Proceedings of the Fifteenth International Conference on
Computational Structures Technology
Edited by: P. Iványi, J. Kruis and B.H.V. Topping
Civil-Comp Conferences, Volume 9, Paper 4.1
Civil-Comp Press, Edinburgh, United Kingdom, 2024
ISSN: 2753-3239, doi: 10.4203/ccc.9.4.1

Derivative-Free Trust-Region-Guided Explicit Level Set Topology Optimisation

E. K. Bontoft[1], D. Jia[1,2], Y. Zhang[1,3] and V. Toropov[1]

[1] School of Engineering and Material Science, Queen Mary University of London, London, United Kingdom
[2] State IJR Center of Aerospace Design and Additive Manufacturing, Northwestern Polytechnical University, Xi'an, China
[3] School of Aeronautics, Northwestern Polytechnical University, Xi'an, China

Abstract

This work investigates the use of explicit level set parameterisation for topology optimisation using a metamodel-based trust region strategy optimiser. The explicit level set parameterisation consists of building a uniform Design of Experiments using a Permutation Genetic Algorithm, followed by building the Level Set Function using Kriging. Through decoupling the parameterisation from the simulation physics, the use of sensitivity data becomes optional thus enabling computationally complex disciplines (where sensitivity data is not available, e.g. crashworthiness, electromagnetics) to be included. This is achieved through the use of a sequence of approximations to the functions of the original optimisation problems based on a trust region strategy. The method is demonstrated on a benchmark 2D topology optimisation problem to examine the effectiveness of the technique.

Keywords: topology optimisation, explicit level set method, derivative-free, trust region strategy, multipoint approximation method, metamodel, kriging, design of experiments, permutation genetic algorithm

1 Introduction

Metamodel-based optimisation is a methodology used to reduce the computational cost and numerical noise in optimisation problems that require a large number of complex simulations. It is particularly advantageous for problems where sensitivity data is not available and when the response functions have significant numerical noise. The quality of a metamodel is dependent on several factors: the reliability and accuracy of the response data, the effectiveness of the Design of Experiments (DoE) for gathering information for metamodel building, the size of the domain in which approximations are built relative to the entire design domain, the simulation data accuracy and the number of DoE points used to build the model [1, 2, 3].

One strategy to achieve a higher quality and more reliable metamodel is to investigate a sub-domain of the design space and employ a strategy to iteratively update the size and location of this region - known as trust region strategies. A trust region strategy was introduced as early as 1944, by Levenberg [4], where it was implemented to solve a nonlinear least squares problem using a Gauss-Newton optimiser. The trust region strategy was later rediscovered independently by Morrison [5] and Marquardt [6] hence later was termed the Levenberg-Morrison-Marquadt method in acknowledgement of all of their contributions. The Marquardt [6] method modifies Levenberg's method to incorporate a damping term to restrict the step length, and thus prevent an oversized step being taken; therefore is now also known as the Damped Least-Squares method. It was not until the review paper by Moré [7] in 1983 that 'trust region' became widely accepted terminology [8].

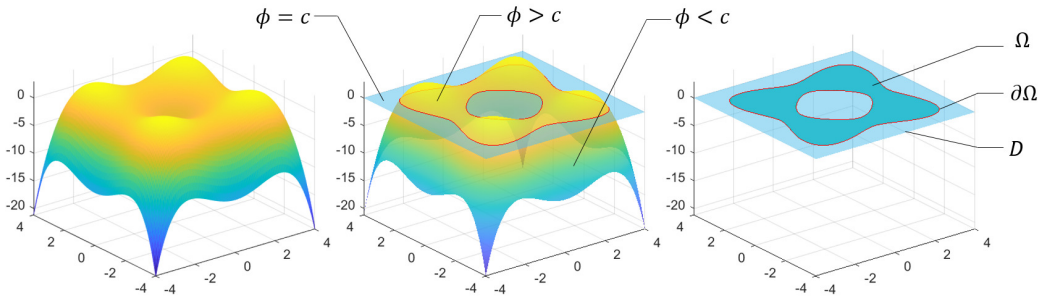

Figure 1: Level Set Function contour extraction for a 2D application

The Multipoint Approximation Method (MAM), also referred to as the Mid-range Approximation Method, is a trust region strategy metamodel-based optimiser. The heritage of the MAM dates back to Toropov [9, 10], Toropov et al. [11] and has been continuously developed to include new features [12, 13, 14]. The MAM algorithm arrives at a solution by iteratively solving approximated sub-problems in trust regions that translate and resize as the search progresses. Each iteration builds an approximated model from the simulation response data, solves

the optimisation problem, employs the trust region strategy to update the trust region's size and location, and begins the next iteration by populating the updated trust region with a DoE – with adaptions to prevent the algorithm becoming stuck in non-converging loops or converging to local minima [15, 16, 17].

Topology optimisation has taken many forms since it's inception, including: Homogenisation [18], Solid Isotropic with Material Penalisation (SIMP) [19], Level Set Method (LSM) [20], Evolutionary Structural Optimisation (ESO) [21], Ground Structure [22], Moving Morphable Components/Voids (MMC/V) [23] also referred to as Feature-Driven Optimisation (FDO) [24]. It is commonplace to use derivatives of the simulation physics' objectives and constraints – known as design sensitivities – to drive the optimisation, reducing the computational expense and enabling use of large numbers of design variables – though derivative-free topology optimisation has been in development as well [25, 26]. However, the use of sensitivities comes with a caveat, when derivatives are not available sensitivity-based approaches cannot perform. This is particularly common in Multidisciplinary Design Optimisation (MDO) applications where numerically noisy and mathematically intricate disciplines are used, e.g. crashworthiness and electromagnetics.

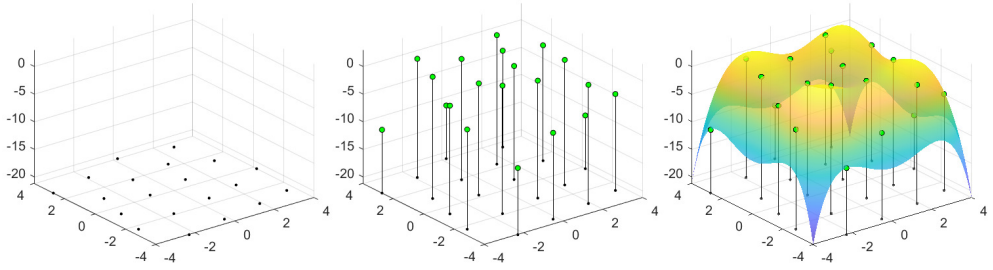

Figure 2: Design of Experiments (left), respective experiment point values (centre), and metamodel fit (right)

2 Topology Optimisation

Topology optimisation takes several forms. The most common is density-based topology optimisation, such as the SIMP method. However, density-based methods require a large number of design variables (one for each finite element within the mesh) – thus, adjoint differential equations must be solved for the response sensitivities to enhance the computational efficiency. This proves problematic for handling disciplines where the adjoint equation for the state variables is not easily derived, also when coupling several solvers is required for multidisciplinary applications [27, 28, 29].

Another primary approach to topology optimisation is the Level Set Method (LSM). The LSM is a boundary variation technique, where a level plane intersects a function, the contours depicted by the contact between the function and the 'level

set' are implicitly extracted. The mathematical description of the LSF ϕ for the boundaries at a contour can be expressed as

$$\phi(\boldsymbol{X}) > c \quad \Longleftrightarrow \quad \boldsymbol{X} \in \Omega \quad \text{(Material)}, \tag{1}$$

$$\phi(\boldsymbol{X}) = c \quad \Longleftrightarrow \quad \boldsymbol{X} \in \partial\Omega \quad \text{(Contour)}, \tag{2}$$

$$\phi(\boldsymbol{X}) < c \quad \Longleftrightarrow \quad \boldsymbol{X} \in D \quad \text{(Void)}, \tag{3}$$

where c denotes a constant (commonly set to be 0, thus sometimes referred to as the 'zero-level set'), and $\boldsymbol{X}$ denotes a position in the domain. The LSF and contour are illustrated by Figure 1 - where Ω, $\partial\Omega$ and D represent the material, contour and void regions, respectively.

2.1 Level Set Method variations

Conventional Level Set Methods (LSM) for topology optimisation employ the Hamilton-Jacobi (H-J) equation to describe the motion of implicit boundaries at the material-void interface. This process iteratively progresses to optimise the topology within a given domain, subject to specified loads and constraints. This approach has been developed in recent years by the level set community, to include diffusive operator and reaction term – which act as regularisation methods to smooth the Level Set Function (LSF) and promote stability within the technique [30]. Due to the implicit parameterisation of the LSF, these methods do not face the same limitations of parametrically or explicitly parameterised LSM techniques, such as limitations in design freedom and requirement of re-parameterisation throughout the optimisation process [31]. The H-J equation is a first order partial differential equation (PDE) that describes the motion of extremal geometries in optimisation problems. The H-J PDE is mathematically described as

$$\frac{\partial \phi}{\partial \tau} + \boldsymbol{v}_n \mid \bigtriangledown \phi \mid = 0, \tag{4}$$

where τ represents the pseudo-time (the iterations in the optimisation process), and $\boldsymbol{v}_n$ is the normal velocity field. Any particular contour configuration for the Level Set material-void boundary can be described by an infinite number of potential LSFs, therefore to ensure regularity in the LSF it is usually re-initialised every few steps. However, the process of re-initialisation is expensive, a potential source of errors in the boundary and can also prevent the nucleation of new holes in the material domain (if not addressed through an alternate scheme). Topological derivatives have been successfully incorporated into the Level Set scheme as a mechanism to enable hole nucleation, but may present a challenge due to the discontinuous nature of the topological derivatives when combined with the continuous shape derivatives [32, 33].

The LSF of conventional LSM is commonly parameterised by locally supported shape functions

$$\phi(X, t) = \sum_i \varphi_i(t) N_i(X), \tag{5}$$

where $N_i(X)$ denotes the local shape function at a node of the finite element mesh in the domain, and $\varphi(t)$ is the nodal level set value. Through decoupling the parameterisation of the LSF from the discretisation of the structural model the LSM problem can be described parametrically. This is achieved by introducing the parameterised LSF into the H-J equation. By treating the LSF as a parameterised implicit surface, the optimisation problem can be treated as a conventional parametric optimisation problem thus enabling the use of nonlinear mathematical programming techniques. This has been referred to as the Parametric Level Set Method (PLSM) in the literature and acts as solution to limitations innate to the conventional LSM, such as: difficulties in nucleating new holes, discrete representation and slow design evolution due to the Courant–Friedrichs–Lewy condition requirement in order to satisfy stability and convergence criteria of the H-J equation [34, 35, 36]. In the PLSM the LSF is parameterised by any form of basis function that is substituted into the H-J equation, such as RBF

$$\phi(\boldsymbol{X}, t) = \Phi(\boldsymbol{X})^T \boldsymbol{\alpha}(t) = \sum_i \alpha_i(t)\varphi_i(\boldsymbol{X}), \tag{6}$$

where Φ is the vector of basis functions, $\alpha_i(t)$ is the RBF expansion coefficient (weight) and $\varphi_i(\boldsymbol{X})$ is the corresponding RBF that exists in the domain space defined by $\boldsymbol{X}$ [34, 35, 36]. The governing equation for $\varphi_i(\boldsymbol{X})$ depends on the type of RBF used, such as compactly-supported [36, 37, 38], multi-quadratic [34, 35, 39], inverse multi-quadratic [40], Gaussian [41], etc. Non-uniform rational B-spline (NURBs) can also be used via the same principle [42, 43]. By substituting this equation back into the H-J equation,

$$\Phi(\boldsymbol{X})^T \frac{d\boldsymbol{\alpha}(t)}{dt} - \boldsymbol{v}_n \mid (\bigtriangledown\Phi)^T \boldsymbol{\alpha}(t) \mid = 0, \tag{7}$$

which can be broken down into a system of first order Ordinary Differential Equations (ODEs) and transformed to a parameter optimisation problem that can be solved via nonlinear programming techniques. Additionally, this parameterisation removes the requirement for re-initialisation that is innate to the conventional LSM – and thus eliminates the limitation in nucleating new holes [34, 35]. The use of shape and/or parameter sensitivity analysis is intrinsic within the H-J-based LSM; thus sharing the same limitations as density-based topology optimisation methods discussed previously [20, 44, 45, 46].

A variation of the LSM that does not utilise the H-J equation in the update procedure entirely, but rather uses an explicit parameterisation of the LSF is referred to as Explicit Level Set Methods (ELSM). However, this requires an alternative approach to describing the material-void contour – that is conventionally described by the H-J equation. Research by de Ruiter and van Keulen [47] referred to this approach as the Topology Description Function (TDF), using Radial Basis Functions (RBFs) to parameterise the LSF and a genetic algorithm as the optimiser. Gomes and Suleman [48] parameterised the LSF using coefficients of the Fourier series expansion. Kreissl et al. [49], also used RBFs to parameterise

the LSF but introduced a geometric Immersed Boundary Technique (IBT) to discretise the domain coupled with explicit smoothing. The use of the MMC/FDO to explicitly parameterise the LSF was performed by Zhang et al. [50], where the Method of Moving Asymptotes (MMA) was used to drive the optimisation.

The use of function derivatives (shape sensitivities) to drive the optimisation process within ELSM techniques is almost assumed. van Dijk et al. [51], wrote "explicit level-set methods use the sensitivities of the response functions to the nodal values of the level-set function directly in the optimisation algorithm". However, as mentioned previously, in MDO applications and when addressing disciplines with *difficult* responses, calculating sensitivities is not always possible. This work proposes a methodology to perform topology optimisation without a requirement for the sensitivity analysis. Gomes and Suleman [48, 25] demonstrated a sensitivity-free ELSM methodology via the use of a trust region strategy optimiser to treating coefficients of a Fourier series as design variables, though concluded further work in linear static topology optimisation would include parameter sensitivities.

	Conventional LSM	Parametric LSM	Explicit LSM
Parameterisation	Local shape functions	Basis Functions	RSM or MMC/V
LSF variable dependence	Implicit	Explicit	Explicit
Update Procedure	Hamilton-Jacobi equation	Nonlinear programming	Nonlinear programming
Governing equations	Partial differential equation	Ordinary differential equations	Dependent on parameterisation

Table 1: Comparison of Level Set Methods for shape and topology optimisation techniques

3 Multipoint Approximation Method

The Multipoint Approximation Method (MAM) is a trust region strategy optimiser that builds an approximated model (metamodel) of the objective function and constraint data at every iteration [9, 10, 11, 12, 52]. This work utilises the MAM as the update procedure for the ELSM for topology optimisation in order to find the design variables (LSF experiment response values displayed in Figure 2) – and in turn optimise the material domain of the Level Set plane.

Trust region strategies create a series of region within which the responses can be *trusted*. The optimum point of the trust region is found and is used to decide upon the direction and size of the next step. With each step, the trust region can be transformed and/or translated. The MAM's trust region strategy takes the centre of the trust region for the following iteration as the optimum point determined by the current step. However, in scenarios where optimum point lays on or near the boundary of the design space, the resultant trust region will exceed the design space limits. Thus, by removing the region of the trust region breaching the design space boundary, the resultant trust region shape changes and therefore so does the centre point location. The trust region translation cannot exceed the boundaries

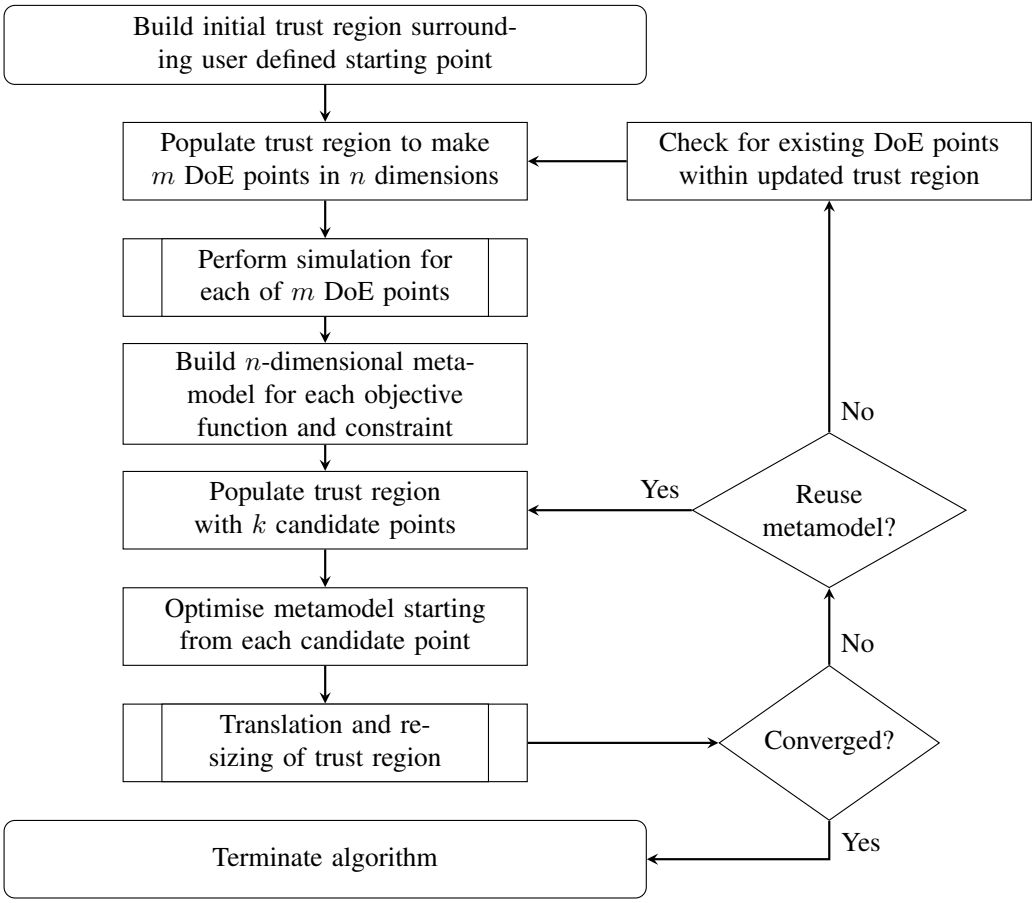

Figure 3: Multipoint Approximation Method design optimisation algorithm, where n, m and k are user-defined integers for the number of design variables, DoE points and candidate points, respectively

of the trust region – this prevents excessively large steps being taken beyond the validity of the approximated models. The use of a trust region strategy optimiser, such as the MAM, within the ELSM for topology optimisation framework improves the chances of converging to a non-local optimum, and enables solving optimisation problems with disciplines for which sensitivities are not available [53, 54, 16].

Metamodels are used to approximate a model using limited amount of data from the original model, i.e. a 'model of a model' [55]. The effects of this on a function's response are: reduced numerical noise, decrease in computational simulation time, reduced computation expense, and a reduced risk of simulation failure [56]. Metamodels are built from a series of a data points distributed throughout the current trust region - the method for distributing those points uniformly as possible is known as the Design of Experiments (DoE).

Two separate types of metamodels are used within this implementation of the Explicit Level Set representation for topology optimisation: one is a level set function metamodel (defined in the physical space), and the other is a metamodel built within a current trust region in the N-dimensional design variable space while solving a parametric optimisation problem. Here N is the number of sampling points used to define the level set function. Several metamodelling techniques have been implemented, chosen to match their advantageous attributes to their purpose. A set of design variables that describe the LSF represent an experiment point within a trust region, where the respective response function values are used to build an N-dimensional optimisation problem.

3.1 Design of Experiments (DoE)

The DoE and metamodel building used for establishing the terrain of the level set function are independent from the metamodels that are used sequentially in the trust regions of the design variable space used within MAM. The MAM objective function DoE is built via a novel Non-collapsible Latin Hypercube (NLH) method. This method entails iteratively introducing experiment points and imposing *forbidden* regions with respect to each experiment point – regions in which next experiment points should not be positioned. These regions consist of *strips* of a prescribed width that extend along the constant value for each dimension, and n-dimensional hyper-sphere sections with a defined radius surrounding the experiment points – illustrated for a two-dimensional problem in Figure 4. This method of obtaining a DoE is computationally inexpensive, and advantageous when adding points to a domain with existing points, but does not necessarily have the best space-filling properties. The forbidden regions are scaled to the size of the trust region the NLH is populating – enabling DoE points to be closer together as the MAM progresses [57, 52].

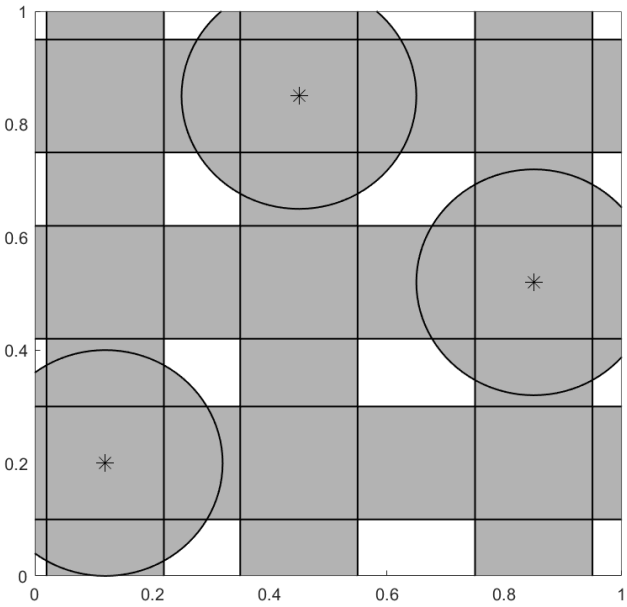

Figure 4: Schematic of the forbidden regions implemented by the Non-collapsible Latin Hypercube Design of Experiments technique in two-dimensions

3.2 Trust region metamodel

A metamodel within an n-dimensional trust region is built from a minimum of $n+1$ DoE points. The MAM algorithm is not limited to particular metamodelling techniques, thus the choice of metamodel technique is dependent on the desirable properties of the problem, e.g. lower fidelity and fast simulation, or higher fidelity with justifiably higher computational processing time. In this work, the Moving Least Square Method (MLSM) is used for this purpose, as in the work of Polynkin and Toropov [58]. The MLSM is a adaption of the Weighted Least Squares (WLS) method, which is a generalisation of the Ordinary Least Squares (OLS) method – a commonly used approach for linear regression in statistics. Least Squares methods, as the name suggests, minimise the sum of squares; i.e.

$$OLS : \sum_{i=1}^{N} \big[\tilde{f}(x_i, \boldsymbol{a}) - f(x_i)\big]^2, \tag{8}$$

$$WLS : \sum_{i=1}^{N} w_i \big[\tilde{f}(x_i, \boldsymbol{a}) - f(x_i)\big]^2, \tag{9}$$

where $f(x_i)$ and $\tilde{f}(x_i)$ denote the experiment point's values and approximated values, respectively, $\boldsymbol{a}$ denotes the vector of regression coefficients used to minimise the sum of squares, and w_i represents the weight associated with each sampling point, i. Within the MLSM, the weight is dependent on the distance between the DoE point, x_i and the point being evaluated, x_e. In this approach the weights are

calculated from a Gaussian weight decay function,

$$w_i = e^{-\theta r_i^2}, \tag{10}$$

where r_i is the distance (Euclidean norm) between the experiment point and the evaluation point, i.e. $r_i = ||x_i - x_e||$. The metamodel's surface sensitivity to the function value at experiment points is controlled by a 'closeness of fit' parameter, θ. The greater the value of θ, the more rapid is the weight decay, and therefore, the closer the fit. However, if the value of θ is too large then artificial noise can be introduced and the accuracy of the metamodel is reduced [59, 60, 61].

3.3 Optimisation within a trust region

The optimum point in the trust region is to be found. This is achieved by using a multi-start of a nonlinear programming method, which produces a set of candidate points. The candidate points are distributed throughout the trust region with the same DoE method used to populate the experiment points (NLH in this work). Starting from the candidate points a gradient-based line search optimisation via the Sequential Quadratic Programming (SQP) method [62]. The candidate points are then examined, removing candidates that converged to be duplicates (existing within predefined range), and cross-examined to the DoE points to ensure that there was no better solution. The resulting data is then used to influence the trust region strategy decision. In this work 10 candidate points have been used in each trust region.

Inequality constraints are used in this work, as equality constraints can impose too large an influence on the optimisation problem, leading to premature convergence to local minima. As one would expect improved objective function values as the constraint(s) responses increase, the optimiser will move towards the upper limit of the constraint regardless. Constraint violations are treated by a p-norm penalty function. The penalised objective function value for a set of design variables $f(\boldsymbol{s})$ is the sum of the objective function and the penalty function P, defined as

$$f(\boldsymbol{s}) = f(\boldsymbol{s}) + P, \tag{11}$$

$$P = \alpha \times \sum_i [max(0, g_i - 1)]^\beta, \tag{12}$$

where P is the sum of constraint violations for i constraints, subject to coefficient and exponent, α and β. Constraint values are scaled such that feasible (non-violating) values exist in the range of $0 \leq gi \leq 1$. In this work, the coefficient $\alpha = 1$, and the exponent $\beta = 1$ resulting in a linear trend with a low gradient. This is done to provide the optimisation problem with as much design freedom as possible, to reduce likelihood of converging to a local minima, yet still includes a convex vertex to encourage the optimiser to converge to exactly the upper limit of the constraint(s).

3.4 Trust region strategy

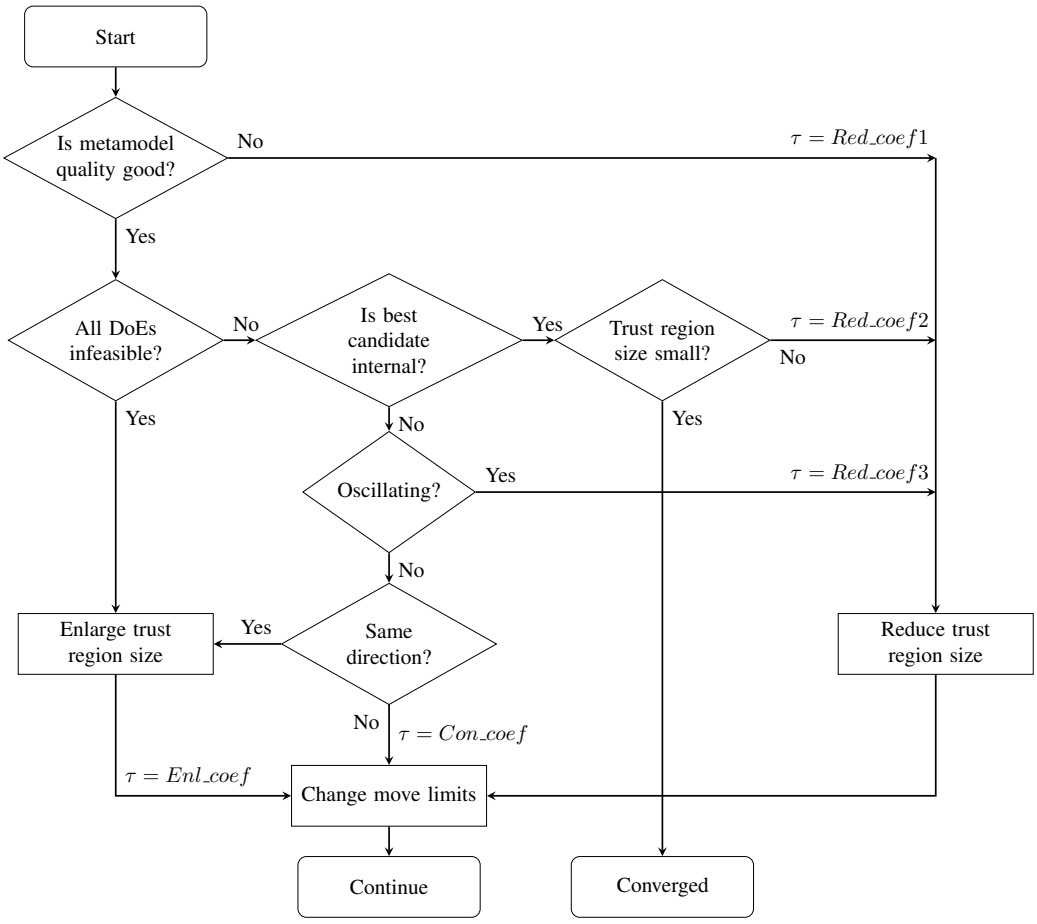

Figure 5: Trust region strategy within the Multipoint Approximation Method

The trust region strategy is the definitive decision making process of the MAM algorithm. The trust region strategy decides: the termination or continuation of the optimisation loop; the transformation and translation of the trust region; and the reuse of metamodels. This utilises information from the metamodel quality, trust region size, optimal point (location and direction) and DoE points to drive the decisions made by the trust region strategy at each iteration, as displayed in Figure 5.

The quality of the produced metamodel is then evaluated, calculated from the Root Mean Square Error (RMSE), ϵ,

$$\epsilon = \sqrt{\frac{\sum_{i=1}^{m}(y_i - \tilde{y}_i)^2}{m}}, \tag{13}$$

for m experiment points; where $\tilde{y}_i$ and y_i represent the approximated and actual response values, respectively. Predefined quality values for a *good* (ϵ_{good}) and *very good* ($\epsilon_{v.good}$) RMSE are used by the trust region strategy.

The trust region size, r_k, for the current iteration, k, is calculated as a percentage of the entire design space (set by the bounds of the design variables). As the trust region size is used to make decisions within the trust region strategy, threshold values need to be set. A sufficiently small trust region size, r_{small}, for acceptable convergence is defined by the user. Additionally, a trust region minimum size, r_{min}, is defined to that indicate when $r_k < r_{min}$ convergence has not been met due to an error in the problem formulation, and to terminate the MAM.

The location of the optimal point of the current iteration is evaluated to determine if it is in internal, or lies on the bounds of the trust region. An internal point, where $A_i^k < x_i < B_i^k$ (where A_i^k and B_i^k denote the lower and upper bounds of the trust region at iteration k, respectively), indicates that the optimisation process is approaching the true optimum.

The angle, α_k, between the prior two move vectors (using three prior optimal points) of the trust region's movement is calculated. The magnitude of this angle is defined as a non-dimensional parameter,

$$\Theta_k = cos(\alpha_k) = \frac{\boldsymbol{v}_{k-1} \cdot \boldsymbol{v}_k}{\mid \boldsymbol{v}_{k-1} \mid \cdot \mid \boldsymbol{v}_k \mid}, \tag{14}$$

where $\boldsymbol{v}$ denotes the move vector for the current iteration, k, and the prior iteration, $k-1$. The value of Θ_k is monitored to account for predictable patterns in the movement of the trust region; such as oscillating directions ($\Theta_k \leq \Theta_{min}$) and continuous movement in one direction ($\Theta_k \geq \Theta_{max}$ for l iterations), as shown in Figure 6.

Next, it is determined whether all DoE and candidate points within the current iteration's trust region violate a constraint. If this is the case then it suggests that the trust region has moved into a region of the design space that does not have any feasible solutions.

Parameter	Value
ϵ_{good}	5%
$\epsilon_{v.good}$	0.5%
r_{small}	1%
r_{min}	0.2%
Θ_{max}	0
Θ_{min}	0.8
l	10

Table 2: Trust region strategy quality parameters

The convergence criteria of the MAM are defined within the trust region strategy. The MAM will converge when the following criteria are met:

1. the trust region size is sufficiently small ($r \leq r_{small}$)
2. feasible point has been found ($g_j(\boldsymbol{s}) \leq 0$)

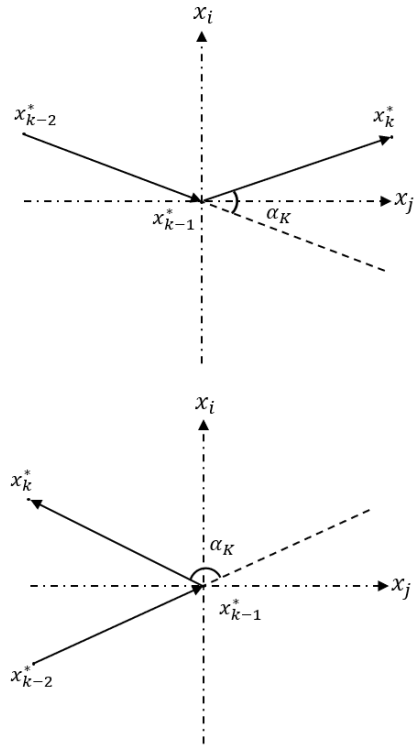

Figure 6: Oscillation angles of optimum points, where top and bottom denote when $\Theta_k > 0$ and $\Theta_k < 0$, respectively

3. the metamodel is of good quality ($\epsilon \leq \epsilon_{good}$)

4. the optimal point is internal ($A_i^k < x_i < B_i^k$)

where r denotes the trust region size; ϵ is the Root Mean Squared Error of the metamodel (a quantitative parameter for the metamodel quality); and the boundaries of the point of the domain are represented by A_i^k and B_i^k, all calculated for point x_i in iteration i.

τ	Value	Reuse metamodel	Reuse condition
Red_coef1	8.0×10^-1	No	N/A
Red_coef2	7.5×10^-1	No	N/A
Red_coef3	8.0×10^-1	No	N/A
Enl_coef	1.5×10^-0	Yes	$\Theta_k \geq \Theta_{max}$
Con_coef	1.0×10^-0	Yes	$\epsilon \leq \epsilon_{v.good}$

Table 3: Trust region strategy sizing coefficients used in this work

3.5 Capabilities

The MAM has been shown to have a nearly linear dependence of the number of function evaluations to the number on design variables. Thus, the ability to handle large numbers of design variables with a relatively low number of function calls thus reducing the computational cost. Recent research performed by Gergel et al. [63] tailored the MAM's parallel computing capabilities, testing problems up to 1000 design variables.

In this work the MAM architecture has been utilised to perform multidisciplinary topology optimisation. This is feasible due to the MAM's handling of function approximations for individual disciplines. Such metamodels can be combined in a single optimisation problem, which can then be solved to determine the solution of a multidisciplinary problem [17, 14].

4 Methodology and Results

In the following section the methodology for ELSM for topology optimisation using the MAM is proposed. Results for a simple linear static topology optimisation benchmark case – Michell single load – are presented.

4.1 Parameterisation

The proposed methodology is an Explicit Level Set representation for topology optimisation – where the function values obtained at the Design of Experiment (DoE) points are used to build a metamodel that represents the Level Set Function (LSF). The parameterisation process involves an initial DoE within the LSF design space, that remains the same for all sets of design variables, and a metamodel build with respect to each set of design variables.

4.1.1 Design of Experiments

Capturing the LSF efficiently is paramount to this methodology – as the number of DoE points used represents the number of design variables within the optimisation process (MAM). Thus, an effective space-filling DoE should be established. To achieve this, this work uses a DoE obtained by a permutation Genetic Algorithm (GA) [64]. The permutation GA is coupled with an Optimum Latin Hypercube DoE. This principle refers to the distribution of DoE points in each dimension being separated by uniform intervals, with only one DoE point positioned at each interval – illustrated by Figure 7.

The optimisation problem the genetic algorithm solves is the minimisation of an Audze-Eglais objective function. This function represents a physical analogy: points within the domain exert repulsive forces on each other representing a

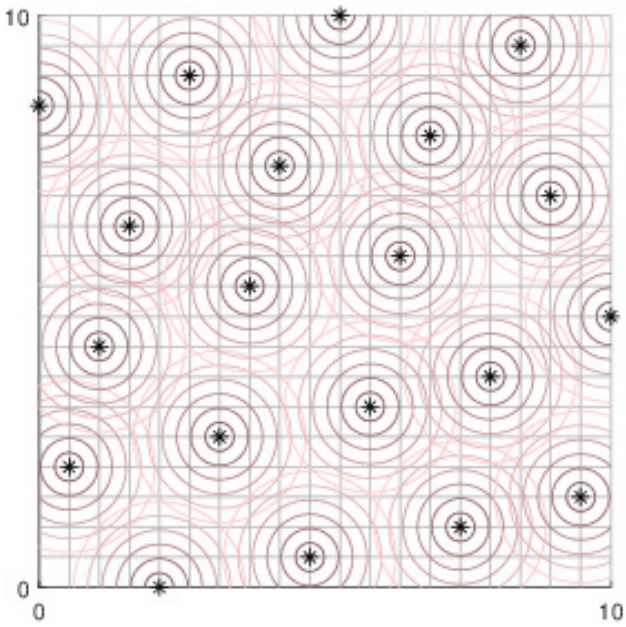

Figure 7: Design of Experiments via the Permutation Genetic Algorithm for 20 points in two-dimensions, displaying the Latin Hypercube distribution and pseudo-potential energy acting as a repulsive force

pseudo-potential energy, U. It can be mathematically expressed as

$$minU = min \sum_{p=1}^{P} \sum_{q=p+1}^{P} \frac{1}{L_{pq}^2}, \tag{15}$$

where L_{pq} is the distance between points p and q, for a system containing P points. The term 'permutation' refers to the rearrangement or change of order of existing elements – this is preferable for DoE application as it can be used to satisfy Optimum Latin Hypercube criteria. Although Genetic Algorithms are expensive when using large numbers of design variables they have desirable properties when working with (1) binary-based problems, and (2) populations of points. However, within this work, the DoE that describes the LSF is only required to occur once, at the very beginning of the algorithm. Thus, the computational expense can be accepted in return for an optimally distributed DoE, which in-turn reduces the cost of the optimisation procedure by reducing the number of design variables required to produce a LSF metamodel of sufficient quality [65, 64, 66].

4.1.2 Level Set Function metamodel

The LSF metamodel is built using the Ordinary Kriging method [67, 68] – a spacial correlation-based technique that builds an interpolating metamodel. The approximated Kriging metamodel, $f'(\boldsymbol{x})$, is built from an estimated mean, $\hat{\mu}$, deter-

mined by a weighted least squares problem, and an error term, $\epsilon(\boldsymbol{x})$,

$$f'(\boldsymbol{x}) = \hat{\mu} + \epsilon(\boldsymbol{x}). \tag{16}$$

Kriging runs on the underlying assumption that experiment point data is deterministic, and approximation errors only exist due to missing terms in the model. Additionally, Kriging assumes the error is continuous for a continuous function. The error, ψ, between two points (i & j) is assumed to be correlated with their distance, and (in this work) modelled as Gaussian basis functions

$$\psi(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp\left[\sum_{k=1}^{n} -\theta^{(k)} \left(\left|\left|\boldsymbol{x}_i^{(k)} - \boldsymbol{x}_j^{(k)}\right|\right|\right)^2\right], \tag{17}$$

where $\boldsymbol{x}$ denotes the location of a point and θ and k are tuning parameters. The error models between an evaluation point, e, and all experiment points ($i = 1, ..., p$) are used to build the vector, $\boldsymbol{r}$,

$$\boldsymbol{r}(\boldsymbol{x}_e, \boldsymbol{x}_i) = \begin{bmatrix} \psi(\boldsymbol{x}_e, \boldsymbol{x}_1), \\ \vdots \\ \psi(\boldsymbol{x}_e, \boldsymbol{x}_p) \end{bmatrix}. \tag{18}$$

The error term for any experiment point, ϵ_i, is calculated from a stochastic process, described as

$$\epsilon(\boldsymbol{x}_i) = \boldsymbol{w}^T \boldsymbol{r}, \tag{19}$$

where $\boldsymbol{w}$ denotes a matrix comprised of the weights for each experiment point basis function $\boldsymbol{r}$. The weights, $\boldsymbol{w}$, are calculated from the estimated spacial correlation (ψ_i) of all training points, and the function values of each experiment point [69].

Kriging acts as an exact interpolator, which is attractive in deterministic simulation [70, 3]. It can be noted that Kriging loses it's computational efficiency when approaching problems with several thousand DoE points [71], however, such number of design variables is far beyond the scope of the proposed methodology.

4.2 Mechanical modelling

Erzatz material [72, 35], also known as element material fraction [73] and a density field [74, 30], is a method for mapping and discretising the mechanical model simultaneously. A fixed regular grid is used for the discretised grid, where a pseudo-density value is used to denote the fraction of material present in each element. This method is the most commonly used in the literature, motivated by its ease of implementation and efficiency.

This investigation opted for an Ersatz material mechanical model. This was due to the likelihood of material becoming disconnected when a mesh does not encompass the entirety of the domain. The Ersatz material *density* values are

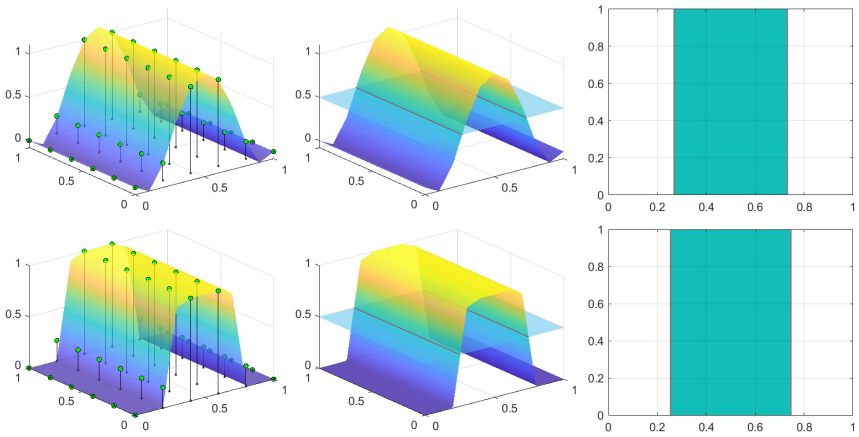

Figure 8: Smoothed Heaviside function with β values of 1 (top) and 10 (bottom), displaying the resulting level set cut and material domain

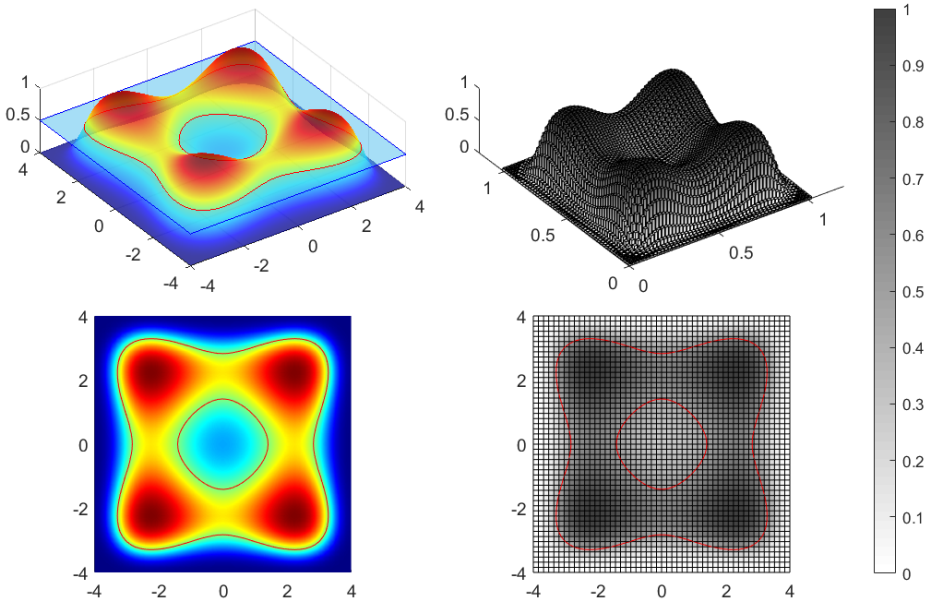

Figure 9: Parameterisation of a surface function as an Ersatz material model

calculated using a series of *integration* points throughout the mesh. Where each element consists of a set (S) of: 4, 9 or 16 integration points – shown in Fig. 10. The integration points evaluate the function value of the LSF, $\phi(\boldsymbol{x})$, and the pseudo-density element value is calculated as the mean value of each element's respective integration points, as

$$\rho_i\big(\phi(\boldsymbol{x})\big) = \frac{\sum\limits_{j \in S_i} \phi_j(\boldsymbol{x})}{\sum\limits_{j \in S_i} 1}, \tag{20}$$

where $\phi(\boldsymbol{x}) \in [0, 1]$, and ρ_i denotes the i^{th} element's pseudo-density value.

4.3 Update procedure

The update procedure for the proposed method is driven by the MAM, as discussed in Section 3. Each DoE point within the MAM's trust regions denotes a set of design variables that are used to build the LSF, and thus describe the topological configuration.

Using the contours from the Level Set cut to map the geometry, it is common at the initial stages of the MAM for the DoE points to provide topological configurations that do not produce material on the regions where boundary conditions are applied; or if material is present in these regions, it is disconnected from the rest of the structure. This results in poorly formulated topology optimisation problems that are not properly constrained, causing the topology to not satisfy static equilibrium conditions, i.e. in motion. In such instances, the objective function cannot be calculated. The MAM can handle such DoE points by assigning an artificially large objective response, but still calculating any other response values that are available (e.g. the volume fraction). This information provides the trust region metamodel with the data to move away from these instances – thus, after the first few iterations they become infrequent enough not to hamper the optimisation process. However, this method is not computationally efficient, as an artificially large

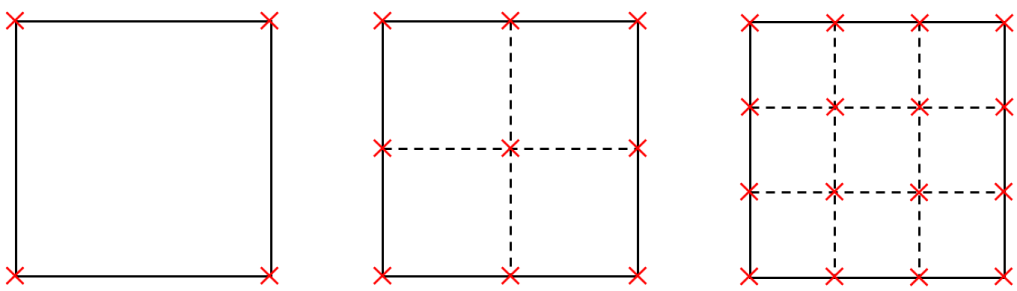

Figure 10: Element integration point locations for: 4 (left), 9 (centre), and 16 (right)

objective function can encourage convergence to local minima. The Ersatz material model overcomes this limitation by introducing a minimum element density, thus, all resultant models are feasible to solve for the response. This motivated the choice of an Ersatz material model within this research.

The MAM trust region strategy utilises a parameter that is the minimum acceptable number of DoE points within a trust region. This is to ensure the trust region has sufficient data for the current optimisation problem. The MAM will first check if any existing DoE points are present within the trust region and include them, counting them towards the total number of DoE points within the trust region. The trust region is then populated with the remaining number of DoE points required to make the minimum acceptable number.

4.4 Regularisation

Schemes to manipulate the optimisation problem to create a well-posed problem are known as regularisation. Regularisation can be performed at any stage in the Level Set Method. In this work, a smoothed Heaviside function is used to reduce intermediate values within the LSF. In doing so, this creates steep gradients on the walls of the LSF, providing a clear indication of the where the level set is cut, and thus the material-void boundary will lie. The Level Set Function transformed by the smoothed Heaviside function, $\hat{H}$, can be described as

$$\bar{\phi} = \hat{H}(\phi) = \frac{tanh(\beta\eta) + tanh(\beta(\phi - \eta))}{tanh(\beta\eta) + tanh(\beta(1 - \eta))} \tag{21}$$

where β and η are coefficients that control the sharpness of the projection, as shown in Figure 8. In this work, the value of β is linked to the size the of the trust region. The value of β increases linearly from an initial value of 1, at the initial trust region size, to a final value of 10, at the final trust region size. This is performed to ensure that by the end of the optimisation process the LSF has clearly defined boundaries.

Density filtering schemes can be implemented to impose a relationship wherein the density of an element is dependent on distances to surrounding elements and their respective densities. Surrounding elements are considered within a given radius. A set of elements, N, can be expressed as

$$N_i = j \ : \ d(i,j) \leq R, \tag{22}$$

where

$$d(i,j) = ||\boldsymbol{x}_i - \boldsymbol{x}_j||, \tag{23}$$

for a given point (i) within a neighbourhood bounded by the radius, R. The aug-

mented density ($\tilde{\rho}$) value for the given point is calculated as

$$\tilde{\rho}_i = \frac{\sum_{j \in N_i} w_{ij}^{(d)} v_j \rho_j}{\sum_{j \in N_i} w_{ij}^{(d)} v_j}, \tag{24}$$

where

$$w_{ij}^{(d)} = \begin{cases} f\big(d(i,j)\big) & \text{for } j \in N_i, \\ 0 & \text{for } j \notin N_i, \end{cases} \tag{25}$$

v denotes the volume of an element, and $w_{ij}^{(d)}$ is the distance-dependent weight. This formulation produces the condition that

$$\sum_{j \in N_i} w_{ij} = 1 \; \forall \; i, \tag{26}$$

which is consistent for the weighting functions explored in this work [75].

A Gaussian distribution can be utilised to provide a smoothed nonlinear weighting function in the shape of a bell-curve. The Gaussian exponential function is mathematically described by

$$w_{ij}^{(d)} = \exp\left(-\frac{1}{2}\left(\frac{d(i,j)}{\sigma_d}\right)^2\right), \tag{27}$$

where σ_d denotes a scaling constant, with a value commonly set to be $R/2$, and $R/3$ – in this work $R/3$ is used.

Density-based weight functions can be incorporated to determine a relationship between the density of the point being considered and that of the points within a given neighbourhood: wherein the less the change in the density, the greater the weight of the function. The Bi-lateral Density Filter was introduced by [76], the filtering scheme utilises a distance-based and density-based weight function. This can be mathematically expressed as

$$\tilde{\rho}_i = \frac{\sum_{j \in N_i} w_{ij}^{(d)} w_{ij}^{(\rho)} v_j \rho_j}{\sum_{j \in N_i} w_{ij}^{(d)} w_{ij}^{(\rho)} v_j}, \tag{28}$$

where $w_{ij}^{(\rho)}$ denotes the density-based weight function. The distance-based weighting is calculated via the Gaussian distribution introduced above. The density-based weights, too, are calculated using a Gaussian distribution, as

$$w_{ij}^{(\rho)} = \exp\left(-\frac{1}{2}\left(\frac{||\rho_i - \rho_j||}{\sigma_\rho}\right)^2\right), \tag{29}$$

where $w_{ij}^{(\rho)} = 0 \ \forall \ j \notin N_i$, and σ_ρ denotes a scaling constant that $\in [0, 1]$. The greater the value of σ_ρ, the less the weight of the filter and, thus, the greater the region of intermediate densities [75]. A value of 0.5 has been used for σ_ρ in this work. Another scheme implemented to regularise the optimisation problem is a re-initialisation of the trust region throughout the optimisation process. This is performed due to the highly non-convex data present in topology optimisation problems which can lead to convergence to local minima. Re-initialising the problem can guide the trust region effectively regardless of it's starting position. When the trust region becomes sufficiently small, the trust region size is restarted ($r_{restart}$) to the initial size, r_0, but multiplied by a reduction coefficient, $\tau_{restart}$, as

$$r_{restart} = r_0 \tau_{restart}^N \tag{30}$$

where N denotes the number of times the trust region has been restarted. The re-initialisation scheme is included into the trust region strategy by checking the best response before restart, and comparing to the best response from the prior restart. If difference between the responses is less than a small tolerance of each other, the restart scheme is stopped. Alternatively, if the restart trust region size becomes smaller than the threshold size to trigger the re-initialisation the scheme is stopped, and the trust region strategy proceeds as normal.

4.5 Michell single load benchmark

The method proposed in this work is applied to a two-dimensional linear static structural topology optimisation problem – Michell one-load. The Michell benchmark problem was initially formulated as a truss optimisation problem that could be analytically solved. This has lead to the extensive use of Michell structure-based problems in topology optimisation benchmarking [77, 78].

It should be noted that although it is common practice to use the terminology 'optimal' within the structural optimisation community, that two-dimensional topology optimisation Michell problems do not effectively demonstrate the optimal configuration of material in three-dimensional space. Instead, the material distribution should be a closed wall structure with a variable topography [77]. However, two-dimensional solutions can serve as an effective influence to such reinforcement patterns.

The Michell one load benchmark case for topology optimisation (displayed in Figure 11) simulates a centrally loaded beam via imposing boundary conditions that act as a line of symmetry. It is an effective problem for testing topology optimisation techniques, as it requires the material to be connected and in contact with both constraints in order to satisfy static equilibrium conditions. The material used for the benchmark is steel ASTM A-36, in accordance with the benchmark data for topology optimisation performed by Valdez et al. [79] with the required data presented in table 4.

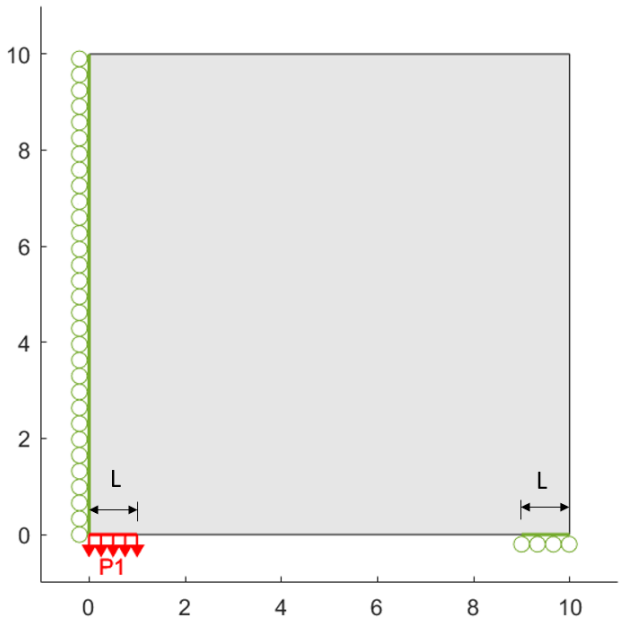

Figure 11: Michell benchmark topology optimisation problem

Parameter	Value
Domain $x \times y$ (m)	1×1
L (m)	0.1
P1 (N)	-5.6×10^4
Young's Modulus (Pa)	2.11×10^9
Poisson's ratio	0.29
Material density (kg/m^3)	7874

Table 4: Michell single load benchmark problem set up

4.5.1 General Problem Formulation

The general problem formulation is to minimise the compliance, c, (i.e. achieve maximum global stiffness), with a constraint on the volume fraction, V, set to be 50% ($\bar{V} = 0.5$). This optimisation problem can be expressed as

$$\text{minimise:} \qquad f(\boldsymbol{x}) = c\big(\boldsymbol{\rho}(\boldsymbol{x})\big) = F^T U, \qquad \boldsymbol{x}, \boldsymbol{\rho} \in \mathbb{R}^n, \quad (31)$$

$$\text{subject to:} \qquad g_V(\boldsymbol{\rho}) = \int_\Omega f dV = \sum_{e=1}^{p} \rho_e \leq \bar{V}, \qquad e = 1, \ldots, p, \quad (32)$$

$$\text{w.r.t.:} \qquad \underline{x}_i \leq x_i \leq \overline{x}_i, \qquad i = 1, \ldots, n, \quad (33)$$

where F and U are the load and displacement vectors, respectively. The design variables, $\boldsymbol{x}$, exists within limits, set to be $[\underline{x}_i, \overline{x}_i] = [0, 1]$. The LSF domain ($\Omega$) size is defined in Table 4, where the limits are $[0, 1]$ in both dimensions.

The problem is composed of a set of elements, e, each with a respective density value, ρ_e, where $e = 1, \ldots, p$, and $0 \leq \rho \leq 1$. As the density of such elements serve as a fraction of the material present in this region, the sum of the elements' density values equates to the effective volume fraction for the domain.

The optimisation problem can be further developed to consider another constraint with respect to the quantity of element densities in an intermediate region – this will be referred to informally as the *discreteness* constraint. The discreteness constraint (g_D) is a function of the density of the elements, where elements with density values in the intermediate regions of the density limits have a greater value. This can be mathematically described as

$$g_D(\boldsymbol{\rho}) = \frac{\sum_{e=1}^{p} - sin(\pi\rho_e - \frac{\pi}{2})^{2\beta} + 1}{p} \leq \bar{D} \quad (34)$$

where p denotes the total quantity of elements (e) in the domain, and β is a penalisation parameter that influences the size of the *intermediate* density region. The discreteness constraint limit, $\bar{D}$, is investigated through this work, but a common value used for the constraint is 0.2.

4.5.2 Multipoint Approximation Method Problem Formulation

The MAM framework is formulated as a series of optimisation sub-problems – the trust region metamodels. This involves reformulating the general optimisation

problem as

$$\begin{aligned}
&\text{minimise:} && \tilde{f}^{(\kappa)}\big(\boldsymbol{\rho}(\boldsymbol{x})\big), && \boldsymbol{x}, \boldsymbol{\rho} \in \mathbb{R}^n, && (35)\\
&\text{subject to:} && g_V(\boldsymbol{\rho}) \leq \bar{V}, && && (36)\\
& && g_D(\boldsymbol{\rho}) \leq \bar{D}, && && (37)\\
&\text{w.r.t.:} && \left.\begin{aligned} A_i^{(\kappa)} \leq x_i \leq B_i^{(\kappa)},\\ A_i^{(\kappa)} \leq A_i,\\ B_i^{(\kappa)} \leq B_i, \end{aligned}\right\} && i = 1, \ldots, n, && (38)
\end{aligned}$$

where $\tilde{f}^{(\kappa)}$ denotes the approximated response function by the metamodel, at iteration, κ. $A_i^{(\kappa)}$ and $B_i^{(\kappa)}$ respectively denote the trust region lower and upper limits for each design variable, i. The relative trust region size, r, at each iteration can be calculated by

$$r^{(\kappa)} = \frac{1}{n}\sum_{i=1}^{n} \frac{\|B_i^{(\kappa)} - A_i^{(\kappa)}\|}{\|B_i^{(1)} - A_i^{(1)}\|}. \tag{39}$$

4.5.3 Post-Processing

A scheme is implemented in this method's framework to further refine the topology optimisation results. This scheme can be thought of as post-processing - although it is a part of the algorithm - but occurs when the initial topology optimisation problem converges. The post-processing involves introducing a new set of design variables to act as the Euclidean coordinates of the DoEs that build the LSF. Theoretically, by enabling the DoEs points to 'move' in the domain space, they have a greater capacity to shape the contours of the topology defined by the Level Set cut. The design variables of the optimisation problem are concatenated as

$$\boldsymbol{s} = \{\boldsymbol{x}||\boldsymbol{y}||\boldsymbol{z}\} = \{x_1, ..., x_n, y_1, ..., y_n, z_1, ..., z_n\}, \tag{40}$$

where $\boldsymbol{y}$ and $\boldsymbol{z}$ denote the two-dimensional Euclidean coordinates for each respective design variable's DoE point. The design variables from the initial optimisation problem, $\boldsymbol{x}$, are set to have the value of the optimum solution, $\boldsymbol{x}^*$; the new design variables are set to have starting values of the DoE coordinates from the prior optimisation problem. The optimisation problem can be reformulated using the new set of design variables, $\boldsymbol{s}$.

4.6 Results

The starting postion of an optimisation problem is an important feature to be considered. It can heavily influence the likelihood of convergence to a local minimum. The starting postion of the optimisation method LSMs corresponds to the initial LSF used.

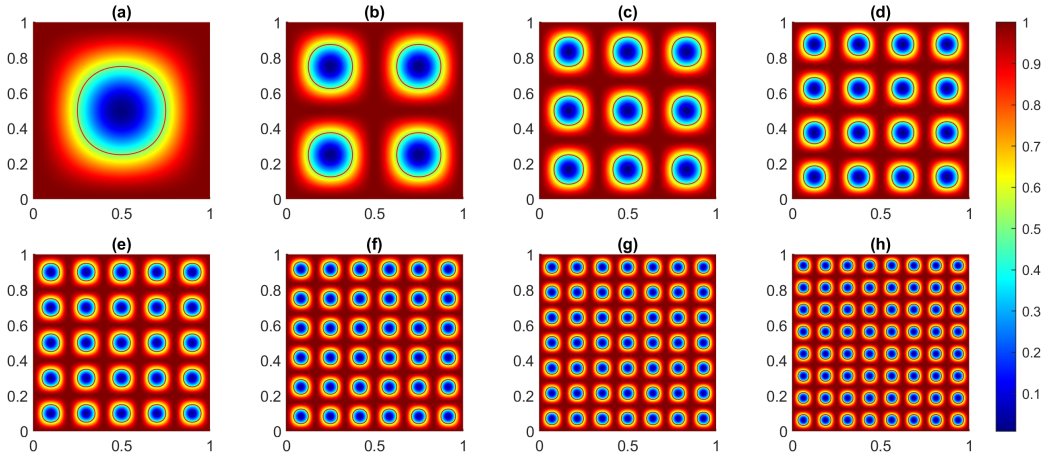

Figure 12: Initial LSF for a variable number of holes, where (a)-(h) respectively correspond to a set of $\{1, 4, 9, 16, 25, 36, 49, 64\}$ holes

In this study, a variable quantity of holes within the initial LSF and how this influences the solution of the optimisation problem is examined. Figure 12 presents the initial function used to calculate the design variable values that build the initial LSF. Figure 13 displays the topological configuration produced by the solution of the optimisation problem, wherein (a)-(h) correspond to the initial functions in Fig. 12.

The results of the study, presented in Tables 5 & 6, show that the initial configuration of the LSF does influence the likelihood of convergence to a local minimum using the proposed method. Where plot (a) of Fig. 13, that corresponds to an initial LSF with one hole, suffers significantly from this initial configuration. However, beyond this, the initial configuration has a reduced significance. The best solution was produced by the problem with an initial LSF that has 9 holes, closely followed by the solution produced by 64 holes – presented in plots (c) & (h). Both solutions resemble a Michell structure like configuration, with the same quantity of truss-like members.

5 Summary and conclusions

The work done in this research looked to develop a method to realise topology optimisation solutions without the requirement of sensitivity analysis and using relatively few design variables. This was motivated by the suitability of use for the method developed – the MAM4TO – to be applied to problems considering engineering disciplines where sensitivites are not available (such as crashworthiness and electromagnetics). The MAM4TO was developed using an Explicit-LSM parameterisation – where the design variables directly correspond to the basis functions used to build the LSF. This was coupled with an Ersatz material model, where the LSF was projected onto the discretised grid as pseudo-density values.

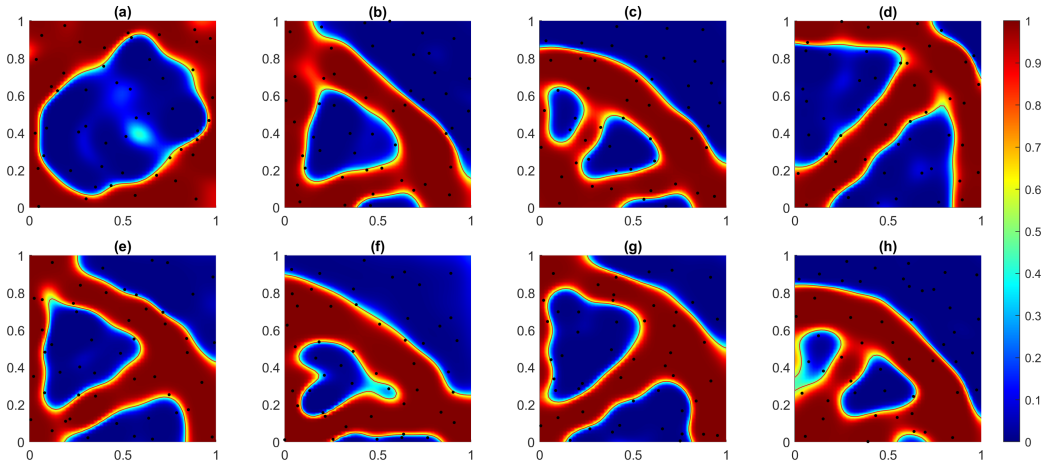

Figure 13: Solutions corresponding to the initial functions presented for in Fig. 12, using the baseline optimisation problem with a Bi-lateral filter and filter radius of 0.05 and 50 desgin variables

Holes	$f(\boldsymbol{x}_0)$	$g_V(\boldsymbol{x}_0)$	$g_D(\boldsymbol{x}_0)$	$f(\boldsymbol{x}_\kappa^{(*)})$	$g_V(\boldsymbol{x}_\kappa^{(*)})$	$g_D(\boldsymbol{x}_\kappa^{(*)})$	κ
1	0.2126	0.7414	0.4757	0.5323	0.5020	0.2002	151
4	0.2140	0.7157	0.6074	0.2770	0.4998	0.1991	136
9	0.2637	0.5759	0.9158	0.2589	0.4998	0.1986	135
16	0.2531	0.6362	0.9018	0.3156	0.4999	0.1928	116
25	0.3825	0.4971	0.9355	0.2699	0.4999	0.1990	111
36	0.5068	0.5616	0.8949	0.2672	0.4999	0.1945	108
49	0.2154	0.6708	0.8108	0.2826	0.4999	0.1993	121
64	0.2389	0.6582	0.7926	0.2655	0.4996	0.1999	124

Table 5: Solutions to the initial optimisation problem for a variable quantity of holes in the initial LSF

Holes	$f(\boldsymbol{s}_\kappa^{(*)})$	$g_V(\boldsymbol{s}_\kappa^{(*)})$	$g_D(\boldsymbol{s}_\kappa^{(*)})$	κ	$\Delta f^{(*)}(\%)$	$\Delta g_V^{(*)}(\%)$	$\Delta g_D^{(*)}(\%)$
1	0.5795	0.5001	0.1984	50	8.8580	-0.3828	-0.8954
4	0.2709	0.4998	0.1995	39	-2.1996	-0.0123	0.1688
9	0.2589	0.4999	0.1998	44	0.0374	0.0300	0.5972
16	0.3041	0.4999	0.1992	48	-3.6311	0.0033	3.2998
25	0.2634	0.5000	0.1988	49	-2.3968	0.0249	-0.1254
36	0.2822	0.4998	0.2000	49	5.6353	-0.0117	2.7976
49	0.2739	0.5000	0.1966	39	-3.0922	0.0172	-1.3547
64	0.2594	0.4998	0.1997	35	-2.3158	0.0473	-0.0671

Table 6: Comparison of the initial and post-processing optimisation solutions for the problems presented in Fig. 13 and Table 5.

The update proceedure was achieved using the MAM TRS.

A methodology of techniques to regularise the topology optimisation proceedure was presented – wherein a study on the influence of such techniques was then performed. The results examined the performance using a benchmark topology optimisation problem, the Michell single load problem. The results achieved are in accordance with those presented in the literature. However, certain decificies were highlighted in elements of the MAM4TO method where numerical instabilities became present in larger optimisation problems.

The results achieved in this work presented that the solutions produced by the MAM4TO benefited from the inclusion of a post-processing stage in the optimisation process. During this stage, the basis functions used to build the LSF metamodel are endowed with the capacity to move within the design domain. This aided the refinement of the topological configurations produced by the initial stage of the optimisation process (wherein the basis functions are static). Furthermore, the presence of a *discreteness* constraint was seen to considerably benefit the optimisation solutions, by encouraging the design variables to converge to near discrete values – therein reducing the presence of intermediate density material.

Acknowledgements

Elliot Bontoft would like to thank Chenée Psaros, for the unwavering compassion and kindness she exhibits in her coaching.

References

[1] W. J. Roux, N. Stander, and R. T. Haftka. Response surface approximations for structural optimization. *International Journal for Numerical Methods in Engineering*, 42(3):517–534, 1998.

[2] F. A. C. Viana and R. T. Haftka. Cross validation can estimate how well prediction variance correlates with error. *AIAA Journal*, 47(9):2266–2270, 2009.

[3] M. G. Fernández-Godino, C. Park, N. H. Kim, and R. T. Haftka. Issues in deciding whether to use multifidelity surrogates. *AIAA Journal*, 57(5): 2039–2054, 2019.

[4] K. Levenberg. A method for the solution of certain nonlinear problems in the least squares. *Quarterly Applied Mathematics*, 2(2):164–168, 1944.

[5] D. D. Morrison. Methods for nonlinear least squares problems and convergence proofs. In *Proceedings of Tracking Programs and Orbit Determination, Jet Propulsion Lab.*, pages 1–9, 1960.

[6] D. Marquardt. An algorithm for least squares estimation of nonlinear parameters. *SIAM Journal of Applied Mathematics*, 11(431-441), 1963.

[7] J. J. Moré. *Recent Developments in Algorithms and Software for Trust Region Methods*, book section 10, pages 258–287. Springer Berlin Heidelberg, Berlin, Heidelberg, 1983.

[8] Y. X. Yuan. Recent advances in trust region algorithms. *Mathematical Programming*, 151(1):249–281, 2015.

[9] V. V. Toropov. Multipoint approximation method in optimization problems with expensive function values. In *Proceedings of the 4th international symposium on Systems analysis and simulation*, pages 207–212, 1992. ISBN 0-444-89780-1.

[10] V. V. Toropov. Multipoint approximation method for structural optimization problems with noisy function values abstract. *Stochastic Programming*, 1995.

[11] V. V. Toropov, A. A. Filatov, and A. A. Polynkin. Multiparameter structural optimization using fem and multi-point explicit approximations. *Structural Optimization*, 6(1):7–14, 1993.

[12] V. V. Toropov, F. van Keulen, V. L. Markine, and L. F. Alvarez. Multipoint approximations based on response surface fitting: A summary of recent developments. In *Association for Structural and Multidisciplinary Optimization*, pages 371–380. International Society of Structural and Multidisciplinary Optimization, 1999.

[13] S. Shahrokh, A. A. Polynkin, and V. V. Toropov. Large scale optimization of transonic axial compressor rotor blades. In *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2008.

[14] J. Ollar, R. D. Jones, and V. V. Toropov. Sub-space metamodel-based multidisciplinary optimization of an aircraft wing subjected to bird strike. In *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2017.

[15] V. V. Toropov, F. van Keulen, V. Markine, and H. de Boer. Refinements in the multi-point approximation method to reduce the effects of noisy structural responses. *AIAA Journal*, 1994.

[16] F. van Keulen, V. V. Toropov, and V. Markine. Recent refinements in the multi-point approximation method in conjunction with adaptive mesh refinement. In *ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, volume 3, 1996.

[17] A. A. Polynkin, V. V. Toropov, and S. Shahrokh. Adaptive and parallel capabilities in the multipoint approximation method. In *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2008.

[18] M. P. Bendsoe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197–224, 1988.

[19] M. Zhou and G. Rozvany. The coc algorithm, part ii: Topological, geometrical and generalized shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 89:309–336, 1991.

[20] J. A. Sethian. *Level Set and Fast Marching Methods: evolving interfaces in computational geometry, fluid mechanics, computer vision and material science*. Cambridge University Press, Cambridge, 1999.

[21] Y. M. Xie and G. P. Steven. A simple evolutionary procedure for structural optimization. *Computers & Structures*, 49(5):885–896, 1993.

[22] V. O. Balabanov and R. T. Haftka. Topology optimization of transport wing internal structure. *Journal of aircraft*, 33(1):232–233, 1995.

[23] X. Guo, W. S. Zhang, and W. L. Zhong. Doing topology optimization explicitly and geometrically-a new moving morphable components based framework. *Journal of Applied Mechanics-Transactions of the Asme*, 81(8), 2014.

[24] Y. Zhou, W. H. Zhang, J. H. Zhu, and Z. Xu. Feature-driven topology optimization method with signed distance function. *Computer Methods in Applied Mechanics and Engineering*, 310:1–32, 2016.

[25] A. A. Gomes and A. Suleman. Application of spectral level set methodology in topology optimization. *Structural and Multidisciplinary Optimization*, 31 (6):430–443, 2006.

[26] O. Querin, M. Victoria, C. Alonso, R. Ansola, and M. Pascual. *Topology Design Methods for Structural Optimization*. Elsevier Ltd., 2017. ISBN 978-0-08-100916-1.

[27] M. Stolpe and K. Svanberg. On the trajectories of penalization methods in topology optimization. *Structural and Multidisciplinary Optimization*, 21 (128-139), 2001.

[28] M. P. Bendsoe and O. Sigmund. *Topology Optimisation - Theory, Methods and Applications*. Springer, 2003. ISBN 3-540-42992-1.

[29] P. D. Dunning, C. J. Brampton, and H. A. Kim. Multidisciplinary level set topology optimization of the internal structure of an aircraft wing. *10th World Congress on Structural and Multidisciplinary Optimization*, 2013.

[30] N. P. van Dijk, K. Maute, M. Langelaar, and F. van Keulen. Level-set methods for structural topology optimization: a review. *Structural and Multidisciplinary Optimization*, 48(3):437–472, 2013.

[31] M. Y. Wang, X. M. Wang, and D. M. Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1-2):227–246, 2003.

[32] M. Burger, B. Hackl, and W. Ring. Incorporating topological derivatives into level set methods. *Journal of Computational Physics*, 194(1):344–362, 2004.

[33] G. Allaire and F. Jouve. Coupling the level set method and the topological gradient in structural optimization. In *IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials*, volume 137 of *Solid Mechanics and Its Applications*, pages 3–12. Springer, Dordrecht, 2006. ISBN 978-1-4020-4729-9.

[34] M. Wang and S. Wang. Parametric shape and topology optimization with radial basis functions. In *IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials*, volume 137, pages 13–22, 2006.

[35] S. Wang and M. Y. Wang. Radial basis functions and level set method for structural topology optimization. *International Journal for Numerical Methods in Engineering*, 65(12):2060–2090, 2006.

[36] Z. Luo, L. Tong, M. Y. Wang, and S. Wang. Shape and topology optimization of compliant mechanisms using a parameterization level set method. *Journal of Computational Physics*, 227(1):680–705, 2007.

[37] Z. Luo, M. Y. Wang, S. Wang, and P. Wei. A level set-based parameterization method for structural shape and topology optimization. *International Journal for Numerical Methods in Engineering*, 76(1):1–26, 2008.

[38] Z. Luo, L. Tong, and H. Ma. Shape and topology optimization for electrothermomechanical microactuators using level set methods. *Journal of Computational Physics*, 228(9):3173–3181, 2009.

[39] S. Y. Wang, K. M. Lim, B. C. Khoo, and M. Y. Wang. An extended level set method for shape and topology optimization. *Journal of Computational Physics*, 221(1):395–421, 2007.

[40] X. Xing, M. Y. Wang, and B. F. Y. Lui. Parametric shape and topology optimization with moving knots radial basis functions and level set methods. In *Proceedings of 7th World Congress of Structural and Multidisciplinary Optimization (WCSMO7)*, pages 1928–1936, 2007.

[41] H. Li, Z. Luo, L. Gao, and J. Wu. An improved parametric level set method for structural frequency response optimization problems. *Advances in Engineering Software*, 126:75–89, 2018.

[42] Y. Wang and D. J. Benson. Isogeometric analysis for parameterized lsm-based structural topology optimization. *Computational Mechanics*, 57(1): 19–35, 2015.

[43] H. A. Jahangiry and S. M. Tavakkoli. An isogeometrical approach to structural level set topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 319:240–257, 2017.

[44] G. Allaire, F. Jouve, and A. M. Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363–393, 2004.

[45] S. Osher and R. P. Fedkiw. *The Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2004. ISBN 0-387-95482-1.

[46] L. Jiang and S. K. Chen. Parametric structural shape & topology optimization with a variational distance-regularized level set method. *Computer Methods in Applied Mechanics and Engineering*, 321:316–336, 2017.

[47] M. J. de Ruiter and F. van Keulen. Topology optimization: Approaching the material distribution problem using a topological function description. In *Computational Techniques for Materials, Composites and Composite Structures*, pages 111–119, 372497, 2000. Civil-Comp press.

[48] A. A. Gomes and A. Suleman. Spectral level set methodology in mdo. In *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference*, pages 1–14, 2004.

[49] S. Kreissl, G. Pingen, and K. Maute. An explicit level set approach for generalized shape optimization of fluids with the lattice boltzmann method. *International Journal for Numerical Methods in Fluids*, 65(5):496–519, 2011.

[50] W. S. Zhang, W. Y. Yang, J. H. Zhou, D. Li, and X. Guo. Structural topology optimization through explicit boundary evolution. *Journal of Applied Mechanics-Transactions of the ASME*, 84(1), 2017.

[51] N. P. van Dijk, M. Langelaar, and F. Keulen. Explicit level-set-based topology optimization using an exact heaviside function and consistent sensitivity analysis. *International Journal for Numerical Methods in Engineering*, 91 (1):67–97, 2012.

[52] V. V. Toropov, Y. Korolev, K. Barkalov, E. Kozinov, and V. Gergel. Hpc implementation of the multipoint approximation method for large scale design optimization problems under uncertainty. In *EngOpt 2018 Proceedings of*

the 6th International Conference on Engineering Optimization, pages 296–306. Springer International Publishing, 2018.

[53] V. V. Toropov and V. L. Markine. The use of simplified numerical models as mid-range approximations. In *6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, pages 952–958, 1996.

[54] F. van Keulen and V. V. Toropov. New developments in structural optimization using adaptive mesh refinement and multipoint approximations. *Eng. Opt.*, 29, 1997.

[55] J. P. C. Kleijnen. A comment on blanning's metamodel for sensitivity analysis: The regression metamodel in simulation. *Interfaces*, 5:21–23, 1975.

[56] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. K. Tucker. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41(1):1–28, 2005.

[57] Y. Korolev, V. V. Toropov, and S. Shahpar. Large-scale cfd optimization based on the ffd parametrization using the multipoint approximation method in an hpc environment. In *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, pages 1–13, 2015.

[58] A. A. Polynkin and V. V. Toropov. Mid-range metamodel assembly building based on linear regression for large scale optimization problems. *Structural and Multidisciplinary Optimization*, 45(4):515–527, 2011.

[59] V. V. Toropov, U. Schramm, A. Sahai, R. D. Jones, and T. Zeguer. Design optimization and stochastic analysis based on the moving least squares method. In *6th World Congresses of Structural and Multidisciplinary Optimization*, 2005.

[60] C. Bucher and T. Most. A comparison of approximate response functions in structural reliability analysis. *Probabilistic Engineering Mechanics*, 23 (2-3):154–163, 2008.

[61] R. Garimella. A simple introduction to moving least squares and local regression estimation. Report LA-UR-17-24975, Los Alamos National Laboratory, 2017.

[62] R. B. Wilson. *A simplicial algorithm for concave programming*. Phd thesis, Harvard University, 1963.

[63] V. P. Gergel, K. A. Barkalov, E. A. Kozinov, and V. V. Toropov. Parallel multipoint approximation method for large-scale optimization problems. In L. Sokolinsky and M. Zymbler, editors, *Parallel Computational Technologies*, pages 174–185. Springer International Publishing, 2018. ISBN 978-3-319-99673-8.

[64] S. J. Bates, J. Sienz, and V. V. Toropov. Formulation of the optimal latin hypercube design of experiments using a permutation genetic algorithm. In *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference*, pages 1–7, 2004.

[65] S. J. Bates, J. Sienz, and D. S. Langley. Formulation of the audze-eglais uniform latin hypercube design of experiments. *Advances in Engineering Software*, 34(8):493–506, 2003.

[66] D. Z. Liu and V. V. Toropov. A lamination parameter-based strategy for solving an integer-continuous problem arising in composite optimization. *Computers & Structures*, 128:170–174, 2013.

[67] D. Krige. *A statistical approach to some mine valuation and allied problems on the Witwatersrand*. Masters thesis, University of Witwatersrand, 1951.

[68] G. Matheron. Principles of geostatics. *Economic Geology*, 58:1246–1266, 1963.

[69] J. Ollar, C. Mortished, R. D. Jones, J. Sienz, and V. V. Toropov. Gradient based hyper-parameter optimisation for well conditioned kriging metamodels. *Structural and Multidisciplinary Optimization*, 55(6):2029–2044, 2016.

[70] J. P. C. Kleijnen. Kriging metamodeling in simulation: A review. *European Journal of Operational Research*, 192(3):707–716, 2009.

[71] V. Dubourg, B. Sudret, and J. M. Bourinet. Reliability-based design optimization using kriging surrogates and subset simulation. *Structural and Multidisciplinary Optimization*, 44(5):673–690, 2011.

[72] G. Allaire, F. Jouve, and A. M. Toader. A level-set method for shape optimization. *Comptes Rendus Mathematique*, 334(12):1125–1130, 2002.

[73] P. D. Dunning, B. K. Stanford, and H. A. Kim. Aerostructural level set topology optimization for a common research model wing. In *10th AIAA Multidisciplinary Design Optimization Conference, AIAA SciTech Forum*, 2014.

[74] O. Sigmund and K. Maute. Topology optimization approaches a comparative review. *Structural and Multidisciplinary Optimization*, 48(6):1031–1055, 2013.

[75] O. Sigmund. Morphology-based black and white filters for topology optimization. *Structural and Multidisciplinary Optimization*, 33(4-5):401–424, 2007.

[76] M. Y. Wang and S. Wang. Bilateral filtering for structural topology optimization. *International Journal for Numerical Methods in Engineering*, 63 (13):1911–1938, 2005.

[77] O. Sigmund, N. Aage, and E. Andreassen. On the (non-) optimality of michell structures. *Structural and Multidisciplinary Optimization*, 54(2): 361–373, 2016.

[78] A. V. Pichugin, A. Tyas, and M. Gilbert. Michell structure for a uniform load over multiple spans. In *9th World Congress on Structural & Multidisciplinary Optimization*. Sheffield, 2011.

[79] S. I. Valdez, V. Cardoso, S. Botello, M. A. Ochoa, and J. L. Marroquin. Topology optimization benchmarks in 2d: Results for minimum compliance and minimum volume in planar stress problems. *Archives of Computational Methods in Engineering*, 24(4):803–839, 2016.