Computers and Chemical Engineering 200 (2025) 109189

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/cace

A novel framework for flexibility assessment in design spaces defined by a set of affine bounds

Diogo A.C. Narciso [a,*], Steven Sachio [b,c], Maria M. Papathanasiou [b,c]

[a] Centro de Recursos Naturais e Ambiente, Department of Chemical Engineering, Instituto Superior Técnico, University of Lisbon, Av. Rovisco Pais 1, 1049-001, Lisboa, Portugal
[b] The Sargent Centre for Process Systems Engineering, Imperial College London, London, SW7 2AZ, United Kingdom
[c] Department of Chemical Engineering, Imperial College London, London, SW7 2AZ, United Kingdom

ARTICLE INFO

Keywords:
Flexibility
Process design
Multi-parametric programming

ABSTRACT

A novel framework for flexibility assessment in the context of system design is proposed. We deal with the case when the design space is bounded by a set of affine bounds defining a convex-hull. For this class of problems several flexibility metrics can be calculated, which are related to the minimum/maximum distances between any point in the design space and the less/most distant points at its bounds, respectively. In the first case, the distance functions are obtained via projection to individual bounds, and in the second case, the distance functions are defined via the Euclidean distance to the corners of convex hulls. These two sets of functions can then be used separately to calculate the minimum/maximum of the complete set of minimum/maximum distance functions over the full design space. This approach effectively enables the definition of four multi-parametric programming problems, and deliver four flexibility maps from their solutions. Flexibility maps based on the average of the two sets of distance functions are also delivered. This offers a plethora of complementary metrics for flexibility assessment, which extend beyond the classic approach based on the definition of feasible boxes. From the full set of solutions enabled by this framework, the minimum–minimum and maximum–maximum distance-based flexibility maps stand out as the extreme and most useful cases for flexibility assessment; the initial experimentation with these maps suggest that the average-minimum and particularly the average-maximum distance cases also provide useful information as an overall score on flexibility for any points within the design space.

1. Introduction

Process design is a challenging task that often requires conflicting objectives to be met. Both at a unit- and at a whole-process- operation level, engineers need to consider physical constraints and/or system specifications that will broadly define the feasible space within which design decisions can be made (Grossmann et al., 2014). In this context, process feasibility can be defined as the operation within a set of pre-defined constraints. Swaney and Grossmann (1985) recognised the need to further quantify the ability of a process to operate within the feasible region under uncertainty. This is known as the flexibility index problem and is an extension to the flexibility test problem presented by Halemane and Grossmann (1983). The mathematical formulation considers a set of equality (h) and inequality (g) constraints (Eqs. (1) and (2)), dependent of the process specifications:

$$h(d, z, s, \theta) = 0 \tag{1}$$

$$g(d, z, s, \theta) \leq 0 \tag{2}$$

Swaney and Grossmann (1985) define h as a vector of equations that describes the constraints of any given process at steady state. In this context, d corresponds to a vector of design variables, while z and s are the vectors of the control and state variables respectively; θ is the vector of uncertain parameters. This problem has been revisited and demonstrated in several works in the open literature. Grossmann and Floudas (1987) proposed an active set strategy that solves the flexibility index problem without assuming critical points to be at the vertices, employing mixed-integer optimisation. In 1990, Pistikopoulos and Mazzuchi presented a stochastic flexibility index problem to identify how the impact of multivariate uncertainty distribution can affect process flexibility. In a different approach, Kabatek and Swaney (1992) investigated how localised uncertainties can affect the overall

* Corresponding author.
E-mail addresses: diogo.narciso@tecnico.ulisboa.pt (D.A.C. Narciso), steven.sachio19@imperial.ac.uk (S. Sachio), maria.papathanasiou11@imperial.ac.uk (M.M. Papathanasiou).

https://doi.org/10.1016/j.compchemeng.2025.109189
Received 7 February 2025; Received in revised form 21 April 2025; Accepted 11 May 2025
Available online 21 May 2025
0098-1354/© 2025 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

process performance. Through an improved branch-and-bound algorithm they demonstrated how the worst-case scenarios can be identified and compensated for via suitable control actions. For the first time, in 1995, Dimitriadis and Pistikopoulos presented a flexibility analysis of dynamic systems. They achieved this by solving a dynamic optimisation problem for time-invariant uncertainty and/or by assuming a given uncertainty profile over time.

Different to the aforementioned approaches, there have been works that propose the identification of the full feasible space directly — known as design space identification (DSI). DSI methodologies can be broadly categorised into (1) nominal design centred, (2) geometrical envelope-based, (3) feasibility function-based, and (4) probabilistic. Table 1 shows an indicative list of various contributions from the literature. Nominal design centred DSI methods are based on solving the flexibility index problem to find the design with the largest flexibility index. Samsatli et al. (2001) proposed a reduced-vertex optimisation algorithm in which a hyper-rectangle is used to define the design space. On the other hand, Rooney and Biegler (2001) utilised likelihood ellipsoidal confidence regions to define the design space which is often less conservative when applied to non-convex problems. Applicable to non-convex problems, Banerjee and Ierapetritou (2005) proposed a geometrical envelope-based methodology via shape reconstruction, using α-shapes and the feasibility function as the objective. Diez et al. (2015) presented a geometrical approach combined with a dimensionality reduction to tackle high-dimensionality challenges in shape optimisation. Similarly, Sachio et al. (2023) developed three methodologies to construct the α-shape, which integrate prior process knowledge and data-driven interpolation to minimise the computational effort. Geometrical envelope-based methods provide an explicit mathematical definition of the design space boundary which allows for the calculation of a volumetric flexibility metric useful for process design. However, such methods can result computationally demanding especially at higher input dimensions.

Feasibility function-based DSI methods define the design space based on a binary classifier function which indicate the feasibility of a combination of input variables. To reduce the computational complexity that may be arising from the process model formulation, several contributions of feasibility function-based methods have been focused on data-driven approaches. Those include kriging-based surrogate modelling (Boukouvala and Ierapetritou, 2012; Boukouvala et al., 2010; Rogers and Ierapetritou, 2015; Ding and Ierapetritou, 2021), adaptive sampling techniques (Geremia et al., 2023) and Artificial Neural Networks (ANNs) (Metta et al., 2020). Others have also investigated probabilistic definitions of design spaces in an effort to propagate parameter uncertainty. Instead of a binary classifier function, probabilistic DSI methods outputs the probability of feasibility of a given input combination. Such approaches are often computationally expensive due to the need of carrying out exhaustive robust optimisation procedures (Close et al., 2014). Kusumo et al. (2020) proposed to lower this computational cost by implementing a Bayesian nested adaptive sampling approach, while Kucherenko et al. (2024) utilised surrogate meta modelling with rejection filtering.

It is evident that all the aforementioned works have created a strong foundation in the way design space identification and feasibility analysis problems can be approached. Nevertheless, increased process understanding and the ability to obtain tailored online measurements present novel challenges and needs. First, is the understanding that the boundaries defining the flexibility of a process are process-specific and a box-like approach may not be fully representative. Second, as we gain knowledge by operating the process repeatedly, we may wish to employ a more (or less) risk-taking control strategy. For this, one would need quantitative information of the flexibility metric not only with respect to how far is any given point from the boundary, but with respect to how process flexibility is changing as a function of the distance from the boundaries. To address these two questions, here we present a novel methodology that leverages multi-parametric programming (mp-P) to express the feasibility problem, in a mixed-integer quadratic form (mp-MIQP). To the best of our knowledge, this is the first time that design space identification problem is combined with flexibility analysis and expressed in an mp-MIQP formulation. Further to this, we introduce six different flexibility metrics to provide information on the sensitivity of the problem to each of the constraints as well as the dependency of the average process flexibility to the distance from the boundaries.

1.1. Brief overview of multi-parametric programming

Multi-parametric programming addresses optimisation problems involving uncertainties, by treating the uncertain parameters as independent from the optimisation variables (Narciso et al., 2022). This approach enables the explicit characterisation of solutions as piece-wise affine functions of the uncertain parameters, by dividing the parameter space into critical regions. Their solutions include a list of optimal active sets (of active constraints), and the matching critical regions and optimiser functions over the full space of uncertain parameters. Critical region maps can then be queried to obtain optimal solutions for any given vector of parameters without the need of solving any optimisation problem. Consider the typical multi-parametric quadratic programming (mp-QP) problem in Eq. (3).

$$\begin{aligned} \min_{x} \quad & 1/2x'Qx + q'x \\ s.t. \quad & Ax \le b + F\theta \\ & P_A\theta \le P_b \\ & x \in X \subset \mathbb{R}^n \\ & \theta \in \Theta \subset \mathbb{R}^p \end{aligned} \tag{3}$$

where x is the vector of n optimisation variables, while θ represents the vector of p uncertain parameters. Q is a positive definite matrix, ensuring convexity of the objective function, q is a constant vector and $'$ denotes a transpose vector or matrix. A, b and F are constant matrices and vectors defining the m bounds enclosing the feasible space X. P_A and P_b are a constant matrix and vector defining the bounds of the parameter space Θ, which effectively demands the solution of this optimisation problem for all $\theta \in \Theta$.

The solution of Eq. (3) is based on the definition of the Karush–Kuhn–Tucker (KKT) conditions for all active sets yielding optimal solutions in Θ. For any such active set, their optimiser functions are obtained directly from solving the corresponding KKT conditions as affine functions of the parameters. By enforcing a set of optimality and feasibility conditions on the optimiser functions, the matching critical region where the given active set yields the optimal solutions in Θ is obtained as a set of linear inequalities of the parameters. If required, the optimiser functions can be replaced directly on the mp-QP's objective function to obtain the corresponding optimal objective functions per active set. One of the most significant challenges in mp-P lies in efficiently identifying these regions and their associated optimiser functions without exhaustively enumerating all active sets (Narciso et al., 2022; Narciso and Pistikopoulos, 2025).

Multi-parametric programming has been applied to solve a diverse range of problems (Pistikopoulos and Tian, 2024; Pistikopoulos et al., 2015; Pappas et al., 2021). In explicit model predictive control (MPC), mp-QP has been used to derive control laws offline, enabling real-time decision-making without requiring online optimisation (Diangelakis et al., 2017; Papathanasiou et al., 2016; Pistikopoulos et al., 2015). Burnak and Pistikopoulos (2020) also developed an integrated process design, scheduling, and control framework by exploiting multi-parametric programming to obtain design dependent offline control and scheduling strategies. Notably, Bansal et al. (2002) presented a unified approach whereby flexibility analysis and design optimisation are approached simultaneously via multi-parametric programming. However, it uses hyper-rectangles to quantify the flexibility index which may not be representative for all types of problems. The methodology proposed in this work leverages multi-parametric mixed-integer quadratic programming (mp-MIQP) to integrate design space identification with six novel flexibility metrics.

Table 1
Indicative list of design space identification methodologies and contributions from literature.

Type of design space Identification (DSI) Technique	Contributions
Nominal Design Centred	Samsatli et al. (2001), Rooney and Biegler (2001), Georgakis et al. (2003), Harwood and Barton (2017), Pulsipher et al. (2019), Ochoa et al. (2021), Zhao et al. (2022b),
Geometrical Envelope-based	Ierapetritou (2001), Goyal and Ierapetritou (2003), Banerjee and Ierapetritou (2005), Bates et al. (2007), Diez et al. (2015), Adi et al. (2016), Zhao and Chen (2018), Kotidis et al. (2019), Zhao et al. (2022a), Sachio et al. (2023, 2024a,b)
Feasibility Function-based	Boukouvala et al. (2010), Boukouvala and Ierapetritou (2012), García-Muñoz et al. (2010), Chatzizacharia and Hatziavramidis (2014), Facco et al. (2015), Rogers and Ierapetritou (2015), Wang and Lakerveld (2017), Wang and Ierapetritou (2017), Bano et al. (2018), Ding and Ierapetritou (2021), Metta et al. (2020), Owoyele and Pal (2020), Geremia et al. (2023), Zhang et al. (2024),
Probabilistic	Close et al. (2013), Laky et al. (2022), Kusumo et al. (2020), Kucherenko et al. (2020), Hirono et al. (2022), Kucherenko et al. (2024)

2. Metrics for flexibility analysis

Consider the n-dimensional space defined via a set of m affine bounds in Eq. (4).

$$Ax \leq b \tag{4}$$

where, consistently with Eq. (3), $x \in \mathbb{R}^n$ is a vector of variables, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ are a constant matrix and vector, respectively. In the context of system design, x is in fact the vector of design variables, and Eq. (4) defines the complete set of feasible design specifications: the feasible space, abbreviated for convenience as X. It is assumed that X defines a (i) non-empty, (ii) closed convex hull in $\mathbb{R}^n$; the first assumption is assessed via a check in Section 3.1 and the second is revisited at the end of Sections 4 and 5.

In this work we refer to a given design in X as θ, which takes the role of the uncertain parameter vector in the framework developed, since flexibility metrics must be computed for all $\theta \in X$. Index i is used to refer to each of the n coordinates of x or θ, and index j to refer to each of the m bounds of X via $Ax \leq b$; A_j and b_j denote the jth row vector of A and the jth coordinate of b, respectively, whereas $A_{j,i}$ corresponds to the element of A at its jth row and ith column.

2.1. Minimum and maximum design flexibilities

In the context of system design, flexibility is closely associated with the computation of distances between a given design θ and the bounds of the feasible space. Firstly, we examine two extreme cases concerning flexibility, as follows:

- *Minimum design flexibility:* the minimum distance between design $\theta \in X$ and the bounds of X at a feasible point $x^{min-flex}$.
- *Maximum design flexibility:* the maximum distance between design $\theta \in X$ and the bounds of X at a feasible point $x^{max-flex}$.

The first metric above represents the worst-case scenario concerning flexibility: disturbances on design θ along the direction of $x^{min-flex}$ are those nearest to the bounds of the feasible space; in principle, this direction represents the highest risk of selecting an infeasible design, and thus the distance between θ and $x^{min-flex}$ is a clear indication of the minimum design flexibility for θ. Similarly, disturbances along the direction of $x^{max-flex}$ represent the lowest risk of delivering an infeasible design, and the distance between θ and $x^{max-flex}$ represents the maximum design flexibility for θ. A schematic 2D motivation for both metrics is depicted in Fig. 1 (A) and (B).

From geometrical intuition, the minimum distance between $\theta \in X$ and the bounds of X is obtained for the smallest n-dimensional sphere with centre in θ such that its surface includes at least one point of the bounds of the feasible space. This definition is self evident: by drawing a set of circumferences with centre in θ with increasingly larger radius as illustrated in Fig. 1 (A), the minimum distance to the bounds is found when the selected radius effectively delivers the first interception between the corresponding circumference and the bounds of X: smaller radiuses do not deliver any interception (not relevant to compute the minimum distance), while larger radiuses deliver interception points which are necessarily more distant than the reference case. Note that, the first instance when a radius delivers an interception with the bounds of X does not necessarily have to include a single point, but may actually include multiple points equally distant to θ as illustrated in Fig. 1 (C).

Following a similar approach, the maximum distance between $\theta \in X$ and the bounds of the feasible space is obtained for the largest n-dimensional sphere with centre in θ such that its surface includes at least one point of the bounds of the feasible space. In this case, by drawing a set of circumferences with centre in θ with increasingly smaller radius as illustrated in Fig. 1 (B), the maximum distance to the bounds is found when the selected radius effectively delivers the first interception between the corresponding circumference and the bounds of the feasible space. Multiple equally distant points can also be found in the maximum distance case as illustrated earlier for minimum distance calculation.

2.2. Flexibility assessment goals and supporting metrics

A full flexibility assessment for a given design θ would consist of a detailed map of all the distances between $\theta \in X$ and all points at the bounds of X. Clearly, for all continuous (infinite) set of directions around θ, the distance to the bounds of X are bounded between the minimum and maximum distances as defined above. Obtaining such a detailed map presents several challenges, but perhaps more importantly, this level of information is likely not required for flexibility analysis: as discussed in Section 1, authors have focused on providing a single representative flexibility metric, which tend to align more closely with the minimum design flexibility case — generally the most critical information from a design perspective.

In this work, we provide a set of flexibility metrics, extending beyond the minimum and maximum design flexibilities. We show that the minimum distances to the m single bounds, and the distances to the full set of corners of X are particular cases of the full set of

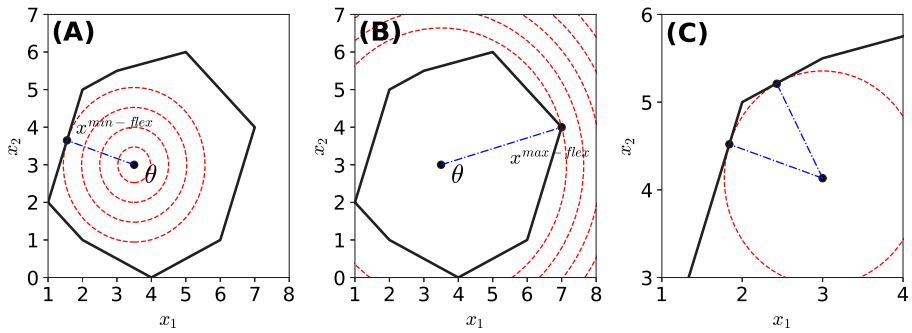


Fig. 1. A 2D geometrical motivation for design flexibilities in convex hulls. (A) minimum design flexibility; (B) maximum design flexibility. (C) multiple points for minimum design flexibility.

$2^m - 1$ active sets of Eq. (4) including at least one active bound of the feasible space. The first case enforces a single bound active at the nearest point to θ located at the bounds of the feasible space, and is closely associated with the minimum design flexibility (Section 4). The second case enforces n simultaneously active bounds defining a unique corner of X and is directly related to the maximum design flexibility (Section 5). A set of flexibility metrics is defined for these two extreme cases, from the solutions of the active sets enumerated above. We refer to the full set of solutions in X of a given flexibility metric as the corresponding flexibility map. Collectively, these maps deliver a comprehensive framework for flexibility analysis on design spaces formulated via $Ax \leq b$.

On a conceptual level, we remark that minimum design flexibility is closely related to the utilisation of back-off methods in optimal design under uncertainty (Rafiei-Shishavan et al., 2017; Koller et al., 2018). Both approaches capture the risk of process operation at the bounds of the design space, where any perturbation can lead to the violation of one or more process constraints. On one hand, back-off methods aim to deliver robust design by moving the selected point towards the interior of the design space; on the other hand, the maps proposed in this work provide quantitative flexibility information for all points in the design space to assess the need of a similar design adjustment.

3. Feasible space: pre-processing

Three initial pre-processing steps deliver key information on the feasible space, which is then used in the calculation of flexibility maps in Sections 4 and 5.

3.1. Empty feasible space check

To check if X defines a non-empty space in $\mathbb{R}^n$, the following Linear Programming (LP) problem is solved:

$$\begin{aligned} \min_{x} \quad & \begin{bmatrix} 1 & \dots & 1 \end{bmatrix} x \\ s.t. \quad & Ax \leq b \\ & x \in \mathbb{R}^n \end{aligned} \tag{5}$$

where a unitary cost vector of size n is used in the definition of the objective function. Since the purpose of this LP is to exclusively test feasibility, any vector of size n can be selected in its place. The proposed check is summarised below:

Algorithm 1: Empty feasible space check.

Step 1: Define boolean ϕ to denote the empty status of X and solve Equation (5).
Step 2: If no feasible solution is returned, set ϕ = True; else set ϕ = False. Terminate.

3.2. Redundant bounds of the feasible space

To check if the jth bound of the feasible space ($j = 1, \dots, m$) is redundant with respect to the remaining bounds of X, the following LP problem is solved:

$$\begin{aligned} \min_{x} \quad & \begin{bmatrix} 1 & \dots & 1 \end{bmatrix} x \\ s.t. \quad & Ax \leq b \\ & A_j x = b_j \\ & x \in \mathbb{R}^n \end{aligned} \tag{6}$$

Note that additionally to the m bounds defining the feasible space, an equality constraint adapted from its jth bound is also added to Eq. (6). The results obtained from this LP are then used to draw a conclusion on the redundancy of the m bounds of X as follows:

Algorithm 2: Redundancy check on the m bounds of the feasible space.

Step 1: Initialise a boolean vector ψ of size m with all entries set to False to denote the redundancy status of the m bounds of the feasible space. Set $o = 1$.
Step 2: Solve Equation (6) with $j = o$.
Step 3: If a feasible solution is returned, go to Step 4 (non redundant bound); else, set ψ_j = True (redundant bound).
Step 4: If $o < m$, set $o = o + 1$ and go to Step 2; else, terminate.

A feasible/infeasible solution for Eq. (6) indicates the jth bound is active/not active on at least a single point of X, thus providing a clear indication on its redundancy on the definition of X. The total number of non redundant bounds is denoted as m' and the significance of ψ is covered in greater detail in Sections 3.3 and 4.4.

3.3. Corners of the feasible space

A corner is defined as a unique point in the feasible space where n linearly independent bounds of $Ax \leq b$ are simultaneously active. Corners play an important role in delivering the maximum design flexibility, thus, making the calculation of their coordinates in X a critical initial step. Algorithm 3 delivers the full list of corners in the feasible space and their norms:

Algorithm 3: Calculation of the coordinates of all corners of the feasible space.

Step 1: Compute all combinations of n bounds, from the m' non redundant bounds of the feasible space, as an integer index matrix S (including $C_n^{m'}$ rows and n columns).
Step 2: Initialise output matrix C with 0 rows and n columns to store all corners' coordinates.
Step 3: Set $o = 1$.
Step 4: Set the current candidate combination of bounds to be tested from the indices of S at its row in position o - denoted as S_o; perform the following steps in sequence:
  Step 4.1: Define a matrix comprising only the n rows of A identified in S_o.
  Step 4.2: Calculate the rank (r) of the matrix in Step 4.1, and: if $r = n$, go to step 4.3; else, set $o = o + 1$ and go to Step 5.
  Step 4.3: Solve the system of equations $Ax = b$, including only the set of rows identified in S_o, to obtain a point $c \in \mathbb{R}^n$.
  Step 4.4: If $Ac \leq b$, $c \in X$: add row vector c to matrix C; else, go to step 4.5.
  Step 4.5: Set $o = o + 1$ and go to Step 5.
Step 5: If $o < C_n^{m'}$, go to Step 4; else, go to Step 6.
Step 6: Remove any duplicate corners' coordinates from C, calculate the norms of all corners - $\|C\|$ - and terminate.

All candidate combinations of n bounds are computed implicitly in Step 1 via a set of indices, and two checks are then performed for all candidates such that: (i) all combinations where $r < n$ (infeasible or high-dimensional subspace, Step 4.2), and (ii) all solutions not included in X (Step 4.4) are excluded from the set of corners of X. Algorithm 3 returns the full list of unique corners' coordinates as a constant matrix $C \in \mathbb{R}^{t \times n}$ and the corresponding vector of (row-wise) norms of C defined as $\|C\|$; t denotes the total number of corners. Index k is used in this work to refer to the kth vector of corners' coordinates C_k, or the kth norm $\|C_k\|$; $C_{k,i}$ denotes the ith coordinate of the kth corner in matrix C.

Algorithm 3 was developed to reduce as much as possible unnecessary computational burden. However, for large m' and n the total number of corners may increase very significantly. Results on solution complexity, namely the calculation of corners, is covered in greater detail in Section 6.3.3 via a set of 10 examples ranging from 1D to 10D.

4. Flexibility analysis via the minimum distances to the individual bounds of the feasible space

4.1. Partial minimum design flexibilities

Partial minimum flexibilities, as defined in this work, capture the minimum distances between a design θ and each of the m bounds of the feasible space individually. As discussed in Section 2, these are closely associated, but not limited, to the calculation of the minimum design flexibility. Explicit functions for these distances are derived in a general form in this section using geometric arguments. To this end, firstly we define $x^{proj,j}$ as the unique point where the minimum distance between θ and the jth bound of X is found (for any $j = 1, \dots, m$). This solution must be active at $A_j x = b_j$ and is obtained from the projection of θ in the normal direction of the jth bound using vector A_j. Mathematically:

$$\begin{aligned} x^{proj,j} - A_j l^j &= \theta \\ -A_j x^{proj,j} &= -b_j \end{aligned} \tag{7}$$

where l^j is a multiplicative factor enabling the projection of θ along the normal vector A_j. Note that formally, $\theta + A_j l^j = x^{proj,j}$ represents the projection of θ to the jth bound of the feasible space; $x^{proj,j} - A_j l j = \theta$ is an equivalent statement, such that vector $-A_j$ points towards the interior of the feasible space, and ensuring positive values for l^j are obtained in this case. The theoretical consistency of Eq. (7) is covered in greater detail in Appendix A. Eq. (7) may be written equivalently in a detailed or compact matrix notation as shown in Eqs. (8) and (9), respectively.

$$\left[\begin{array}{ccccc|c} 1 & 0 & 0 & \dots & 0 & -A_{j,1} \\ 0 & 1 & 0 & \dots & 0 & -A_{j,2} \\ 0 & 0 & 1 & \dots & 0 & -A_{j,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -A_{j,n} \\ \hline -A_{j,1} & -A_{j,2} & -A_{j,3} & \dots & -A_{j,n} & 0 \end{array}\right] \left[\begin{array}{c} x_1^{proj,j} \\ x_2^{proj,j} \\ x_3^{proj,j} \\ \vdots \\ x_n^{proj,j} \\ \hline l^j \end{array}\right] = \left[\begin{array}{c} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_n \\ \hline -b_j \end{array}\right] \tag{8}$$

$$\left[\begin{array}{c|c} I & -A_j \\ \hline -A_j' & 0 \end{array}\right] \left[\begin{array}{c} x^{proj,j} \\ \hline l^j \end{array}\right] = \left[\begin{array}{c} \theta \\ \hline -b_j \end{array}\right] \tag{9}$$

where I is the identity matrix of size n and the full matrix in the left hand side of Eq. (9) denoted as $M^{proj,j}$. The explicit solution for $x^{proj,j}$ and l^j in Eqs. (7)–(9) is given below:

$$\left[\begin{array}{c} x^{proj,j} \\ \hline l^j \end{array}\right] = \left[\begin{array}{c|c} I & -A_j \\ \hline -A_j' & 0 \end{array}\right]^{-1} \left[\begin{array}{c} \theta \\ \hline -b_j \end{array}\right] \tag{10}$$

Consider a non-null vector A_j, which is represented in simplified form in Eqs. (11) and (12) as $[a_1, a_2, \dots, a_n]$ to facilitate their reading; the inverse matrix in Eq. (10) - $(M^{proj,j})^{-1}$ - may be obtained numerically, or computed explicitly, as a function of these simplified coefficients as shown in Eq. (11) (for any n). (See Box I).

This result was first derived analytically for problems up to $n = 4$, and then generalised to any n. To confirm the matrices in Eqs. (8) and (11) are inverse, it suffices to compute their matrix product: (see Box II) which yields the identity matrix of size $n+1$, and thus confirms this important result.

Note that matrix $(M^{proj,j})^{-1}$ does not need to be calculated; instead, the distance between θ and $x^{proj,j}$, is obtained from l^j: this variable is defined as a multiplicative factor for the projection vector $-A_j$ in Eq. (7), from where it follows that the distance between the two points is obtained as a product between l^j and the norm of vector A_j: $\sqrt{\sum_{i=1}^{n} a_i^2} = \|A_j\|$. Combining this information with Eqs. (10) and (11), an explicit minimum distance function - d_j^{min} - between a design θ and the jth bound of the feasible space is derived:

$$d_j^{min} = \frac{1}{\|A_j\|} \left[\begin{array}{c|c} -A_j & b_j \end{array}\right] \left[\begin{array}{c} \theta \\ \hline 1 \end{array}\right], j = 1, \dots, m \tag{13}$$

The full set of m partial minimum flexibilities - d^{min} - are listed in matrix form in Eq. (14).

$$d^{min} = \left[\begin{array}{c} d_1^{min} \\ d_2^{min} \\ \vdots \\ d_m^{min} \end{array}\right] = \left[\begin{array}{ccc|c} \frac{-A_{1,1}}{\|A_1\|} & \frac{-A_{1,2}}{\|A_1\|} & \dots & \frac{-A_{1,n}}{\|A_1\|} & \frac{b_1}{\|A_1\|} \\ \frac{-A_{2,1}}{\|A_2\|} & \frac{-A_{2,2}}{\|A_2\|} & \dots & \frac{-A_{2,n}}{\|A_2\|} & \frac{b_2}{\|A_2\|} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{-A_{m,1}}{\|A_m\|} & \frac{-A_{m,2}}{\|A_m\|} & \dots & \frac{-A_{m,m}}{\|A_m\|} & \frac{b_m}{\|A_m\|} \end{array}\right] \left[\begin{array}{c} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \\ \hline 1 \end{array}\right] \tag{14}$$

Note that following the convention highlighted above, any points in the halfspace defined by the jth bound return a non-negative distance via Eqs. (13) or (14), while any points in the opposite halfspace return a non-positive distance. The derivation of a minimum distance function in Eq. (13) is the most important piece of information in this section, since this relates directly to flexibility analysis. If required, explicit optimiser functions for the projected point at the bounds - $x^{proj,j}$ - may also be recorded directly from the first n rows of Eqs. (10) and (11). The m distance functions in Eq. (14) enable the definition of a set of additional minimum-distance flexibility related metrics which are covered in Sections 4.2 and 4.3.

$$(M^{proj,j})^{-1} = \frac{1}{\sum_{i=1}^{n} a_i^2} \left[\begin{array}{ccccc|c} a_2^2+a_3^2+\cdots+a_n^2 & -a_2a_1 & -a_3a_1 & \cdots & -a_na_1 & -a_1 \\ -a_1a_2 & a_1^2+a_3^2+\cdots+a_n^2 & -a_3a_2 & \cdots & -a_na_2 & -a_2 \\ -a_1a_3 & -a_2a_3 & a_1^2+a_2^2+\cdots+a_n^2 & \cdots & -a_na_3 & -a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_1a_n & -a_2a_n & -a_3a_n & \cdots & a_1^2+a_2^2+\cdots+a_{n-1}^2 & -a_n \\ \hline -a_1 & -a_2 & -a_3 & \cdots & -a_n & -1 \end{array}\right] \tag{11}$$

Box I.

$$M^{proj,j}(M^{proj,j})^{-1} = \frac{1}{\sum_{i=1}^{n} a_i^2} \left[\begin{array}{ccccc|c} \sum_{i=1}^{n} a_i^2 & -a_2a_1+a_2a_1 & -a_3a_1+a_3a_1 & \cdots & -a_na_1+a_na_1 & a_1-a_1 \\ -a_1a_2+a_1a_2 & \sum_{i=1}^{n} a_i^2 & -a_3a_2+a_3a_2 & \cdots & -a_na_2+a_na_2 & a_2-a_2 \\ -a_1a_3+a_1a_3 & -a_2a_3+a_2a_3 & \sum_{i=1}^{n} a_i^2 & \cdots & -a_na_3+a_na_3 & a_3-a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_1a_n+a_1a_n & -a_2a_n+a_2a_n & -a_3a_n+a_3a_n & \cdots & \sum_{i=1}^{n} a_i^2 & a_n-a_n \\ \hline (a_1-a_1)\sum_{i=1}^{n} a_i^2 & (a_2-a_2)\sum_{i=1}^{n} a_i^2 & (a_3-a_3)\sum_{i=1}^{n} a_i^2 & \cdots & (a_n-a_n)\sum_{i=1}^{n} a_i^2 & \sum_{i=1}^{n} a_i^2 \end{array}\right] \tag{12}$$

Box II.

4.2. Minimum minimum-distance based design flexibility

We define flexibility metric $d^{min-min}$ as the minimum distance between a given θ and the set of m minimum distance functions in Eq. (14). Mathematically:

$$d^{min-min}(\theta) = \min\{d_1^{min}(\theta), d_2^{min}(\theta), \ldots, d_m^{min}(\theta)\} \tag{15}$$

In this section we consider the calculation of the flexibility map for $d^{min-min}$, which must deliver explicit functions for this metric for all $\theta \in X$.

4.2.1. Problem formulation and solution

Formally, the flexibility map for $d^{min-min}$ is obtained from the solution of the following optimisation problem:

$$\begin{aligned} \min_{x,y} \quad & \sum_{i=1}^{n} (\theta_i - x_i)^2 \\ s.t. \quad & Ax - b \le 0 \\ & Ax - b \ge \alpha y \\ & \sum_{i=1}^{n} y_i = m - \beta \\ & A\theta - b \le 0 \\ & x \in X \subset \mathbb{R}^n \\ & \theta \in \Theta \subset \mathbb{R}^n \\ & y \in \{0,1\}^m \end{aligned} \tag{16}$$

where α is a large negative number, y is a binary vector of size m and β is a positive integer lower or equal to n - set to 1 in this particular case. The objective function is the square of the (minimum) distance between θ and an optimal x at the bounds of the feasible space. The first constraint enforces the feasible space for x as in Eq. (4). The second constraint in Eq. (16) either enforces a given bound of the feasible space to include x when the corresponding entry for y is set to 0 (e.g. $A_j x \le b_j$ and $A_j x \ge b_j$ is equivalent to $A_j x = b_j$, or becomes redundant when y is set to 1 (right hand side multiplied by a very large negative number α). Since only the single nearest bound to θ must be allowed to be active, the third constraint in Eq. (16) achieves this (only a single entry of y is set to 0, using $\beta = 1$). At last, this problem is to be solved parametrically, that is not only for a given θ, but for all θ in the parameters space (Θ), which in this case matches the feasible space (X) and is enforced by the last constraint.

Since the objective function in Eq. (16) is quadratic, and all constraints are linear with respect to the continuous and binary variables, this problem fits in the class of multi-parametric mixed integer quadratic programming (mp-MIQP) problems. To the best of the authors' knowledge there are no algorithms available in the literature to compute the solution of Eq. (16). The proposed rationale to deliver the flexibility map for $d^{min-min}$ results, instead, directly from its definition in Eq. (15). Since this problem fits in the class of mp-P problems, four elements may be optionally considered in the calculation of their solutions as covered in Section 1.1:

- Optimal active sets
- Optimiser functions
- Critical regions
- Optimal objective functions

In this case, only m active sets are relevant: one for each of the m bounds of the feasible space, representing the candidate nearest bounds to a given θ. Furthermore, the calculation of the optimiser functions and optimal objective functions (partial minimum distances) per each of the m bounds was already covered in Section 4.1; critical regions' calculation is the remaining element of the full solution of Eq. (16) for which a solution strategy must still be developed.

The rationale in multi-parametric programming concerning critical regions' calculation is based on two key types of constraints: (i) optimality constraints, which bound the set of points in Θ where a given active set outperforms all other active sets (delivers the optimal values for the objective function), and (ii) feasibility constraints, which ensure the optimiser functions return feasible solutions for all points in critical regions. The same general principles are also applicable to this class of problems: in this case, and for the j_1^{th} single bound active set

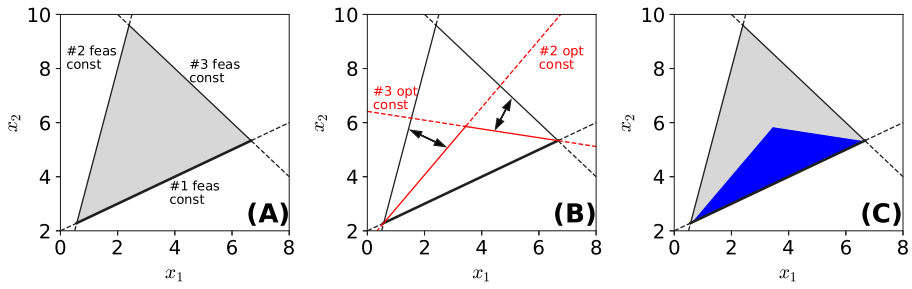


Fig. 2. Illustration of critical region construction for $d^{min-min}$. (A) the feasible space is the grey convex hull, where a critical region for the active set including the first bound – highlighted – is to be calculated. (B) the optimality constraints for this active set from distance comparison are shown, which are more stringent than the corresponding feasibility constraints. (C) the critical region is bounded by the two optimality constraints and the corresponding bound of X as the single feasibility constraint.

$(j_1 = 1, \dots, m)$, a total of $m-1$ optimality constraints are required for critical region definition, via the direct comparison of the associated distance function with the remaining $m-1$ distance functions:

$$d_{j_1}^{min} \leq d_{j_2}^{min}, \quad \begin{matrix} j_2 = 1, \dots, m \\ j_2 \neq j_1 \end{matrix} \tag{17}$$

Combining the individual distance functions in Eq. (14) with Eq. (17), a set of $m-1$ affine halfspaces for critical region construction per active set are obtained using simple matrix operations. Since only the set of parameters included in Θ are relevant for critical region construction, their bounds must also be considered; not all m bounds of X are strictly necessary, and in fact only its j_1^{th} bound must be added as a feasibility constraint, since the $m-1$ optimality constraints are necessarily more stringent than the corresponding feasibility constraints (the latter become redundant). These principles are illustrated in Fig. 2, where the construction of a critical region in 2D is depicted.

As a final remark to this section, we note that a methodology to obtain the solution to this problem from the KKT conditions is also possible. The KKT conditions for the active set including only the jth bound of the feasible space is defined as follows:

$$\left[\begin{array}{cccc|c} 2 & 0 & \dots & 0 & -A_{j,1} \\ 0 & 2 & \dots & 0 & -A_{j,2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 2 & -A_{j,n} \\ \hline -A_{j,1} & -A_{j,2} & \dots & -A_{j,n} & 0 \end{array}\right] \left[\begin{array}{c} x_1^{opt,j} \\ x_2^{opt,j} \\ \vdots \\ x_n^{opt,j} \\ \hline \lambda_j \end{array}\right] = \left[\begin{array}{c} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \\ \hline -b_j \end{array}\right] \tag{18}$$

where $x^{opt,j}$ denotes the optimiser and λ_j the Lagrange multiplier. It is trivial to show that Eq. (18) is fully equivalent to Eq. (8), with $\lambda_j = 2l$, from where the optimiser functions for $x^{opt,j}$ and $x^{proj,j}$ from both routes are also the same. While the utilisation of the KKT conditions is the more conventional route in the context of mp-P, we have opted to use Eq. (8) from which the distance functions for flexibility calculations can be more straightforwardly calculated.

4.2.2. Minimum design flexibility

We consider again Fig. 1 (A). Observe that the circumference depicted is tangent to a single point on (at least) a single bound, from where the minimum design flexibility can then be calculated. Since the bounds of the feasible space consist of a set of affine inequalities of x, the minimum distance between any design θ and the bounds of the feasible space always conforms to the scenario where a single bound is active at its nearest point(s); any active set including two or more bounds defines a more constrained subspace, and necessarily delivers a more distant nearest point to θ than to their bounds considered individually. Note that this does not mean that the same bound always delivers the minimum distance for all points in the feasible space, but for all designs, the minimum distance is always obtained by one (or more) of the m partial minimum flexibilities presented in Section 4.1; an illustration of this result is presented in Fig. 3, and a formal discussion on this subject can be found in Murty (2009).

This is a key result for minimum design flexibility, since it proves that it suffices to assess the m bounds of the feasible space to calculate it for any design θ. This avoids working with the $2^m - 1$ active sets where at least one of the m bounds is active, which significantly reduces the methodology's complexity. Finally, we conclude that the minimum minimum-distance based design flexibility - $d^{min-min}$ - covered in Section 4.2.1 conforms to the minimum design flexibility condition as presented in Section 2.1. Its flexibility maps comprise the full set of critical regions in X and the associated partial minimum design flexibilities as discussed above.

4.3. Average and maximum minimum-distance based design flexibilities

We now consider the calculation of flexibility maps delivering the average distance of the set of m minimum distance functions - $d^{avg-min}$. This map is obtained directly from Eq. (14), as follows:

$$d^{avg-min} = \frac{1}{m} \left[\begin{array}{c} -\frac{A_{1,1}}{\|A_1\|} - \frac{A_{2,1}}{\|A_2\|} - \dots - \frac{A_{m,1}}{\|A_m\|} \\ -\frac{A_{1,2}}{\|A_1\|} - \frac{A_{2,2}}{\|A_2\|} - \dots - \frac{A_{m,2}}{\|A_m\|} \\ \vdots \\ -\frac{A_{1,n}}{\|A_1\|} - \frac{A_{2,n}}{\|A_2\|} - \frac{A_{m,m}}{\|A_m\|} \\ \hline \frac{b_1}{\|A_1\|} + \frac{b_2}{\|A_2\|} + \dots + \frac{b_m}{\|A_m\|} \end{array}\right]' \left[\begin{array}{c} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \\ \hline 1 \end{array}\right] \tag{19}$$

where the row vector including the column-wise average of the coefficients in Eq. (14) is transposed for presentation convenience in Eq. (19). Note that not all projections on the m bounds for a design θ necessarily yield a feasible point; rather, this distance must be interpreted as the average of partial flexibilities, and thus providing a more global indicator of flexibility with respect to the edges of the feasible space taken individually.

The maximum of the m minimum distances – $d^{max-min}$ – concerns the calculation of the projections most distant to the bounds of the feasible space for a given θ in X. Obtaining flexibility maps for this metric results also from solving a mp-P problem, which can be defined via a simple adaptation of the guidelines presented in Section 4.2.1, namely by solving for the maximum in Eqs. (15) or (16), and reversing the inequality sign in Eq. (17) (finding the most distant projection). The usefulness of this metric and the corresponding flexibility maps, however, is questionable since $d^{max-min}$ may correspond in some cases to projections far beyond the bounds of the feasible space. Its calculation may be enabled via the Python code developed for this project to confirm this assessment; given its limited scope for flexibility assessment, it will not be considered further in this work.

4.4. Algorithm

Starting from the results delivered in Algorithms 1 and 2 (ϕ and ψ), all steps leading to the calculation of the partial minimum design

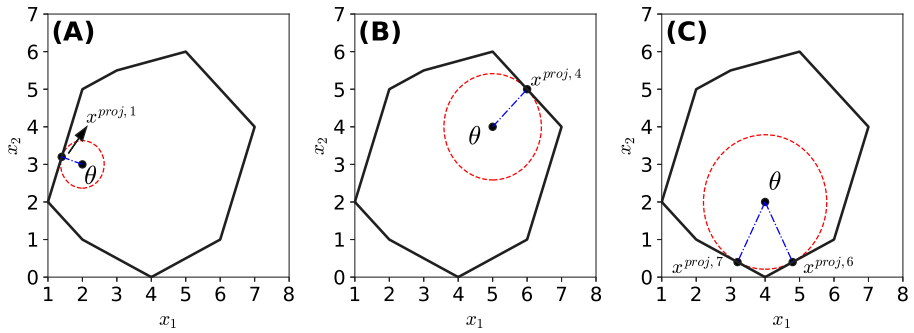


Fig. 3. Examples of the minimum distance to the bounds of the feasible space. Minimum distance at the: (A) 1st bound. (B) 4th bound. (C) 6th and 7th bounds.

flexibilities - d^{min} - and the flexibility maps for $d^{min-min}$ and $d^{avg-min}$ are summarised below:

Algorithm 4: Calculation of minimum distance-based flexibility maps.

Step 1: If ϕ = True, terminate.[(a)]

Step 2: Compute d^{min} using Equation (14).[(b)] Set $j = 1$.

Step 3: For the j^{th} single-constraint active set in the map for $d^{min-min}$, perform the following steps in sequence:

Step 3.1: If ψ_j = True, go to Step 3.2[(c)]; else, calculate the critical region for the j^{th} single-constraint active set defined as $CR_j^A x \leq CR_j^b$ [(d)], using: (i) the $m-1$ bounds listed in Equation (17) (optimality), and (ii) $A_j x \leq b_j$ (single feasibility constraint).

Step 3.2: If $j < m$, set $j = j + 1$ and go to Step 3; else, go to Step 4.

Step 4: Calculate $d^{avg-min}$ from Equation (19) and terminate.

[(a)] X is empty, and flexibility calculations are not applicable.

[(b)] The corresponding optimiser functions may also be calculated via Equations (10) and (11) (optional).

[(c)] The j^{th} bound is redundant in X, and d_j^{min} is never optimal (empty critical region).

[(d)] Where CR_j^A and CR_j^b are a constant matrix and vector of matching sizes, respectively, denoting the critical region for the j^{th} single-constraint active set.

For a more compact representation and efficient online identification of critical regions, it is convenient to identify which bounds in the m critical regions of the flexibility map for $d^{min-min}$ are redundant. This can be achieved via the adaptation of the methodology presented in Section 3.2, which for brevity is not discussed in this paper, but fully implemented on the code developed.

We remark that relaxing the (closed) convex-hull assumption made in Section 2 such that X becomes a partially unbounded subspace of $\mathbb{R}^n$ has no impact on the calculation of the minimum-distance based flexibility maps: all these maps are based on the calculation of the minimum distance to a set of m bounds, which are not affected in any way by this relaxation.

5. Flexibility analysis via the maximum distances to the corners of the feasible space

5.1. Partial maximum design flexibilities

In the same way as the m individual bounds of the feasible space are closely related with the minimum design flexibility, so are its t corners related with the maximum design flexibility. To deliver a set of maximum distance-based metrics, the partial maximum design flexibilities are defined in this work as the distance functions between a design θ and all corners of the feasible space; for the kth corner of the feasible space ($k = 1, \ldots, t$) these distance functions - d_k^{max} - are computed trivially, as follows:

$$d_k^{max} = \sqrt{\sum_{i=1}^{n} \left(\theta_i - C_{k,i}\right)^2} \tag{20}$$

where C is the matrix of corners' coordinates calculated via Algorithm 3. Eq. (20) may be written equivalently as:

$$(d_k^{max})^2 = \theta'\theta - 2(C_k)'\theta + \|C_k\|^2 \tag{21}$$

where both sides of Eq. (21) are squared for convenience. The complete set of distance functions between design θ and the t corners of X are presented in Eq. (22):

$$(d^{max})^2 = \begin{bmatrix} (d_1^{max})^2 \\ (d_2^{max})^2 \\ \vdots \\ (d_t^{max})^2 \end{bmatrix} = \theta'\theta \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} - 2 \begin{bmatrix} C_{1,1} & C_{1,2} & \cdots & C_{1,n} \\ C_{2,1} & C_{2,2} & \cdots & C_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{t,1} & C_{t,2} & \cdots & C_{t,n} \end{bmatrix} \times \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} + \begin{bmatrix} \|C_1\|^2 \\ \|C_2\|^2 \\ \vdots \\ \|C_t\|^2 \end{bmatrix} \tag{22}$$

5.2. Maximum maximum distance-based design flexibility

We define flexibility metric $d^{max-max}$ as the maximum distance between a given θ and the set of t maximum distance functions in Eq. (22):

$$d^{max-max}(\theta) = \max\{d_1^{max}(\theta), d_2^{max}(\theta), \ldots, d_t^{max}(\theta)\} \tag{23}$$

In this section we consider the calculation of the flexibility map for $d^{max-max}$, which must deliver explicit functions for this metric for all $\theta \in X$.

5.2.1. Problem formulation and solution

Taking the mp-P problem in Eq. (16) as reference, it suffices to: (i) update the goal of the optimisation problem from minimisation to maximisation, and (ii) set $\beta = n$ (enforces n bounds active at the optimal solution) to obtain a formal mathematical statement. The solution of this problem includes: (i) the optimiser functions – equal to corners' coordinates (Section 3.3), (ii) the optimal objective functions – the distance functions presented in Eq. (22), and (iii) critical regions, which must now be computed following the guidelines presented in Section 4.2.1. In this case, taking corner C_{k_1} as the reference active set ($k_1 = 1, \ldots, t$), the necessary optimality conditions for delivering its critical region are defined in Eq. (24):

$$d_{k_1}^{max} \geq d_{k_2}^{max}, \quad \begin{matrix} k_2 = 1, \ldots, t; \\ k_2 \neq k_1 \end{matrix} \tag{24}$$

These constraints deliver a critical region where the distance to the k_1^{th} corner is the maximum of all partial maximum distances. Combining Eqs. (21) and (24) the following inequality is obtained (comparison of corners at positions k_1 and k_2):

$$\theta'\theta - 2(C_{k_1})'\theta + \|C_{k_1}\|^2 \geq \theta'\theta - 2(C_{k_2})'\theta + \|C_{k_2}\|^2 \tag{25}$$

where the quadratic terms ($\theta'\theta$) on both sides of the inequality cancel each other, yielding optimality conditions in a simplified form, and where the linear dependence with θ is restored:

$$2\left(C_{k_1} - C_{k_2}\right)'\theta \leq \left(\|C_{k_1}\|^2 - \|C_{k_2}\|^2\right) \tag{26}$$

To complete the definition of critical regions, and for all corners (active sets), all bounds of the feasible space are also added as feasibility constraints.

5.2.2. Maximum design flexibility

Following the same reasoning as presented in Section 4.2.2, we first build on the geometrical analysis presented in Fig. 1 (B): observe that the smallest circumference intercepts a single point at n bounds of the feasible space, defining a corner. Since the bounds of the feasible space are defined via a set of affine inequalities of x, the maximum distance between any design θ and the bounds of the feasible space always conforms to the scenario where n bounds are active at its furthest point(s); any active set including a subset of $n-1$ or less bounds of the original n selected (non redundant) bounds, defines a less constrained subspace and includes at least a less distant point to θ than the corresponding corner.

This is a key result for maximum design flexibility, since it proves that it suffices to assess the t corners of the feasible space to calculate it for any design θ. This avoids working with the $2^m - 1$ active sets where at least one of the m bounds is active, which significantly reduces the methodology's complexity. Finally, we conclude that the maximum maximum-distance based design flexibility – $d^{max-max}$ – covered in Section 5.2.1 conforms to the maximum design flexibility condition as presented in Section 2.1. Accordingly, its flexibility maps comprise the full set of critical regions in X and the associated partial maximum design flexibilities as discussed above.

5.3. Average and minimum maximum-distance based design flexibilities

The flexibility map for the average distance of the set of t maximum distance functions - $d^{avg\text{-}max}$ - is calculated directly from Eq. (22) to form Eq. (27) as follows:

$$(d^{avg,max})^2 = \theta'\theta - \frac{2}{t}\begin{bmatrix} C_{1,1}+C_{2,1}+\cdots+C_{t,1} \\ C_{1,2}+C_{2,2}+\cdots+C_{t,2} \\ \vdots \\ C_{1,n}+C_{2,n}+\cdots+C_{t,n} \end{bmatrix}'\begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} + \frac{1}{t}\left(\|C_1\|^2 + \|C_2\|^2 + \cdots + \|C_t\|^2\right) \tag{27}$$

where the vector obtained from the column-wise averages of the matrix in Eq. (22) is shown in transpose format for convenience. This average distance takes a total of t individual distances to the bounds of X (at their corners), thus representing a very useful global flexibility indicator.

The minimum of the t maximum distances - $d^{min-max}$ - captures the minimum distance between a given θ and the nearest of the t corners of the feasible space for a given θ in X. Flexibility maps for this metric result from the definition of a mp-P problem, and these solutions may be obtained via a simple adaptation of the guidelines presented in Section 5.2.1, namely by solving for the minimum in Eq. (23), and reversing the inequality sign in Eq. (24) (finding the nearest corner). The usefulness of such metric, however, is also questionable since $d^{min-max}$ refers to the local distances to the nearest corner: this is neither a minimum or a maximum flexibility. Its meaning is not particularly insightful, but mostly academic and not considered further in this work. The calculation of flexibility maps for $d^{min-max}$ may be optionally enabled in the code developed to confirm this assessment.

5.4. Algorithm

Starting from the results delivered in Algorithms 1 and 3 (ϕ, C and $\|C\|$), all steps leading to the calculation of the partial maximum design flexibilities - d^{max} - and the flexibility maps for $d^{max-max}$ and $d^{avg-max}$ are summarised below:

Algorithm 5: Calculation of maximum distance-based flexibility maps.

Step 1: If ϕ = True, terminate.[(a)]
Step 2: Compute d^{max} using Equation (21). Set $k = 1$.
Step 3: For the k^{th} active set (corner) in the map for $d^{max-max}$, perform the following steps in sequence:
  Step 3.1: Calculate the critical region for the k^{th} active set as $CR_k^A x \leq CR_k^b$, using the: (i) the $t-1$ bounds listed in Equation (24) (optimality), and (ii) $Ax \leq b$ (feasibility constraints).
  Step 3.2: If $k < t$, set $k = k+1$ and go to Step 3; else, go to Step 4.
Step 4: Calculate $d^{avg,max}$ from Equation (27) and terminate.

(a) X is empty, and flexibility calculations are not applicable.

Relaxing the convex hull assumption made in Section 2 has a more significant impact on the set of maximum-distance based metrics presented in this section. When X extends to infinity, the partial maximum design flexibilities no longer remain viable candidates in the calculation of maximum distances: in this case, the maximum distance is unbounded and the metrics and flexibility maps presented in this section are not applicable.

6. Results

6.1. Dodecagon

We consider firstly the 2D design space bounded by 12 constraints as defined in Eq. (28). This low-dimension problem enables a detailed illustration of the application of Algorithms 1–5, including the display of the critical regions for $d^{min-min}$ and $d^{max-max}$, and the flexibility maps for $d^{min-min}$, $d^{avg-min}$, $d^{max-max}$ and $d^{avg-max}$.

$$\begin{bmatrix} -1 & 0 \\ -3/2 & 1 \\ -5/4 & 1 \\ -3/4 & 1 \\ -1/2 & 1 \\ 0 & 1 \\ 1 & 0 \\ 3/2 & -1 \\ 5/4 & -1 \\ 3/4 & -1 \\ 1/2 & -1 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 2 \\ 5/2 \\ 9/2 \\ 6 \\ 10 \\ 10 \\ 7 \\ 5 \\ 2 \\ 1 \\ 0 \end{bmatrix} \tag{28}$$

6.1.1. Feasible space processing

Firstly, we solve Eq. (5) (Algorithm 1), from where ϕ = True: Eq. (28) defines a non-empty space. Then, Eq. (6) is solved for all bounds (Algorithm 2), where ψ_j = False for $j = 1, \ldots, 12$: all bounds are non redundant. All corners of the feasible space are calculated (Algorithm 3) via the $C_2^{12} = 66$ combinations of two bounds; after processing all candidates, a total of 12 corners of the feasible space and their norms

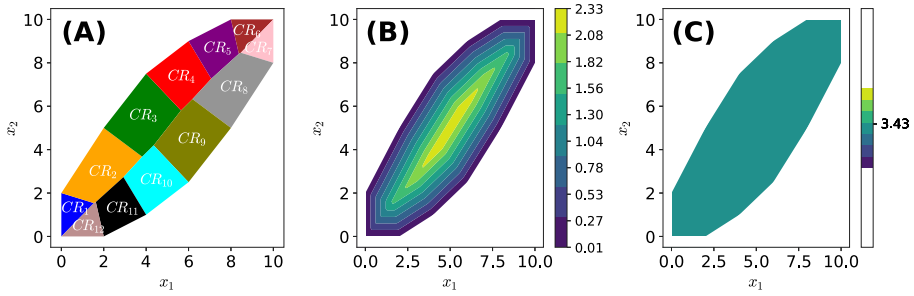


Fig. 4. Minimum-based distance results for dodecagon example. (A) critical regions for $d^{min-min}$, (B) flexibility map for $d^{min-min}$, (C) flexibility map for $d^{avg-min}$. In (B) and (C) a colorbar is presented to capture the contours of $d^{min-min}$ and $d^{avg-min}$, respectively, in the feasible space.

are delivered as the outputs (Eq. (29)).

$$C = \begin{bmatrix} 4.000 & 1.000 \\ 2.000 & 0.000 \\ 0.000 & 0.000 \\ 0.000 & 2.000 \\ 2.000 & 5.000 \\ 4.000 & 7.500 \\ 6.000 & 9.000 \\ 8.000 & 10.000 \\ 10.000 & 10.000 \\ 10.000 & 8.000 \\ 8.000 & 5.000 \\ 6.000 & 2.500 \end{bmatrix} \quad \|C\| = \begin{bmatrix} 4.123 \\ 2.000 \\ 0.000 \\ 2.000 \\ 5.385 \\ 8.500 \\ 10.816 \\ 12.806 \\ 14.142 \\ 12.806 \\ 9.433 \\ 6.5 \end{bmatrix} \tag{29}$$

6.1.2. Minimum-based distances

Next, we illustrate the calculation of distance functions d^{min} and $d^{avg-min}$, and the critical region map for $d^{min-min}$ via Algorithm 4. Since ϕ = False (Step 1), the calculation of these metrics is applicable; in Step 2, d^{min} is calculated from Eq. (14):

$$\begin{bmatrix} d_1^{min} \\ d_2^{min} \\ d_3^{min} \\ d_4^{min} \\ d_5^{min} \\ d_6^{min} \\ d_7^{min} \\ d_8^{min} \\ d_9^{min} \\ d_{10}^{min} \\ d_{11}^{min} \\ d_{12}^{min} \\ \hline d^{avg-min} \end{bmatrix} = \left[\begin{array}{ccc} 1.000 & 0.000 & 0.000 \\ 0.832 & -0.554 & 1.109 \\ 0.780 & -0.624 & 1.561 \\ 0.600 & -0.800 & 3.600 \\ 0.447 & -0.894 & 5.366 \\ 0.000 & -1.000 & 10.000 \\ -1.000 & 0.000 & 10.000 \\ -0.832 & 0.554 & 3.882 \\ -0.780 & 0.624 & 3.123 \\ -0.600 & 0.800 & 1.600 \\ -0.447 & 0.894 & 0.894 \\ 0.000 & 1.000 & 0.000 \\ \hline 0.000 & 0.000 & 3.428 \end{array}\right] \begin{bmatrix} \theta_1 \\ \theta_2 \\ \hline 1 \end{bmatrix} \tag{30}$$

Step 2, optionally enables the calculation of the optimiser functions (projections) for each of the m bounds, which in this case is omitted for brevity. In Step 3, the critical regions for $d^{min-min}$ are obtained: the distance functions for each of the 12 active sets (single bounds) are compared against the remaining 11 distance functions using Eq. (17), from which a total of 12 matrices (CR^A) and 12 vectors (CR^b) are obtained as the critical regions for the corresponding 12 active sets. This information is depicted graphically in Fig. 4 (A). In Step 4, the flexibility map for $d^{avg-min}$ is calculated via Eq. (14) as shown in the last row of Eq. (30).

Formally, the flexibility map for $d^{min-min}$ consists of the 12 critical regions and the matching distance functions in Eq. (30). In the 2D case, it is possible to deliver this result as a heatmap for $d^{min-min}$ as shown in Fig. 4 (B). This map was created via a grid of points in the 2D space for which the corresponding distances were calculated. This requires firstly the identification of the relevant critical regions for each point in the grid, based on which the corresponding distance functions are used to calculate $d^{min-min}$. A similar approach is employed for the flexibility map of $d^{avg-min}$ (Fig. 4 (C)), which in this case is obtained directly from the last row of Eq. (30).

Observe that all critical regions in Fig. 4 (A) extend from the corresponding bounds of the feasible space, where $d^{min-min} = 0$, and are limited by two or more optimality constraints. These constraints define the internal bounds within the feasible space where the distance functions for the pairs of neighbouring critical regions match. For this reason, and even though the full set of distances is captured individually by these 12 critical regions, the flexibility map in Fig. 4 (B) exhibits a continuous gradient-like shape, confirming that $d^{min-min}$ is 0 at the bounds, and increasing progressively towards the centre where a maximum value for $d^{min-min}$ is found. These results clearly confirm the relevance and usefulness of this map for flexibility assessment.

The flexibility map for $d^{avg-min}$ is insensitive to θ, which can be confirmed both via the last row of Eq. (30) or Fig. 4 (C). As pointed in Section 4.3, this metric relies on the average of all projections to individual bounds, some of which may be beyond the bounds of the feasible space as in this case, and thus not fully aligned with classic flexibility definition. Our initial experimentation suggests that this metric/flexibility map is more sensitive to the number and proximity of bounds, and thus could serve as an indication of the overall flexibility towards all bounds. Additional results are required to confirm and expand on this assessment.

6.1.3. Maximum-based distances

Finally, the calculation of distance functions d^{max} and $d^{avg-max}$, and the critical regions map for $d^{max-max}$ via Algorithm 5 is presented. Since ϕ = False (Step 1), the calculation of these metrics is applicable. In Step 2, d^{max} is calculated from Eq. (22): the first vector in this equation is pre-calculated (all ones in the 12 coordinates), the second matrix is C and the last row vector is $\|C\|^2$ (Eq. (29) from Algorithm 3). In Step 3, the critical regions for $d^{max-max}$ are obtained: the distance functions for each of the 12 active sets (corners) are compared against the remaining 11 partial maximum design flexibilities using Eq. (24), to obtain all critical regions' bounds via a set of matrices CR^A and vectors CR^b. This result is depicted graphically in Fig. 5 (A). In Step 4, the flexibility map for $d^{avg-max}$ is calculated via Eq. (22) and presented in Eq. (31). The flexibility maps for $d^{max-max}$ and $d^{avg-max}$ are depicted as heatmaps in Fig. 5 (B) and (C) respectively, following the guidelines presented in Section 6.1.2.

$$(d^{avg-max})^2 = \theta'\theta + \left[\begin{array}{cc|c} -10.00 & -10.00 & 75.21 \end{array}\right] \begin{bmatrix} \theta_1 \\ \theta_2 \\ \hline 1 \end{bmatrix} \tag{31}$$

Observe that the critical regions' map in Fig. 5 (A) includes only two non-empty regions. CR_3, for instance, is associated with the third corner listed in C, at $\theta = [0, 0]$. To facilitate the interpretation of this map, note that corners are highlighted with the same colour as the corresponding critical region. A graphical analysis confirms that for all θ in CR_3, the corner at $[0, 0]$ is the most distant point (highest flexibility), thus confirming that the intended functionality is delivered

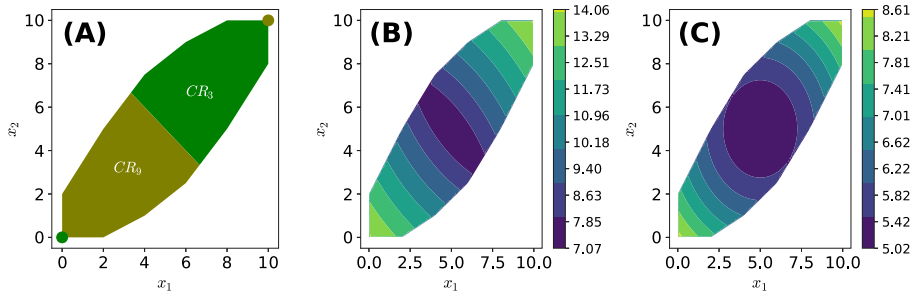


Fig. 5. Maximum-based distance results for dodecagon example. (A) critical regions for $d^{max-max}$, (B) flexibility map for $d^{max-max}$, (C) flexibility map for $d^{avg-max}$. In (B) and (C) a colorbar is presented to capture the contours of $d^{max-max}$ and $d^{avg-max}$, respectively, in the feasible space.

as expected. An equivalent analysis can be made for the corner at $\theta = [10, 10]$ and CR_9.

The flexibility map for $d^{max-max}$ is formally defined as the collection of critical regions and their matching distance functions; in this particular case, the corresponding heatmap shown in Fig. 5 (B) provides useful insights into this map: consistently with the analysis of critical regions presented above, note for instance that $d^{max-max}$ in CR_3 defines a set of contour circles with centre in $\theta = [0, 0]$, where the same distance to this corner is found. Note also that the flexibility map exhibits a clear gradient-like shape centred in both corners. The flexibility map is fully continuous across the areas delimited by CR_3 and CR_9, and also at the boundary line between these two critical regions (at $x_2 = 10 - x_1$), where the two distance functions match.

Fig. 5 (C) also provides valuable insights for flexibility analysis. The flexibility map for $d^{avg-max}$ is dependent on the quadratic term $\theta'\theta$, thus defining a set of circles of constant distance around a centre point. This centre point is implicitly defined via the average of all corners' coordinates as shown in Eqs. (22) or (31). Since $d^{avg-max}$ represents the average distance to the full set of corners effectively included at the bounds of the feasible space, it bears the quality of a global flexibility metric, where corners act as representatives of the full set of points at their bounds.

Two important remarks on the flexibility map for $d^{avg-max}$ are in order: firstly, the occurrence of corners depends solely on $Ax \leq b$, and thus it is not possible to ensure a priori that the set of corners define a representative network of points: some regions of the feasible space's bounds may include a higher density of corners than others, which will naturally impact these flexibility maps; secondly, this set of representative points are directly associated with the partial maximum design flexibilities, and thus, and to some extent, this global flexibility metric is biased towards the maximum distance, while $d^{avg-max}$ is still an average-based calculation.

6.2. CSTR design

In this Section, a practical illustration of the developed framework for flexibility assessment in the design of a Continuous Stirred Tank Reactor (CSTR) is considered as shown in Fig. 6.

In this process, product A is fed to the CSTR and is converted irreversably to product B (A → B). F^{in} and F^{out} are the inlet and outlet volumetric flowrates (m^3/h) and C_A^{in} and C_A^{out} are the inlet and outlet molar concentrations of A (mol/m^3). V denotes the reactor volume (m^3). The reaction rate is given by $r_A = -kC_A^{out}$, where k is a first-order kinetic factor (h^{-1}). The mass balance for A and the global mass balance for the CSTR at steady-state are presented in Eqs. (32), and (33), respectively:

$$F^{in}C_A^{in} = F^{out}C_A^{out} + VkC_A^{out} \tag{32}$$

$$F^{in} = F^{out} \tag{33}$$

For illustration purposes it is assumed that $C_A^{in} = 1 \text{mol/m}^3$ and $k = 1\ \text{h}^{-1}$. The design variables for this problem are F^{in} and V and

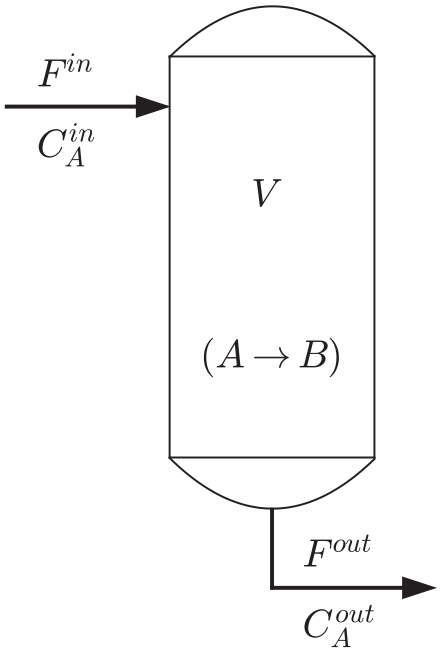


Fig. 6. CSTR process.

the design space defined as: $2 \leq F^{in} \leq 20$, $10 \leq V \leq 100$. Additionally, three process design constraints are defined for this problem as follows:

First constraint: reactant conversion ≥ 80%

Reactant conversion for this system is given by:

$$Conversion = \frac{F^{in}C_A^{in} - F^{out}C_A^{out}}{F^{in}C_A^{in}} = \frac{C_A^{in} - C_A^{out}}{C_A^{in}} \tag{34}$$

from where a minimum of 80% on conversion translates to $C_A^{out} \leq 0.2\ \text{mol/m}^3$. By combining Eqs. (32) and (33), C_A^{out} is calculated explicitly as a function of the two design variables:

$$C_A^{out} = \frac{F^{in}C_A^{in}}{F^{in} + Vk} \tag{35}$$

After replacing process parameters and simplifying Eq. (35), the minimum conversion requirement is expressed equivalently as a function of the design variables:

$$0.8F^{in} - 0.2V \leq 0 \tag{36}$$

Second constraint: annualised cost ≤ 40 M€

Assuming a feed cost of 10€/m^3 and an annualised cost of reactor capacity per unit volume of 0.4M€/m^3, the total cost for 1 year of operation (8000 h) is expressed as:

$$Cost(€) = F^{in}(\text{m}^3/\text{h}) * 10(€/\text{m}^3) * 8000(\text{h}) + V(\text{m}^3) + 400,000(€/\text{m}^3) \tag{37}$$

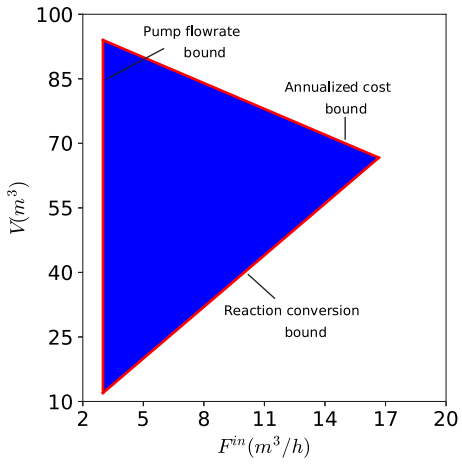


Fig. 7. Design space for CSTR example, highlighting the three design constraints and feasible and infeasible combinations of design variables (blue and white, respectively).

The cost constraint is then obtained and simplified from Eq. (37) as:

$$2F^{in} + V \leq 100 \tag{38}$$

Third constraint: pump flowrate ≥ 3 m^3/h

This design constraint is posed simply as:

$$F^{in} \geq 3 \tag{39}$$

The impact of these three process constraints on the design of this system is shown in Fig. 7, where multiple design specifications are considered in the window of interest and then labelled as feasible or infeasible.

The corners of this design space and all distance functions are calculated as illustrated in Section 6.1. For brevity, only the critical regions and corresponding flexibility maps for the minimum and maximum-based distance cases are presented in Fig. 8.

The flexibility map for $d^{min-min}$ in Fig. 8 (B) shows that the minimum distance to the bounds of the design space is 0 at the process constraints, and increases towards its centre, exhibiting a pyramid-like shape. From a design perspective, the region around the pyramid's apex exhibits the most flexibility, meaning that if F^{in} or V shift from the selected design, a feasible design is still achieved. For instance, if F^{in} is selected close to the minimum pump flowrate of 3 m^3/h, any minor decrease in flowrate results in an operation violating the defined bounds.

The flexibility map for $d^{max-max}$ in Fig. 8 (E) shows that the higher values are obtained around both corners at $F^{in} = 3\mathrm{m}^3/\mathrm{h}$. This information is not necessarily critical from a design perspective. Rather, it shows that a significant flexibility exists along the most favourable directions for these points, while still being at the bounds of the design space. The flexibility map for $d^{avg-max}$ (F) is based on d^{max}, and hence conveying a similar meaning as the flexibility map for $d^{max-max}$, where the maximum values for this map are typically found at one or more of the corners. Since it is based on all corners of X, $d^{avg-max}$ represents in fact a global maximum flexibility metric. In the future, this metric can be updated to consider a more representative set of points (and not only the most distant points at the corners) to provide more insights on global minimum flexibility.

6.3. n-dimensional boxes

In this Section we consider design spaces in the n-dimensional space, where $n = 1, \ldots, 10$, and where all coordinates of x are bounded between 0 and 1. This arrangement delivers a line segment in 1D, a square in 2D, a cube in 3D, and more generally, n-dimensional boxes in n-dimensional spaces. This set of examples illustrates how the methodology developed can be applied to spaces of any size, and not restricted to 2D or 3D spaces only. Matrices and vectors defining the feasible spaces for these examples are available in the Supplementary Materials. For brevity, the results for the 1D and 2D boxes are presented in Sections 6.3.1 and 6.3.2, respectively; results for the 3D box are presented in Appendix B.

6.3.1. 1D box

Minimum-based distances

The minimum distances to the 2 bounds of the feasible space defining the 1D box ($\theta_1 \geq 0$ and $\theta_1 \leq 1$), and the average minimum distance is presented in Eq. (40). The critical regions for $d^{min-min}$ are presented in Table 2.

$$\left[\begin{array}{c} d_1^{min} \\ d_2^{min} \\ \hline d^{avg\text{-}min} \end{array}\right] = \left[\begin{array}{c|c} 1.000 & 0.000 \\ -1.000 & 1.000 \\ \hline 0.000 & 0.500 \end{array}\right] \left[\begin{array}{c} \theta_1 \\ \hline 1 \end{array}\right] \tag{40}$$

Maximum-based distances

The maximum distances to the 2 bounds/corners of the feasible space ($\theta = [0]$ and $\theta = [1]$), and the average maximum distance is presented in Eq. (41). The critical regions for $d^{max-max}$ are presented in Table 3.

$$\left[\begin{array}{c} (d_1^{max})^2 \\ (d_2^{max})^2 \\ \hline (d^{avg\text{-}max})^2 \end{array}\right] = \theta'\theta \left[\begin{array}{c} 1 \\ 1 \\ \hline 1 \end{array}\right] + \left[\begin{array}{c|c} 0.000 & 0.000 \\ -2.000 & 1.000 \\ \hline -1.000 & 0.500 \end{array}\right] \left[\begin{array}{c} \theta_1 \\ \hline 1 \end{array}\right] \tag{41}$$

6.3.2. 2D box

Minimum-based distances

The minimum distances to the 4 bounds of the feasible space defining the 2D box ($\theta_1 \geq 0$, $\theta_1 \leq 1$, $\theta_2 \geq 0$ and $\theta_2 \leq 1$), and the average minimum distance is presented in Eq. (42). The critical regions for $d^{min-min}$ are presented in Table 4. The map of critical regions for $d^{min-min}$ and the flexibility maps for $d^{min-min}$ and $d^{avg-min}$ are graphically depicted in Fig. 9.

$$\left[\begin{array}{c} d_1^{min} \\ d_2^{min} \\ d_3^{min} \\ d_4^{min} \\ \hline d^{avg\text{-}min} \end{array}\right] = \left[\begin{array}{ccc|c} 1.000 & 0.000 & 0.000 & \\ -1.000 & 0.000 & 1.000 & \\ 0.000 & 1.000 & 0.000 & \\ 0.000 & -1.000 & 1.000 & \\ \hline 0.000 & 0.000 & 0.500 & \end{array}\right] \left[\begin{array}{c} \theta_1 \\ \theta_2 \\ \hline 1 \end{array}\right] \tag{42}$$

Maximum-based distances

The maximum distances to the 4 corners of the feasible space ($\theta = [0,0]$, $\theta = [0,1]$, $\theta = [1,1]$ and $\theta = [1,0]$), and the average maximum distance is presented in Eq. (43). The critical regions for $d^{max-max}$ are presented in Table 5. The map of critical regions for $d^{max-max}$ and the flexibility maps for $d^{max-max}$ and $d^{avg-max}$ are graphically depicted in Fig. 9.

$$\left[\begin{array}{c} (d_1^{max})^2 \\ (d_2^{max})^2 \\ (d_3^{max})^2 \\ (d_4^{max})^2 \\ \hline (d^{avg\text{-}max})^2 \end{array}\right] = \theta'\theta \left[\begin{array}{c} 1 \\ 1 \\ 1 \\ 1 \\ \hline 1 \end{array}\right] + \left[\begin{array}{ccc|c} 0.000 & 0.000 & 0.000 & \\ 0.000 & -2.000 & 1.000 & \\ -2.000 & -2.000 & 2.000 & \\ -2.000 & 0.000 & 1.000 & \\ \hline -1.000 & -1.000 & 1.000 & \end{array}\right] \left[\begin{array}{c} \theta_1 \\ \theta_2 \\ \hline 1 \end{array}\right] \tag{43}$$

6.3.3. Solution complexity

The solutions for the 3D box are listed in Appendix B, and the solutions for the remaining n-dimensional boxes can be calculated from the Python code and the example files in the Supplementary Materials. Results on solution complexity for these examples are presented in Table 6.

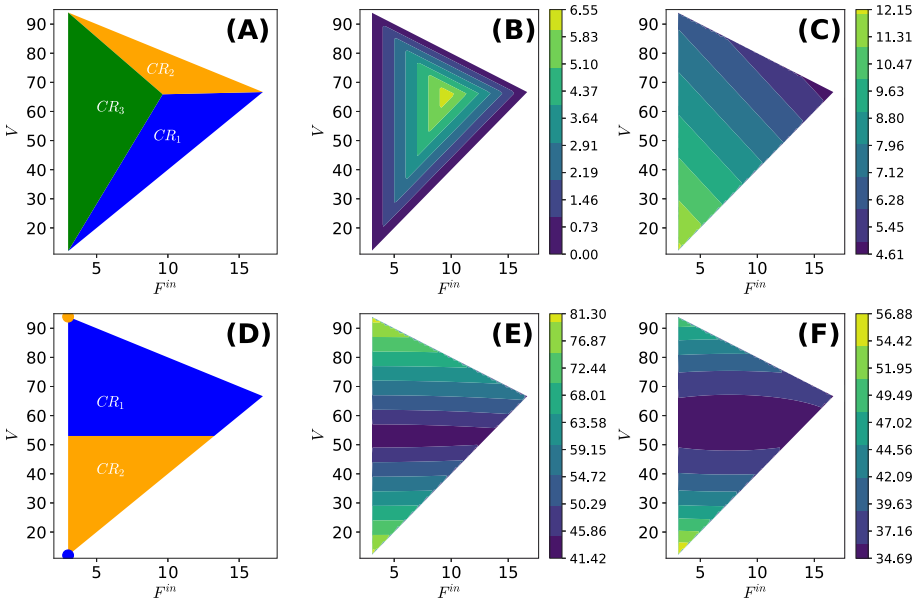


Fig. 8. Minimum and maximum-based distance results for CSTR example. (A) critical regions for $d^{min-min}$, (B) flexibility map for $d^{min-min}$, (C) flexibility map for $d^{avg-min}$, (D) critical regions for $d^{max-max}$, (E) flexibility map for $d^{max-max}$, (F) flexibility map for $d^{avg-max}$. In (B), (C), (E) and (F) a colorbar is presented to capture the contours of $d^{min-min}$, $d^{avg-min}$, $d^{max-max}$ and $d^{avg-max}$, respectively, in the feasible space.

Table 2
Critical regions for $d^{min-min}$ in the 1D box ($CR^A\theta \leq CR^b$).

CR	Active bound	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$	CR	Active bound	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$
1	$\theta_1 = 0$	$\begin{bmatrix} -1.000 & 0.000 \\ 2.000 & -1.000 \end{bmatrix}$	2	$\theta_1 = 1$	$\begin{bmatrix} -2.000 & 1.000 \\ 1.000 & -1.000 \end{bmatrix}$

Table 3
Critical regions for $d^{max-max}$ in the 1D box ($CR^A\theta \leq CR^b$).

CR	Active corner	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$	CR	Active corner	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$
1	$\theta = [0]$	$\begin{bmatrix} -2.000 & 1.000 \\ -1.000 & 0.000 \\ 1.000 & -1.000 \end{bmatrix}$	2	$\theta = [1]$	$\begin{bmatrix} 2.000 & -1.000 \\ -1.000 & 0.000 \\ 1.000 & -1.000 \end{bmatrix}$

Table 4
Critical regions for $d^{min-min}$ in the 2D box ($CR^A\theta \leq CR^b$).

CR	Active bound	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$	CR	Active bound	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$
1	$\theta_1 = 0$	$\begin{bmatrix} -1.000 & 0.000 & 0.000 \\ 2.000 & 0.000 & -1.000 \\ 1.000 & -1.000 & 0.000 \\ 1.000 & 1.000 & -1.000 \end{bmatrix}$	2	$\theta_1 = 1$	$\begin{bmatrix} -2.000 & 0.000 & 1.000 \\ 1.000 & 0.000 & -1.000 \\ -1.000 & -1.000 & 1.000 \\ -1.000 & 1.000 & 0.000 \end{bmatrix}$
3	$\theta_2 = 0$	$\begin{bmatrix} -1.000 & 1.000 & 0.000 \\ 1.000 & 1.000 & -1.000 \\ 0.000 & -1.000 & 0.000 \\ 0.000 & 2.000 & -1.000 \end{bmatrix}$	4	$\theta_2 = 1$	$\begin{bmatrix} -1.000 & -1.000 & 1.000 \\ 1.000 & -1.000 & 0.000 \\ 0.000 & -2.000 & 1.000 \\ 0.000 & 1.000 & -1.000 \end{bmatrix}$

The geometry of these n-dimensional boxes allows the explicit calculation of the total number of critical regions and bounds necessary for their definition. Note that solution complexity for minimum design flexibility scale more favourably than maximum design flexibility: the number of critical regions in the first case is equal to the number of non redundant bounds of the design space, while the total number of corners and matching critical regions scales exponentially with n in the second case. Since the calculation of critical regions' bounds requires the comparison of distance functions, their calculation scales much more favourably in minimum design flexibility (m^2) than in maximum design flexibility ($2^n(2^n - 1 + m)$). This is not necessarily the case in all problems as shown in Section 6.1.3, where out of the maximum 66 pairs of bounds combinations, only 12 yield feasible corners, and only 2 corners are relevant solutions of the mp-P problem.

A refined analysis of Table 6 shows that the computational burden of the proposed methodology for flexibility assessment, particularly in large problems, is associated mainly with Algorithms 3 and 5, confirming the assessment above. Algorithm 3 requires the calculation of up to $C_n^{m'}$ combinations of the following elements: (i) design bounds' indices, (ii) auxiliary matrices and their ranks, (iii) inverse matrices and their products, and (iv) function evaluations to deliver all corners. By contrast, Algorithms 1 and 2 require the solution of a single and m LPs, respectively. Algorithms 4 and 5 consist essentially of vector subtractions for critical regions construction; while in Algorithm 4 this amounts to roughly $(m')^2$ vector subtractions, in Algorithm 5 it may amount to a very large $(C_n^{m'})^2$, which grows much more rapidly for large n and m.

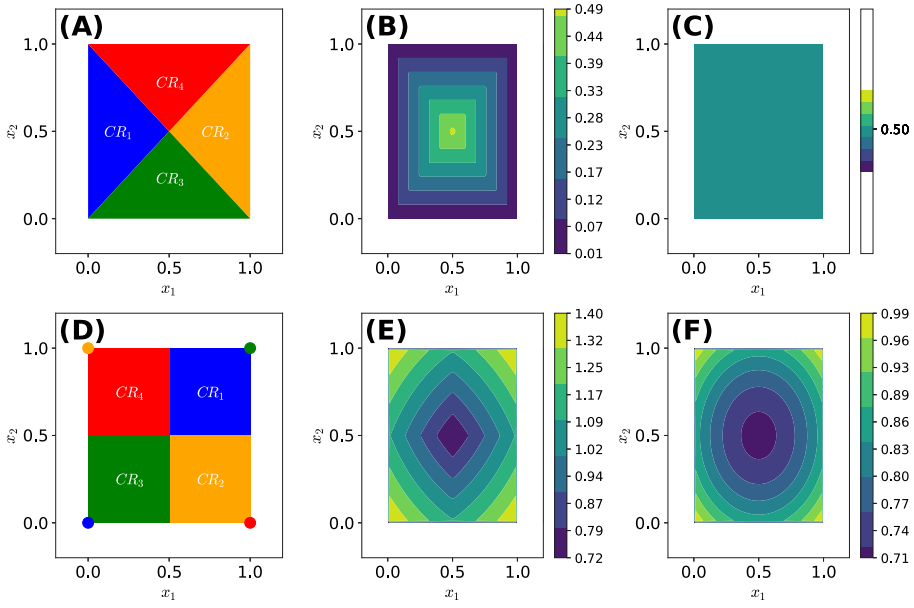


Fig. 9. Minimum and maximum-based distances for the 2D box. (A) critical regions for $d^{min-min}$, (B) flexibility map for $d^{min-min}$, (C) flexibility map for $d^{avg-min}$, (D) critical regions for $d^{max-max}$, (E) flexibility map for $d^{max-max}$, (F) flexibility map for $d^{avg-max}$. In (B), (C), (E) and (F) a colorbar is presented to capture the contours of $d^{min-min}$, $d^{avg-min}$, $d^{max-max}$ and $d^{avg-max}$, respectively, in the feasible space.

Table 5
Critical regions for $d^{max-max}$ in the 2D box ($CR^A\theta \leq CR^b$).

CR	Active corner	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$			CR	Active corner	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$		
1	$\theta = [0, 0]$	0.000	−2.000	1.000	2	$\theta = [0, 1]$	0.000	2.000	−1.000
		−2.000	−2.000	2.000			−2.000	0.000	1.000
		−2.000	0.000	1.000			−2.000	2.000	0.000
		−1.000	0.000	0.000			−1.000	0.000	0.000
		1.000	0.000	−1.000			1.000	0.000	−1.000
		0.000	−1.000	0.000			0.000	−1.000	0.000
		0.000	1.000	−1.000			0.000	1.000	−1.000
3	$\theta = [1, 1]$	2.000	2.000	−2.000	4	$\theta = [1, 0]$	2.000	0.000	−1.000
		2.000	0.000	−1.000			2.000	−2.000	0.000
		0.000	2.000	−1.000			0.000	−2.000	1.000
		−1.000	0.000	0.000			−1.000	0.000	0.000
		1.000	0.000	−1.000			1.000	0.000	−1.000
		0.000	−1.000	0.000			0.000	−1.000	0.000
		0.000	1.000	−1.000			0.000	1.000	−1.000

Table 6
Solution complexity for n-dimensional boxes ($n = 1, \ldots, 10$).

n	Minimum design flexibility		Maximum design flexibility	
	Design bounds	CR bounds	Corners	CR bounds
1	2	4	2	6
2	4	16	4	28
3	6	36	8	104
4	8	64	16	368
5	10	100	32	1312
6	12	144	64	4800
7	14	196	128	18 048
8	16	256	256	69 376
9	18	324	512	270 848
10	20	400	1024	1068032
⋮	⋮	⋮	⋮	⋮
(n)	$(m = 2n)$	(m^2)	$(t = 2^n)$	$(t(t - 1 + m))$

Future work may focus on identifying and excluding irrelevant comparisons of pairs of partial maximum distance functions, which contribute significantly to the complexity of maximum design flexibility. The requirement of including all m bounds of the design space in all critical regions may also be relaxed: while this is important for a formal definition of critical regions, from an online assessment perspective, this requirement may be removed pending an initial feasibility assessment, which also contributes to reducing complexity.

7. Concluding remarks

A novel framework for flexibility assessment has been proposed in this work. It deals specifically with the case when the design space is bounded by a set of m affine constraints defining closed convex hulls in any n-dimensional space. In this case, we have shown that the m single-constraint active sets, and up to C_n^m active sets directly associated with the convex hulls' corners (n active bounds) play an important role in the calculation of the nearest and farthest points at the bounds of the design space with respect to any of its points: this geometry ensures that for any $\theta \in X$, the nearest/farthest points are always found at a bound/corner of X, respectively.

This insight, suggests that the minimum and maximum distances, which translate directly as the minimum and maximum design flexibilities - $d^{min-min}$ and $d^{max-max}$ - can be computed by calculating and processing a restricted set of optimal (distance) functions. These metrics complement each other and present a significant novelty with respect to previous work on this subject, which focuses mainly on the definition of square flexibility boxes around a given design. Additionally, this

work focuses not only on the definition of these metrics but also their calculation in the full design space, which fits in the class of mp-P problems. These solutions have been coined as flexibility maps, and a strategy for critical region construction was defined for both cases via a set of optimality and feasibility constraints as is the common practice in the field.

This framework also enables the calculation of: (i) the average and maximum of the minimum partial design flexibilities, and (ii) the average and minimum of the maximum partial design flexibilities. Average-based calculations - $d^{avg\text{-}min}$ and $d^{avg\text{-}max}$ - are obtained as the average of the corresponding distance functions, whereas the remaining set of flexibility maps are obtained from the solution of multi-parametric problems. Average-based distances have also proved useful as global flexibility metrics, particularly $d^{avg\text{-}max}$, which builds on a set of feasible corners at the bounds of X. An initial screening of results for the remaining metrics - $d^{max-min}$ and $d^{min-max}$ - suggests these are not particularly valuable for flexibility assessments.

Multiple examples were solved and their results analysed, including problems up to $n = 10$. We focused particularly on the illustration of the solution of a 2D illustrative problem and a CSTR design problem, for which rich graphical insights can be obtained via the corresponding map of critical regions and the corresponding flexibility maps delivered as heatmaps. The results obtained fully confirm the intended functionality is delivered via the set of 5 algorithms developed, and offering complementary insights on flexibility. Based on the number of relevant distance functions for the minimum and maximum distance cases, it has been shown and confirmed by preliminary results that solution complexity is expected to grow more significantly with n for the maximum design flexibility case, while minimum design flexibility depends mainly on the number of non redundant bounds.

This work delivers an initial framework for the calculation of multiple flexibility maps when design spaces are defined via a set of affine bounds. For this specific case, all flexibility metrics are calculated explicitly and exactly and delivered as a set of critical regions and matching optimal distance functions. In most practical applications in Chemical Engineering, however, process variables may exhibit nonlinear dependencies between them, which generally result in nonlinear design spaces.

In future work, this more general case will be considered. Finding the nearest/furthest points at the bounds of the design space presents several challenges in the nonlinear case, including also delivering explicit distance functions and bounds of the critical regions. Approximation of the design space via a set of affine bounds is likely the most promising path, from where the methods developed in this work can be used to obtain accurate estimates of flexibility maps.

CRediT authorship contribution statement

Diogo A.C. Narciso: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Steven Sachio:** Writing – review & editing, Writing – original draft, Validation, Investigation. **Maria M. Papathanasiou:** Writing – review & editing, Writing – original draft, Validation, Investigation.

Code availability

A set of Python code files deliver the functionality of the 5 algorithms presented in this article. These files are available in https://github.com/diogonarciso/DSflex. All design space examples presented in this manuscript can be found in the Supplementary Materials.

Funding

Diogo Narciso gratefully acknowledges the support of the CERENA (FCT-UIDB/04028/2025 and FCT-UIDP/04028/2025).

Declaration of competing interest

I have nothing to declare.

Appendix A. Projection to a single bound

In this Appendix, we explain step-by-step how the projection to a single bound as presented in Eq. (8) is consistent with the goals of this work, and namely that it ensures that for any θ included in the halfspace defined via the jth bound of X - $A_j x \le b_j$ $(j = 1, \dots, m)$ - the multiplicative factor l^j in Eq. (8) is non-negative.

Eq. (8) is formulated such that for a given θ, we seek to calculate its projection to the jth bound of X, denoted as $x^{proj,j}$. The definition of the corresponding halfspace is achieved as follows:

$$\begin{bmatrix} -A_{j,1} & -A_{j,2} & \dots & -A_{j,n} \end{bmatrix} \begin{bmatrix} x_1 - x_1^{proj,j} \\ x_2 - x_2^{proj,j} \\ \vdots \\ x_n - x_n^{proj,j} \end{bmatrix} \ge 0 \tag{44}$$

Note that Eq. (44) is defined as an internal product of two vectors: (i) the normal vector for the jth bound of X, $-A_j$, and (ii) the vector defined as the difference between $x^{proj,j}$ and x, thus pointing in the

Table 7
Critical regions for $d^{min-min}$ in the 3D box ($CR^A\theta \le CR^b$).

CR	Active bound	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$	CR	Active bound	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$
1	$\theta_1 = 0$	$\begin{bmatrix} -1.000 & 0.000 & 0.000 & 0.000 \\ 2.000 & 0.000 & 0.000 & -1.000 \\ 1.000 & -1.000 & 0.000 & 0.000 \\ 1.000 & 1.000 & 0.000 & -1.000 \\ 1.000 & 0.000 & -1.000 & 0.000 \\ 1.000 & 0.000 & 1.000 & -1.000 \end{bmatrix}$	2	$\theta_1 = 1$	$\begin{bmatrix} -2.000 & 0.000 & 0.000 & 1.000 \\ 1.000 & 0.000 & 0.000 & -1.000 \\ -1.000 & -1.000 & 0.000 & 1.000 \\ -1.000 & 1.000 & 0.000 & 0.000 \\ -1.000 & 0.000 & -1.000 & 1.000 \\ -1.000 & 0.000 & 1.000 & 0.000 \end{bmatrix}$
3	$\theta_2 = 0$	$\begin{bmatrix} -1.000 & 1.000 & 0.000 & 0.000 \\ 1.000 & 1.000 & 0.000 & -1.000 \\ 0.000 & -1.000 & 0.000 & 0.000 \\ 0.000 & 2.000 & 0.000 & -1.000 \\ 0.000 & 1.000 & -1.000 & 0.000 \\ 0.000 & 1.000 & 1.000 & -1.000 \end{bmatrix}$	4	$\theta_2 = 1$	$\begin{bmatrix} -1.000 & -1.000 & 0.000 & 1.000 \\ 1.000 & -1.000 & 0.000 & 0.000 \\ 0.000 & -2.000 & 0.000 & 1.000 \\ 0.000 & 1.000 & 0.000 & -1.000 \\ 0.000 & -1.000 & -1.000 & 1.000 \\ 0.000 & -1.000 & 1.000 & 0.000 \end{bmatrix}$
5	$\theta_3 = 0$	$\begin{bmatrix} -1.000 & 0.000 & 1.000 & 0.000 \\ 1.000 & 0.000 & 1.000 & -1.000 \\ 0.000 & -1.000 & 1.000 & 0.000 \\ 0.000 & 1.000 & 1.000 & -1.000 \\ 0.000 & 0.000 & -1.000 & 0.000 \\ 0.000 & 0.000 & 2.000 & -1.000 \end{bmatrix}$	6	$\theta_3 = 1$	$\begin{bmatrix} -1.000 & 0.000 & -1.000 & 1.000 \\ 1.000 & 0.000 & -1.000 & 0.000 \\ 0.000 & -1.000 & -1.000 & 1.000 \\ 0.000 & 1.000 & -1.000 & 0.000 \\ 0.000 & 0.000 & -2.000 & 1.000 \\ 0.000 & 0.000 & 1.000 & -1.000 \end{bmatrix}$

Table 8
Critical regions for $d^{max-max}$ in the 3D box ($CR^A\theta \leq CR^b$).

CR	Active corner	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$	CR	Active corner	$[\mathbf{CR^A} \mid -\mathbf{CR^b}]$
1	$\theta = [0,0,0]$	0.000 0.000 −2.000 1.000 0.000 −2.000 0.000 1.000 0.000 −2.000 −2.000 2.000 −2.000 0.000 0.000 1.000 −2.000 0.000 −2.000 2.000 −2.000 −2.000 0.000 2.000 −2.000 −2.000 −2.000 3.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000	2	$\theta = [0,0,1]$	0.000 0.000 2.000 −1.000 0.000 −2.000 2.000 0.000 0.000 −2.000 0.000 1.000 −2.000 0.000 2.000 0.000 −2.000 0.000 0.000 1.000 −2.000 −2.000 2.000 1.000 −2.000 −2.000 0.000 2.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000
3	$\theta = [0,1,0]$	0.000 2.000 0.000 −1.000 0.000 2.000 −2.000 0.000 0.000 0.000 −2.000 1.000 −2.000 2.000 0.000 0.000 −2.000 2.000 −2.000 1.000 −2.000 0.000 0.000 1.000 −2.000 0.000 −2.000 2.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000	4	$\theta = [0,1,1]$	0.000 2.000 2.000 −2.000 0.000 2.000 0.000 −1.000 0.000 0.000 2.000 −1.000 −2.000 2.000 2.000 −1.000 −2.000 2.000 0.000 0.000 −2.000 0.000 2.000 0.000 −2.000 0.000 0.000 1.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000
5	$\theta = [1,0,0]$	2.000 0.000 0.000 −1.000 2.000 0.000 −2.000 0.000 2.000 −2.000 0.000 0.000 2.000 −2.000 −2.000 1.000 0.000 0.000 −2.000 1.000 0.000 −2.000 0.000 1.000 0.000 −2.000 −2.000 2.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000	6	$\theta = [1,0,1]$	2.000 0.000 2.000 −2.000 2.000 0.000 0.000 −1.000 2.000 −2.000 2.000 −1.000 2.000 −2.000 0.000 0.000 0.000 0.000 2.000 −1.000 0.000 −2.000 2.000 0.000 0.000 −2.000 0.000 1.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000
7	$\theta = [1,1,0]$	2.000 2.000 0.000 −2.000 2.000 2.000 −2.000 −1.000 2.000 0.000 0.000 −1.000 2.000 0.000 −2.000 0.000 0.000 2.000 0.000 −1.000 0.000 2.000 −2.000 0.000 0.000 0.000 −2.000 1.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000	8	$\theta = [1,1,1]$	2.000 2.000 2.000 −3.000 2.000 2.000 0.000 −2.000 2.000 0.000 2.000 −2.000 2.000 0.000 0.000 −1.000 0.000 2.000 2.000 −2.000 0.000 2.000 0.000 −1.000 0.000 0.000 2.000 −1.000 −1.000 0.000 0.000 0.000 1.000 0.000 0.000 −1.000 0.000 −1.000 0.000 0.000 0.000 1.000 0.000 −1.000 0.000 0.000 −1.000 0.000 0.000 0.000 1.000 −1.000

direction from $x^{proj,j}$ to x: all points x such that the internal product with $-A_j$ is non-negative define the halfspace of interest. This may be formulated equivalently as:

$$\begin{bmatrix} -A_{j,1} & -A_{j,2} & \dots & -A_{j,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \geq \begin{bmatrix} -A_{j,1} & -A_{j,2} & \dots & -A_{j,n} \end{bmatrix} \begin{bmatrix} x_1^{proj} \\ x_2^{proj} \\ \vdots \\ x_n^{proj} \end{bmatrix} \tag{45}$$

where the internal product $-A_j x^{proj,j}$ necessarily equals the independent term of the jth bound, and so:

$$\begin{bmatrix} -A_{j,1} & -A_{j,2} & \dots & -A_{j,n} \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}' \geq -b_j \tag{46}$$

which can be written in the more standard notation using a less or equal to sign used in this work to list all bounds of the feasible space:

$$\begin{bmatrix} A_{j,1} & A_{j,2} & \dots & A_{j,n} \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}' \leq b_j \tag{47}$$

The projection of θ to the jth bound of X is actually defined in the opposite direction: from $x^{proj,j}$ to θ, such that in this way the normal vector used in this calculation points towards the interior of X, and thus ensuring l^j is non-negative for all θ satisfying $A_j\theta \leq b_j$. This is defined as:

$$\left[\begin{array}{cccc|c} 1 & 0 & \dots & 0 & -A_{j,1} \\ 0 & 1 & \dots & 0 & -A_{j,2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -A_{j,n} \\ \hline A_{j,1} & A_{j,2} & \dots & A_{j,n} & 0 \end{array}\right] \left[\begin{array}{c} x_1^{proj,j} \\ x_2^{proj,j} \\ \vdots \\ x_n^{proj,j} \\ \hline l^j \end{array}\right] = \left[\begin{array}{c} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \\ \hline b_j \end{array}\right] \tag{48}$$

From the discussion above and the first n rows of Eq. (48), all θ in the halfspace $A_j x \leq b_j$ must be obtained by a non-negative l. To finalise this note, it suffices to multiply the last equality constraint in Eq. (48)

by −1 to obtain Eq. (8). Both problem statements are fully equivalent; Eq. (8) is preferable since the matrix defined - M - is such that $M = M'$.

Appendix B. Results for 3D box

Minimum-based distances

The minimum distances to the 6 bounds of the feasible space defining the 3D box ($\theta_1 \geq 0$, $\theta_1 \leq 1$, $\theta_2 \geq 0$, $\theta_2 \leq 1$, $\theta_3 \geq 0$ and $\theta_3 \leq 1$), and the average minimum distance is presented in Eq. (49). The critical regions for $d^{min-min}$ are presented in Table 7.

$$\begin{bmatrix} d_1^{min} \\ d_2^{min} \\ d_3^{min} \\ d_4^{min} \\ d_5^{min} \\ d_6^{min} \\ \hline d^{avg,min} \end{bmatrix} = \left[\begin{array}{cccc} 1.000 & 0.000 & 0.000 & 0.000 \\ -1.000 & 0.000 & 0.000 & 1.000 \\ 0.000 & 1.000 & 0.000 & 0.000 \\ 0.000 & -1.000 & 0.000 & 1.000 \\ 0.000 & 0.000 & 1.000 & 0.000 \\ 0.000 & 0.000 & -1.000 & 1.000 \\ \hline 0.000 & 0.000 & 0.000 & 0.500 \end{array}\right] \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \hline 1 \end{bmatrix} \tag{49}$$

Maximum-based distances

The maximum distances to the 8 corners of the feasible space ($\theta = [0,0,0]$, $\theta = [0,0,1]$, $\theta = [0,1,0]$, $\theta = [0,1,1]$, $\theta = [1,0,0]$, $\theta = [1,0,1]$, $\theta = [1,1,0]$ and $\theta = [1,1,1]$), and the average maximum distance is presented in Eq. (50). The critical regions for $d^{max-max}$ are presented in Table 8.

$$\begin{bmatrix} (d_1^{max})^2 \\ (d_2^{max})^2 \\ (d_3^{max})^2 \\ (d_4^{max})^2 \\ (d_5^{max})^2 \\ (d_6^{max})^2 \\ (d_7^{max})^2 \\ (d_8^{max})^2 \\ \hline (d^{avg,max})^2 \end{bmatrix} = \theta'\theta \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ \hline 1 \end{bmatrix} + \left[\begin{array}{cccc} 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & -2.000 & 1.000 \\ 0.000 & -2.000 & 0.000 & 1.000 \\ 0.000 & -2.000 & -2.000 & 2.000 \\ -2.000 & 0.000 & 0.000 & 1.000 \\ -2.000 & 0.000 & -2.000 & 2.000 \\ -2.000 & -2.000 & 0.000 & 2.000 \\ -2.000 & -2.000 & -2.000 & 3.000 \\ \hline -1.000 & -1.000 & -1.000 & 1.500 \end{array}\right] \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \hline 1 \end{bmatrix} \tag{50}$$

Appendix C. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.compchemeng.2025.109189.

Data availability

No data was used for the research described in the article.

References

Adi, V.S.K., Laxmidewi, R., Chang, C.-T., 2016. An effective computation strategy for assessing operational flexibility of high-dimensional systems with complicated feasible regions. Chem. Eng. Sci. 147, 137–149. http://dx.doi.org/10.1016/j.ces.2016.03.028.

Banerjee, I., Ierapetritou, M.G., 2005. Feasibility evaluation of nonconvex systems using shape reconstruction techniques. Ind. Eng. Chem. Res. 44, 3638–3647. http://dx.doi.org/10.1021/ie049294d.

Bano, G., Wang, Z., Facco, P., Bezzo, F., Barolo, M., Ierapetritou, M., 2018. A novel and systematic approach to identify the design space of pharmaceutical processes. Comput. Chem. Eng. 115, 309–322. http://dx.doi.org/10.1016/j.compchemeng.2018.04.021, URL:<Go to ISI>://WOS:000439701900026.

Bansal, V., Perkins, J.D., Pistikopoulos, E.N., 2002. Flexibility analysis and design using a parametric programming framework. AIChE J. 48, 2851–2868. http://dx.doi.org/10.1002/aic.690481213.

Bates, R.A., Wynn, H.P., Fraga, E.S., 2007. Feasible region approximation: a comparison of search cone and convex hull methods. Eng. Optim. 39, 513–527. http://dx.doi.org/10.1080/03052150701351680.

Boukouvala, F., Ierapetritou, M.G., 2012. Feasibility analysis of black-box processes using an adaptive sampling Kriging-based method. Comput. Chem. Eng. 36, 358–368. http://dx.doi.org/10.1016/j.compchemeng.2011.06.005.

Boukouvala, F., Muzzio, F.J., Ierapetritou, M.G., 2010. Design space of pharmaceutical processes using data-driven-based methods. J. Pharm. Innov. 5, 119–137. http://dx.doi.org/10.1007/s12247-010-9086-y, URL:<Go to ISI>://WOS:000297067800008.

Burnak, B., Pistikopoulos, E.N., 2020. Integrated process design, scheduling, and model predictive control of batch processes with closed-loop implementation. AIChE J. 66, http://dx.doi.org/10.1002/aic.16981.

Chatzizacharia, K.A., Hatziavramidis, D.T., 2014. Design space approach for pharmaceutical tablet development. Ind. Eng. Chem. Res. 53, 12003–12009. http://dx.doi.org/10.1021/ie5005652.

Close, E., Bracewell, D.G., Sorensen, E., 2013. A model based approach to an adaptive design space in chromatography. In: 23rd European Symposium on Computer Aided Process Engineering. In: Computer Aided Chemical Engineering, pp. 115–120. http://dx.doi.org/10.1016/b978-0-444-63234-0.50020-8.

Close, E.J., Salm, J.R., Bracewell, D.G., Sorensen, E., 2014. A model based approach for identifying robust operating conditions for industrial chromatography with process variability. Chem. Eng. Sci. 116, 284–295. http://dx.doi.org/10.1016/j.ces.2014.03.010, URL:<Go to ISI>://WOS:000340330500027.

Diangelakis, N.A., Burnak, B., Katz, J., Pistikopoulos, E.N., 2017. Process design and control optimization: A simultaneous approach by multi-parametric programming. AIChE J. 63, 4827–4846. http://dx.doi.org/10.1002/aic.15825, URL:<Go to ISI>://WOS:000412727500009.

Diez, M., Campana, E.F., Stern, F., 2015. Design-space dimensionality reduction in shape optimization by Karhunen–Loève expansion. Comput. Methods Appl. Mech. Engrg. 283, 1525–1544. http://dx.doi.org/10.1016/j.cma.2014.10.042.

Dimitriadis, V.D., Pistikopoulos, E.N., 1995. Flexibility analysis of dynamic systems. Ind. Eng. Chem. Res. 34, 4451–4462. http://dx.doi.org/10.1021/ie00039a036.

Ding, C., Ierapetritou, M., 2021. A novel framework of surrogate-based feasibility analysis for establishing design space of twin-column continuous chromatography. Int. J. Pharm. 609, 121161. http://dx.doi.org/10.1016/j.ijpharm.2021.121161.

Facco, P., Dal Pastro, F., Meneghetti, N., Bezzo, F., Barolo, M., 2015. Bracketing the design space within the knowledge space in pharmaceutical product development. Ind. Eng. Chem. Res. 54, 5128–5138. http://dx.doi.org/10.1021/acs.iecr.5b00863.

García-Muñoz, S., Dolph, S., Ward, H.W., 2010. Handling uncertainty in the establishment of a design space for the manufacture of a pharmaceutical product. Comput. Chem. Eng. 34, 1098–1107. http://dx.doi.org/10.1016/j.compchemeng.2010.02.027.

Georgakis, C., Uztürk, D., Subramanian, S., Vinson, D.R., 2003. On the operability of continuous processes. Control Eng. Pract. 11, 859–869. http://dx.doi.org/10.1016/s0967-0661(02)00217-4.

Geremia, M., Bezzo, F., Ierapetritou, M.G., 2023. A novel framework for the identification of complex feasible space. Comput. Chem. Eng. http://dx.doi.org/10.1016/j.compchemeng.2023.108427.

Goyal, V., Ierapetritou, M.G., 2003. Framework for evaluating the feasibility/operability of nonconvex processes. AIChE J. 49, 1233–1240. http://dx.doi.org/10.1002/aic.690490514.

Grossmann, I.E., Calfa, B.A., Garcia-Herreros, P., 2014. Evolution of concepts and models for quantifying resiliency and flexibility of chemical processes. Comput. Chem. Eng. 70, 22–34. http://dx.doi.org/10.1016/j.compchemeng.2013.12.013.

Grossmann, I.E., Floudas, C.A., 1987. Active constraint strategy for flexibility analysis in chemical processes. Comput. Chem. Eng. 11, 675–693. http://dx.doi.org/10.1016/0098-1354(87)87011-4.

Halemane, K.P., Grossmann, I.E., 1983. Optimal process design under uncertainty. AIChE J. 29, 425–433. http://dx.doi.org/10.1002/aic.690290312.

Harwood, S.M., Barton, P.I., 2017. How to solve a design centering problem. Math. Methods Oper. Res. 86, 215–254. http://dx.doi.org/10.1007/s00186-017-0591-3.

Hirono, K., A. Udugama, I., Hayashi, Y., Kino-oka, M., Sugiyama, H., 2022. A dynamic and probabilistic design space determination method for mesenchymal stem cell cultivation processes. Ind. Eng. Chem. Res. 61, 7009–7019. http://dx.doi.org/10.1021/acs.iecr.2c00374.

Ierapetritou, M.G., 2001. New approach for quantifying process feasibility: Convex and 1-D quasi-convex regions. AIChE J. 47, 1407–1417. http://dx.doi.org/10.1002/aic.690470616.

Kabatek, U., Swaney, R.E., 1992. Worst-case identification in structured process systems. Comput. Chem. Eng. 16, 1063–1071. http://dx.doi.org/10.1016/0098-1354(92)80024-4.

Koller, R.W., Ricardez-Sandoval, L.A., Biegler, L.T., 2018. Stochastic back-off algorithm for simultaneous design, control, and scheduling of multiproduct systems under uncertainty. AIChE J. 64, 2379–2389. http://dx.doi.org/10.1002/aic.16092.

Kotidis, P., Demis, P., Goey, C.H., Correa, E., McIntosh, C., Trepekli, S., Shah, N., Klymenko, O.V., Kontoravdi, C., 2019. Constrained global sensitivity analysis for bioprocess design space identification. Comput. Chem. Eng. 125, 558–568. http://dx.doi.org/10.1016/j.compchemeng.2019.01.022, URL:<Go to ISI>://WOS:000471232800038.

Kucherenko, S., Giamalakis, D., Shah, N., García-Muñoz, S., 2020. Computationally efficient identification of probabilistic design spaces through application of metamodeling and adaptive sampling. Comput. Chem. Eng. 132, http://dx.doi.org/10.1016/j.compchemeng.2019.106608.

Kucherenko, S., Sopittakamol, N., Shah, N., 2024. Design space identification of a coupled two-stage batch reactor system. Comput. Chem. Eng. 187, http://dx.doi.org/10.1016/j.compchemeng.2024.108727.

Kusumo, K.P., Gomoescu, L., Paulen, R., García Muñoz, S., Pantelides, C.C., Shah, N., Chachuat, B.t., 2020. Bayesian approach to probabilistic design space characterization: A nested sampling strategy. Ind. Eng. Chem. Res. 59, 2396–2408. http://dx.doi.org/10.1021/acs.iecr.9b05006.

Laky, D., Casas-Orozco, D., Rossi, F., Mackey, J.S., Reklaitis, G.V., Nagy, Z.K., 2022. Determination of probabilistic design spaces in the hybrid manufacture of an active pharmaceutical ingredient using pharmapy. Comput. Aided Chem. Eng. 49, 2131–2136. http://dx.doi.org/10.1016/B978-0-323-85159-6.50355-9.

Metta, N., Ramachandran, R., Ierapetritou, M., 2020. A novel adaptive sampling based methodology for feasible region identification of compute intensive models using artificial neural network. AIChE J. 67, http://dx.doi.org/10.1002/aic.17095.

Murty, K.G., 2009. Ball centers of special polytopes.

Narciso, D.A.C., Pappas, I., Martins, F.G., Pistikopoulos, E.N., 2022. A new solution strategy for multiparametric quadratic programming. Comput. Chem. Eng. 164, http://dx.doi.org/10.1016/j.compchemeng.2022.107882.

Narciso, D.A.C., Pistikopoulos, E.N., 2025. Novel solution strategies for multiparametric nonlinear optimization problems with convex objective function and linear constraints. Optim. Eng. 26, 1–29. http://dx.doi.org/10.1007/s11081-024-09888-2.

Ochoa, M.P., García-Muñoz, S., Stamatis, S., Grossmann, I.E., 2021. Novel flexibility index formulations for the selection of the operating range within a design space. Comput. Chem. Eng. 149, http://dx.doi.org/10.1016/j.compchemeng.2021.107284.

Owoyele, O., Pal, P., 2020. A novel active optimization approach for rapid and efficient design space exploration using ensemble machine learning. J. Energy Resour. Technol. 143, http://dx.doi.org/10.1115/1.4049178.

Papathanasiou, M.M., Avraamidou, S., Oberdieck, R., Mantalaris, A., Steinebach, F., Morbidelli, M., Mueller-Spaeth, T., Pistikopoulos, E.N., 2016. Advanced control strategies for the multicolumn countercurrent solvent gradient purification process. AIChE J. 62, 2341–2357. http://dx.doi.org/10.1002/aic.15203, URL:<Go to ISI>://WOS:000377274600008.

Pappas, I., Kenefake, D., Burnak, B., Avraamidou, S., Ganesh, H.S., Katz, J., Diangelakis, N.A., Pistikopoulos, E.N., 2021. Multiparametric programming in process systems engineering: Recent developments and path forward. Front. Chem. Eng. 2, http://dx.doi.org/10.3389/fceng.2020.620168.

Pistikopoulos, E.N., Diangelakis, N.A., Oberdieck, R., Papathanasiou, M.M., Nascu, I., Sun, M.X., 2015. PAROC-an integrated framework and software platform for the optimisation and advanced model-based control of process systems. Chem. Eng. Sci. 136, 115–138. http://dx.doi.org/10.1016/j.ces.2015.02.030, URL:<Go to ISI>://WOS:000361408600011.

Pistikopoulos, E.N., Mazzuchi, T.A., 1990. A novel flexibility analysis approach for processes with stochastic parameters. Comput. Chem. Eng. 14, 991–1000. http://dx.doi.org/10.1016/0098-1354(90)87055-t.

Pistikopoulos, E.N., Tian, Y., 2024. Advanced modeling and optimization strategies for process synthesis. Annu. Rev. Chem. Biomol. Eng. 15, 81–103. http://dx.doi.org/10.1146/annurev-chembioeng-100522-112139.

Pulsipher, J.L., Rios, D., Zavala, V.M., 2019. A computational framework for quantifying and analyzing system flexibility. Comput. Chem. Eng. 126, 342–355. http://dx.doi.org/10.1016/j.compchemeng.2019.04.024.

Rafiei-Shishavan, M., Mehta, S., Ricardez-Sandoval, L.A., 2017. Simultaneous design and control under uncertainty: A back-offapproach using power series expansions. Comput. Chem. Eng. 99, 66–81. http://dx.doi.org/10.1016/j.compchemeng.2016.12.015.

Rogers, A., Ierapetritou, M., 2015. Feasibility and flexibility analysis of black-box processes part 1: Surrogate-based feasibility analysis. Chem. Eng. Sci. 137, 986–1004. http://dx.doi.org/10.1016/j.ces.2015.06.014.

Rooney, W.C., Biegler, L.T., 2001. Design for model parameter uncertainty using nonlinear confidence regions. AIChE J. 47, 1794–1804. http://dx.doi.org/10.1002/aic.690470811.

Sachio, S., Kontoravdi, C., Papathanasiou, M.M., 2023. A model-based approach towards accelerated process development: A case study on chromatography. Chem. Eng. Res. Des. 197, 800–820. http://dx.doi.org/10.1016/j.cherd.2023.08.016.

Sachio, S., Likozar, B., Kontoravdi, C., Papathanasiou, M.M., 2024a. Computer-aided design space identification for screening of protein A affinity chromatography resins. J. Chromatogr. A 1722, 464890. http://dx.doi.org/10.1016/j.chroma.2024.464890.

Sachio, S., Ward, A., Pini, R., Papathanasiou, M.M., 2024b. Operability-economics trade-offs in adsorption-based CO_2 capture processes. Commun. Eng. 3, http://dx.doi.org/10.1038/s44172-024-00244-x.

Samsatli, N.J., Sharif, M., Shah, N., Papageorgiou, L.G., 2001. Operational envelopes for batch processes. AIChE J. 47, 2277–2288. http://dx.doi.org/10.1002/aic.690471013.

Swaney, R.E., Grossmann, I.E., 1985. An index for operational flexibility in chemical process design. Part I: Formulation and theory. AIChE J. 31, 621–630. http://dx.doi.org/10.1002/aic.690310412.

Wang, Z., Ierapetritou, M., 2017. A novel feasibility analysis method for black-box processes using a radial basis function adaptive sampling approach. AIChE J. 63, 532–550. http://dx.doi.org/10.1002/aic.15362.

Wang, J., Lakerveld, R., 2017. Continuous membrane-assisted crystallization to increase the attainable product quality of pharmaceuticals and design space for operation. Ind. Eng. Chem. Res. 56, 5705–5714. http://dx.doi.org/10.1021/acs.iecr.7b00514.

Zhang, Z., Yang, S.-B., Huang, B., Li, Z., 2024. Novel feasible set learning and process flexibility analysis method using deep neural networks. Ind. Eng. Chem. Res. 63, 11516–11529. http://dx.doi.org/10.1021/acs.iecr.4c00838.

Zhao, F., Chen, X., 2018. Analytical and triangular solutions to operational flexibility analysis using quantifier elimination. AIChE J. 64, 3894–3911. http://dx.doi.org/10.1002/aic.16207.

Zhao, F., Grossmann, I.E., García-Muñoz, S., Stamatis, S.D., 2022a. Design space description through adaptive sampling and symbolic computation. AIChE J. 68, http://dx.doi.org/10.1002/aic.17604.

Zhao, F., Paz Ochoa, M., Grossmann, I.E., García-Muñoz, S., Stamatis, S.D., 2022b. Novel formulations of flexibility index and design centering for design space definition. Comput. Chem. Eng. 166, http://dx.doi.org/10.1016/j.compchemeng.2022.107969.