Error-Based mesh selection for efficient numerical simulations with variable parameters

Hugo Dornier, Olivier P Le Maître, Pietro Marco Congedo, Itham Salah El Din, Julien Marty, Sébastien Bourasseau

Computers and Fluids 313 (2026) 107081

Contents lists available at ScienceDirect

Computers and Fluids

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

Error-based mesh selection for efficient numerical simulations with variable parameters

Hugo Dornier [a,b,*], Olivier P. Le Maître [c], Pietro M. Congedo [b], Itham Salah el Din [a], Julien Marty [a], Sébastien Bourasseau [d]

[a] DAAA, ONERA, Institut Polytechnique de Paris, 92190 Meudon, France
[b] Inria, Centre de Mathématiques Appliquées, Ecole Polytechnique, IPP, Route de Saclay, 91128 Palaiseau Cedex, France
[c] CNRS, Centre de Mathématiques Appliquées, Ecole Polytechnique, IPP, Route de Saclay, 91128 Palaiseau Cedex, France
[d] DAAA, ONERA, Institut Polytechnique de Paris, 92322 Châtillon, France

ARTICLE INFO

Keywords:
Mesh adaptation
Error estimates
Variable parameters

ABSTRACT

Advanced numerical simulations often rely on mesh refinement techniques to control discretization errors while limiting computational cost. In parametric studies, however, repeatedly adapting the mesh for each operating condition can lead to significant overhead. To address this issue, we introduce the Error-based Mesh Selection (EMS) method, and its practical variant approximate EMS (A-EMS), which selects, for any given condition, a mesh from a precomputed library expected to minimize the discretization error.

The method relies on Gaussian process models of restriction errors to predict, without additional model evaluations, the error associated with each library mesh. While EMS is formulated in a general setting, A-EMSrequires suitable projection operators and error estimators, which may limit its applicability depending on the discretization framework.

The approach is assessed on an analytical oblique shock problem and on a supersonic scramjet configuration. In both cases, A-EMSachieves mesh selections close to the optimal choice, significantly outperforming distance-based strategies. In the scramjet test case, the average computational time per evaluation is reduced by a factor of approximately 6 compared to systematic mesh adaptation, while maintaining comparable accuracy. A library of about 32 meshes is sufficient to achieve the prescribed accuracy.

Accounting for the cost of constructing the mesh library, the method becomes advantageous after a few tens to a few hundred evaluations, depending on the library size. These results demonstrate that A-EMSis particularly well suited for large parametric studies, such as uncertainty quantification, where the number of evaluations is high.

1. Introduction

Numerical simulations based on nonlinear physical models aim to provide accurate predictions by solving discretized governing equations. For steady problems, the quality of the numerical solution primarily depends on the spatial discretization of the computational domain. Resolving small-scale solution features typically requires locally refined meshes, while uniform refinement is computationally prohibitive. Mesh adaptation is therefore essential, yet the lack of a priori knowledge of the solution structure prevents prescribing an appropriate mesh beforehand.

Adaptive Mesh Refinement (AMR [1–3]) addresses this issue by iteratively adjusting the mesh resolution according to solution scales estimated on successive meshes. Combined with robust error estimators and refinement criteria, AMR enables the construction of meshes that meet accuracy requirements at controlled computational cost, and has been successfully applied to a wide range of complex problems [4,5]. However, AMR incurs additional cost due to repeated solves, error evaluations, and mesh generation steps. This overhead becomes substantial in parametric, optimization, or sensitivity studies where many conditions must be evaluated [6–8]. For AMR methods, efficient strategies have been proposed to reduce the cost of each individual adaptation [9,10], without eliminating the need to iteratively adapt every mesh. To alleviate this burden, one would ideally employ a single mesh across all conditions. A common approach is to adapt the mesh to a nominal condition (Nominal Mesh Adaptation, NMA), but when the solution is highly parameter-dependent, this may lead

* Corresponding author at: DAAA, ONERA, Institut Polytechnique de Paris, 92190 Meudon, France.
E-mail address: hugo.dornier@onera.fr (H. Dornier).

https://doi.org/10.1016/j.compfluid.2026.107081
Received 21 January 2026; Received in revised form 27 March 2026; Accepted 7 April 2026
Available online 8 April 2026

to large errors compared with systematic adaptation [11,12]. Adjoint-based enrichments [13] partially extend the validity of nominal meshes but remain fundamentally local and cannot guarantee error control when solution structures vary markedly with the parameters.

Alternative strategies aim at constructing a single mesh representative of several conditions. In [14], local mesh metrics obtained from independently adapted meshes were combined to generate a unified mesh. This approach is more robust than nominal adaptation, though its performance depends on the choice of construction conditions. More recently, Mean Mesh Adaptation (MMA [15]) was introduced to control the average error over a continuous parameter set via Monte Carlo estimates. MMA performs well when refinement remains confined to limited regions of the domain; however, when solution features shift widely across the domain, the method yields near-uniform meshes and becomes less efficient than systematic AMR. Overall, methods relying on a single mesh are inherently restricted to problems with relatively stable solution structures.

The present work adopts a different perspective by considering multiple meshes, each adapted to a specific condition. These adapted meshes, together with their solutions and parameter values, form a library assumed to be given. We propose an Error-based Mesh Selection (EMS) method called Approximate EMS or A-EMS that, for any new condition, selects *a priori* the mesh from this library expected to yield the smallest error.

EMS is inspired by quantization theory [16,17], which maps high-dimensional input data onto a finite codebook. Specifically, EMS parallels vector quantization techniques that partition a domain by minimizing a metric, as in clustering methods (e.g., k-means). Here, the mapping assigns each condition in a continuous parameter space to one mesh in the library while minimizing the associated discretization error. Assuming the availability of projection and interpolation operators between the approximation spaces, as well as a computable *a posteriori* error estimator, A-EMS constructs error models for each mesh directly from the library. A key advantage is that no additional flow solutions are required during the selection phase.

The remainder of the paper is organized as follows. Section 2 reviews the discretization framework, the class of problems considered, and the mesh adaptation strategies relevant to this work. Section 3 introduces the EMS formulation and the proposed approximation (A-EMS) strategy. Section 4 presents an analytical computational fluid dynamics (CFD) test case involving a supersonic oblique shock with exact solution available to finely investigate the method's performance. Section 5 extends the study to a realistic supersonic scramjet configuration [18], demonstrating the robustness and practical relevance of the approach.

2. Background materials

The EMS method proposed below is largely independent of the specific numerical schemes and adaptation strategies employed to obtain the discrete solutions and generate the adapted meshes. In this section, we state the general requirements for these methods to be compatible with EMS and introduce the notations used throughout the article.

2.1. Mesh and discrete solution

We consider a fixed domain $\Omega \subset \mathbb{R}^d$, $d \geq 1$, with boundary $\partial\Omega$. Let $u(x) \in \mathcal{V}(\Omega)$ denote the unique solution of the problem, which satisfies the equations

$$\mathscr{P}(u) = 0, \tag{1}$$

Here, the operator $\mathscr{P}$ encompasses both the governing equations and the boundary conditions. The solution space $\mathcal{V}(\Omega)$ depends on the particular problem and is assumed to be equipped with a norm $\|\cdot\|_{\mathcal{V}}$. In what follows, the dependence of $\mathcal{V}$ on Ω is omitted for convenience.

To solve (1), we employ a discretization of Ω represented by a mesh $\mathcal{H}$. Without loss of generality, $\mathcal{H}$ is defined as a collection of non-overlapping computational elements c_i that partition Ω:

$$\Omega = \overline{\bigcup_{i\in\mathcal{H}} c_i}, \quad \forall (c_i, c_j) \in \mathcal{H}^2 \; c_i \cap c_j = \emptyset \text{ for } i \neq j.$$

This definition encompasses, for instance, the finite-volume (FVM) and finite-element (FEM).[1] As is standard in the literature, $h(\mathcal{H})$ denotes the size of the largest computational element, $h(\mathcal{H}) = \max_{c_i\in\mathcal{H}} |c_i|$, while $|\mathcal{H}|$ denotes the number of computational elements in the mesh. We introduce the approximation space $\mathcal{V}_\mathcal{H} \subset \mathcal{V}(\Omega)$ constructed on the mesh, and denote by $u_\mathcal{H} \in \mathcal{V}_\mathcal{H}$ the discrete solution. The dimension of the approximation space, $\dim(\mathcal{V}_\mathcal{H})$, is typically larger than $|\mathcal{H}|$, even for scalar problems.

We assume that the discrete problem is well-posed with a unique solution, and that $\|u_\mathcal{H} - u\|_\mathcal{V} \to 0$ as $h(\mathcal{H}) \to 0$.

In practice, it is neither possible nor desirable to let $h(\mathcal{H})$ tend to zero to achieve a sufficiently accurate solution. A more effective strategy is to control the size of individual computational elements to attain a prescribed accuracy with minimal computational effort. For the discussion that follows, we assume that the objective is to control the approximation error measured in the $L_2(\Omega)$ norm:

$$\varepsilon_\mathcal{H}^2 = \|u - u_\mathcal{H}\|_{L_2(\Omega)}^2 = \sum_{c_i\in\mathcal{H}} \|u - u_\mathcal{H}\|_{L_2(c_i)}^2. \tag{2}$$

However, error control based on other error definitions may also be considered. For instance, in goal-oriented adaptation, one may wish to control the error,

$$\varepsilon_\mathcal{H} = \left|\Phi(u) - \Phi(u_\mathcal{H})\right|,$$

where $\Phi : \mathcal{V} \mapsto \mathbb{R}$ is a quantity of interest.

2.2. Adaptive mesh refinement (AMR)

Mesh adaptation aims to determine a mesh $\mathcal{H}$ that (approximately) minimizes the mesh size $|\mathcal{H}|$ while satisfying the error condition $\varepsilon_\mathcal{H} \leq \varepsilon_{\text{ad}}$, where $\varepsilon_{\text{ad}} > 0$ is a prescribed target error. Here, we assume that the computational cost of obtaining $u_\mathcal{H}$ increases monotonically with $\dim(\mathcal{V}_\mathcal{H})$, which is proportional to $|\mathcal{H}|$.

In AMR methods, a sequence of refined meshes $\mathcal{H}^{(0)}, \mathcal{H}^{(1)}, \ldots$ is constructed until the error criterion is satisfied. In the simplest algorithms, the current mesh $\mathcal{H}^{(k)}$ is refined to $\mathcal{H}^{(k+1)}$ by subdividing selected computational elements, so that $\mathcal{V}_{\mathcal{H}^{(k)}} \subset \mathcal{V}_{\mathcal{H}^{(k+1)}}$. A criterion is required to prioritize the elements that need refinement and to avoid refining elements c_i that already contribute little to the error. A common choice is to select for refinement a subset comprising a fraction α_{ad} of the computational elements in $\mathcal{H}^{(k)}$, with $0 < \alpha_{\text{ad}} < 1$, based on the local error contributions.

For instance, in the case of the L_2 error norm in (2) and a scalar solution u, the decomposition reads

$$\varepsilon_{\mathcal{H}^{(k)}}^2 = \sum_{c_i\in\mathcal{H}^{(k)}} \varepsilon_{\mathcal{H}^{(k)}}^{\text{loc}}(c_i)^2, \quad \varepsilon_{\mathcal{H}^{(k)}}^{\text{loc}}(c_i)^2 = \|u - u_{\mathcal{H}^{(k)}}\|_{L_2(c_i)}^2 = \int_{c_i} |u - u_{\mathcal{H}^{(k)}}|^2 dx. \tag{3}$$

In practice, the exact solution u is unknown, and estimators $\widehat{\varepsilon}_\mathcal{H}$ of $\varepsilon_\mathcal{H}$ are introduced, for instance using *a posteriori* error estimation [2,19–21]. In the case of goal-oriented adaptation, an additional adjoint problem may need to be solved to estimate the contribution of each computational element to the error.

Once the fraction α_{ad} of computational elements carrying the highest estimated error $\widehat{\varepsilon}_{\mathcal{H}^{(k)}}$ is determined, the next mesh $\mathcal{H}^{(k+1)}$ is generated through a refinement procedure. Various types of refinement can be

[1] The proposed method could also be applied to other discretization methods, *e.g.* finite-difference and discontinuous Galerkin, provided that error estimators are available.

employed. One approach is to subdivide the selected elements c_i, possibly including some neighboring elements to maintain a smooth variation of cell sizes $|c_i|$ and ensure mesh quality. More generally, isotropic or anisotropic metrics can be defined to constrain the local sizes of the computational elements in the refined mesh $\mathcal{H}^{(k+1)}$, in which case the nested character of the meshes and approximation spaces may no longer hold.

During the iterative process, the parameter α_{ad} plays a crucial role. For small values of α_{ad}, only a few elements are refined at each iteration, so that $\mathcal{H}^{(k)}$ and $\mathcal{H}^{(k+1)}$ differ only slightly. Many iterations may then be required to obtain a mesh satisfying the target error ε_{ad}, but the resulting mesh will be sharply adapted. In contrast, for larger α_{ad}, more elements are refined per iteration, which may yield a mesh that is less sharply adapted, though it can be reached in fewer iterations. In the present work, we assume that α_{ad} is chosen appropriately to ensure well-adapted meshes and stays fixed throughout adaptation.

The structure of the AMR algorithm is summarized in Algorithm 1.

Algorithm 1 AMR algorithm.

Require: $\mathcal{H}^{(0)}$, ε_{ad}, α_{ad}
1: Set $k = 0$
2: Compute $u_{\mathcal{H}^{(k)}}$ and local error estimates $\widehat{\varepsilon}^{\text{loc}}_{\mathcal{H}^{(k)}}(c_i)$ ▷ *Using Eq. (1) and Eq. (3)*
3: **while** $\sum_{c_i \in \mathcal{H}^{(k)}} \widehat{\varepsilon}^{\text{loc}}_{\mathcal{H}^{(k)}}(c_i)^2 > \varepsilon_{\text{ad}}^2$ **do**
4: Flag the $\alpha_{\text{ad}}|\mathcal{H}^{(k)}|$ elements with highest error estimates
5: Refine the flagged elements of $\mathcal{H}^{(k)}$ to generate $\mathcal{H}^{(k+1)}$ ▷ *Here, an isotropic subdivision*
6: $k \leftarrow k + 1$
7: Compute $u_{\mathcal{H}^{(k)}}$ and local error estimates $\widehat{\varepsilon}^{\text{loc}}_{\mathcal{H}^{(k)}}(c_i)$ ▷ *Using Eq. (1) and Eq. (3)*
return $\mathcal{H}^{(k)}$, $u_{\mathcal{H}^{(k)}}$

2.3. Present methods

As outlined in the introduction to this section, EMS is rather independent of the specific numerical methods employed for problem resolution and mesh refinement; this includes the type of mesh and elements used (structured or unstructured). Nevertheless, specific choices were made for its implementation and testing in this work. To investigate the performance of the EMS method, we consider CFD problems, and more specifically, supersonic inviscid configurations. These configurations are particularly challenging for studies involving variable conditions, as they feature localized phenomena such as shocks and expansion fans that are highly sensitive to boundary conditions. Consequently, this study employs a standard finite volume method (FVM) on unstructured polyhedral meshes, combined with a first-order Roe scheme [22], which ensures stable handling of solution discontinuities such as shocks while maintaining a satisfactory level of accuracy. The local error estimator guiding the adaptation process is based on an *a posteriori* estimation of the interpolation error on $u_{\mathcal{H}}$. The specific estimator used here is a piecewise constant interpolation error estimator derived from our previous work [15], tailored for piecewise constant schemes such as the Roe scheme employed in this study. The error in a given cell c_i is defined in Equations 3 and 4 of [15]. More generally, for nth order schemes, the interpolation error estimate relies on the $(n+1)$th derivative of the solution.

For adaptation, we adopt a standard isotropic subdivision strategy, in which cells marked for refinement are subdivided into 2^d subcells. A smoothing procedure is then applied to the resulting mesh to ensure a gradual evolution of cell sizes, so that the difference in refinement level between two neighboring cells does not exceed 1. This approach produces a sequence of nested meshes $\mathcal{H}^{(k)}$ starting from an initial coarse mesh $\mathcal{H}^{(0)}$, which can be conveniently represented using a tree structure. This structure can be efficiently exploited for compact adaptation procedures, storage, and various operations such as initial solution transfer, similarity measurements, and multi-resolution computations. Finally, we note that mesh coarsening is not considered in this study.

3. Error-based mesh selection

In this section, we extend the problem of predicting u to multiple conditions. We introduce a vector of parameters $q \in \mathcal{Q} \subseteq \mathbb{R}^{d_q}$ that defines these conditions. For example, q may parametrize boundary conditions, model constants, or external forcings. Let μ_Q denote the probability measure of q, such that

$$\mathbb{E}_q[f(q)] = \int_{\mathcal{Q}} f(q) d\mu_Q(q),$$

for any quantity of interest $f(q)$.

We restrict ourselves to a fixed computational domain Ω (independent of q) and denote by $u(q) \in \mathcal{V}$ the solution satisfying the parameter-dependent problem

$$\mathscr{P}(u, q) = 0. \tag{4}$$

We assume that (4) is almost surely well-posed, and denote by $u_{\mathcal{H}}(q) \in \mathcal{V}$ and $\varepsilon_{\mathcal{H}}(q)$ the discrete solution and associated error corresponding to its resolution on a mesh $\mathcal{H}$.

3.1. Library of meshes

In the following, $\mathcal{H}_q$ denotes the mesh adapted to $u(q)$, i.e., to the resolution of (4) for a given q. The mesh $\mathcal{H}_q$ is constructed such that the estimated error

$$\|u(q) - u_{\mathcal{H}(q)}(q)\| \approx \widehat{\varepsilon}_{\mathcal{H}(q)}(q) \leq \varepsilon_{\text{ad}}. \tag{5}$$

Instead of computing $\mathcal{H}_q$ for each queried parameter $q \in \mathcal{Q}$, or using a single mesh $\overline{\mathcal{H}}$ for all q, we propose to address the problem by selecting a mesh from a finite library of $1 < N_{\text{L}} < \infty$ meshes.

Let $\{q_i, i = 1, \dots, N_{\text{L}}\}$ be a set of distinct parameter vector values in $\mathcal{Q}$. For each $i = 1, \dots, N_{\text{L}}$, we define $\mathcal{H}_i \doteq \mathcal{H}(q_i)$ and $u_i \doteq u_{\mathcal{H}_i}(q_i)$. The task considered in this section is then to select the appropriate mesh $\mathcal{H}_i$ from the library to approximate the solution $u(q)$ for a queried parameter $q \in \mathcal{Q}$.

Assuming that a mesh $\mathcal{H}_i$ is suitable in the neighborhood of q_i, a simple approach for mesh selection relies on the parametric distance. Specifically, we choose the mesh $\mathcal{H}_{i_{\text{dist}}}$ with index i_{dist} given by

$$i_{\text{dist}} \in \underset{i \in \{1, \dots, N_{\text{L}}\}}{\arg\min}\ d_W(q_i, q), \tag{6}$$

where d_W is the weighted Euclidean distance on $\mathbb{R}^{d_q}$,

$$d_W(q, q')^2 = (q - q')^T W (q - q'), \tag{7}$$

where W is a symmetric positive definite matrix. Since the components of the vector q may correspond to different physical quantities, the matrix W is essential for properly scaling these quantities. Transformations that map the components' ranges to the unit interval do not reduce W to the identity matrix, as the distance would still depend on the parametric ranges. In the following we refer to this first approach as Distance-based Mesh Selection (DMS). In practice, W should be fitted to the specific problem under consideration, a task that can require considerable effort and may yield limited efficiency. This observation motivates the development of a more robust approach in the remainder of this section.

3.2. Error-based mesh selection

We formulate the problem in terms of a policy $\mathfrak{p} : \mathcal{Q} \to \{1, \dots, N_{\text{L}}\}$ that aims to minimize the resulting approximation error

$$\varepsilon_{\mathcal{H}_{\mathfrak{p}(q)}}(q) = \|u(q) - u_{\mathcal{H}_{\mathfrak{p}(q)}}(q)\|. \tag{8}$$

The solution $u_{\mathcal{H}_{\mathfrak{p}(q)}}(q)$ represents the approximate solution of the problem with parameter q, computed on the mesh from the library selected by the policy at q.

To determine the optimal policy, we must specify the class of policies under consideration and clarify the sense in which optimality is defined. Since q is a random variable and the error is non-negative, any moment of its distribution could serve as a valid criterion. A natural choice is to minimize the mean error for the policy:

$$\mathcal{E}(\mathfrak{p}) = \mathbb{E}_q\left[\varepsilon_{\mathcal{H}_{\mathfrak{p}(q)}}(q)\right]. \tag{9}$$

However, evaluating the performance of a policy typically requires computing the error for many values of q, which makes this approach impractical.

To derive an alternative approach, we observe that for any policy $\mathfrak{p}$, a lower bound on the mean error in (9) is

$$\mathcal{E}(\mathfrak{p}) \geq \mathbb{E}_q\left[\min_{i=1,\dots,N_\mathrm{L}} \varepsilon_{\mathcal{H}_i}(q)\right], \quad \varepsilon_{\mathcal{H}_i}(q) = \|u(q) - u_{\mathcal{H}_i}(q)\|.$$

This expression highlights the role of the error functions $\varepsilon_{\mathcal{H}_i}(q)$ in determining the performance of a policy. Rather than directly modeling $\mathfrak{p}$, we propose to model selected error functions to construct an approximate policy. The motivation is that these error functions can be inferred from the library without any additional problem resolution, as explained below and illustrated in the results section.

Computing the error functions for each library mesh over all queried q is not feasible, since $u(q)$ is unknown. The choice is made instead to rely on the error estimates $\hat{\varepsilon}_{\mathcal{H}_i}(q)$, which would require solving $u_{\mathcal{H}_i}(q)$ on the library meshes. Performing these N_L resolutions may not be practical compared to performing a full specific adaptation, unless N_L is small. To overcome this limitation, we introduce a proxy error model for $\varepsilon_{\mathcal{H}_i}(q)$, which can be determined directly from the library.

3.3. Error models

A first approach to constructing the error models from the library is to compute, for all $1 \leq i, j \leq N_\mathrm{L}$, the estimated errors $e_{i,j} \doteq \hat{\varepsilon}_{\mathcal{H}_i}(q_j)$, and use them to build a model $e_i(\mathfrak{p})$. This approach requires $N_\mathrm{L}(N_\mathrm{L}-1)$ additional resolutions on known meshes. To reduce this computational effort, we further approximate $\hat{\varepsilon}_{\mathcal{H}_i}(q_j)$ using the so-called restriction error [23] formulation.

We define Π_i as the projection operator from $\mathcal{V}$ onto $\mathcal{V}_{\mathcal{H}_i}$, given by

$$\Pi_i u = \arg\min_{v \in \mathcal{V}_{\mathcal{H}_i}} \|u - v\|. \tag{10}$$

This defines an orthogonal projection since $\mathcal{V}_{\mathcal{H}_i} \subset \mathcal{V}$. This projection can be computed efficiently in the case of MRA (multi-resolution analysis) methods, particularly when the meshes in the library are structured and represented by (possibly incomplete) trees. In fact, most MRA libraries provide utilities to project functions from one mesh to another. Writing the error $\varepsilon_{\mathcal{H}_i}(q)$ at $q = q_j$, we obtain

$$\begin{aligned}\varepsilon_{\mathcal{H}_i}(q_j)^2 &= \|u(q_j) - u_{\mathcal{H}_i}(q_j)\|^2 \leq \|u(q_j) - u_j\|^2 + \|u_j - u_{\mathcal{H}_i}(q_j)\|^2 \\ &\leq \varepsilon_{\mathrm{ad}} + \|u_j - \Pi_i u_j\|^2 + \|\Pi_i u_j - u_{\mathcal{H}_i}(q_j)\|^2.\end{aligned} \tag{11}$$

The second term, $\|u_j - \Pi_i u_j\|$, is known as the restriction error and is denoted by e_i in the following. It is defined such that the restriction error vanishes when $\mathcal{V}_{\mathcal{H}_i}$ is locally richer (more refined) than $\mathcal{V}_{\mathcal{H}_j}$. Consequently, it quantifies the loss of quality in $u(q_j)$ when projected onto $\mathcal{V}_{\mathcal{H}_i}$. This quantity can be computed with limited computational effort, since the adapted solutions u_j are readily available.

The final term, $\|\Pi_i u_j - u_{\mathcal{H}_i}(q_j)\|$, is generally expected to be dominated by the restriction error, as it quantifies the difference between a reasonably accurate approximation of the projection of $u(q_j)$ onto $\mathcal{V}_{\mathcal{H}_i}$ and the actual solution computed on $\mathcal{H}_i$.

The next task is to construct surrogate models $\hat{e}_i(q)$ for the restriction errors. These restriction errors can be efficiently computed from the library for $q = q_j$, $j = 1, \dots, N_\mathrm{L}$, as explained above, providing the data necessary to train the models. Here, we employ Gaussian process (GP) regression [24] to approximate the restriction error functions. This choice is motivated by the robustness of GP models, which can accommodate limited training data. Additionally, GP models provide confidence intervals that can be leveraged to develop infilling strategies for enhancing the library in future work.

The models take the following generic form:

$$e_i(q) \approx \hat{e}_i(q) = \sum_{k=0}^{n} \beta_k \widehat{f_k}(q) + Z(q). \tag{12}$$

In (12), the $\widehat{f_k}$ are prescribed trend functions, and Z is a Gaussian process. The GP, Z, has a prior distribution, which is updated via Bayes' formula using the N_L observations $e_i(q_j)$. The prior distribution consists of a mean function and a covariance function $C(q, q')$, generally involving hyperparameters that must be set alongside the trend coefficients β_k. In the present work, we rely on the library `smt` [25], which provides several choices of trend functions and covariance structures, as well as different algorithms for regression and hyperparameter optimization. Based on numerical tests, we train the GP models on standardized data (centered and scaled), using linear trend functions, zero-mean priors, and simple squared exponential covariance functions.

$$C(q, q') = \sigma^2 \exp\left[-\frac{\|q - q'\|^2}{l_c^2}\right], \tag{13}$$

with variance (σ^2) and correlation length (l_c) as hyperparameters.

One challenge in approximating the restriction error e_i from a limited dataset $e_i(q_j)$ is that the approximation must be sufficiently accurate when the error is small, to discriminate between meshes that induce low errors. For a large library, the restriction error e_i may be small only in a limited neighborhood of q_i and large elsewhere, so that most data points $e_i(q_j)$ take large values. Training a GP on a dataset dominated by high-error points can compromise the quality of the prediction. To mitigate this effect, we found it more effective to train the GP model $\hat{e}_i(q)$ using only the restriction errors corresponding to parameters q_j within a fixed neighborhood $\mathcal{N}_i$ of q_i. Defining this neighborhood geometrically based on the distance $\|q_i - q_j\|$ encounters the same difficulties discussed in 3.1, and we adopted the following definition,

$$\mathcal{N}_i = \{j = 1, \dots, N_\mathrm{L} \,|\, e_i(q_j) \leq \delta_e\}, \tag{14}$$

where δ_e is a user-prescribed error threshold. With this definition, the GP training disregards data points with excessively large errors, thereby improving the predictive capability in regions where the error is small.

3.4. Approximate EMS method

Given a library of meshes (and their associated solutions), the approximate EMS (A-EMS) method proceeds in two stages. In the offline stage, the restriction error models $\hat{e}_i(q)$ are constructed. In the online stage, the index of the mesh to use for a queried parameter value q is selected from

$$\mathfrak{p}(q) = \arg\min_{i=1,\dots,N_\mathrm{L}} \hat{e}_i(q).$$

The selected mesh will therefore be the one with the lowest restriction error predicted by the GP.

To assess the performance of the A-EMS method, we focus on the resulting average error

$$\mathcal{E}(\mathfrak{p}) = \mathbb{E}_q\left[\varepsilon_{\mathcal{H}_{\mathfrak{p}(q)}}(q)\right]. \tag{15}$$

The A-EMS error can be compared to the average minimal error achievable with the considered library. Let $i^*(q)$ denote the true optimal mesh selection,

$$i^*(q) = \arg\min_{i=1,\dots,N_\mathrm{L}} \|u(q) - u_{\mathcal{H}_i}(q)\|, \tag{16}$$

and denote corresponding $\varepsilon^*(q)$ the minimal error,

$$\varepsilon^*(q) = \|u(q) - u_{\mathcal{H}_{i^*(q)}}(q)\|.$$

With these notations, the local sub-optimality of the A-EMS method is defined by

$$\rho_{\text{A-EMS}}(q) = \frac{\varepsilon_{\mathcal{H}_{\mathfrak{p}(q)}}(q) - \varepsilon^*(q)}{\varepsilon_{\text{ad}}}, \tag{17}$$

and the global sub-optimality is

$$\overline{\rho}_{\text{A-EMS}} = \frac{\|\varepsilon_{\mathcal{H}_{\mathfrak{p}(q)}}(q) - \varepsilon^*(q)\|^2}{\varepsilon_{\text{ad}}}. \tag{18}$$

To complete the assessment of the A-EMS method, we provide a brief analysis of its computational cost and compare it with the cost of performing a systematic mesh adaptation for all queried parameters q, referred to as Specific Mesh Adaptation (SMA). The computational time required to generate the library in this analysis is not taken into account, as it is subsequently amortized over a large number of queried values q, which is assumed to be much greater than N_{L}. Accounting for the library construction cost requires adding N_{L} SMA costs to the total cost (see Eq. (19)). The impact on the average cost of A-EMS depends on the ratio between N_{L} and the number of queried values of q.

Under the working assumption of an MRA method, where the mesh $\mathcal{H}^{(k+1)}$ is obtained by refining a fixed fraction α_{ad} of the cells in $\mathcal{H}^{(k)}$ into 2^d sub-cells, we obtain

$$|\mathcal{H}^{(k)}| = |\mathcal{H}^{(0)}| G^k_{d,\alpha_{\text{ad}}}, \quad G_{d,\alpha_{\text{ad}}} = (1 + (2^d - 1)\alpha_{\text{ad}})$$

where $|\mathcal{H}^{(0)}|$ is the number of cells in the initial mesh, $G_{d,\alpha_{\text{ad}}} > 1$ is the mesh size increase factor per adaptive iteration, d is the number of spatial dimensions, and α_{ad} is the fraction of cells refined at each iteration.

To estimate the computational cost of the methods, we assume that the cost of error estimation and mesh refinement is negligible compared to solving the problem on a given mesh, which is often the case with modern (possibly parallel) mesh adaptation frameworks. Therefore, for systematic mesh adaptation, the total cost to obtain a solution on a sufficiently refined mesh corresponds to the cost of solving the problems on the sequence of refined meshes. We further assume that the solution time for a given mesh can be expressed as

$$T(\mathcal{H}) = C|\mathcal{H}|^\gamma,$$

with $\gamma \geq 1$ characterizing the solver's scalability. The total computational time to obtain the solution on the adapted mesh $\mathcal{H}^{(k_{\text{ad}})}$ is then

$$T_{k_{\text{ad}}} = C|\mathcal{H}^{(0)}|^\gamma \sum_{k=0}^{k_{\text{ad}}} \left(G^k_{d,\alpha_{\text{ad}}}\right)^\gamma,$$

where k_{ad} is the number of adaptation iterations required to reach the prescribed accuracy. In practice, k_{ad} depends on the parameter q, the target precision ε_{ad}, and the refinement fraction α_{ad}. We are interested in the average computational time to solution, $\overline{T}_{\text{SMA}} = \mathbb{E}_q\left[T_{k_{\text{ad}}}\right]$. For small k_{ad} variations with q, the SMA average time to solution can be approximated by

$$\overline{T}_{\text{SMA}} \approx C|\mathcal{H}^{(0)}|^\gamma \sum_{k=0}^{\overline{k}_{\text{ad}}} \left(G^k_{d,\alpha_{\text{ad}}}\right)^\gamma, \tag{19}$$

where $\overline{k}_{\text{ad}}$ is the (rounded) average number of adaptive iterations.

For the A-EMS method, only a single solution needs to be computed for each queried q, with a cost

$$T_{\text{A-EMS}}(q) = C|\mathcal{H}_{\mathfrak{p}(q)}|^\gamma.$$

When the library consists of N_{L} samples drawn uniformly from $d\mu_{\mathcal{Q}}$, we have $\mathbb{E}_q\left[|\mathcal{H}_i|\right] \approx |\mathcal{H}^{(0)}| G^{\overline{k}_{\text{ad}}}_{d,\alpha_{\text{ad}}}$. It is also reasonable to assume that $\mathbb{E}_q\left[|\mathcal{H}_{\mathfrak{p}(q)}|\right] \approx |\mathcal{H}^{(0)}| G^{\overline{k}_{\text{ad}}}_{d,\alpha_{\text{ad}}}$ if the policy performs correctly. Under these assumptions, the average time to solution for the A-EMS method is

$$\overline{T}_{\text{A-EMS}} \approx C|\mathcal{H}^{(0)}|^\gamma \left(G^{\overline{k}_{\text{ad}}}_{d,\alpha_{\text{ad}}}\right)^\gamma. \tag{20}$$

The averaged gain in time per solution of the A-EMS method, compared to SMA, is

$$\overline{T}_{\text{SMA}} - \overline{T}_{\text{A-EMS}} \approx C|\mathcal{H}^{(0)}|^\gamma \sum_{k=0}^{\overline{k}_{\text{ad}}-1} \left(G^k_{d,\alpha_{\text{ad}}}\right)^\gamma = C|\mathcal{H}^{(0)}|^\gamma \frac{1 - \left(G^\gamma_{d,\alpha_{\text{ad}}}\right)^{\overline{k}_{\text{ad}}}}{1 - G^\gamma_{d,\alpha_{\text{ad}}}}.$$

This expression indicates that the differences will primarily depend on the refinement fraction α_{ad} and the average number of adaptation iterations $\overline{k}_{\text{ad}}$, two quantities that are not independent, since reducing α_{ad} generally requires more adaptation iterations. The reduction in computational time will also increase as the target precision ε_{ad} decreases, because $\overline{k}_{\text{ad}}$ will increase. However, the average error of A-EMS does not necessarily decrease unless the library is sufficiently rich.

4. EMS investigation on oblique shock

This section examines the proposed selection method using an analytical problem that permits the exact projection of the solution onto a prescribed mesh and the evaluation of the resulting error at low computational cost.

4.1. Oblique shock problem presentation

We consider the two-dimensional flow of an inviscid, compressible fluid around a symmetric wedge of fixed angle θ_w, see Fig. 1(a). For a supersonic inflow, an attached oblique shock forms symmetrically at the wedge tip and extends outward at a constant angle β, which depends on the inflow Mach number M_0, the wedge angle θ_w, and the ratio of specific heats of the perfect gas, γ.

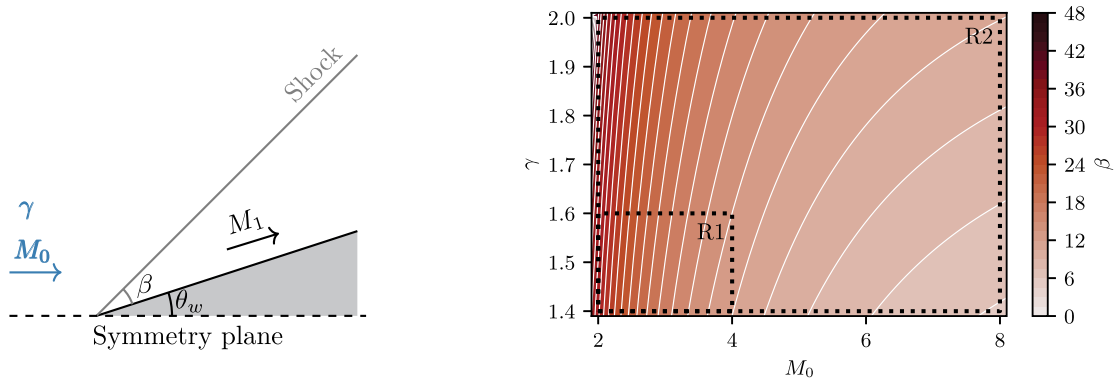
The oblique shock separates two constant regions: the inflow region and the post-shock region. Both are known explicitly as functions of the problem parameters (θ_w, M_0, γ), see [26]. Fig. 1(b) illustrates the dependence of the shock angle β on M_0 and γ for $\theta_w = 15°$. The shock angle decreases with M_0 and increases with γ. The effect of γ becomes more pronounced at higher Mach numbers: it is almost negligible around $M_0 \approx 2$, but dominant near $M_0 \approx 8$.

This variation in the dependence of β on the flow parameters poses a challenge for mesh selection methods, since a simple parametric distance cannot capture such heterogeneous dependencies. For the analysis, we consider two ranges of uncertain conditions. The reference range is $(M_0, \gamma) \in R_2 = [2, 8] \times [1.4, 2]$, and the restricted range is $R_1 = [2, 4] \times [1.4, 1.6]$, both assumed with uniform probability distributions.

This simple flow configuration enables a cost-effective and accurate piecewise-constant approximation on any prescribed mesh, enabling precise evaluation of discretization errors. For mesh adaptation, we focus only on the Mach field and employ a standard AMR strategy that isotropically subdivides a fraction α_{ad} of the mesh cells with the largest local error estimates, continuing until the estimated L_2 error falls below $\varepsilon_{\text{ad}} = 0.005$. All computations presented use the interpolation error for error estimation. The refinement fraction is fixed at $\alpha_{\text{ad}} = 0.02$ to produce sharply adapted meshes. An adapted mesh is shown in Fig. 2.

4.2. Selection maps and error analysis

In this section, we consider a library of $N_{\text{L}} = 16$ meshes associated with parameter values q_i, generated by a Sobol sequence in R_2. Fig. 3 reports the positions of the parameters q_i using symbols, while subdomains of R_2 sharing the same minimizer i^* in Eq. (16) are shown with a common color. We distinguish between used and unused meshes by the marker shape: meshes selected at least once are represented at their parameter location q_{i^*} with hexagons, whereas meshes never



(a) Schematic of the oblique shock problem.

(b) Shock angle β function of M_0 and γ.

Fig. 1. Presentation of the oblique shock problem.

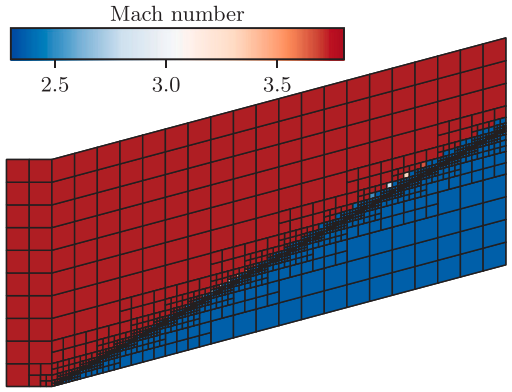


Fig. 2. Mesh adapted at $M_0 = 4$ and $\gamma = 1.75$ (library sample) along with mach number field for $M_0 = 3.7$ and $\gamma = 1.8$.

selected are indicated with triangles. The three maps in Fig. 3 are generated by evaluating the errors over an 80×80 uniform grid of parameter values.

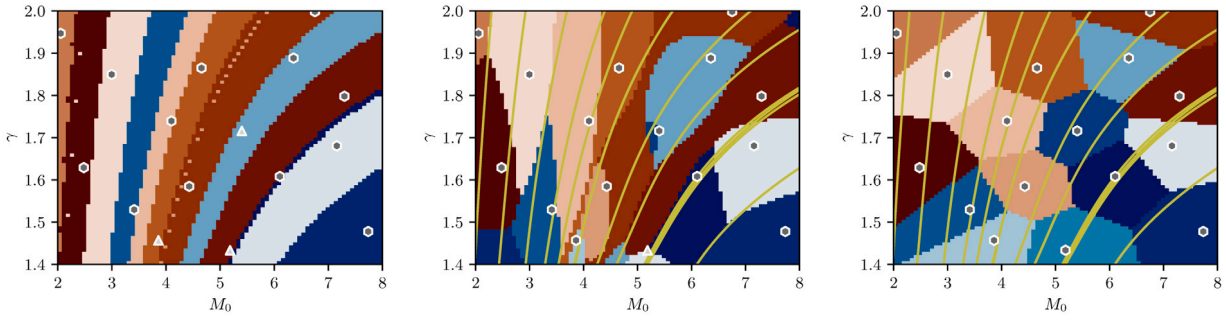
The first plot, Fig. 3(a), shows the optimal selection map. A first observation is that some library meshes are never selected. This occurs because a more refined mesh, adapted to another condition, may yield a similar shock angle β. Since the jump across the shock increases with M_0, the optimal strategy favors the mesh adapted to the higher Mach number when two candidates produce comparable shock angles. Furthermore, because the q_i are drawn randomly, the resulting selection domains exhibit highly disparate sizes, reflecting the nonlinear parametric behavior highlighted above. Mesh selection strategies rely on solving the problem on a mesh from a precomputed library, which is generally not fully adapted to the specific parameter of interest. For localized structures such as the oblique shock, this imperfect adaptation is clearly visible in Fig. 2, where the adapted region does not align well with the shock farther from its origin. Enriching the library enhances the alignment of the refinements with the flow structure.

Fig. 3(b) presents the selection map produced by the A-EMS method. For comparison, the contours of the optimal map are also shown. One observes that, while the A-EMS map captures some features of the optimal map, such as the Mach-aligned anisotropy and the increasing influence of γ as M_0 grows, the boundaries of the selection domains are generally misplaced. In addition, all but one mesh are selected, and the resulting domains are sometimes disconnected or small. The GP modeled errors are likely responsible for these complex selection domains, as they can lead to a noisy policy in areas where the restriction errors of different meshes are comparable, particularly at the boundaries of the optimal selection domains. Nevertheless, as discussed below, selecting a non-optimal mesh remains acceptable when the induced error is close to that of the optimal choice.

This relatively coherent behavior contrasts with the selection map of the Distance-based Mesh Selection (DMS) method, shown in Fig. 3(c). Here, the map is constructed using a diagonal weight matrix corresponding to the standard Euclidean distance in normalized M_0 and γ (i.e., $W_{M_0,M_0} = (M_{0\max} - M_{0\min})^{-1}$ and $W_{\gamma,\gamma} = (\gamma_{\max} - \gamma_{\min})^{-1}$). The resulting partition of $\mathcal{Q}$ is the Voronoi partition of the space induced by W. In this case, all meshes are selected within their parametric neighborhood; consequently, the boundaries of the selection domains differ substantially from those of the optimal map. We emphasize that, although this choice is consistent with standard practice, the use of this weight matrix W is not optimal for the DMS strategy. In [27], several approaches for optimizing W are discussed, leading to DMS methods that still underperform compared to A-EMS. The superior performance of the A-EMS method can be attributed to the greater expressiveness of Gaussian Process (GP) models, compared with the use of a constant weight matrix W over the entire parameter domain in DMS.

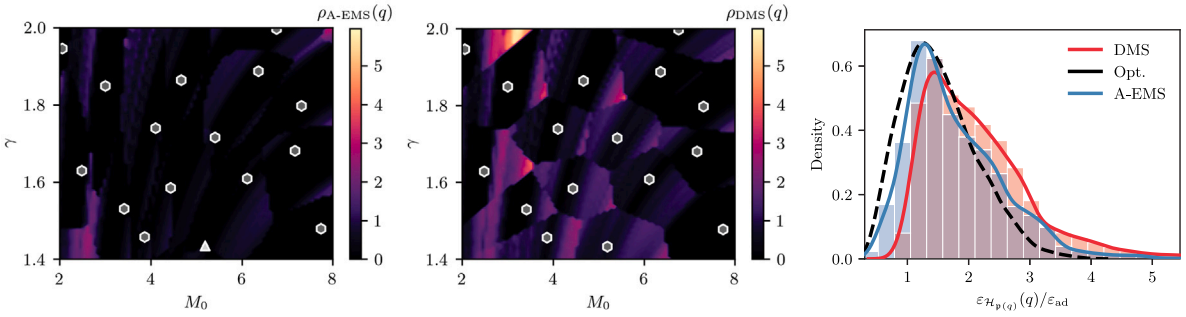
While indicative of proximity to the optimal selection, the maps are not directly informative of the errors achieved by the A-EMS and DMS strategies. To complete the analysis, we examine the local errors for each method in Fig. 4. The figure reports the local sub-optimality errors normalized by the target precision ε_{ad}, defined in (17) for A-EMS. The sub-optimality of the DMS approach, ρ_{DMS}, follows the same definition as the A-EMS one in (17). Comparing Figs. 4(a) and 4(b), we observe that the sub-optimality of A-EMS remains below ε_{ad} across most of the domain, whereas for DMS it exceeds $2\varepsilon_{\text{ad}}$ in several regions. Moreover, the regions of high sub-optimality for A-EMS tend to concentrate along the boundaries of the selection domains, i.e., far from the library meshes. For both methods, the sub-optimality gradients are primarily aligned with the Mach direction, underscoring the greater mesh discrepancies along this parameter and the resulting reduction in accuracy of both modeling and selection in this direction.

Overall, the sub-optimality map of A-EMS (Fig. 3(b)) shows that when a non-optimal mesh is selected, the induced error is typically not significantly higher than for the optimal one, particularly in regions sufficiently populated. Notable exceptions occur in the bottom-left and top-right portions of $\mathcal{Q}$, where large sub-optimality values are observed. In these areas, the policy based on the modeled restriction errors selects meshes adapted to significantly different shock angles. This behavior stems from the degradation of the GP approximations near domain



(a) Optimal selection map. (b) A-EMS map. (c) DMS map.

Fig. 3. Selection maps for the oblique shock problem on R_2 with $N_L = 16$. The library meshes parameters q_i are shown with symbols (hexagons are selected meshes while triangles are never selected meshes).



(a) A-EMS sub-optimality map. (b) DMS sub-optimality map. (c) Error distributions for the optimal selection, A-EMS and DMS methods.

Fig. 4. Sub-optimality maps and error distributions. Oblique shock on R_2 with $N_L = 16$. The library meshes parameters q_i are shown with symbols (hexagons are selected meshes while triangles are never selected meshes).

boundaries due to extrapolation. It could be mitigated by sequentially enriching the mesh library.

Fig. 4(c) further compares the error distributions obtained with the optimal selection, A-EMS, and DMS. Interestingly, the probability of achieving an error smaller than ε_{ad} is nonzero for all methods, particularly for the optimal selection and A-EMS. This effect arises from overlapping shock angles, which lead to resolution on overadapted meshes. The A-EMS error distribution exhibits a much lighter tail than that of DMS, with an almost negligible probability of exceeding $4\varepsilon_{\text{ad}}$. This result highlights the robustness of A-EMS, which is unlikely to produce excessively large errors.

To obtain a more quantitative comparison of the selection strategies, Table 1 reports the evolution of the averaged error for different library sizes and for the two parametric ranges. For convenience, these averaged errors are normalized by the target error ε_{ad}. A first observation is that the averaged error decreases as N_L increases and converges toward ε_{ad}. As expected, the convergence of A-EMS is slightly slower than that of the optimal selection but significantly faster than that of DMS, with this trend being more pronounced for the larger range. Finally, the lower averaged error of the optimal selection for R_2 compared to R_1 at $N_L = 64$ is explained by the stronger effect of shock angle overlap in the wider range.

4.3. Error models investigation

The previous results have shown that A-EMS can select, for most queried values of q, a mesh that yields an error comparable to the optimal error and significantly smaller than that obtained with DMS. This selection relies on substituting the true error with the restriction error and predicting values at arbitrary parameters using a GP model.

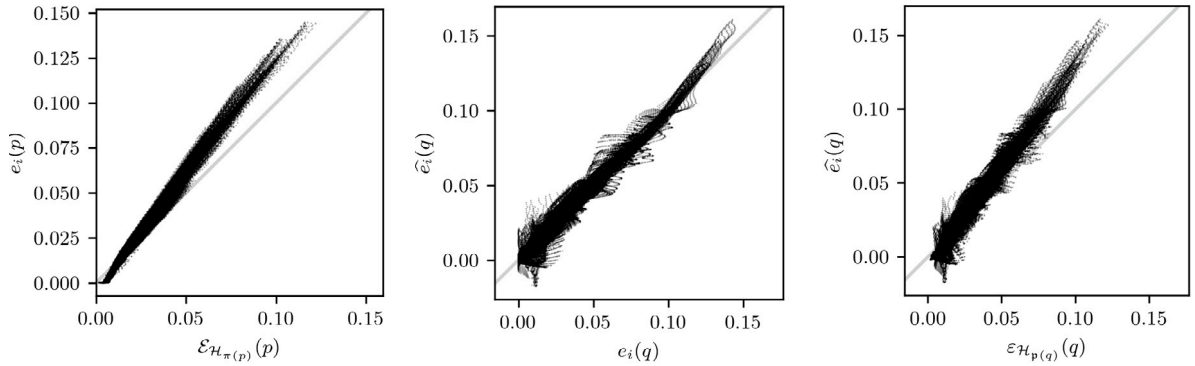
Table 1
Averaged error $\mathcal{E}(\mathfrak{p})$, normalized by the target error ε_{ad} for the optimal selection (Opt.), A-EMS, and DMS methods. Using different library sizes N_L and the two parametric ranges. Oblique shock problem.

$\times\varepsilon_{\text{ad}}$		R_1			R_2	
N_L	Opt	A-EMS	DMS	Opt	A-EMS	DMS
1	2.70	2.70	2.70	5.15	5.15	5.15
8	1.45	1.52	1.83	1.85	2.26	2.59
16	1.32	1.41	1.65	1.52	1.78	2.16
32	1.19	1.29	1.51	1.29	1.58	1.94
64	1.12	1.22	1.36	1.08	1.28	1.63

Consequently, the performance of the A-EMS method depends directly on the quality of the GP models.

To assess the validity of this approach, Fig. 5(a) presents a q-q plot comparing exact and restriction errors. Each point corresponds to the restriction error $\|u(q) - \Pi_i u(q)\|$ against the true error $\|u(q) - u_{\mathcal{H}_i}(q)\|$, for one of the N_L meshes and one parameter value from the 80×80 uniform grid covering R_2. The plot thus contains 16×80^2 data points. A strong correlation between exact and restriction errors is evident, particularly at low error levels, where precision is critical for accurate mesh selection. We also note that while the exact error typically remains above ε_{ad}, the restriction error may vanish since it does not incorporate ε_{ad}.

Fig. 5(b) compares the restriction errors predicted by the GP models with the computed restriction errors. The correlation remains clear but exhibits greater dispersion than in the previous plot, and the GP models may even predict negative restriction errors. Finally, Fig. 5(c) compares the GP predictions directly with the exact errors. The combined effect of



(a) Restriction error against exact error. (b) GP error model against restriction error. (c) GP error model against exact error.

Fig. 5. Comparison of exact, restriction, and GP error model. Oblique shock on R_2 with $N_L = 16$.

trend errors and the dispersion introduced by restriction and GP modeling does not significantly degrade overall accuracy, thereby explaining the correct behavior of the method.

The conclusion from our experiments on the oblique shock problem is that precise error prediction is not required to correctly decide on the mesh from the library to be used at a new query point. Indeed, confusion between meshes yielding comparable errors has little impact on the resulting accuracy. At the same time, even relatively crude GP models can reliably distinguish between well-adapted and poorly adapted meshes.

To further examine the approximation of the error by GP models, Fig. 6 analyzes the influence of the neighborhood size considered for GP construction. For simplicity, we adopt a geometric definition of the neighborhood, retaining the closest library parameters q_j to build each GP model $e_i(q)$. The figure reports the dependence of the A-EMS averaged sub-optimality on the neighborhood size for both the large and reduced parametric ranges. The results show that, beyond a sufficient neighborhood size, a minimum sub-optimality is attained. Two distinct behaviors then emerge as the neighborhood size increases. If the library is sufficiently rich to achieve an average error close to ε_{ad}, the sub-optimality remains nearly constant and independent of neighborhood size. Conversely, if the library cannot achieve an average error close to ε_{ad}, the sub-optimality stays low but increases once the neighborhood size exceeds a certain threshold. This increase is explained by the relatively sparse sampling in low-error regions, which leads to a higher rate of non-optimal selections as the error models become biased by longer-distance trends.

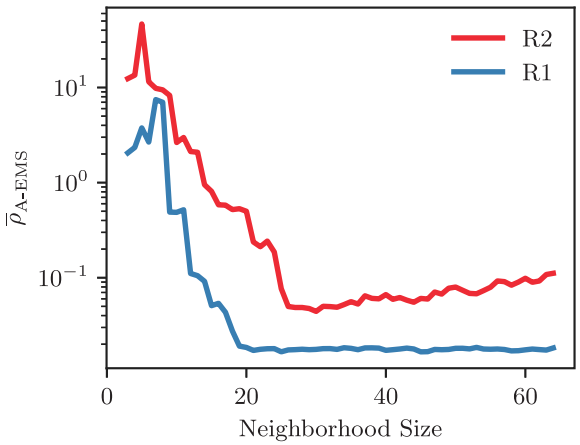


Fig. 6. Averaged sub-optimality of A-EMS as a function of the neighborhood size in the GP models construction and for the two parametric ranges. Oblique shock on R_2 with $N_L = 64$.

4.4. Cost analysis

To further assess the efficiency of A-EMS, we provide a brief analysis of the computational cost and a comparison with other adaptation strategies. Specifically, Fig. 7 reports the effective average errors as a function of computational cost for Nominal Mesh Adaptation (NMA), the naive single-mesh approach; Mean Mesh Adaptation (MMA), which constructs a unique mesh designed to minimize the mean error; A-EMS, the present method; and Specific Mesh Adaptation (SMA), the most accurate but also the most computationally expensive approach.

To generate these error-to-cost curves, we progressively decrease the target precision ε_{ad} in the mesh adaptations. We then compute the average error achieved with the adapted meshes and the considered strategies. For SMA, the computational cost is defined by Eq. (19) as the sum of costs of all adaptation iterations required to reach the average number of iterations $\overline{k}_{ad}$, estimated over the library meshes for each error target. Similarly, the cost of A-EMS is computed using Eq. (20), corresponding to the cost of solving on the final grid of a $\overline{k}_{ad}$-iteration process. For A-EMS, $\overline{k}_{ad}$ is taken as the weighted average of the selected meshes' k_{ad} according to the selection policy, reflecting potential bias toward specific library meshes. For NMA, the cost corresponds to the last grid of a mesh adapted at the median parameter values, while for MMA the cost derives from solving on a single mesh adapted to ε_{ad} [15]. In all cases, the same refinement fraction $\alpha_{ad} = 0.08$ is used.

Fig. 7 highlights the limitation of the NMA approach: it cannot achieve arbitrary levels of averaged error, because a mesh adapted to a nominal condition is generally unsuitable for other conditions unless the refinement is uniform. Consequently, the NMA cost increases even when the averaged error stagnates. A-EMS exhibits a similar trend, with a rapid initial reduction of the averaged error followed by a plateau, while the cost continues to rise. The asymptotic error level of A-EMS depends on N_L for the same reason as for NMA, which is a particular case of A-EMS with $N_L = 1$. The convergence rate before stagnation is largely independent of N_L, as it depends on the average size of the adapted library meshes. In the limit $N_L \to \infty$, A-EMS achieves the same convergence rate as SMA but at a fraction of its computational cost, corresponding to the saved adaptation iterations. In the examples presented, A-EMS is asymptotically five times less expensive than SMA. Nonetheless, very large libraries may introduce storage and handling challenges during mesh selection.

These results do not account for the cost of constructing the library mesh required by A-EMS. When this cost is included, it is useful to determine the number of evaluations required for A-EMS to become advantageous compared with SMA. This threshold is characterized by

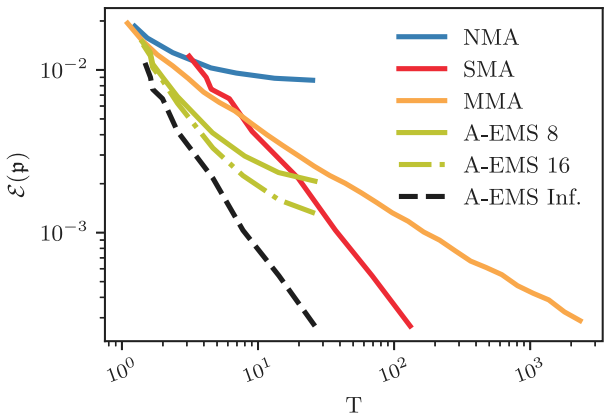


Fig. 7. Cost-efficiency of the different adaptation strategies, using $\alpha_{\text{ad}} = 0.08$. Oblique shock on R_2.

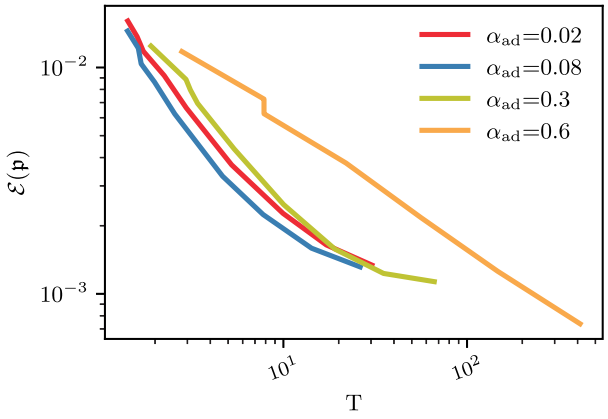


Fig. 8. Cost-efficiency of the A-EMS method for different refinement fraction. Oblique shock on R_2 with $N_{\text{L}} = 16$.

the break-even point,

$$k_{\text{eq}} = \frac{N_{\text{L}}, \overline{T}\text{SMA}}{\overline{T}\text{SMA} - \overline{T}_{\text{A-EMS}}}. \tag{21}$$

For instance, to achieve an average error of $3{\times}10^{-3}$, a library of $N_{\text{L}} = 16$ meshes is sufficient, leading to a positive return after only $k_{\text{eq}} = 25$ evaluations. This highlights the advantage of A-EMS over SMA, even for relatively small numbers of evaluations in this case.

Regarding MMA, we observe that in this example it outperforms SMA at large error levels but does not surpass A-EMS for the considered values of N_{L}. Still, being a single-mesh method, MMA is simpler to implement and allows for average error control, without encountering convergence plateaus like NMA or A-EMS.

Finally, Fig. 8 illustrates the impact of the refinement fraction α_{ad} on A-EMS performance. For $\alpha_{\text{ad}} \in [0.02, 0.3]$, the computational cost remains essentially constant for a fixed averaged error. However, excessively large α_{ad} values (e.g., $\alpha_{\text{ad}} = 0.6$) reduce adaptation sharpness and therefore the efficiency of A-EMS. Accordingly, it is generally advantageous to use A-EMS with a low refinement fraction.

5. Validation on scramjet flow simulation

A two-dimensional supersonic scramjet flow, originally introduced in [18] and widely employed to assess adaptation methods [28–31], is considered in this section to validate the behavior of EMS.

5.1. Scramjet case presentation

The symmetric scramjet domain consists of five regions, as illustrated in Fig. 9(b): an upstream section, a central duct, two outer ducts (top and bottom), and a downstream section where the duct flows merge. The boundary conditions consist of a supersonic inlet and outlet, with slip conditions imposed on the solid walls. The Euler equations for a perfect gas are solved in the flow domain using the finite-volume, cell-centered solver *elsA* [32] (ONERA-SAFRAN property). Fluxes are discretized using Roe scheme [22], and the unsteady governing equations are time-integrated until steady state using an implicit backward Euler scheme coupled with a LUSSOR-type linear solver.

The variable flow parameters are the inflow Mach number M_0 and the inlet incidence angle α_{in}, such that $q(\theta) = (M_0(\theta), \alpha_{\text{in}}(\theta))$. These two parameters govern the flow structure and determine the position of the shock waves in the domain. The Mach number varies in the range $3 \leq M_0 \leq 5$, which ensures a constant flow regime and a supersonic outlet across the entire interval. The incidence angle is restricted to $0 \leq \alpha_{\text{in}} \leq 6$, since viscous effects would become non-negligible at higher values. The two parameters are assumed to vary independently, with uniform distributions over their respective ranges.

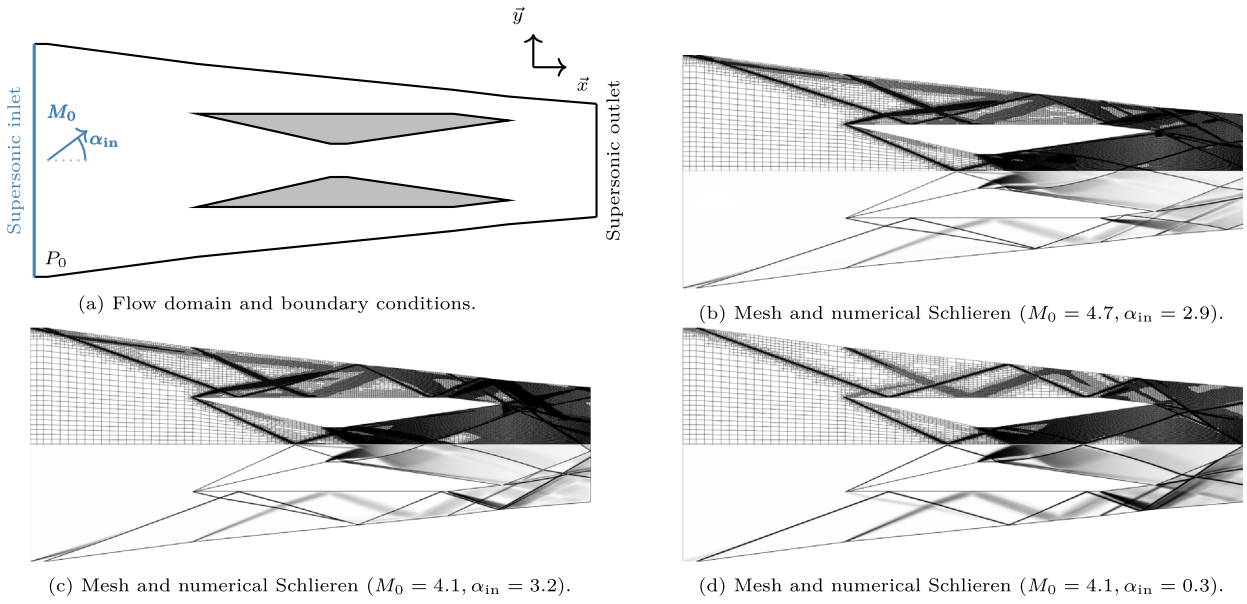
To apply the EMS method, flow solutions and adapted meshes are computed for N_{L} flow conditions $(M_0, \alpha_{\text{in}})_i$ generated from a Sobol sequence in the rectangle $[3, 5] \times [0, 6]$. Unless specified otherwise, $N_{\text{L}} = 16$. Each computation is initialized from the same coarse mesh $\mathcal{H}_0$, containing $|\mathcal{H}_0| = 3,000$ cells. Mesh adaptation relies on an interpolation error estimate of the Mach number field to guide refinement, and it terminates when the error estimate $\widehat{\varepsilon}_{\mathcal{H}_k}$ falls below $\varepsilon_{\text{ad}} = 0.004$. This threshold corresponds approximately to a one-order-of-magnitude reduction of the error estimate compared to the initial mesh $\mathcal{H}_0$.

Figs. 9(a) to 9(d) show three adapted meshes (upper part of the domain) and their solutions' Schlieren fields (bottom of the domain). The flows present oblique shocks and expansion fans along with complex interactions in the central and outer ducts. The upstream section reduction induces asymmetric oblique shocks that hit the struts past their leading edge. In the outer ducts, their reflections bounce off the walls up to the outlet. The varying duct section creates several expansion fans and oblique shocks interacting with each other in the downstream section. In the central duct, the shocks induced by the struts' leading edges meet and reflect in the converging section. The throat creates a system of interacting expansion fans and shocks that develop and interact in the diverging section down to the outlet. The merging of the central and outer flows and their interaction with the trailing edge structures yield a complex flow in the downstream section. To capture the flow features, the meshes show local adaptations along the shocks and expansion fans in the upstream section and entrance of the central and outer ducts. Further downstream, the diffusion of the expansion fans and the complex interactions lead to more uniform refinement meshes. In addition, comparing the three meshes highlights the displacement and the changes in the flow structures with the conditions.

For the analysis of EMS, a validation set of 100 reference solutions is generated to evaluate the mesh selection strategies and the resulting errors. The validation samples correspond to flow conditions $(M_0, \alpha_{\text{in}})$ drawn randomly within their respective ranges. Reference solutions are computed on a fine mesh $\mathcal{H}_{\text{ref}}$, obtained by uniformly refining the initial mesh $\mathcal{H}_0$ six times, resulting in $|\mathcal{H}_{\text{ref}}| \approx 12 \times 10^6$ cells. For consistency, a maximum of six successive refinements is permitted in the AMR procedure, ensuring that all adapted meshes in the EMS library are sub-meshes of $\mathcal{H}_{\text{ref}}$.

5.2. Selection maps and error analysis

We first examine the quality of the selection maps for the scramjet problem. The results are presented in Fig. 10 for $N_{\text{L}} = 16$. The first plot in Fig. 10(a) illustrates the optimal selection, with segments connecting



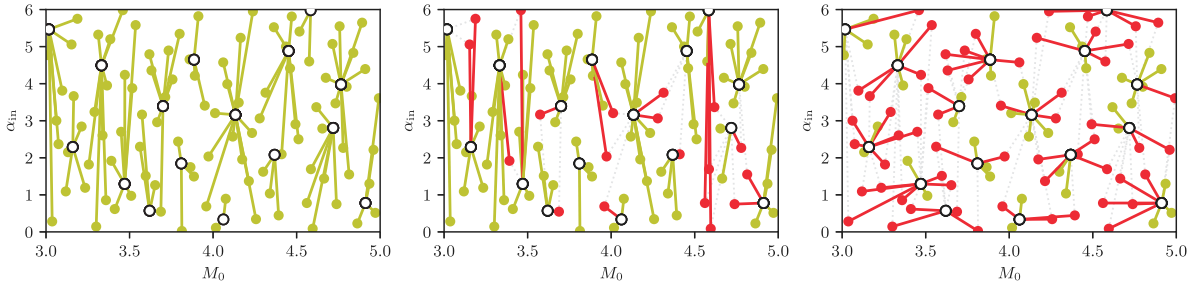
(a) Flow domain and boundary conditions.

(b) Mesh and numerical Schlieren ($M_0 = 4.7, \alpha_{\text{in}} = 2.9$).

(c) Mesh and numerical Schlieren ($M_0 = 4.1, \alpha_{\text{in}} = 3.2$).

(d) Mesh and numerical Schlieren ($M_0 = 4.1, \alpha_{\text{in}} = 0.3$).

Fig. 9. Scramjet problem description.



(a) Optimal selection map.

(b) GP-based A-EMS selection map.

(c) Distance-based mesh (DMS) selection map.

Fig. 10. Selection maps for the scramjet problem with $N_{\text{L}} = 16$. The maps show the association between the validation samples (full circles) and an element of the library (white-core circles). The optimal associations are shown with green segments, and non-optimal are shown in red.

each validation sample (solid circles) to the adapted mesh in the library (white-core circles) that yields the lowest error in the Mach field. For this case, the error is evaluated using the reference solutions. The plot emphasizes the dominant influence of the Mach number on the optimal selection across all incidence values, which is expected since the Mach number primarily determines the longitudinal position of the shocks, whereas the inlet incidence produces only minor vertical displacements of the shocks within the domain.

Next, we apply the A-EMS method, which relies on Gaussian process models of the restriction errors. Fig. 10(b) displays the discrete map, where green segments indicate selections that match the optimal ones (from the previous figure), and red segments indicate selections that differ from the optimal. Light gray dashed lines indicate the optimal selections for reference. Approximately 80% of the validation samples are associated with their optimal mesh from the library. The A-EMS model successfully captures the dominant influence of the inflow Mach number on the selection process. Moreover, no significant correlation is observed between the selection error and the distance between a sample and its (erroneously) selected mesh, suggesting that the errors in the Gaussian process models exhibit no systematic structure.

The performance of A-EMS contrasts sharply with that of the parametric distance-based mesh selection (DMS), as shown in Fig. 10(c). Specifically, associating each sample with the library element having the closest flow conditions (accounting for the relative ranges of the parameters) yields the optimal selection for only about 30% of the samples.

In addition to selection performance, the errors achieved by the A-EMS method have been analyzed. Fig. 11 shows histograms of the true errors for the optimal selection (black), A-EMS (blue), and DMS (red), along with the corresponding estimated distributions. We first observe that the error density of A-EMS closely matches that of the optimal selection, albeit with a slightly heavier tail toward higher error values. This is expected, as A-EMS correctly selects the optimal mesh for the majority of samples. The comparison of the densities indicates, however, that non-optimal selections do not lead to a substantial increase in the true error; these mis-selected meshes remain effective choices, though not optimal. Quantitatively, A-EMS exhibits a sub-optimality of less than 1% relative to the optimal selection. In contrast, the true error distribution for the distance-based selection is considerably flatter and exhibits a pronounced high-error tail, corresponding to a sub-optimality of 26.5%. Increasing the library size to $N_{\text{L}} = 32$ reduces the optimal averaged error from 0.0061 to 0.0053; A-EMS again approaches this optimal error within 1% sub-optimality, while the distance-based method improves but remains 24% sub-optimal.

The error analysis above indicates that the averaged error (≈ 0.0061) exceeds the target error $\varepsilon_{\text{ad}} = 0.004$ for all selection approaches, suggesting that $N_{\text{L}} = 16$ is insufficient to fully cover the parameter domain.

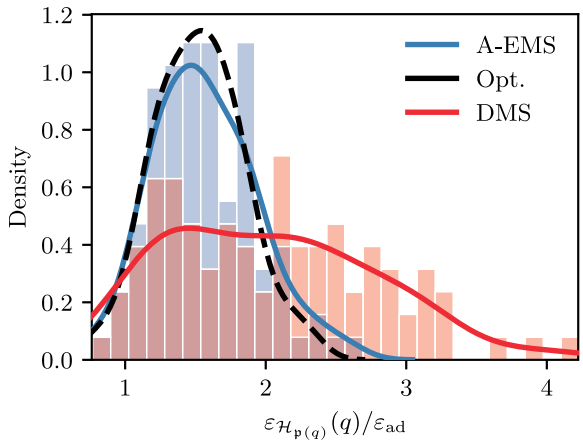


Fig. 11. True error distribution and histogram using A-EMS, DMS, and optimal selections. Scramjet case with $N_{\text{L}} = 16$.

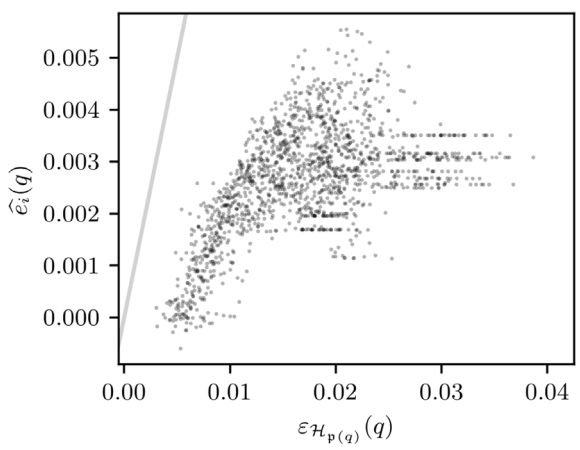


Fig. 12. Predicted error of A-EMS (GP models) against true error. Scramjet case with $N_{\text{L}} = 16$.

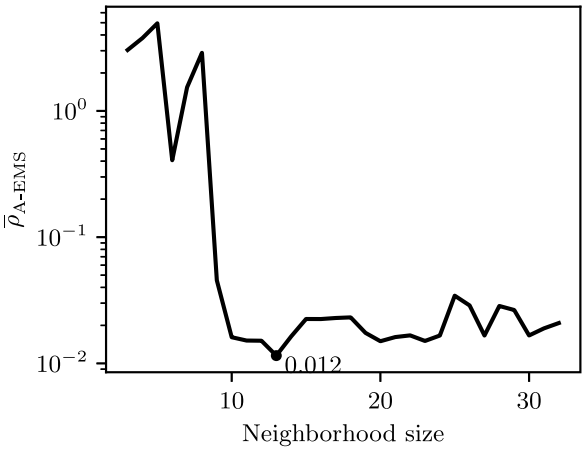


Fig. 13. Evolution of the sub-optimality measure with the size of the neighborhood used for A-EMS GP models construction. Scramjet case with $N_{\text{L}} = 32$.

To complete the analysis, we compare with alternative approaches. A naive single-mesh approach based on a hierarchical mesh containing all library meshes (the union mesh) would require roughly 10^6 cells, which is computationally prohibitive. Alternatively, the MMA method [15] can generate a single mesh capable of achieving an averaged error comparable to the optimal selection (0.0061), with $|\mathcal{H}_{\text{MMA}}| = 3 \times 10^5$ cells approximately, a size 50% larger than the average size of the adapted meshes in the library. Furthermore, to ensure an average error close to the adaptation threshold $\varepsilon_{\text{ad}} = 0.004$, MMA requires a mesh with 5×10^5 cells, an even bigger overshoot of the library's average mesh size. In contrast, by increasing the library density, A-EMS can reach ε_{ad} without increasing the average mesh size. Therefore, for this problem and the considered error level, the A-EMS method enables the use of meshes that are at least 2–3 times coarser than a single-mesh approach, provided that the library densely samples the parameter domain.

To compare with the SMA approach, we measured and compared the computational costs of both SMA and A-EMS. The cost of performing SMA to reach ε_{ad} is, on average, approximately 13.5 CPU hours per parameter value. In contrast, with A-EMS, the average cost per evaluation is only about 2.2 CPU hours and remains nearly independent of the library size N_{L}. Based on these measurements, break-even points can be estimated for different library sizes using Eq. (21). For this problem, A-EMS becomes advantageous in terms of computational cost after approximately 20 evaluations for $N_{\text{L}} = 16$, 39 evaluations for $N_{\text{L}} = 32$, and 153 evaluations for $N_{\text{L}} = 128$. As highlighted previously, relatively large libraries (i.e., 32 meshes or more) are required to reach ε_{ad} for this test case. Therefore, a realistic break-even point for the method in this example lies in the range of a few hundred evaluations. Since typical uncertainty quantification (UQ) studies involve thousands or more evaluations, A-EMS can significantly reduce their computational cost, asymptotically approaching the ratio of the average CPU times of A-EMS and SMA evaluations.

5.3. Error models investigation

To better assess the effectiveness of A-EMS, Fig. 12 presents an analysis of the predicted errors. The plot shows the GP-predicted errors at the validation samples as a function of the true error for each mesh in the library, yielding $N_{\text{L}} \times 100 = 1,600$ data points. As expected, the predicted errors tend to underestimate the true errors, since they are constructed from the restriction errors and neglect additional contributions (see Eq. (11)). In particular, the GP models predict errors below ε_{ad}, which is impossible for the true errors. Two distinct trends are observed depending on the true error magnitude. For errors smaller than ≈ 0.015, there is a reasonable correlation between the GP predictions and the true errors. This correlation deteriorates for larger errors, and the predictive capability of the GP models quickly degrades. Nevertheless, A-EMS selects sufficiently close meshes to ensure that the true error rarely exceeds 0.01, as seen in Fig. 11. Thus, the GP models provide rough error estimates that are adequate to correctly guide the mesh selection from the library.

Finally, Fig. 13 illustrates the dependence of the sub-optimality $\overline{\rho}_{\text{A-EMS}}$ on the neighborhood size used to construct the Gaussian process models. Here, $N_{\text{L}} = 32$ is employed to allow for a sufficient range of neighborhood sizes. The observed trends are consistent with those reported for the oblique shock case. For small neighborhoods, the sub-optimality is high but decreases rapidly as the neighborhood size increases. Beyond a certain size, the sub-optimality stagnates and then gradually rises. This behavior reflects the trade-off between GP models designed to capture local error structures versus those optimized for a more global fit. These results highlight the importance of carefully choosing the neighborhood size to balance local accuracy with the broader generalization capability of the GP models.

6. Conclusion

We introduced Error-based Mesh Selection (EMS) methods, to improve the efficiency of simulations performed across varying conditions. Direct application of EMS being computationally intractable, we

propose an approximate strategy: A-EMS. While EMS is formulated generically, and is therefore largely independent of the numerical method and mesh type, A-EMS relies on the availability of suitable projection operators and error estimators, which may limit its applicability depending on the discretization framework.

The method predicts, for any given condition, the discretization error associated with each mesh in a precomputed library and selects the mesh expected to minimize this error. These predictions rely on Gaussian process models of restriction errors, enabling error estimation across all library meshes without additional model solves. The approach was assessed on an analytical oblique shock problem and on a finite-volume CFD configuration involving a supersonic flow. In both cases, the restriction error proved to be a reliable surrogate for the discretization error, and the Gaussian process models enabled mesh selections close to the optimal choice, significantly outperforming simpler strategies such as parametric distance-based selection.

In the two example problems, the numerical experiments show that the average discretization error decreases rapidly as the library size increases. For instance, in the scramjet test case, a library containing approximately 32 meshes is sufficient to achieve the prescribed accuracy. Owing to the structure of A-EMS, increasing the library size has only a limited impact on the cost per evaluation, which remains low (about 2.2 CPU hours compared to 13.5 CPU hours for SMA).

However, constructing the library requires performing a set of condition-specific mesh adaptations. In our tests, the break-even point at which the benefits of A-EMS outweigh the initial library construction cost ranges from a few tens to a few hundred evaluations, depending on the library size. This makes the method particularly advantageous for large parametric studies, such as uncertainty quantification, which typically involve thousands of evaluations. In addition, moderately large libraries (e.g., up to 128 meshes for the scramjet case) are recommended, as they provide good average error convergence while preserving the computational efficiency of the method.

Despite these advantages, several challenges remain. Gaussian process modeling becomes difficult when the restriction error exhibits strong anisotropy or when the library is unevenly distributed across the parameter space. Sequential construction of the library is, therefore, a promising direction. Future work will focus on designing infilling criteria guided by GP uncertainty, to quantify mis-selection likelihood and suboptimality risk. More advanced GP modeling techniques may further improve the selection quality. Additionally, coupling A-EMS with Mean Mesh Adaptation (MMA) to compute a mean adapted mesh within each selection region represents a potential avenue to enhance robustness.

CRediT authorship contribution statement

Hugo Dornier: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Olivier P. Le Maître:** Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Formal analysis, Conceptualization. **Pietro M. Congedo:** Writing – review & editing, Methodology, Conceptualization. **Itham Salah el Din:** Writing – review & editing, Methodology, Conceptualization. **Julien Marty:** Writing – review & editing, Methodology, Conceptualization. **Sébastien Bourasseau:** Writing – review & editing, Methodology, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Hugo Dornier reports financial support was provided by direction générale de l'aviation civile. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank Niels Brun for his contribution to the development of the library creation and assessment tools used in this work.

This work was partially supported by the SONICE project, which is a France Relance project co-funded by the French Civil Aviation Authority (DGAC). France Relance benefits from EU funding via the NextGenerationEU initiative.

Data availability

The authors do not have permission to share data.

References

[1] Alauzet F, Loseille A. A decade of progress on anisotropic mesh adaptation for computational fluid dynamics. Computer-Aided Des 2016;72:13–39.
[2] Habashi WG, Dompierre J, Bourgault Y, Ait-Ali-Yahia D, Fortin M, Vallet M-G. Anisotropic mesh adaptation: Towards user-independent, mesh-independent and solver-independent CFD. Part I: General principles. Internat J Numer Methods Fluids 2000;32(6):725–44. http://dx.doi.org/10.1002/(SICI)1097-0363(20000330)32:6<725::AID-FLD935>3.0.CO;2-4.
[3] Baker TJ. Mesh adaptation strategies for problems in fluid dynamics. Finite Elem Anal Des 1997;25:243–73.
[4] Alauzet F, Frazza L, Papadogiannis D. Periodic adjoints and anisotropic mesh adaptation in rotating frame for high-fidelity RANS turbomachinery applications. J Comput Phys 2022;450:110814.
[5] Vivarelli G, Qin N, Shahpar S, Radford D. Efficient adjoint-based mesh adaptation applied to turbo-machinery flows. In: Proceedings of ASME turbo expo 2018. 2018, p. 1–14.
[6] Bijl H. Uncertainty quantification in computational fluid dynamics, In: Lecture notes in computational science and engineering ser, first ed.. (v.92). Cham: Springer International Publishing AG; 2013.
[7] Burt JM, Hagenmaier MA, Eklund DR, Wilkin HL, Josyula E. Uncertainty Quantification for a Scramjet Inlet Flow. In: 50th AIAA/ASME/SAE/ASEE joint propulsion conference. Cleveland, OH: American Institute of Aeronautics and Astronautics; 2014, p. 1–13. http://dx.doi.org/10.2514/6.2014-3844.
[8] Zhang B, Ju Y, Li Z, Zhang X, Liu Y, Zhang C. Reconstruction and reduction of realistic manufacturing error field for uncertainty quantification of transonic axial compressor rotor. J Turbomach 2026;148(1):11002. http://dx.doi.org/10.1115/1.4069132.
[9] Gillette A, Keith B, Petrides S. Learning robust marking policies for adaptive mesh refinement. SIAM J Sci Comput 2024;46(1):A264–89. http://dx.doi.org/10.1137/22M1510613.
[10] Foucart C, Charous A, Lermusiaux PF. Deep reinforcement learning for adaptive mesh refinement. J Comput Phys 2023;491:112381. http://dx.doi.org/10.1016/j.jcp.2023.112381.
[11] Van Langenhove J, Lucor D, Alauzet F, Belme A. Goal-oriented error control of stochastic system approximations using metric-based anisotropic adaptations. J Comput Phys 2018;374:384–412. http://dx.doi.org/10.1016/j.jcp.2018.07.044.
[12] Capriati M, Cortesi A, Magin TE, Congedo PM. Stagnation point heat flux characterization under numerical error and boundary conditions uncertainty. Eur J Mech / B Fluids 2022;95:221–30.
[13] Palacios F, Duraisamy K, Alonso JJ, Zuazua E. Robust Grid Adaptation for Efficient Uncertainty Quantification. AIAA J 2012;50(7):1538–46. http://dx.doi.org/10.2514/1.J051379.
[14] Barral N, Taddei T, Tifouti I. Registration-based model reduction of parameterized PDEs with spatio-parameter adaptivity. J Comput Phys 2024;499:112727. http://dx.doi.org/10.1016/j.jcp.2023.112727.
[15] Dornier H, Le Maître O, Congedo P, Salah El Din I, Marty J, Bourasseau S. Mean Mesh Adaptation for Efficient Cfd Simulations with Operating Conditions Variability. 2024, http://dx.doi.org/10.2139/ssrn.5039807.
[16] Gersho A, Gray RM. Vector quantization and signal compression. Boston, MA: Springer US; 1992, http://dx.doi.org/10.1007/978-1-4615-3626-0.
[17] Gray R. Vector quantization. IEEE ASSP Mag 1984;1(2):4–29. http://dx.doi.org/10.1109/MASSP.1984.1162229.
[18] Kumar A. Numerical analysis of the scramjet-inlet flow field by using two-dimensional Navier-Stokes equations. Technical Report NASA-TP-1940, NASA Langley Research Center; 1981.
[19] Löhner R, Morgan K, Zienkiewicz O. An adaptive finite element procedure for compressible high speed flows. Comput Methods Appl Mech Engrg 1985;51(1–3):441–65. http://dx.doi.org/10.1016/0045-7825(85)90042-8.

[20] Oden JT, Strouboulis T, Devloo P. Adaptive finite element methods for high-speed compressible flows. Internat J Numer Methods Fluids 1987;7(11):1211–28. http://dx.doi.org/10.1002/fld.1650071105.

[21] Peraire J, Vahdati M, Morgan K, Zienkiewicz O. Adaptive remeshing for compressible flow computations. J Comput Phys 1987;72(2):449–66. http://dx.doi.org/10.1016/0021-9991(87)90093-3.

[22] Roe P. Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 1981;43(2):357–72. http://dx.doi.org/10.1016/0021-9991(81)90128-5.

[23] Wesseling P. An introduction to multigrid methods. In: Pure and applied mathematics, Chichester: Wiley; 1995.

[24] Rasmussen CE, Williams CKI. Gaussian processes for machine learning, In: Adaptive computation and machine learning, 3. print. Cambridge, Mass.: MIT Press; 2008.

[25] Saves P, Lafage R, Bartoli N, Diouane Y, Bussemaker J, Lefebvre T, Hwang JT, Morlier J, Martins JR. SMT 2.0: A Surrogate Modeling Toolbox with a focus on hierarchical and mixed variables Gaussian processes. Adv Eng Softw 2024;188:103571. http://dx.doi.org/10.1016/j.advengsoft.2023.103571.

[26] Anderson JD. Fundamentals of aerodynamics, In: mcGraw-hill series in aeronautical and aerospace engineering, Sixth ed.. New York, NY: McGraw-Hill Education; 2017.

[27] Dornier H. Development of adaptive mesh refinement strategies for problems with variable parameters (Ph.D. thesis), Institut Polytechnique de Paris; 2025.

[28] Castro-Díaz MJ, Hecht F, Mohammadi B, Pironneau O. Anisotropic unstructured mesh adaption for flow simulations. Internat J Numer Methods Fluids 1997;25(4):475–91. http://dx.doi.org/10.1002/(SICI)1097-0363(19970830)25:4<475::AID-FLD575>3.0.CO;2-6.

[29] Alauzet F. Adaptation de maillage anisotrope en trois dimensions. application aux simulations instationnaires en mécanique des fluides (Ph.D. thesis), Montpellier II; 2003.

[30] Loseille A, Dervieux A, Frey P, Alauzet F. Achievement of global second order mesh convergence for discontinuous flows with adapted unstructured meshes. AIAA J 2007.

[31] Van Langenhove J. Adaptive control of deterministic and stochastic approximation errors in simulations of compressible flow (Ph.D. thesis), Paris: Université Pierre et Marie Curie - Paris VI; 2017.

[32] Cambier L, Heib S, Plot S. The ONERA elsA CFD software : Input from research and feedback from industry. Mech Ind EDP Sci 2013;14(3):159–74.