Interval Predictor Models for Robust System Identification*

Luis G. Crespo†*, Sean Kenny†, Brendon Colbert†, and Tanner Slagel†

***Abstract*— This paper proposes a framework for the identification and uncertainty quantification of plant models according to multivariable data. The only restriction imposed upon such models is for their outputs to depend continuously on their parameters. An Interval Predictor Model (IPM) prescribes the parameters of a computational model as a path-connected set thereby making each predicted output an interval-valued function of its inputs. The formulation proposed seeks the parameter set for which the predicted outputs tightly enclose the data. This set, which is modeled as a semi-algebraic set of low-degree polynomials, enables the characterization of possibly strong parameter dependencies commonly found in practice. This uncertainty characterization makes the resulting plant model amenable to robust control approaches using polynomial optimization. Furthermore, we use scenario theory to assess the reliability of the resulting IPM. This assessment yields a distribution-free upper bound on the probability that future data will fall outside the predicted intervals.**

I. Introduction

The dynamics of physical systems are often uncertain thereby making non-robust computational models intrinsically inaccurate. The classic approach to robust identification entails generating an uncertainty model of the plant, e.g., norm-bounded perturbations of a nominal plant in the frequency domain, whose outputs tightly enclose available measurements. This is also the case for the Interval Predictor Models (IPM) proposed herein. IPMs characterize the parameters of a computational model as a path-connected set thereby yielding predicted output intervals. Means to estimate IPMs for computational models having an affine parameter dependency are available [1]. This paper proposes an identification framework applicable to models having a continuous but otherwise arbitrary dependency on its parameters according to multivariate data. This setting includes models having a nonlinear and implicit structure whose predictions require simulation or numerical integration, e.g., the outputs of a Linear Time-Invariant (LTI) model derived from a finite element model.

The parameters of the model will be characterized as a semi-algebraic set. Parameters prescribed by a set are regarded as dependent when the range of variation of any of them depends on the value taken by the others. Modeling these commonly found dependencies is instrumental in reducing unnecessary conservatism in the uncertain plant model and therefore, in the robust controllers designed for it [2]. The uncertain plant models resulting from the proposed framework are able to characterize such dependencies with varying levels of complexity. Furthermore, we use scenario theory [3], [4] to compute a distribution-free upper bound on the probability that future data will fall outside the predicted intervals. This is a certificate of correctness quantifying the generalization properties of the identified model.

*This effort was supported by the Advanced Air Transport Technology Project from NASA.

†Dynamic Systems and Controls Branch, NASA Langley Research Center, Hampton, VA, 23681, USA.

*Corresponding author, luis.g.crespo@nasa.gov.

II. Problem Statement

A dynamic system is postulated to act on an input variable, $x \in X \subset \mathbb{R}^{n_x}$, to produce an output variable, $y \in Y \subset \mathbb{R}^{n_y}$. Assume that n independent and identically distributed input-output pairs are drawn from an uncertain dynamical system, and denote by

$$\mathcal{D} \triangleq \left\{ \left(x^{(i)}, y^{(i)} \right) \right\}_{i=1}^{n}, \tag{1}$$

the corresponding data sequence. Each element of this sequence will be called an observation or a scenario. Variability in the observations might stem from a changing environment, varying physical properties, measurement error, unmodeled/unmeasurable inputs, or non-linearities. As such, multiple observations are taken to account for the variability of the conditions to which future decisions will be applied. Furthermore, assume that a parametric computational model of the underlying dynamical system is available. This input-output model will be given by the continuous function $y : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_y}$, where

$$y(x, p), \tag{2}$$

represents the output given some input x and parameter vector p. These functions are commonly represented in either time or frequency domains. The uncertainty models developed below cast the variability in the measured responses as parametric uncertainty.

The goals of this article are two fold. First, we want to identify an uncertain input-output model according to the data sequence $\mathcal{D}$ that captures parameter dependencies. Second, we want assess the reliability of such a model with respect to future data drawn from the same underlying process governing $\mathcal{D}$ without actually modeling it.

A few remarks regarding notation are in order. A scenario, to be denoted as $\mathcal{D}^{(i)} = (x^{(i)}, y^{(i)})$, might correspond to a single value or to a set of values. For instance, if each of the n input-output pairs corresponds to a different specimen in an ensemble of test articles, and the data describes the Frequency Response Function (FRF) resulting from modal analysis, the input variable x will contain n_w frequencies whereas the output variables y_1 and y_2 will contain the corresponding magnitude and phase. For simplicity in the

notation we assume that all scenarios have the same input, so $x^{(i)} = \hat{x} \in \mathbb{R}^{n_w}$ for $i = 1, \dots n$, and that each output has n_w elements, so $y^{(i)} \in \mathbb{R}^{n_w \times n_y}$. Particular elements of the output will be referenced using subindices. In particular, $y^{(i)}_{j,k}$ will denote the j-th element of the k-th output of the i-th scenario. Therefore, the data subsequence corresponding to the k-th output is $\mathcal{D}_k = \{\hat{x}, y^{(i)}_{j,k}\}_{i=1}^{n}$ with $j = 1 \dots n_w$.

Finding a point p in (2) leading to a predicted output that closely approximates the measurements is a classical model calibration problem. Techniques for identifying this parameter point can be cast as

$$p^\star(\mathcal{D}) = \underset{p}{\operatorname{argmin}} \left\{ \sum_{i=1}^{m} e(\mathcal{D}^{(i)}, p) : g(p) \leq 0 \right\}, \quad (3)$$

where

$$e(\mathcal{D}^{(i)}, p) = \left\| y(\hat{x}, p) - y^{(i)} \right\|, \quad (4)$$

is the prediction error, $\| \cdot \|$ refers to a norm in the corresponding output space(s), and the constraint $g(p) \leq 0$, with $g : \mathbb{R}^{n_p} \to \mathbb{R}^{n_g}$, enforces performance and stability requirements.

In contrast to classical system identification approaches, the prediction error e in (4) will not be characterized upfront as measurement noise having a known distribution [5]. Instead, uncertainty will be characterized parametrically according to the spread of the measured outputs. A key attribute of the resulting uncertainty characterization is its compatibility with the input-output model. For instance, if (2) is the finite element model of a flexible structure, any uncertain LTI model the analyst might derive from it will correspond to a feasible pair of mass and stiffness matrices. This is in contrast to schemes for which the uncertain LTI model contains physically unrealizable plants.

III. Interval Predictor Models

The single-output $n_y = 1$ case will be considered first. The IPM corresponding to the input-output model (2), the input $x \in X$, and the set $\mathbb{P} \subset \mathbb{R}^{n_p}$ is defined as

$$I_y(y; x, \mathbb{P}) \triangleq \{ y(x, p) \text{ for all } p \in \mathbb{P} \}. \quad (5)$$

Lemma 1: I_y is an interval when $y(x, p)$ is continuous in p for a fixed x, and $\mathbb{P}$ is closed, bounded, and path-connected.

Proof: $\mathbb{P}$ is path-connected if for every pair of points p_a and p_b in $\mathbb{P}$ there exists a continuous function $f : [0, 1] \to \mathbb{P}$ such that $f(0) = p_a$ and $f(1) = p_b$. Suppose $n_y = 1$, y is continuous in the p variable, and $\mathbb{P}$ is path-connected. Let $y_a, y_b, \in I_y$. There are $p_a, p_b \in \mathbb{P}$ such that $y_a = y(x, p_a)$ and $y_b = y(x, p_b)$. Since $\mathbb{P}$ is path-connected there is a continuous function $f : [0, 1] \to \mathbb{P}$ such that $f(0) = p_a$ and $f(1) = p_b$. Define $g : [0, 1] \to I_y$, where $g(t) = y(x, f(t))$ for each $t \in [0, 1]$. Since f and y are continuous, and the composition of two continuous functions is continuous, g is continuous. Since $g(0) = x_a$, and $g(1) = x_b$, I_y is path-connected. Being a path-connected subset of $\mathbb{R}$ is equivalent to being an interval. However this interval could be open, closed, half open, half closed, or unbounded. To ensure that I_y is bounded, it is sufficient to require that y is bounded. To ensure that I_y is a closed interval, it is sufficient to require that $\mathbb{P}$ is compact. □

The conditions of Lemma (1) will be assumed to hold hereafter. The IPM in (5) can be written as

$$I_y(y; x, \mathbb{P}) = \left[\underline{y}(x, \mathbb{P}),\ \overline{y}(x, \mathbb{P}) \right], \quad (6)$$

where the functions

$$\underline{y}(x, \mathbb{P}) = \min_{p \in \mathbb{P}} y(x, p), \quad (7)$$

and

$$\overline{y}(x, \mathbb{P}) = \max_{p \in \mathbb{P}} y(x, p), \quad (8)$$

are the lower and upper IPM boundaries respectively. Therefore, an IPM is an interval-valued function of x given by the mapping of all possible values of p in $\mathbb{P}$ through the input-output model. For each $x \in X$, (5) is an interval in Y.

IPMs admit a functional interpretation: the graph of each member of the family of infinitely many output functions that results from evaluating (2) at all possible realizations of p in $\mathbb{P}$ and x in X lies between the lower and upper IPM boundaries and no tighter boundaries exist.

Equations (5) and (6) can be naturally extended to the multi-output case $n_y > 1$, which leads to the collection of n_y interdependent IPMs given by $I_{y_k}(y_k; x, \mathbb{P})$, where $k = 1, \dots n_y$. This interdependency stems from all n_y IPMs using the same parameter set $\mathbb{P}$.

The formulation below seeks a set $\mathbb{P}$ of a given class leading to a data-enclosing IPM having a minimal width. This set will be characterized in the two-step process detailed below. The first step entails mapping each input-output measurement onto a collection of parameter points. This mapping is built such that the ensemble of predictions corresponding to these points tightly enclose such a measurement. The second step consists in finding a semi-algebraic set tightly enclosing such points.

IV. Data Mapping

A. Ignoring the Prediction Error

In this section we use a single parameter point to represent each input-output datum. A natural choice for such a point is $p^{(i)} = p^\star(\mathcal{D}^{(i)})$ from (3). Hence, the parameter points corresponding to all scenarios in $\mathcal{D}$ comprise the sequence

$$\mathcal{P} \triangleq \left\{ p^{(i)} \right\}_{i=1}^{n}. \quad (9)$$

Hence, each input-output pair in (1) is mapped to a parameter point in (9), i.e., $\left(x^{(i)}, y^{(i)}\right) \to p^{(i)}$. The unobservable elements of (9) can be interpreted as "surrogate data."

B. Quantifying the Prediction Error

IPMs that account for the prediction error represent each input-output pair in (1) as a parameter set. The process of calculating this set, which entails first seeking a point-based inner bound, is presented next.

The multi-point IPMs corresponding to the parameter sequence $\mathcal{Q} = \left\{q^{(j)}\right\}_{j=1}^{r}$ with $q^{(j)} \in \mathbb{R}^{n_p}$ are defined as

$$I_{y_k}(y_k; x, \mathcal{Q}) \triangleq \left[\min_{j=1\ldots r} y_k\left(x, q^{(j)}\right),\ \max_{j=1\ldots r} y_k\left(x, q^{(j)}\right)\right], \\ = \left[\ \underline{y}_k(x, \mathcal{Q}),\ \overline{y}_k(x, \mathcal{Q})\ \right], \tag{10}$$

where $k = 1, \ldots n_y$. The spread or width of (10) is defined as

$$S_k(\mathcal{Q}) \triangleq \mathbb{E}_x\left[\overline{y}_k\left(x, \mathcal{Q}\right) - \underline{y}_k\left(x, \mathcal{Q}\right)\right], \tag{11}$$

where $\mathbb{E}_x[\cdot]$ is the expectation with respect to the input x in X. The input distribution needed to compute (11), often prescribed by the experimentalist, renders the measured input $\hat{x}$. A figure of merit for the spread of all n_y IPMs is

$$S(\mathcal{Q}) = \sum_{k=1}^{n_y} \omega_k S_k(\mathcal{Q}), \tag{12}$$

where $\omega > 0$ are weights.

Our goal is to identify the parameter sequence $\mathcal{Q}$ that minimizes $S(\mathcal{Q})$ while satisfying $\mathcal{D}_k \subseteq I_{y_k}(y_k; x, \mathcal{Q})$ for $k = 1, \ldots n_y$. This sequence can be written as the union of n smaller subsequences, each corresponding to a datum-containing IPM, i.e.,

$$\mathcal{Q} = \bigcup_{i=1}^{n} \mathcal{Q}^{(i)}, \tag{13}$$

where $\mathcal{D}^{(i)} \subseteq I_y(y; x, \mathcal{Q}^{(i)})$. An algorithm for estimating $\mathcal{Q}^{(i)}$ from $\mathcal{D}^{(i)}$ is presented next.

Algorithm 1 (Multipoint IPM): Set $\mathcal{Q}^{(i)} \leftarrow p^\star(\mathcal{D}^{(i)})$.

1) Evaluate

$$\underline{\lambda}^\star = \max_{\substack{j=1\ldots n_w \\ k=1\ldots n_y}} \left\{\underline{y}_k(\hat{x}_j, \mathcal{Q}^{(i)}) - y_{j,k}^{(i)}\right\},$$

and

$$\overline{\lambda}^\star = \max_{\substack{j=1\ldots n_w \\ k=1\ldots n_y}} \left\{y_{j,k}^{(i)} - \overline{y}_k(\hat{x}_j, \mathcal{Q}^{(i)})\right\}.$$

2) If $\underline{\lambda}^\star \leq 0$ and $\overline{\lambda}^\star \leq 0$ stop. Otherwise, continue.
3) Denote $j^\star$ and $k^\star$ as the j and k indices corresponding to the greatest value between $\underline{\lambda}^\star$ and $\overline{\lambda}^\star$.
4) If $\underline{\lambda}^\star \geq \overline{\lambda}^\star$ solve

$$q^\star = \operatorname*{argmin}_p \left\{S(A) : y_{k^\star}(\hat{x}_{j^\star}, p) \leq y_{j^\star,k^\star}^{(i)},\ g(p) \leq 0\right\}$$

where $A = \mathcal{Q}^{(i)} \cup p$. Otherwise, solve

$$q^\star = \operatorname*{argmin}_p \left\{S(A) : y_{k^\star}(\hat{x}_{j^\star}, p) \geq y_{j^\star,k^\star}^{(i)},\ g(p) \leq 0\right\}$$

5) $\mathcal{Q}^{(i)} \leftarrow \mathcal{Q}^{(i)} \cup q^\star$. Go to Step 1.

Hence, $I_y(y; x, \mathcal{Q}^{(i)})$ is the multi-point IPM of minimal spread enclosing the datum[1] $\mathcal{D}^{(i)}$. An equivalent but simpler representation is obtained by removing the elements of $\mathcal{Q}^{(i)}$ not corresponding to any portion of the IPM boundaries in (10). The parameter points corresponding to all the elements in $\mathcal{D}$ comprise the sequence

$$\mathcal{P} \triangleq \left\{\mathcal{Q}^{(i)}\right\}_{i=1}^{n}. \tag{14}$$

Therefore, each input-output pair in (1) is mapped to a set of parameter points in (14), i.e., $\left(x^{(i)}, y^{(i)}\right) \rightarrow \mathcal{Q}^{(i)}$, whose elements can be interpreted as "surrogate data". Note that (9) is a particular case of (14).

Means to estimate the set $\mathbb{P}$ for (5) from the sequence $\mathcal{P}$ are presented next. When the additional requirements in (3) are present, the desired set is given by the intersection of any of the sets in the next section, and $\mathcal{S} = \{p : g(p) \leq 0\}$.

V. Data-enclosing Sets

Any path-connected set containing the ensemble of parameter points $\mathcal{Q} = \{p^{(i)}\}_{i=1}^{m}$ comprising $\mathcal{P}$ yields a data-enclosing IPM, i.e., if $\mathcal{Q} \subset \mathbb{P}$ then $I_y(y; x, \mathcal{Q}) \subseteq I_y(y; x, \mathbb{P})$ for all $x \in X$. The strategies below seek semi-algebraic sets that maximize the tightness of the enclosure. To this end, consider the family of sets given by $\mathbb{P}(\theta)$, where the decision variable θ is given by

$$\theta^\star(\mathcal{Q}) = \operatorname*{argmin}_\theta \left\{f\left(\mathbb{P}(\theta)\right) : \mathcal{Q} \subset \mathbb{P}(\theta)\right\}, \tag{15}$$

and $f(\mathbb{P})$ returns a value proportional to the size of $\mathbb{P}$. Hence, $\mathbb{P}(\theta^\star(\mathcal{Q}))$ is the element of the chosen class that encloses the surrogate data while having minimal size. The multi-point IPM $I_y(y; x, \mathcal{Q})$, and the IPM $I_y(y; x, \mathbb{P}(\theta^\star(\mathcal{Q})))$ often differ, with their spreads satisfying $S\left(\mathcal{Q}\right) \leq S\left(\mathbb{P}(\theta^\star(\mathcal{Q}))\right)$.

The desired set $\mathbb{P}$ will be designed to contain not only the elements of $\mathcal{Q}$ but also their neighborhood thereby making the identified model robust to uncertainty. The extent of this neighborhood should be set according to engineering judgement. Insufficiently large sets might yield to uncertain plant models that fail to accurately characterize the unknown system dynamics thereby incurring a possibly high risk, whereas the conservatism resulting from overly large sets might render unnecessarily complex controllers having a poor performance.

Formulations seeking to size optimal data-enclosing sets having various geometries are presented next. Each of the resulting sets alone, as well as their intersection, comply with the required path-connectedness property. This framework enables the analyst to adjust the level of conservatism in the identified plant model.

[1]The process of calculating $\mathcal{Q}$ in (13) can be cast in terms of the solution to the following n optimization programs

$$\mathcal{Q}^{(i)} = \operatorname*{argmin}_{\mathcal{R}} \left\{S(\mathcal{R}) : \mathcal{D}_k^{(i)} \subset I_{y_k}(y_k; x, \mathcal{R}), g(\mathcal{R}) \leq 0,\ \forall k\right\},$$

for $i = 1, \ldots n$. One could instead seek $\mathcal{Q}$ by solving the single optimization program

$$\mathcal{Q} = \operatorname*{argmin}_{\mathcal{R}} \left\{S(\mathcal{R}) : \mathcal{D}_k \subset I_{y_k}(y_k; x, \mathcal{R}),\ g(\mathcal{R}) \leq 0,\ \forall k\right\}.$$

This formulation, however, might lead to predictions that do not resemble individual scenarios, i.e., there might not be points p in $\mathcal{Q}$ for which $y^{(i)} \approx y(x^{(i)}, p)$, thereby making $\mathcal{Q}$ inadequate.

A. Polytopes

The data-enclosing box of minimum volume is

$$\mathbb{P}(\theta^\star(\mathcal{Q})) = \left\{ p : \min_i p_j^{(i)} \leq p_j \leq \max_i p_j^{(i)},\ \forall j \right\}. \quad (16)$$

Even though this set is trivial to compute, it fails to model the parameter dependencies commonly present in $\mathcal{Q}$, thereby making the identified plant model overly conservative.

Another choice is the convex hull of $\mathcal{Q}$, given by

$$\mathbb{P}(\theta^\star(\mathcal{Q})) = \left\{ \sum_{i=1}^{m} \lambda_i\, p^{(i)} : \sum_{i=1}^{m} \lambda_i = 1,\ \lambda_i \geq 0\ \ \forall i \right\}, \quad (17)$$

which is the tightest data-enclosing convex polytope. This set can be computed using the Quickhull algorithm [6]. In addition to its high computational cost, given by $\mathcal{O}(m\log(m))$ in average and $\mathcal{O}(m^2)$ in the worst-case [7], and to the possibly large number of linear inequalities often needed to represent $\mathbb{P}$, the identified plant model might be regarded as insufficiently conservative and overly sensitive to the data.

B. Sum of Squares of Polynomials

Sliced-Normal (SN) distributions enable the characterization of multivariate data as both a vector of possibly dependent random variables, or as a tight, data-enclosing semi-algebraic set. A brief summary of developments in [8] leading to the latter characterization is presented next.

Consider the polynomial mapping from physical space p to feature space z given by $z = t(p;d)$, where $t : \mathbb{R}^{n_p} \times \mathbb{N} \to \mathbb{R}^{n_z}$ is the vector of monomials in variables p of positive degree less than or equal to d, so $n_z = \binom{n_p+d}{n_p} - 1$. A Sum of Squares (SOS) of polynomials in p of degree $2d$ is

$$\phi(z;\theta) = (z-\mu)^\top P (z-\mu), \quad (18)$$

where $\mu \in \mathbb{R}^{n_z}$ and $P \in S^{n_z}_{++}$, with S^n_{++} denoting the space of symmetric positive definite matrices in $\mathbb{R}^{n\times n}$. The joint density of a SN is

$$f_p\,(p;\theta) = \begin{cases} \frac{1}{c(\theta)} \exp\left(-\frac{1}{2}\phi(t(p;d);\theta)\right) & \text{if } p \in \Delta, \\ 0 & \text{otherwise,} \end{cases} \quad (19)$$

where $\theta = \{\mu, P\}$ is the shaping parameter, Δ is the support set, and $c(\theta)$ is the normalization constant. The super-level sets of (19) are

$$\mathbb{L}\,(\theta, d, \kappa) = \left\{ p \in \Delta : \phi(t(p;d),\theta) \leq \kappa \right\}, \quad (20)$$

where $\kappa \geq 0$. The maximization of the worst-case log-likelihood of $\mathcal{Q}$, defined as $\min_i \log f_p\left(p^{(i)};\theta\right)$, yields

$$\theta^\star(\mathcal{Q}) = \operatorname*{argmax}_{\theta} \min_{i=1,\dots m} l\left(p^{(i)};\theta,d\right), \quad (21)$$

where

$$l(p;\theta,d) = -\frac{1}{2}\phi\big(t(p;d);\theta\big) - \log c(\theta), \quad (22)$$

is the log-likelihood function. A choice for the set $\mathbb{P}$ is the tightest member of (20) that encloses $\mathcal{Q}$, i.e.,

$$\mathbb{P}(\theta^\star(\mathcal{Q})) = \mathbb{L}\big(\theta^\star, d, \kappa\left(t(\mathcal{Q};d),\theta^\star\right)\big), \quad (23)$$

where

$$\kappa(\mathcal{R},\theta) \triangleq \max_i \left\{ \phi(r^{(i)};\theta) : r^{(i)} \in \mathcal{R} \right\}. \quad (24)$$

It has been observed that (23) closely approximates the semi-algebraic set of degree $2d$ having minimal volume.

The program in (21) is non-convex in general, and the corresponding data-enclosing set in (23) might not have the desired path-connectedness property. In such a case additional constraints must be added to (21), e.g., the set can be made SOS-convex. Unfortunately, the computational demands of this practice render it unsuitable when n_p is moderately large. However, when $d = 1$ and $\Delta = \mathbb{R}^{n_p}$, (21) reduces to

$$\min_{\mu, P\in S^{n_p}_{++}, \alpha\in\mathbb{R}^+} \{\alpha - \log\det(P) : \phi(\mathcal{Q};\{\mu,P\}) \leq 2\alpha\} \quad (25)$$

An equivalent semi-definite program can be used to find the solution to (25), [8].

Lemma 2: The set in (23) with $\theta^\star$ given by the program in (25) is a data-enclosing ellipsoid of minimum volume.

Proof: Performing the change of variable $R = P/(2\alpha)$ in program (25) yields

$$\min_{\mu,\ R,\ \alpha} \{\alpha - n_p\log(2\alpha) - \log\det(R) : \phi(\mathcal{Q};\{\mu,R\}) \leq 1\},$$

whose solution is given by $\alpha^\star = n_p/2$, and

$$\min_{\mu,\ R\in S^{n_z}_{++}} \{-\log\det(R) : \phi(\mathcal{Q};\{\mu,R\}) \leq 1\}.$$

This optimization program is equivalent to

$$\min_{A\in S^{n_p}_{++},\, b\in\mathbb{R}^{n_p}} \Big\{\ \log\det(A^{-1}) : \\ \|Ap^{(i)} - b\|_2 \leq 1,\ i = 1\dots m\Big\}, \quad (26)$$

with $A = R^{1/2}$ and $b = R^{1/2}\mu$. This convex optimization program yields an ellipsoid of minimal volume [9], thereby completing the proof. □

Specialized algorithms for solving (26), such as Khachiyan's Algorithm [10], are more efficient than those generally used in semi-definite programing to solve (25). Hence, these algorithms should be used to find degree-one SNs. The convexity of (26) makes it suitable when n_p is large. However, the resulting ellipsoid might not enclose the data sufficiently tightly. A strategy that mends for this deficiency is presented next.

B.1. Multi-Ellipsoidal Sets

The formulation below seeks a set $\mathbb{P}$ given by the union of n_e intersecting ellipsoids. The greater n_e the tighter the enclosure of the data, and therefore the smaller the corresponding IPM's spread. To this end, consider the functions

$$s_j(p;\beta) = \begin{cases} 1 & \text{if } j = \operatorname*{argmin}_k \phi\left(p;\{u^{(k)},N\}\right) \\ 0 & \text{otherwise,} \end{cases} \quad (27)$$

with $j = 1,\dots n_e$, where the function ϕ is in (18), and the parameter $\beta = \{\mathcal{C}, N\}$ is comprised of $\mathcal{C} = \{u^{(k)}\}_{k=1}^{n_e}$ with $u^{(k)} \in \mathbb{R}^{n_p}$, and $N \in S^{n_p}_{++}$. The functions in (27) act as

a n_e-classifier for the m elements of $\mathcal{Q}$. In particular, (27) defines a Voronoi diagram in which the parameter space is partitioned into the n_e convex polytopes

$$\gamma_j(\beta) = \{p \in \Delta : s_j(p;\beta) = 1\},$$

with $j = 1, \dots n_e$. The members of this partition will be ordered such that γ_j shares a facet with γ_{j+1}. The corresponding partition of the data is $\{\mathcal{Q}_j\}_{j=1}^{n_e}$, where

$$\mathcal{Q}_j(\beta) = \{p : p \in (\mathcal{Q} \cap \gamma_j(\beta))\}.$$

Note that this is a k-means clustering algorithm, where $k = n_e$ and β is the parameterizing variable.

The data-enclosing set sought is based on the solution to

$$\min_{\theta_j,\,\beta,\,\mathcal{V}} \left\{ \sum_{j=1}^{n_e} \frac{\kappa\Big(\mathcal{U}_j\big(\mathcal{Q}_j(\beta),\mathcal{V}\big),\theta_j\Big)^{n_p}}{\det(P_j)} : g(\mathcal{V}) \le 0 \right\}, \quad (28)$$

where $\theta_j = \{\mu_j, P_j\}$ are the parameters of an ellipsoid, $\mathcal{V} = \{v^{(k)}\}_{k=1}^{n_e-1}$ with $v^{(k)} \in \mathbb{R}^{n_p}$ is a sequence of intersection points, the functions κ and g are given in (24) and (3) respectively, and

$$\mathcal{U}_j(\mathcal{Q},\mathcal{V}) = \mathcal{Q} \cup \left\{ v^{(\max(j,\,n_e-1))}, v^{(\min(j,\,n_e-1))} \right\}. \quad (29)$$

The rationale of program (28), whose solution will be denoted as $\theta_j^\star$, $\beta^\star$, and $\mathcal{V}^\star$, is explained next. The objective function is the sum of the squares of the volume of n_e ellipsoids. At the optimum these ellipsoids are given by

$$\mathbb{L}_j = \mathbb{L}\Big(\theta_j^\star, 1, \kappa\left(\mathcal{U}_j(\mathcal{Q}_j(\beta^\star),\mathcal{V}^\star),\theta_j^\star\right)\Big), \quad (30)$$

for $j = 1, \dots n_e$, where $\mathbb{L}$ is in (20). The ellipsoid $\mathbb{L}_j$ contains the elements of $\mathcal{Q}_j(\beta^\star)$ as well as the elements of $\mathcal{V}$ corresponding to the intersection between $\mathbb{L}_j$ and its neighboring ellipsoids, e.g., $v^{(1)}$ is in the intersection between $\mathbb{L}_1$ and $\mathbb{L}_2$, $v^{(2)}$ is in the intersection between $\mathbb{L}_2$ and $\mathbb{L}_3$, etc. These parameter points comprise $\mathcal{U}_j$ in (29). The non-empty intersection between $\mathcal{U}_j$ and $\mathcal{U}_{j+1}$ for $j = 1, \dots n_e - 1$ ensures that the union of the ellipsoids in (30) is path-connected. Lastly, the constraint in (28) guarantees that at least one of the points where each pair of ellipsoids intersect complies with the requirements introduced in (3).

The parameter set $\mathbb{P}$ to be used by the IPM is

$$\mathbb{P}(\theta^\star(\mathcal{Q})) = \bigcup_{j=1}^{n_e} \mathbb{L}_j, \quad (31)$$

where $\mathbb{L}_j$ is given in (30). Figure 1 shows data-enclosing sets resulting from (28) for $n_e = 1$ and $n_e = 2$ for a data cloud in $n_p = 2$ dimensions.

The optimization program in (28) is non-convex and NP-hard, thereby restricting its efficient application to moderately large n_p values. A suboptimal approximation corresponding to a fixed partition of the parameter space can be found by solving the sequence of convex optimization programs detailed below. This partition, which is fully prescribed by β, can be found by using any clustering algorithm. The efficiency of this approach makes it suitable when n_p is large.

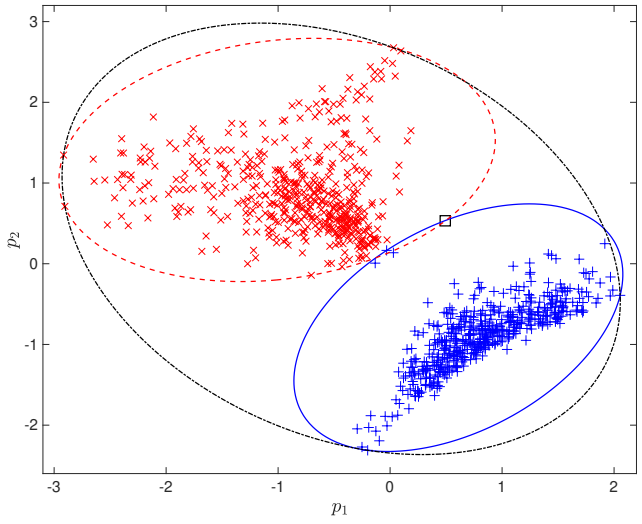

Fig. 1. Data-enclosing ellipsoids for $n_e = 1$ (black) and $n_e = 2$ (red and blue). The data cloud $\mathcal{Q}$ is divided into $\mathcal{Q}_1(\beta^\star)$ ($\times$) and $\mathcal{Q}_2(\beta^\star)$ ($+$). The only element of $\mathcal{V}$ required in the latter case is shown as a square.

Algorithm 2 (Suboptimal multi-ellipsoidal set): Assume that the parameter space has been partitioned onto n_e subsets, that they are ordered such that consecutive elements are adjacent, and that $\mathcal{Q}_j$ contains the elements of $\mathcal{Q}$ falling onto the j-th subset.

1) Use (26) to solve $\theta_j^\star(\mathcal{Q}_j) = \{\mu_j^\star, P_j^\star\}$ for $j = 1, \dots n_e$.
2) Compute the symmetric matrix J, with

$$J_{ij} = \frac{\kappa(p_{ij},\theta_i^\star)^{n_p}}{\det(P_i^\star)} + \frac{\kappa(p_{ij},\theta_j^\star)^{n_p}}{\det(P_j^\star)},$$

for $i = 1, \dots n_e$ and $j = 1, \dots n_e$, where κ is in (24),

$$p_{ij} = \frac{q_{ij}V_i + q_{ji}V_j}{V_i + V_j}, \quad (32)$$

is an intersection point, $V_k = \kappa(\mathcal{Q}_k,\theta_k^\star)^{n_p}/\det(P_k^\star)$ is the volume of the ellipsoid enclosing $\mathcal{Q}_j$, and

$$q_{kl} = \operatorname*{argmin}_{p\in\Delta} \{\phi(p;\theta_k^\star) : \phi(p;\theta_l^\star) \le \kappa(\mathcal{Q}_l,\theta_l^\star)\}.$$

with k and l being either i or j.
3) Generate all the permutations of n_e elements having unique forward and backward orders[2]. Evaluate the cost of each of these permutations by summing the components of J for all overlapping pairs[3].
4) Find the permutation with the lowest cost. Expand the data partition by including the intersection points p_{ij} corresponding to all the pairs in this permutation[4]. Denote the resulting partition $\{\mathcal{Q}_j^\star\}_{j=1}^{n_e}$.
5) Repeat Step 1 using $\mathcal{Q}_j^\star$, and compute the corresponding data-enclosing ellipsoids $\mathbb{L}_j$ using (23).

[2]e.g., all the permutations of interest for $n_e = 3$ are $s_1 = [1,2,3]$, $s_2 = [1,3,2]$, and $s_3 = [2,1,3]$. Note that $[2,3,1]$ is excluded because it coincides with the permutation that results from inverting the order of s_2.

[3]e.g., the cost of $[1,2,3]$ is $J_{12} + J_{23}$.

[4]e.g., the expanded partition for $[1,2,3]$ is $\mathcal{Q}_1^\star = \{\mathcal{Q}_1, p_{12}\}$, $\mathcal{Q}_2^\star = \{\mathcal{Q}_2, p_{12}, p_{23}\}$ and $\mathcal{Q}_3^\star = \{\mathcal{Q}_3, p_{23}\}$.

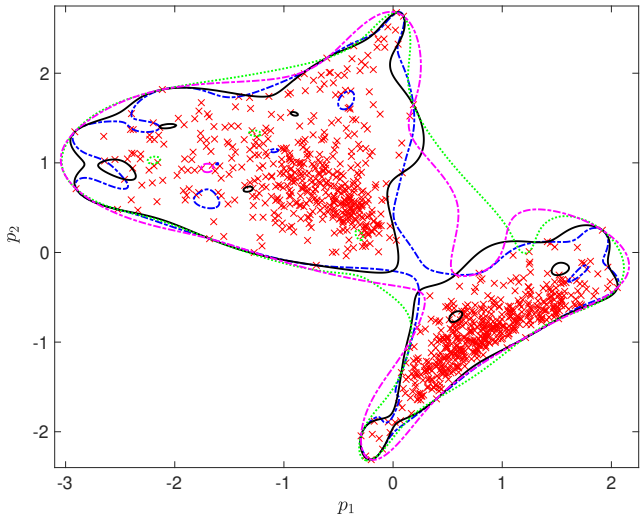

Fig. 2. Data-enclosing sets corresponding to $d = 3$ (magenta/dashed), $d = 4$ (green/dotted), $d = 5$ (black/solid), and $d = 6$ (blue/dashed-dotted). The data points are shown as ($\times$).

Step 1 finds the ellipsoids that enclose each subset of the data. Step 2 evaluates the volume of pairs of ellipsoids each enclosing its corresponding data subsets while having a non-empty intersection. The parameter point in (32), whose calculation requires solving a pair of convex optimization programs, is in this intersection. Step 3 determines the sequence of intersecting ellipsoids having the smallest volume. Step 4 adds elements to the subsets of the data in order to guarantee that the resulting union of ellipsoids is path-connected. Finally, Step 5 calculates the desired ellipsoids. The parameter set to be used by the IPM is given by (31).

B.2. A Convex Formulation for SOS of Higher Degree

SNs that maximize the worst-case likelihood of the data in feature space [8], as given by (21) with $c(\theta) = \sqrt{(2\pi)^{n_z} \det(P^{-1})}$, render a convex optimization program for any polynomial degree d. As before, specialized algorithms for minimizing the volume of data-enclosing ellipsoids are better suited than standard semi-definite programing algorithms to their estimation. Even though the resulting data-enclosing sets in (23) are simpler and possibly tighter than those in (31), they might not be path-connected, thereby precluding their usage in (5).

Figure 2 shows data-enclosing sets of degrees three, four, five, and six for the same surrogate data used in Figure 1. The computational costs of calculating such sets using SDPT3 [11] are 7.7s, 12.9s, 29.5s and 37.6s, whereas Khachiyan's Algorithm takes 4.5s, 3.8s, 3.0s and 2.7s respectively. The significant savings of Khachiyan's algorithm increase rapidly with both n_p and d. Note that the volume of the sets do not decrease monotonically with d, and the set corresponding to $d = 5$ is the only set which is not path-connected. All of these sets enclose the data more tightly than the two ellipsoids shown in Figure 1.

The inability to systematic enforce path-connectedness in the resulting set prevents its usage in (5). Path-connectedness can be evaluated by using Bernstein polynomials. However, the high computational cost of this practice renders it unsuitable when n_p is moderately large.

VI. Examples

The robust system identification of a single-input single-output weakly nonlinear system having a non-collocated sensor actuator pair is considered next. The underlying physical system consists of a spring-mass-damper configuration with 3 degrees of freedom. In this setting, each measured input corresponds to $n_w = 2500$ frequency points in the $X = [40, 500]$ rad/s range, and each measured output consists of the corresponding magnitude and phase. We seek to identify a robustly stable plant model with transfer function

$$H(s,p) = \frac{p_1 s^4 + p_2 s^3 + p_3 s^2 + p_4 s + p_5}{s^6 + p_6 s^5 + p_7 s^4 + p_8 s^3 + p_9 s^2 + p_{10} s + p_{11}},$$

according to data sequences $\mathcal{D}$ of different lengths.

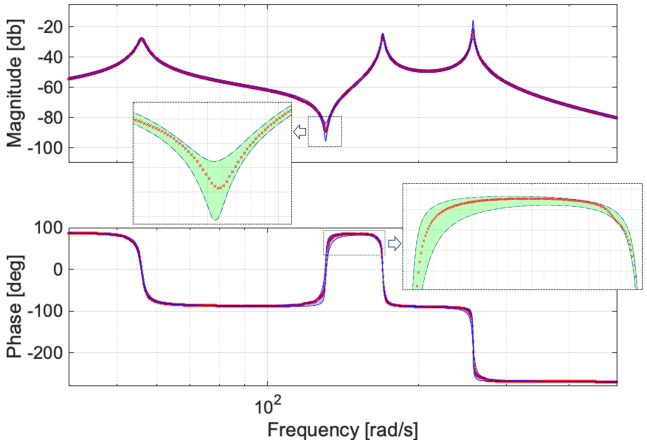

Fig. 3. The data sequence $\mathcal{D}^{(1)}$ ($\times$), the multi-point IPM $I_y(y; x, \mathcal{Q}^{(1)})$ (green), and the multi-point IPM corresponding to member functions of $I_y(y; x, \mathbb{P}(\theta^\star(\mathcal{Q}^{(1)})))$ (blue).

Example 1: IPM for a Single Input-Output Pair

Figure 3 shows the data sequence $\mathcal{D} = \mathcal{D}^{(1)}$ used herein to identify the plant. First, Algorithm 1 was used to obtain $\mathcal{Q}^{(1)}$. The corresponding multi-point IPMs, shown in green, are comprised of 32 FRFs. A data-enclosing ellipsoidal set $\mathbb{P}(\theta^\star(\mathcal{Q}^{(1)}))$ was then calculated based on (26). The resulting IPM, $I_y(y; x, \mathbb{P}(\theta^\star(\mathcal{Q}^{(1)})))$, was simulated by using $n_s = 10000$ uniformly distributed samples in $\mathbb{P}(\theta^\star(\mathcal{Q}^{(1)}))$. The corresponding input-output functions lead to the multi-point IPM shown in blue. Both IPMs enclose the datum tightly while having practically equal spreads. This will not be the case for deficient plant model structures.

Example 2: IPM for Multiple Input-Output Pairs

Next we identify a plant model according to a data sequence with $n = 100$ elements. Figure 4 shows the upper and lower envelopes of the data in red. The spread in the FRFs might be caused by variability in the material properties of an ensemble of test articles or by the effects of model form uncertainty on a single test article, e.g., the

power of the excitation in a modal experiment of a nonlinear structure changes the identified FRFs. The application of Algorithm 1 to each of the $n = 100$ data points yields a set $\mathcal{Q}$ with 4135 parameter points in $n_p = 10$ dimensions. The dispersion of these points, shown in red in Figure 5, indicate that they should be enclosed by a set that characterizes dependencies, thereby making (16) particularly unsuitable. The multi-point IPMs corresponding to $\mathcal{Q}$, shown in green in Figure 4, enclose the input-output data tightly.

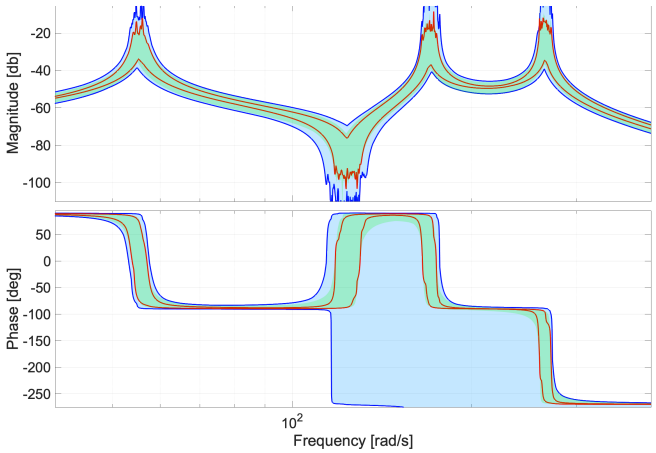

Fig. 4. Data envelopes (red), $I_y(y; x, \mathcal{Q})$ (green), and multi-point IPM corresponding to 5000 member functions of $I_y(y; x, \mathbb{P}(\theta^\star(\mathcal{Q})))$ (blue).

The set $\mathbb{P}$ to be used in (5) is given by the intersection of the ellipsoid in (23), the hyper-rectangle in (16), and the set of asymptotically stable plants. Figure 5 illustrates the tightness of the data enclosure by superimposing $n_s = 5000$ uniformly distributed points in $\mathbb{P}$. In contrast to the chosen set, the convex hull in (17) could not even be computed in finite time. The multi-point IPM corresponding to such points is shown in blue in Figure 4. The magnitude IPM encloses the data tightly, whereas the phase IPM does not. Figure 6 shows the FRFs leading to the phase IPM. Note that some of the FRFs depart from the data at the first natural frequency. This phenomenon is not resolved by wrapping the phase angle. The large spread in the phase might render $\mathbb{P}$ unsuitable.

A means to assess the spread of IPMs is presented next. The deviation of any function in $I_y(y; x, \mathbb{P})$, $y(x, p)$, from $I_y(y; x, \mathcal{Q})$ is measured by

$$\lambda_k(p) = \frac{S_k(\mathcal{Q} \cup p) - S_k(\mathcal{Q})}{S_k(\mathcal{Q})}, \tag{33}$$

where $p \in \mathbb{P}$, and $k = 1, \dots n_y$. The metric $\lambda_k(p)$ is the relative increase in the spread of the k-th IPM that results from adding p to $\mathcal{Q}$. Thus, points for which $\lambda_k(p) = 0$ yield FRFs that are fully contained by the IPM. The evaluation of (33) at uniformly distributed points in $\mathbb{P}$ led to values of λ_1 and λ_2 that vary in the $[0, 0.035]$ and $[0, 10.5]$ ranges respectively. The upper limit of these intervals are proportional to the spreads seen in Figure 4. However, the cumulative distribution of λ_2 shows that more than 98% of its values are less than 0.05. Therefore, less than 2% of the

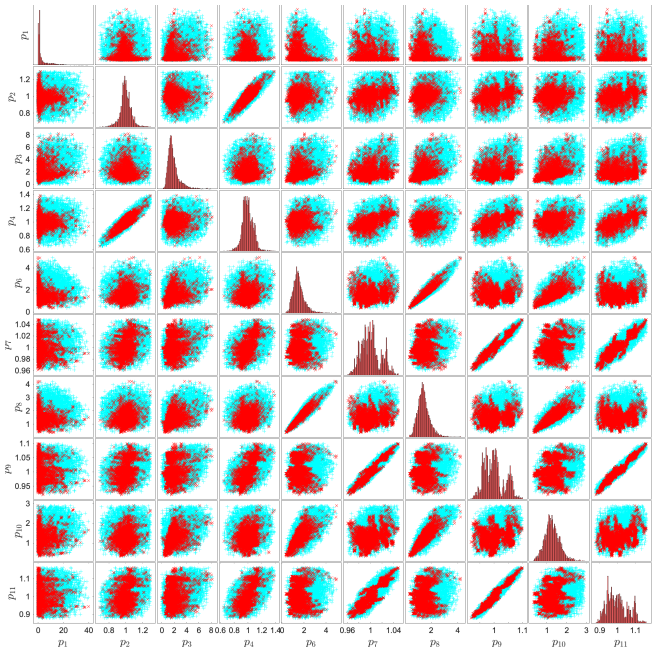

Fig. 5. Elements of $\mathcal{Q}$ ($\times$) and samples of $\mathbb{P}(\theta^\star(\mathcal{Q}))$ (+). Diagonal subplots show the marginal densities whereas off-diagonal subplots show projections over 2-dimensional subspaces. p_5 has been omitted since it takes on a constant value. Values have been normalized to facilitate visualization.

volume of $\mathbb{P}$ causes the large spread in phase seen. This is caused by members of $\mathbb{P}$ leading to $\|y(x, p_1) - y(x, p_2)\| \gg 1$ even though $\|p_1 - p_2\| \approx 0$. Tighter data enclosures are obtained by increasing n_e, by further constraining $\mathbb{P}$, e.g., using $g(p) = \lambda_2(p) - 0.05 < 0$, or by eliminating some of the worse-performing elements of $\mathcal{P}$. These practices, however, will be presented elsewhere due to space limitations.

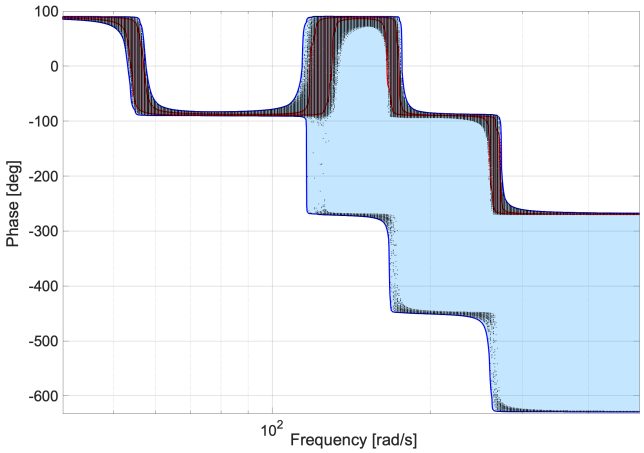

Fig. 6. Data envelopes (red), member functions of $I_y(y; x, \mathbb{P}(\theta^\star(\mathcal{Q})))$ (black dots), and the corresponding multi-point IPM (blue).

VII. Reliability Analysis

This section presents means to formally assess the generalization properties of the identified model as given by its ability to characterize unseen data drawn from the same process from which the training data in $\mathcal{D}$ was obtained.

A. Background

Let $J : \Theta \to \mathbb{R}$ be a cost function of the decision variable $\theta \subset \mathbb{R}^{n_\theta}$, and h_δ be the set of decision points satisfying the design requirements for scenario $\delta \in \Delta$. Consider the scenario program

$$\theta^\star(\mathcal{P}) = \underset{\theta\in\Theta}{\operatorname{argmin}} \left\{ J(\theta) : \theta \in h_{\delta^{(i)}} \text{ for } i = 1, \dots n \right\}, \quad (34)$$

where $\mathcal{D} = \{\delta^{(i)}\}_{i=1}^n$ is drawn from a stationary but otherwise unknown process. Denote by P the unknown probability measure governing the underlying process. The violation probability of θ is defined as $V(\theta) \triangleq \mathsf{P}\left[\delta \in \Delta \mid \theta \notin h_\delta\right]$. The randomness of choosing a particular data sequence $\mathcal{D}$ out of the infinitely many having n scenarios makes $\theta^\star$, thus $V(\theta^\star)$, random. This notion can be quantified by using

$$\mathsf{P}^n \left[V(\theta^\star) \leq \epsilon\right] \geq 1 - \beta. \quad (35)$$

This equation states that the probability of $\theta^\star$ violating a requirement for future data is no greater than $\epsilon \in (0, 1)$ with probability $1-\beta$. The term ϵ is called the reliability parameter whereas β is called the confidence. Scenario theory enables evaluating (35) without making any assumption on P.

When (34) is convex and its solution is unique, ϵ can be obtained from

$$\binom{k + n_\theta - 1}{k} \sum_{i=0}^{k+n_\theta-1} \binom{n}{i} \epsilon^i (1-\epsilon)^{n-i} \leq \beta, \quad (36)$$

where $k < n - n_\theta$ is the number of scenarios one might choose to remove from $\mathcal{D}$ before $\theta^\star$ was calculated. If (34) is also non-degenerate, and $k = 0$, a tighter bound can be obtained [3]. This bound is given by $\epsilon = 1 - \hat{t}$, where $\hat{t}$ is the zero of the polynomial in $t \in [0, 1]$,

$$\frac{\beta}{n+1} \sum_{i=\ell}^{n} \binom{i}{\ell} t^{i-\ell} - \binom{n}{\ell} t^{n-\ell} = 0, \quad (37)$$

and $0 \leq \ell \leq n_\theta$ is the number of support constraints. The support constraints of (34) correspond to a sub-sample $\mathcal{S} \subset \mathcal{D}$ for which $\theta^\star(\mathcal{D}) = \theta^\star(\mathcal{S})$.

When (34) is non-convex ϵ can be obtained from

$$\epsilon(\ell, n, \beta) = \begin{cases} 1 & \text{if } \ell = n, \\ 1 - \left(\frac{\beta}{n\binom{n}{\ell}}\right)^{\frac{1}{n-\ell}} & \text{otherwise.} \end{cases} \quad (38)$$

where $0 \leq \ell \leq n$. Non-convex programs might admit several irreducible support sets, with the one having minimal cardinality yielding the tightest bound ϵ.

B. Reliability of IPMs

The reliability analysis of (5) entails bounding the probability of future data falling outside $I_y(y; x, \mathbb{P}(\theta^\star(\mathcal{Q})))$. The means needed to bound this probability depend upon the process used to compute the value of θ prescribing $\mathbb{P}$. This process requires applying Algorithm 1 to each of the n elements of $\mathcal{D}$ to obtain the $\mathcal{Q}^{(i)}$s, forming the sequence of surrogate data $\mathcal{P}$, and then finding a set $\mathbb{P}$ that optimally encloses it. The first step may be regarded as finding a non-convex map between each input-output datum, and a set of parameter points. Because this mapping is fixed, a scenario can be interpreted as either an element of $\mathcal{D}$ in (1), or as an element of $\mathcal{P}$ in (14). As such, the convexity of the scenario program used to compute $\theta^\star$ from $\mathcal{P}$ determines whether convex or non-convex scenario theory are applicable. Therefore, (36) and (37) apply to (26), whereas (38) applies to (31) and (23) for $d > 1$. Thus, (35) is a probabilistic statement on $\mathcal{Q}^{(n+1)} \in \mathbb{P}$. On the other hand, $\mathcal{Q}^{(n+1)} \in \mathbb{P}$ implies $y^{(n+1)}(x^{(n+1)}) \in I_y(y; x, \mathbb{P})$. Hence, the mapping $\left(x^{(i)}, y^{(i)}\right) \to \mathcal{Q}^{(i)}$ is immaterial from a reliability standpoint.

Evaluating ℓ entails calculating $\theta^\star$ for clouds of parameter points corresponding to subsets of $\mathcal{P}$. It is noteworthy that this process does not require recalculating the $\mathcal{Q}^{(i)}$s. Further notice that the required value of ℓ is generally different than the value of ℓ corresponding to the scenario program having the enclosure of each parameter point as a constraint, e.g., the $2n_p$ parameter points supporting an optimal data-enclosing ellipsoid might correspond to the same input-output datum.

Example 3: Reliability of an IPM

The reliability of the IPM in Example 2 is considered next. In this case ℓ can be calculated by solving n times the convex program (26), each time excluding an element of (14). In particular, the number of support scenarios is the number of times $\theta^\star(\mathcal{Q} \setminus \mathcal{Q}^{(i)})$ for $i = 1, \dots n$ differ from $\theta^\star(\mathcal{Q})$. This practice yields the number of support scenarios $\ell = 21$, which along with $n = 100$, $\beta = 1\text{e}{-3}$ and Equation (37) yield $\epsilon = 0.4010$. Therefore, the probability of an input-output pair falling outside the predicted intervals, $V(\theta^\star)$, is no greater than 0.4010 with confidence $1 - \beta = 0.999$.

References

[1] L. Crespo, S. Kenny, and D. Giesy, "Interval predictor models with a linear parameter dependency," *ASME Journal of verification, validation and uncertainty quantification*, vol. 1, no. 2, pp. 1–10, 2016.

[2] L. Crespo, D. Giesy, S. Kenny, and J. Deride, "A scenario optimization approach to system identification with reliability guarantees," *American Control Conference*, 2019.

[3] M. Campi and A. Care, "Wait-and-judge scenario optimization," *Mathematical Programming*, vol. 167, no. 1, 2018.

[4] M. Campi, S. Garatti, and F. Ramponi, "A non-convex scenario theory," *IEEE Transactions in Automatic Control*, vol. 63, no. 12, 2018.

[5] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, New Jersey, USA: Prentice Hall, 1999.

[6] C. B. Barber, D. P. Dobkin, and H. T. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469–483, 1996.

[7] D. Avis, D. Bremner, and R. Seidel, "How good are convex hull algorithms?" *Computational Geometry*, vol. 7, no. 5, pp. 265–301, 1997, 11th ACM Symposium on Computational Geometry.

[8] L. Crespo, B. Colbert, S. Kenny, and D. Giesy, "On the quantification of aleatory and epistemic uncertainty using sliced-normal distributions," *Systems & Control Letters*, vol. 134 (104560), 2019.

[9] N. Moshtagh, "Minimum volume enclosing ellipsoid," *Convex optimization*, vol. 111, no. 1 (January), pp. 1–9, 2005.

[10] M. J. Todd and E. A. Yildirim, "On khachiyan's algorithm for the computation of minimum-volume enclosing ellipsoids," *Discrete Applied Mathematics*, vol. 155, no. 13, pp. 1731–1744, 2007.

[11] K. Toh, M. Todd, and R. Tutuncu, "SDPT3 - a matlab software package for semidefinite programming," *Optimization Methods and Software*, 1999.