP. (b) Series (⨾). (c) Parallel (⊗). (d) Intersection (∧). (e) Loop (Tr).

Fig. 1: A DP is a monotone relation between posets of functionalities and resources (a), and can be composed via series, parallel, intersection, and loop operations (b)–(e).

Definition 11 (Querying DPs)**.** Given a DP $\mathrm{dp} \in \mathsf{DP}\{F, R\}$, we define two basic queries:

1) *Fix functionalities minimize resources*: for a fixed $x_F \in F$, return the upper set of resources $x_R \in R$ that make $\langle x_F, x_R\rangle$ feasible with respect to dp. We can view this query as a monotone map $q\colon F \to \mathtt{U}(R)$.
2) *Fix resources maximize functionalities*: For a fixed $x_R \in R$, return the upper set of functionalities $x_F \in F$ that make $\langle x_F, x_R\rangle$ feasible with respect to dp. We can view this query as a monotone map $q'\colon R \to \mathtt{U}(F^{\mathrm{op}})$.

In applications we are typically interested in the *minimal* resources in $q(x_F)$ and the *maximal* functionalities in $q'(x_R)$ (in the sense of Pareto optimality).

When F and R are general posets, there might be a *Pareto front* of optimal values and solutions to the queries, instead of a unique one. Using the compositional structure, one can derive efficient algorithms to calculate the query results for complex DPs defined by a multi-graph of sub-systems [3].

C. Linear co-design problems and queries

We next focus on DPs endowed with additional geometric structure. Based on Definition 8, we introduce LDPs.

Definition 12 (Euclidean poset and sub-poset)**.** For $n \in \mathbb{N}$, the *Euclidean poset* $\langle \mathbb{R}^n_{\geq 0}, \preceq\rangle$ is the set $\mathbb{R}^n_{\geq 0} \overset{\text{def}}{=} \{x \in \mathbb{R}^n \mid x_i \geq 0,\ i = 1, \dots, n\}$ equipped with the componentwise order $x \preceq y \Leftrightarrow x_i \leq y_i$ for all i. A subset $S \subseteq \mathbb{R}^n_{\geq 0}$ with the restricted order is called a *Euclidean sub-poset*.

Definition 13 (LDP)**.** ·Let $F \subseteq \left\langle \mathbb{R}^{n_F}_{\geq 0}, \preceq \right\rangle$ and $R \subseteq \left\langle \mathbb{R}^{n_R}_{\geq 0}, \preceq \right\rangle$ be Euclidean sub-posets. A *LDP* is a DP $\mathrm{dp} \in \mathsf{DP}\{F, R\}$ for which the set of all feasible pairs $\langle x_F, x_R\rangle$ forms a polyhedron in the underlying Euclidean space. Equivalently, there exist matrices $A_F \in \mathbb{R}^{m\times n_F}$, $A_R \in \mathbb{R}^{m\times n_R}$, and a vector $b \in \mathbb{R}^m$ such that

$$\mathrm{ldp} = \left\{ \langle x_F, x_R\rangle \in F^{\mathrm{op}} \times R \;\middle|\; A_F x_F + A_R x_R \geq b \right\}. \quad (1)$$

We denote the set of all such LDPs by $\mathsf{LDP}\{F, R\}$. For conciseness, we can write the condition as $A_{\mathrm{ldp}} x_{\mathrm{ldp}} \geq b$, where $A_{\mathrm{ldp}} = [A_F \mid A_R]$ and $x_{\mathrm{ldp}} = \left[x_F^\top, x_R^\top\right]^\top$.

Remark. Since F and F^{op} share the same underlying set, Eq. (1) is written over $F \times R$ and the order reversal appears only in the requirement that dp be an upper set in $F^{\mathrm{op}} \times R$. Not every polyhedron in $F \times R$ therefore defines an LDP: we additionally require the monotonicity encoded by Definition 8.

Following the query formalism in Definition 11 (see also [2, Section 29.3]), the *fix functionalities minimize resources* query asks, for a fixed $x_F \in F$, for the set of *minimal* (Pareto) resources x_R such that $\langle x_F, x_R\rangle$ is feasible; the *fix resources maximize functionalities* query asks, for a fixed $x_R \in R$, for the set of *maximal* (Pareto) functionalities x_F such that $\langle x_F, x_R\rangle$ is feasible. For LDPs, these become MOLPs.

Definition 14 (MOLP)**.** A MOLP in standard form is the optimization problem

$$\operatorname{Min}_{\preceq} \quad Cx \quad \text{subject to} \quad Ax \geq b, \quad x \geq 0,$$

where $C \in \mathbb{R}^{q\times n}$, $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, the variable $x \in \mathbb{R}^n$, and $\preceq$ is the componentwise partial order on $\mathbb{R}^q$. We call the set of feasible points $X := \{x \geq 0 : Ax \geq b\}$ the *feasible set* of the MOLP.

Definition 15 (Efficient point)**.** A feasible point $x^* \in X$ is called *efficient* (or *Pareto optimal*) for the MOLP in Definition 14 if there is no feasible $x \in X$ with $Cx \preceq Cx^*$ and $Cx \neq Cx^*$. Equivalently, $Cx^* \notin CX + \mathbb{R}^q_{\geq 0} \setminus \{0\}$, where $CX := \{Cx \mid x \in X\}$. The set of all efficient points is denoted X_E, and its image CX_E is the *efficient frontier* (or *Pareto front*).

Definition 16 (Upper image of an MOLP)**.** For the MOLP in Definition 14, the *upper image* is

$$\mathcal{P} \overset{\text{def}}{=} \mathrm{cl}\left(CX + \mathbb{R}^q_{\geq 0}\right),$$

In the polyhedral linear case considered here, $\mathcal{P}$ is a polyhedron whose lower boundary coincides with the efficient frontier CX_E. More generally, replacing the ordering cone $\mathbb{R}^q_{\geq 0}$ with an arbitrary polyhedral cone K yields a *vector linear program* (VLP); the solver BENSOLVE [11] computes **V**- and **H**-representations of $\mathcal{P}$ for VLPs.

Lemma 1 (Queries of LDP are equivalent to MOLP)**.** Consider an LDP $\mathrm{dp} \in \mathsf{LDP}\{F, R\}$ with feasible set $\mathrm{ldp} = \{\langle x_F, x_R\rangle \in F \times R : A_F x_F + A_R x_R \geq b\}$.

(i) *Fix functionalities minimize resources*: For fixed $\bar{x}_F \in F$, the set of minimal resources that make $\langle \bar{x}_F, x_R\rangle$ feasible coincides with the set of efficient solutions of the MOLP

$$\operatorname{Min}_{\preceq} x_R \quad \text{s.t. } A_R x_R \geq b - A_F \bar{x}_F, \quad x_R \geq 0.$$

(ii) *Fix resources maximize functionalities*: For fixed $\bar{x}_R \in R$, the set of maximal functionalities that make $\langle x_F, \bar{x}_R\rangle$ feasible coincides with the set of efficient solutions of the MOLP

$$\operatorname{Max}_{\preceq} x_F \quad \text{s.t. } A_F x_F \geq b - A_R \bar{x}_R, \quad x_F \geq 0,$$

which is equivalent to $\operatorname{Min}_{\preceq}(-x_F)$ subject to the same constraints (efficient points are unchanged).

(iii) Conversely, every MOLP of the form $\text{Min}_{\preceq} x$ subject to $Ax \geq b,\ x \geq 0$ is the fix-*functionalities*-minimize-*resources* query of the LDP with functionality space $F = \{0\} \subseteq \mathbb{R}_{\geq 0}$ (singleton), resource space $R = \mathbb{R}^n_{\geq 0}$, and feasible set $\text{ldp} = \{\langle 0, x\rangle : Ax \geq b,\ x \geq 0\}$ (i.e., $A_F = 0,\ A_R = A$).

Proof. We prove the three statements in order. **(i)** Fix $\bar{x}_F \in F$ and consider the associated feasible resource set $R(\bar{x}_F) \overset{\text{def}}{=} \{x_R \in R \mid \langle \bar{x}_F, x_R\rangle \in \text{ldp}\}$. By the polyhedral representation of ldp, a pair $\langle \bar{x}_F, x_R\rangle$ is feasible if and only if

$$A_R x_R \ \geq\ b - A_F \bar{x}_F, \quad x_R \in R \subseteq \mathbb{R}^{n_R}_{\geq 0}.$$

Hence $R(\bar{x}_F)$ coincides with the feasible set of the vector optimization problem

$$\text{Min}_{\preceq}\ x_R \quad \text{s.t. } A_R x_R \geq b - A_F \bar{x}_F, \quad x_R \geq 0,$$

where $\preceq$ is the componentwise order on $\mathbb{R}^{n_R}$. By Definition 11, the query answer at $\bar{x}_F$ is precisely the set of $\preceq$–minimal elements of $R(\bar{x}_F)$. These are exactly the efficient (Pareto–optimal) solutions of the above MOLP with objective matrix $C = I$, establishing (i).

(ii) The statement for the “fix resources, maximize functionalities” query is proved by an order–duality argument entirely analogous to (i). For fixed $\bar{x}_R$ the feasible functionality set is the polyhedron $F(\bar{x}_R) = \{x_F \in F \mid A_F x_F \geq b - A_R \bar{x}_R,\ x_F \geq 0\}$. Taking $\preceq$–maximal elements of $F(\bar{x}_R)$ is equivalent to taking $\preceq$–minimal elements of $-F(\bar{x}_R)$, so exactly as in (i) one obtains that the query result coincides with the efficient set of the MOLP in (ii).

(iii) Let a MOLP of the form

$$\text{Min}_{\preceq} x \quad \text{subject to} \quad Ax \geq b, \quad x \geq 0$$

be given, where $x \in \mathbb{R}^n_{\geq 0}$ and $\preceq$ is the componentwise order. Define an LDP with functionality space $F \overset{\text{def}}{=} \{0\} \subseteq \mathbb{R}_{\geq 0}$, resource space $R \overset{\text{def}}{=} \mathbb{R}^n_{\geq 0}$, and feasible set

$$\text{ldp} \overset{\text{def}}{=} \{\langle 0, x\rangle \in F \times R \mid Ax \geq b\}.$$

This is an LDP, since ldp is a polyhedron in $\mathbb{R}^1 \times \mathbb{R}^n$ and admits the representation

$$\text{ldp} = \{\langle x_F, x_R\rangle \mid A_F x_F + A_R x_R \geq b\},$$

with $A_F = 0$ and $A_R = A$. The fix-*functionalities*-minimize-*resources* query evaluated at the unique functionality $\bar{x}_F = 0$ asks for the set of $\preceq$–minimal x_R such that $\langle 0, x_R\rangle \in \text{ldp}$, that is,

$$\{x_R \in \mathbb{R}^n_{\geq 0} \mid A x_R \geq b\}$$

together with its $\preceq$–minimal elements. By definition, these minimal elements are exactly the efficient solutions of the original MOLP, which proves (iii). ☐

We are now ready to define LCDPs.

Definition 17 (LCDP)**.** Let $\langle F, R, \langle \mathcal{V}, \mathcal{E}\rangle\rangle$ be a CDP in the sense of Definition 10. We call $\langle F, R, \langle \mathcal{V}, \mathcal{E}\rangle\rangle$ a *linear co-design problem (LCDP)* if the following conditions hold:

(i) (*Linear component design problems*) For each node $d \in \mathcal{V}$, the associated design problem

$$d = \langle F_d, R_d, \text{ldp}_d\rangle$$

has functionality and resource posets

$$F_d \subseteq \left\langle \mathbb{R}^{n_{F_d}}_{\geq 0}, \preceq \right\rangle, \qquad R_d \subseteq \left\langle \mathbb{R}^{n_{R_d}}_{\geq 0}, \preceq \right\rangle,$$

and its feasible set $\text{ldp}_d \subseteq F_d \times R_d$ is a polyhedron. Equivalently, there exist matrices $A_{F_d} \in \mathbb{R}^{m_d \times n_{F_d}}$, $A_{R_d} \in \mathbb{R}^{m_d \times n_{R_d}}$, and a vector $b_d \in \mathbb{R}^{m_d}$ satisfying Eq. (1).

(ii) (*Linear interconnection*) Each edge $e \in \mathcal{E}$ is a pair $e = \langle\langle d_1, i_1\rangle, \langle d_2, j_2\rangle\rangle$ that links the i_1-th resource component of node $d_1 \in \mathcal{V}$ to the j_2-th functionality component of node $d_2 \in \mathcal{V}$, with the two components identified by inequality of the corresponding coordinate projections, i.e. $\pi_{i_1}(R_{d_1}) \leq \pi_{j_2}(F_{d_2})$.

In other words, an LCDP is a CDP whose component design problems are all LDPs and interconnections are element-wise inequalities. Consequently, the system-level functionality/resource are Euclidean posets.

The following lemma shows LDPs are closed under the compositions in Definition 9.

Lemma 2 (Closure of LDPs under composition)**.** Let $\mathcal{P}, \mathcal{Q}, \mathcal{R}, \mathcal{P}', \mathcal{Q}'$ be Euclidean sub-posets (Definition 12). Then the following hold:

(a) *Series.* For any $\text{dp}_a \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$ and $\text{dp}_b \in \mathsf{LDP}\{\mathcal{Q}, \mathcal{R}\}$, their series composition $\text{dp}_a \fatsemi \text{dp}_b$ satisfies $\text{dp}_a \fatsemi \text{dp}_b \in \mathsf{LDP}\{\mathcal{P}, \mathcal{R}\}$.
(b) *Parallel.* For any $\text{dp}_a \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$ and $\text{dp}'_a \in \mathsf{LDP}\{\mathcal{P}', \mathcal{Q}'\}$, their parallel composition $\text{dp}_a \otimes \text{dp}'_a$ satisfies $\text{dp}_a \otimes \text{dp}'_a \in \mathsf{LDP}\{\mathcal{P} \times \mathcal{P}', \mathcal{Q} \times \mathcal{Q}'\}$.
(c) *Intersection.* For any $\text{dp}_a, \text{dp}_b \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$, their intersection $\text{dp}_a \wedge \text{dp}_b$ satisfies $\text{dp}_a \wedge \text{dp}_b \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$.
(d) *Feedback.* For any $\text{dp} \in \mathsf{LDP}\{\mathcal{P} \times \mathcal{R}, \mathcal{Q} \times \mathcal{R}\}$, its trace $\text{Tr}(\text{dp})$ satisfies $\text{Tr}(\text{dp}) \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$.

Proof. By Definition 13, a DP over Euclidean sub-posets is an LDP if and only if its feasible set is a polyhedron. Throughout, we use that polyhedra are closed under Cartesian products, intersections, and projections. We give the full argument for series and feedback; the parallel and intersection cases are more direct (Cartesian product and constraint stacking, respectively).

Series. Let $\text{dp}_a \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$ and $\text{dp}_b \in \mathsf{LDP}\{\mathcal{Q}, \mathcal{R}\}$ be LCDPs with feasible sets

$$\begin{aligned}\text{ldp}_a &= \{\langle x_{\mathcal{P}}, x_{\mathcal{Q}}\rangle \mid A_{\mathcal{P}} x_{\mathcal{P}} + A_{\mathcal{Q}} x_{\mathcal{Q}} \geq b_a\},\\ \text{ldp}_b &= \{\langle x_{\mathcal{Q}}, x_{\mathcal{R}}\rangle \mid B_{\mathcal{Q}} x_{\mathcal{Q}} + B_{\mathcal{R}} x_{\mathcal{R}} \geq b_b\},\end{aligned}$$

for suitable matrices $\langle A_{\mathcal{P}}, A_{\mathcal{Q}}, b_a\rangle$ and $\langle B_{\mathcal{Q}}, B_{\mathcal{R}}, b_b\rangle$. By Definition 9, the series connection $\text{dp}_a \fatsemi \text{dp}_b \in \mathsf{DP}\{\mathcal{P}, \mathcal{R}\}$ has feasible set

$$\begin{aligned}\text{ldp}_{\text{series}} = \{\langle x_{\mathcal{P}}, x_{\mathcal{R}}\rangle \mid \exists\, x_{\mathcal{Q}} : A_{\mathcal{P}} x_{\mathcal{P}} + A_{\mathcal{Q}} x_{\mathcal{Q}} \geq b_a,&\\ B_{\mathcal{Q}} x_{\mathcal{Q}} + B_{\mathcal{R}} x_{\mathcal{R}} \geq b_b\}.&\end{aligned} \tag{2}$$

In the extended variable space $x \overset{\text{def}}{=} [x_{\mathcal{P}}^\top, x_{\mathcal{Q}}^\top, x_{\mathcal{R}}^\top]^\top$, the constraints in (2) can be written as

$$\begin{bmatrix} A_{\mathcal{P}} & A_{\mathcal{Q}} & 0 \\ 0 & B_{\mathcal{Q}} & B_{\mathcal{R}} \end{bmatrix} x \geq \begin{bmatrix} b_a \\ b_b \end{bmatrix},$$

which defines a polyhedron in $\langle x_{\mathcal{P}}, x_{\mathcal{Q}}, x_{\mathcal{R}} \rangle$. The feasible set $\mathrm{ldp}_{\text{series}}$ is its projection onto the $\langle x_{\mathcal{P}}, x_{\mathcal{R}} \rangle$ coordinates, hence is again a polyhedron. Since the external functionality and resource posets are still $\mathcal{P}$ and $\mathcal{R}$, which are Euclidean sub-posets by assumption, $\mathrm{dp}_a \mathbin{⨟} \mathrm{dp}_b$ is an LCDP.

Parallel. Let $\mathrm{dp}_a \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$ and $\mathrm{dp}'_a \in \mathsf{LDP}\{\mathcal{P}', \mathcal{Q}'\}$ be LCDPs with feasible sets

$$\mathrm{ldp}_a = \{\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \mid A_{\mathcal{P}} x_{\mathcal{P}} + A_{\mathcal{Q}} x_{\mathcal{Q}} \geq b_a\},$$
$$\mathrm{ldp}'_a = \{\langle x_{\mathcal{P}'}, x_{\mathcal{Q}'} \rangle \mid A_{\mathcal{P}'} x_{\mathcal{P}'} + A_{\mathcal{Q}'} x_{\mathcal{Q}'} \geq b'_a\}.$$

By Definition 9, their parallel connection $\mathrm{dp}_a \otimes \mathrm{dp}'_a \in \mathsf{DP}\{\mathcal{P} \times \mathcal{P}', \mathcal{Q} \times \mathcal{Q}'\}$ has feasible set

$$\begin{aligned} \mathrm{ldp}_{\text{par}} = \Big\{ \langle \langle x_{\mathcal{P}}, x_{\mathcal{P}'} \rangle, \langle x_{\mathcal{Q}}, x_{\mathcal{Q}'} \rangle \rangle \;\Big| \\ \langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \in \mathrm{ldp}_a,\ \langle x_{\mathcal{P}'}, x_{\mathcal{Q}'} \rangle \in \mathrm{ldp}'_a \Big\}. \end{aligned} \tag{3}$$

The right-hand side is the Cartesian product $\mathrm{ldp}_a \times \mathrm{ldp}'_a$ of two polyhedra in disjoint variables, hence a polyhedron. Moreover, $\mathcal{P} \times \mathcal{P}'$ and $\mathcal{Q} \times \mathcal{Q}'$ are Euclidean sub-posets (products of subsets of Euclidean spaces with component-wise order), so $\mathrm{dp}_a \otimes \mathrm{dp}'_a$ is an LCDP.

Intersection. Let $\mathrm{dp}_a, \mathrm{dp}_b \in \mathsf{LDP}\{\mathcal{P}, \mathcal{Q}\}$ be LCDPs with feasible sets

$$\mathrm{ldp}_a = \{\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \mid A_{\mathcal{P}} x_{\mathcal{P}} + A_{\mathcal{Q}} x_{\mathcal{Q}} \geq b_a\},$$
$$\mathrm{ldp}_b = \{\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \mid B_{\mathcal{P}} x_{\mathcal{P}} + B_{\mathcal{Q}} x_{\mathcal{Q}} \geq b_b\},$$

for suitable $\langle A_{\mathcal{P}}, A_{\mathcal{Q}}, b_a \rangle$ and $\langle B_{\mathcal{P}}, B_{\mathcal{Q}}, b_b \rangle$. By Definition 9, the intersection composition $\mathrm{dp}_a \wedge \mathrm{dp}_b \in \mathsf{DP}\{\mathcal{P}, \mathcal{Q}\}$ has feasible set

$$\begin{aligned} \mathrm{ldp}_{\text{int}} = \{\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \mid & A_{\mathcal{P}} x_{\mathcal{P}} + A_{\mathcal{Q}} x_{\mathcal{Q}} \geq b_a, \\ & B_{\mathcal{P}} x_{\mathcal{P}} + B_{\mathcal{Q}} x_{\mathcal{Q}} \geq b_b\}. \end{aligned} \tag{4}$$

The constraints in Eq. (4) can be written as a single system of linear inequalities by stacking the two descriptions:

$$\begin{bmatrix} A_{\mathcal{P}} & A_{\mathcal{Q}} \\ B_{\mathcal{P}} & B_{\mathcal{Q}} \end{bmatrix} \begin{bmatrix} x_{\mathcal{P}} \\ x_{\mathcal{Q}} \end{bmatrix} \geq \begin{bmatrix} b_a \\ b_b \end{bmatrix},$$

which defines a polyhedron directly in $\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle$. Since $\mathcal{P}$ and $\mathcal{Q}$ are Euclidean sub-posets, $\mathrm{dp}_a \wedge \mathrm{dp}_b$ is an LCDP.

Feedback. Let $\mathrm{dp} \in \mathsf{LDP}\{\mathcal{P} \times \mathcal{R}, \mathcal{Q} \times \mathcal{R}\}$ be an LCDP with feasible set

$$\begin{aligned} \mathrm{ldp} = \Big\{ \langle \langle x_{\mathcal{P}}, x_{\mathcal{R}}^{\text{in}} \rangle, \langle x_{\mathcal{Q}}, x_{\mathcal{R}}^{\text{out}} \rangle \rangle \;\Big| \\ A_{\mathcal{P}} x_{\mathcal{P}} + A_{\mathcal{Q}} x_{\mathcal{Q}} + A_{\mathcal{R}}^{\text{in}} x_{\mathcal{R}}^{\text{in}} + A_{\mathcal{R}}^{\text{out}} x_{\mathcal{R}}^{\text{out}} \geq b \Big\}. \end{aligned}$$

By Definition 9, $\mathrm{Tr}(\mathrm{dp}) \in \mathsf{DP}\{\mathcal{P}, \mathcal{Q}\}$ has feasible set

$$\mathrm{ldp}_{\text{fb}} = \{\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \mid \exists x_{\mathcal{R}} :\ \langle \langle x_{\mathcal{P}}, x_{\mathcal{R}} \rangle, (x_{\mathcal{Q}}, x_{\mathcal{R}}) \rangle \in \mathrm{ldp}\}. \tag{5}$$

Equivalently, $\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle \in \mathrm{ldp}_{\text{fb}}$ if and only if there exists $x_{\mathcal{R}}$ such that $(A_{\mathcal{R}}^{\text{in}} + A_{\mathcal{R}}^{\text{out}}) x_{\mathcal{R}} + A_{\mathcal{P}} x_{\mathcal{P}} + A_{\mathcal{Q}} x_{\mathcal{Q}} \geq b$. The set of triples $\langle x_{\mathcal{P}}, x_{\mathcal{Q}}, x_{\mathcal{R}} \rangle$ satisfying this inequality is a polyhedron in the extended space; ldp_{fb} is its projection onto $\langle x_{\mathcal{P}}, x_{\mathcal{Q}} \rangle$, hence a polyhedron. Since $\mathcal{P}$ and $\mathcal{Q}$ remain Euclidean sub-posets, $\mathrm{Tr}(\mathrm{dp})$ is an LCDP. □

We now show that every LCDP induces an LDP at the system level, so that its feasible set is a polyhedron and queries can be cast as MOLP.

Theorem 3 (An LCDP induces an LDP)**.** Let $\langle F, R, \langle \mathcal{V}, \mathcal{E} \rangle \rangle$ be an LCDP in the sense of Definition 17. The CDP obtained by composing the component DPs along the multigraph $\langle \mathcal{V}, \mathcal{E} \rangle$ is an LDP in $\mathsf{DP}\{F, R\}$. In particular, its feasible set $\mathrm{ldp}_{\text{tot}} \subseteq F \times R$ is a polyhedron.

Proof. A DP is constructed from a CDP by applying compositions in Definition 9 to the components according to the multi-graph $\langle \mathcal{V}, \mathcal{E} \rangle$, adding structural DPs of the form

$$\mathrm{dp}_{\mathcal{P}, \preceq, k} \overset{\text{def}}{=} \{\langle x_{\mathcal{P}}, \langle x'_{\mathcal{P},1}, \cdots, x'_{\mathcal{P},k} \rangle \rangle \mid x_{\mathcal{P}} \preceq x'_{\mathcal{P},i},\ \forall i\}$$

and

$$\mathrm{dp}_{\mathcal{P}, k, \preceq} \overset{\text{def}}{=} \{\langle \langle x_{\mathcal{P},1}, \cdots, x_{\mathcal{P},k} \rangle, x'_{\mathcal{P}} \rangle \mid x_{\mathcal{P},i} \preceq x'_{\mathcal{P}},\ \forall i\}$$

when necessary [3]. Since composing LDPs gives LDPs, we only need to prove these structural DPs are LDPs, which follows by their definitions. □

III. Feasible-Set Construction and Query Solving

Theorem 3 guarantees that any LCDP induces a system-level LDP and therefore supports the MOLP-based query semantics of Theorem 1. From a computational standpoint, two distinct tasks arise: (i) *query evaluation*, i.e., computing Pareto-minimal system resources (or Pareto-maximal functionalities) for fixed external ports, and (ii) *explicit feasible-set construction*, i.e., producing an **H**-representation of the induced system-level polyhedron $\mathrm{ldp}_{\text{tot}} \subseteq F \times R$.

Both tasks reduce to polyhedral calculus. An LCDP can be encoded as the Cartesian product of the component polyhedra, intersected with a sparse linear subspace enforcing all wiring equalities; the system-level feasible set is then obtained by a coordinate projection onto the external ports. Lemma 4 and Lemma 5 formalize this encoding and show that queries can be answered directly in the lifted space without explicit projection.

Lemma 4 (Global polyhedral encoding of an LCDP)**.** Consider an LCDP $\langle F, R, \langle \mathcal{V}, \mathcal{E} \rangle \rangle$ in the sense of Definition 17. For each node $d \in \mathcal{V}$, let $\mathrm{ldp}_d = \{z_d \in \mathbb{R}^{n_d} \mid M_d z_d \geq m_d\}$ be an **H**-representation of its component feasible set (including the port-domain constraints $z_d \in F_d \times R_d$). Let $z_{\text{global}} = [z_1^\top, \ldots, z_N^\top]^\top$ and define

$$M_{\text{blk}} \overset{\text{def}}{=} \text{block-diag}(M_1, \ldots, M_N),$$
$$m_{\text{blk}} \overset{\text{def}}{=} [m_1^\top, \ldots, m_N^\top]^\top.$$

Let $E z_{\text{global}} = 0$ be the collection of all wiring equalities induced by the edges $\mathcal{E}$ (each row of E has exactly one $+1$ and one -1). Then the *full-space* system-level feasible set of the LCDP is the polyhedron

$$\mathrm{ldp}_{\text{tot}}^{\text{full}} = \Big\{ z_{\text{global}} \Big| M_{\text{blk}} z_{\text{global}} \geq m_{\text{blk}}, E z_{\text{global}} = 0 \Big\}. \tag{6}$$

Remark. Writing inequalities in Fig. 1 as equalities in Lemma 4 produces the same system-level feasible set, since for any feasible z_{global} where one constraint $x_{\mathcal{P}} < x_{\mathcal{P}}$ is strict, assigning $\hat{x}_{\mathcal{P}} = \hat{x}_{\mathcal{P}} \stackrel{\text{def}}{=} x_{\mathcal{P}}$ results in another feasible global variable because every component is monotone. It also provides and requires the same functionality and resource at the system level.

If S_{ext} is a selection matrix extracting the external ports $z_{\text{ext}} = [x_F^\top, x_R^\top]^\top$ from z_{global}, then the induced *external-port* system-level feasible set is the projection

$$\text{ldp}_{\text{tot}} = \left\{ z_{\text{ext}} \;\middle|\; \exists z_{\text{global}} \in \text{ldp}_{\text{tot}}^{\text{full}} \text{ s.t. } z_{\text{ext}} = S_{\text{ext}} z_{\text{global}} \right\}. \tag{7}$$

Proof. By definition of an LCDP (Definition 17), a system design is feasible if and only if each component design z_d satisfies its local feasibility constraints and all interconnections identify the corresponding connected port coordinates by equality. Stacking the local $\mathbf{H}$-representations yields the block-diagonal system $M_{\text{blk}} z_{\text{global}} \geq m_{\text{blk}}$, while all port identifications yield the linear equalities $E z_{\text{global}} = 0$, giving (6). Finally, the system-level feasible set retains only the external ports, i.e., it is the image of the full-space set under the coordinate projection encoded by S_{ext}, which is exactly (7). □

Lemma 5 (Lifted MOLP queries equal projected DP queries)**.** Let ldp_{tot} and $\text{ldp}_{\text{tot}}^{\text{full}}$ be as in Lemma 4. Let S_F and S_R be selection matrices such that $x_F = S_F z_{\text{global}}$ and $x_R = S_R z_{\text{global}}$. Fix an external functionality $\bar{x}_F \in F$ and consider the two feasible resource sets

$$\mathcal{R}_{\text{ext}}(\bar{x}_F) \stackrel{\text{def}}{=} \{x_R \mid \langle \bar{x}_F, x_R \rangle \in \text{ldp}_{\text{tot}}\},$$
$$\mathcal{R}_{\text{lift}}(\bar{x}_F) \stackrel{\text{def}}{=} \{S_R z_{\text{global}} \mid z_{\text{global}} \in \text{ldp}_{\text{tot}}^{\text{full}},\ S_F z_{\text{global}} = \bar{x}_F\}.$$

Then $\mathcal{R}_{\text{ext}}(\bar{x}_F) = \mathcal{R}_{\text{lift}}(\bar{x}_F)$, and hence the Pareto-minimal (efficient) resources returned by the DP query coincide with the efficient outcomes of the lifted MOLP

$$\begin{aligned} &\operatorname{Min}_{\preceq}\ S_R z_{\text{global}} \\ &\text{s.t.}\quad M_{\text{blk}} z_{\text{global}} \geq m_{\text{blk}},\ E z_{\text{global}} = 0,\ S_F z_{\text{global}} = \bar{x}_F. \end{aligned} \tag{8}$$

An analogous statement holds for the "fix resources, maximize functionalities" query.

Proof. By Lemma 4, $\langle \bar{x}_F, x_R \rangle \in \text{ldp}_{\text{tot}}$ holds if and only if there exists a full assignment z_{global} such that $z_{\text{global}} \in \text{ldp}_{\text{tot}}^{\text{full}}$, $S_F z_{\text{global}} = \bar{x}_F$, and $S_R z_{\text{global}} = x_R$. This is exactly the equality $\mathcal{R}_{\text{ext}}(\bar{x}_F) = \mathcal{R}_{\text{lift}}(\bar{x}_F)$. Since both optimization problems compare resource vectors under the same componentwise order $\preceq$ and have identical attainable resource sets, their efficient outcomes coincide. The second query follows by the same argument with (S_F, S_R) exchanged and the order reversal accounted for as in Theorem 1. □

The monolithic strategy implements the lifted encoding of Lemma 4 by introducing *all* component port variables as decision variables, enforcing (i) each component's local feasibility constraints and (ii) all wiring equalities. This yields the full-space polyhedron $\text{ldp}_{\text{tot}}^{\text{full}}$ in (6), whose constraint matrix is *block-angular*: a block-diagonal part M_{blk} (one block per component) plus a very sparse coupling part E (each wiring

Algorithm 1 Monolithic feasible set for composite LCDP

Require: LCDP $\langle F, R, \langle \mathcal{V}, \mathcal{E} \rangle \rangle$; for each $d \in \mathcal{V}$ ($|\mathcal{V}| = N$), component LDP with $\text{ldp}_d = \{\langle x_{F,d}, x_{R,d} \rangle : A_{F,d} x_{F,d} + A_{R,d} x_{R,d} \geq b_d\}$

Ensure: System-level feasible set $\text{ldp}_{\text{tot}}^{\text{full}}$ in $\mathbf{H}$-representation $M_{\text{blk}} z_{\text{global}} \geq m_{\text{blk}}$ (full-space variables)

1: **Local formulation:** For each node $d \in \mathcal{V}$, set $z_d \leftarrow [x_{F,d}^\top,\ x_{R,d}^\top]^\top$, $M_d \leftarrow [A_{F,d}\ \ A_{R,d}]$, $m_d \leftarrow b_d$

2: **Aggregation:** Form $z_{\text{global}} \leftarrow [z_1^\top, \ldots, z_N^\top]^\top$, $M_{\text{blk}} \leftarrow \text{block-diag}(M_1, \ldots, M_N)$, $m_{\text{blk}} \leftarrow [m_1^\top, \ldots, m_N^\top]^\top$

3: System $M_{\text{blk}} z_{\text{global}} \geq m_{\text{blk}}$ is block-diagonal (one block per node)

4: **Edge constraints:** For each edge $e \in \mathcal{E}$ (series or feedback, Definition 17), the edge links one functionality and one resource. In z_{global} (with $z_d = [x_{F,d}^\top,\ x_{R,d}^\top]^\top$), find the indices i_{f}, i_{r} of these two entries. Append to M_{blk} one row with $+1$ at i_{f}, -1 at i_{r}, zeros elsewhere, and append 0 to m_{blk} (so $z_{i_{\text{f}}} - z_{i_{\text{r}}} \geq 0$; add a second row $-z_{i_{\text{f}}} + z_{i_{\text{r}}} \geq 0$ to enforce equality). Thus each edge adds one (or two) rows to the constraint matrix.

5: **return** $\text{ldp}_{\text{tot}}^{\text{full}} = \{z_{\text{global}} : M_{\text{blk}} z_{\text{global}} \geq m_{\text{blk}}\}$

row has exactly two nonzeros). By Lemma 5, no explicit projection onto external ports is required: the projected DP query is equivalent to solving the lifted MOLP (8) in the full variable space, avoiding polyhedral projection and letting the solver handle internal feasibility implicitly. Algorithm 1 details the construction.

IV. Case Study: Rigid Gripper Co-Design

This section validates the monolithic lifted formulation developed in Section III on a physically grounded multi-objective co-design problem. The gripper model is natively polyhedral after a suitable reparameterization, and therefore lies in the *exact* LCDP regime: the monolithic solver recovers the complete Pareto front with zero approximation error.

Unless stated otherwise, all reported runtimes are end-to-end wall-clock times for a *single* query instance, including model construction and solver calls but excluding plotting.[1] All polyhedral queries are solved with Bensolve [11]. As a baseline, we compare against MCDPL [1], [16], a general discrete solver for DPs.

Remark (Discretization in MCDPL)*.* For any DP dp whose antichains have infinite cardinality (e.g. a continuous relation such as $f_1 \leq r_1 + r_2$), MCDPL replaces the exact relation with a finite-cardinality approximation parameterized by a resolution $R \geq 1$. In this case study, we report only the optimistic antichain dp_R^L, which acts as an outer approximation of the exact relation. Using Van der Corput sampling, the optimistic approximation refines monotonically with R, i.e., $\text{dp}_{R+1}^L \preceq \text{dp}_R^L$, and approaches the exact relation as $R \to \infty$.

A. System and query

We study a tendon-driven rigid gripper with four conceptual design sub-problems: a motor (maps actuator investment

[1] All experiments were run on a macOS Tahoe 26.2 machine with an Apple M2 Pro CPU (10 cores, 3.2 GHz) and 16 GB of RAM.

$\langle m_{\mathrm{mot}}, c_{\mathrm{mot}} \rangle$ to achievable tendon force f_{grasp}), a material-selection module (given a mass and cost budget $\langle m_{\mathrm{f}}, c_{\mathrm{f}} \rangle$, determines achievable segment lengths $\langle L_{\mathrm{Al}}, L_{\mathrm{CF}} \rangle$), a gripper-geometry module (maps lengths $\langle L_{\mathrm{Al}}, L_{\mathrm{CF}} \rangle$ to structural workspace and grasping capacity), and an arm-cost aggregation module (maps workspace requirement f_{ws} to arm mounting cost c_{arm}). For a co-design diagram, refer to Fig. 2. The system-level functionality is $q = \langle f_{\mathrm{grasp}}, f_{\mathrm{ws}} \rangle$, where f_{grasp} is the required grasping force and f_{ws} is the required workspace. The system-level resources are $r = \langle r_{\mathrm{cost}}, r_{\mathrm{mass}} \rangle$, which are minimized in the Pareto sense. We use the query $f_{\mathrm{grasp}} \geq 100$ N, $f_{\mathrm{ws}} \geq 80$ mm.

B. LCDP model formulation

The gripper LCDP is parameterized by two continuous design variables, $\langle L_{\mathrm{Al}}, L_{\mathrm{CF}} \rangle \in \mathbb{R}^2_{\geq 0}$, representing the aluminum and carbon-fibre finger lengths. These variables are the main coupling medium across the architecture, as they jointly determine workspace, structural feasibility, finger mass, and material cost. The resulting model has $n_{\mathrm{f}} = 2$ functionality dimensions, $n_{\mathrm{r}} = 8$ resource dimensions, and seven linear inequalities grouped by subsystem.

a) Actuation (motor sub-problem): The available grasp force increases with both motor mass and motor cost. A heavier motor accommodates a larger rotor, while a costlier motor can achieve higher torque density. We capture these two effects with the supporting halfspace $f_{\mathrm{grasp}} \leq \alpha_{\mathrm{f}}\, m_{\mathrm{mot}} + \beta_{\mathrm{f}}\, c_{\mathrm{mot}}$, where $\alpha_{\mathrm{f}} = 400\,\mathrm{N/kg}$ and $\beta_{\mathrm{f}} = 0.6\,\mathrm{N/USD}$.

b) Kinematics and structure (finger sub-problem): Finger length determines both reach and structural feasibility. The usable workspace grows linearly with total finger length,

$$f_{\mathrm{ws}} \leq k_{\mathrm{w}}\big(L_{\mathrm{Al}} + L_{\mathrm{CF}}\big), \qquad k_{\mathrm{w}} = 0.35, \tag{9a}$$

while sustaining a grasping force requires a minimum total length,

$$L_{\mathrm{Al}} + L_{\mathrm{CF}} \geq \gamma_{\mathrm{L}}\, f_{\mathrm{grasp}}, \qquad \gamma_{\mathrm{L}} = 0.5\,\mathrm{mm/N}. \tag{9b}$$

Thus both functionality requirements couple to the same design pair $\langle L_{\mathrm{Al}}, L_{\mathrm{CF}} \rangle$.

c) Material physics (material sub-problem):

$$m_{\mathrm{f}} \geq \rho_{\mathrm{Al}}\, L_{\mathrm{Al}} + \rho_{\mathrm{CF}}\, L_{\mathrm{CF}}, \tag{9c}$$

$$c_{\mathrm{f}} \geq p_{\mathrm{Al}}\, L_{\mathrm{Al}} + p_{\mathrm{CF}}\, L_{\mathrm{CF}}. \tag{9d}$$

We use $\rho_{\mathrm{Al}} = 0.012\,\mathrm{kg/mm}$, $\rho_{\mathrm{CF}} = 0.004\,\mathrm{kg/mm}$, $p_{\mathrm{Al}} = 2.0\,\mathrm{USD/mm}$, and $p_{\mathrm{CF}} = 8.0\,\mathrm{USD/mm}$. Hence carbon fibre is lighter but more expensive per unit length, which is the main source of the Pareto trade-off.

d) System-level aggregation: The system-level objectives aggregate subsystem resources. Total mass is $r_{\mathrm{mass}} \geq m_{\mathrm{mot}} + m_{\mathrm{f}}$, and total cost combines motor cost, arm payload penalties, finger material cost, and a workspace-dependent term:

$$r_{\mathrm{cost}} \geq c_{\mathrm{mot}} + k_{\mathrm{AP}}\, m_{\mathrm{mot}} + c_{\mathrm{f}} + k_{\mathrm{AP}}\, m_{\mathrm{f}} + k_{\mathrm{AW}}\, f_{\mathrm{ws}} + c_{\mathrm{A0}}, \tag{9e}$$

with $k_{\mathrm{AP}} = 200\,\mathrm{USD/kg}$, $k_{\mathrm{AW}} = 50\,\mathrm{USD/mm}$, and $c_{\mathrm{A0}} = 300\,\mathrm{USD}$. Substituting (9c)–(9d) into (9e) yields effective per-length coefficients $p_{\mathrm{Al}} + k_{\mathrm{AP}}\rho_{\mathrm{Al}} = 4.4\,\mathrm{USD/mm}$, $p_{\mathrm{CF}} + k_{\mathrm{AP}}\rho_{\mathrm{CF}} = 8.8\,\mathrm{USD/mm}$, which compactly combine material price and arm load-bearing penalty. The density–cost asymmetry between aluminum and carbon fiber drives the Pareto trade-off between total mass and total cost.

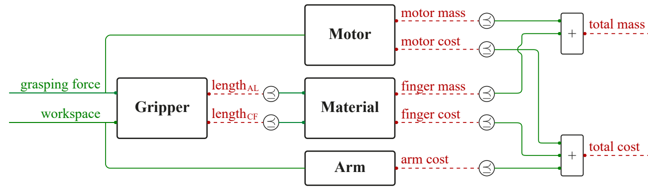


Fig. 2: Gripper co-design architecture. The query $\langle f_{\mathrm{grasp}}, f_{\mathrm{ws}} \rangle$ is propagated to subsystem design choices; the motor and material subsystems are coupled through the shared design variables $\langle L_{\mathrm{Al}}, L_{\mathrm{CF}} \rangle$; and contributions are aggregated into system objectives $\langle r_{\mathrm{cost}}, r_{\mathrm{mass}} \rangle$.

TABLE I: Exact Pareto vertices for the rigid gripper LCDP (monolithic solver). Query: $f_{\mathrm{grasp}} \geq 100\,\mathrm{N}$, $f_{\mathrm{ws}} \geq 80\,\mathrm{mm}$. V_1–V_3 are the only Pareto-optimal solutions; no approximation is introduced.

Vertex	Dominant material mix	r_{cost} (USD)	r_{mass} (kg)
V_1	Pure aluminum (min cost / max mass)	5 371.06	3.0696
V_2	Mixed Al / CF (balanced trade-off)	5 518.10	2.7429
V_3	Pure carbon fibre (min mass / max cost)	9 633.00	0.7677

C. Results

We compare: (i) the *monolithic exact* LCDP pipeline, which solves the induced bi-objective MOLP with BENSOLVE; and (ii) MCDPL optimistic approximations at resolutions $R \in \{5, 10, 20, 40\}$. Because the model is already polyhedral, this benchmark cleanly separates the exact LCDP regime from discretization-based approximation.

Table I reports the three exact Pareto vertices returned by the monolithic solver; Table II summarizes timing and approximation quality. The right panel of Fig. 3 makes the runtime–quality gap explicit in the time–quality plane.

The monolithic LCDP solve enumerates the exact frontier (3 vertices) in approximately 48 ms end-to-end. In contrast, MCDPL optimistic runs require approximately 9–15 s per resolution and produce increasingly dense discretized fronts (from 25 points at $R = 5$ to 944 points at $R = 85$). Approximation quality is measured by the normalised Inverted Generational Distance (IGD), computed against a densified reference front (200 points interpolated along the exact piecewise-linear Pareto curve) and normalised to $[0, 1]^2$ using the ideal–nadir range. The IGD decreases monotonically from 2.15×10^{-1} at $R = 5$ to 1.31×10^{-2} at $R = 85$, confirming theoretical convergence of the optimistic

TABLE II: Rigid gripper: monolithic exact solver vs. MCDPL optimistic approximations. IGD is the Inverted Generational Distance computed against the densified exact frontier (200 reference points), normalised to $[0, 1]^2$ using the ideal–nadir range.

Method	R	#Pts	Time (s)	IGD
Monolithic (exact)	—	3	4.8×10^{-2}	0
MCDPL opt.	5	25	9.59	2.15×10^{-1}
MCDPL opt.	10	84	9.03	9.09×10^{-2}
MCDPL opt.	20	207	9.54	4.16×10^{-2}
MCDPL opt.	40	430	10.09	2.26×10^{-2}
MCDPL opt.	60	642	11.69	1.64×10^{-2}
MCDPL opt.	85	944	14.96	1.31×10^{-2}

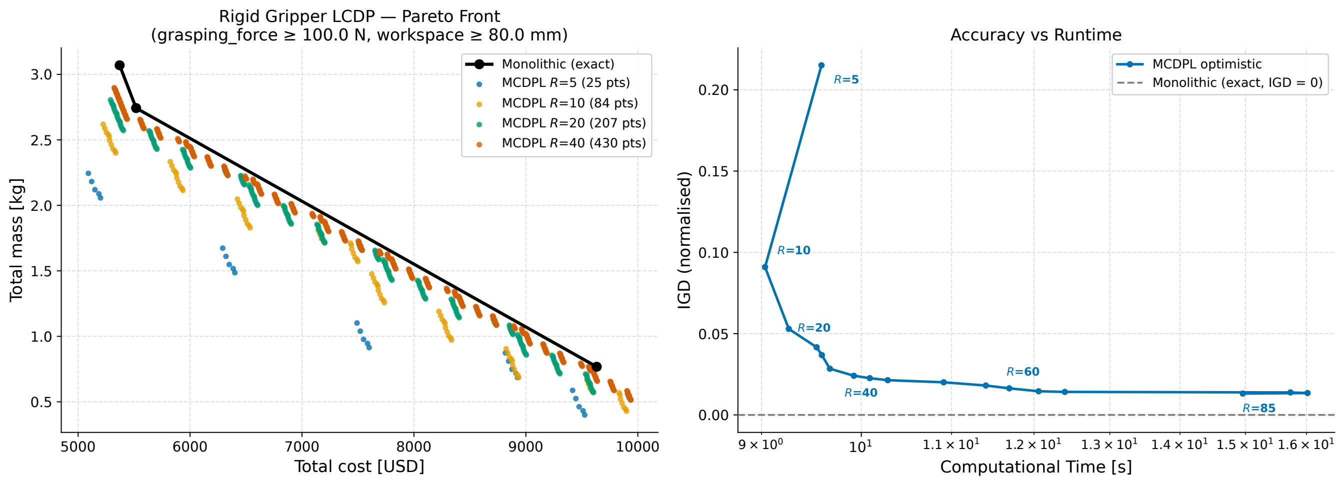


Fig. 3: Pareto-front comparison for the rigid gripper case study. (Left) Monolithic exact frontier (3 vertices) versus MCDPL optimistic approximations at resolutions $R \in \{5, 10, 20, 40\}$. (Right) IGD convergence versus computational time: the normalised Inverted Generational Distance of MCDPL optimistic solutions decreases monotonically with increasing resolution, while the exact monolithic solver achieves IGD $= 0$.

approximation. Even at $R = 85$, the exact monolithic pipeline remains about $300\times$ faster while delivering zero approximation error.

D. Discussion

The gripper case highlights the fully linear setting, where the induced co-design model remains exactly polyhedral. In this regime, the lifted monolithic pipeline exploits sparsity and solves the Pareto query directly, delivering exact fronts in milliseconds and substantially lower runtime than MCDPL. This indicates that, for a broad class of linear component models, exact system-level co-design can be used as a practical inner loop for fast architecture screening.

V. Conclusions and Future Work

We frame linear co-design as a polyhedral modeling problem: an LDP is represented by an upper feasible set in Euclidean posets, and this representation is preserved by series, parallel, intersection, and feedback interconnections (Theorem 2). This structural closure leads directly to a practical solver pipeline, where the lifted system in Algorithm 1 preserves block-angular sparsity and answers queries as MOLPs in the full space, without first computing projected feasible sets. In the rigid-gripper experiment, the exact pipeline removes discretization error and runs substantially faster than SOTA solver MCDPL. Promising next steps include polyhedral outer surrogates for convex nonlinear components, tighter coupling with uncertainty-aware co-design [17], [18], and Dantzig–Wolfe style decomposition to scale the same formulation to larger architectures.

References

[1] A. Censi, "A mathematical theory of co-design," *arXiv preprint arXiv:1512.08055*, 2015.

[2] A. Censi, J. Lorand, and G. Zardini, *Applied Compositional Thinking for Engineering*, 2024, work-in-progress book. [Online]. Available: https://bit.ly/3qQNrdR

[3] G. Zardini, "Co-design of complex systems: From autonomy to future mobility systems," Ph.D. dissertation, ETH Zurich, 2023.

[4] G. Zardini, A. Censi, and E. Frazzoli, "Co-design of autonomous systems: From hardware selection to control synthesis," in *2021 European Control Conference (ECC)*, 2021, pp. 682–689.

[5] G. Zardini, Z. Suter, A. Censi, and E. Frazzoli, "Task-driven modular co-design of vehicle control systems," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 2196–2203.

[6] D. Milojevic, G. Zardini, M. Elser, A. Censi, and E. Frazzoli, "Codei: Resource-efficient task-driven co-design of perception and decision making for mobile robots applied to autonomous vehicles," *IEEE Transactions on Robotics*, vol. 41, pp. 2727–2748, 2025.

[7] J. R. R. A. Martins and A. B. Lambe, "Multidisciplinary design optimization: A survey of architectures," *AIAA Journal*, vol. 51, no. 9, pp. 2049–2075, Sep. 2013.

[8] S. A. Seshia, S. Hu, W. Li, and Q. Zhu, "Design automation of cyber-physical systems: Challenges, advances, and opportunities," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 9, pp. 1421–1434, 2017.

[9] M. Ehrgott, *Multicriteria optimization*. Springer, 2005.

[10] A. Löhne, *Vector optimization with infimum and supremum*. Springer Science & Business Media, 2011.

[11] A. Löhne and B. Weißing, "The vector linear program solver *Bensolve* – notes on theoretical background," *European Journal of Operational Research*, vol. 260, no. 3, pp. 807–813, Aug. 2017.

[12] H. P. Benson, "An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem," *Journal of Global Optimization*, vol. 13, no. 1, pp. 1–24, 1998.

[13] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[14] K. Fukuda, "Lecture: Polyhedral computation, spring 2016," *Institute for Operations Research and Institute of Theoretical Computer Science, ETH Zurich. https://inf. ethz. ch/personal/fukudak/lect/pclect/notes2015/PolyComp2015. pdf*, 2016.

[15] A. Löhne and B. Weißing, "Equivalence between polyhedral projection, multiple objective linear programming and vector linear programming," *Mathematical Methods of Operations Research*, vol. 84, no. 2, pp. 411–426, Oct. 2016.

[16] A. Censi, "Uncertainty in monotone codesign problems," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1556–1563, 2017.

[17] Y. Huang, M. Furter, and G. Zardini, "On composable and parametric uncertainty in systems Co-Design," in *2025 IEEE 64th Conference on Decision and Control (CDC)*. IEEE, 2025, pp. 3049–3056.

[18] Y. Huang and G. Zardini, "Distributional Uncertainty and Adaptive Decision-Making in System Co-design," *arXiv preprint arXiv:2603.14047*, 2026.