Optimization of Composite Structures by Estimation of Distribution Algorithms

Paris Research Center, October 5, 2004

Laurent Grosset

grosset@emse.fr

Advisors:

Raphael T. Haftka, Department of Mechanical and Aerospace Engineering, University of Florida

Rodolphe Le Riche, Département Mécanique et Matériaux, École Nationale Supérieure des Mines de Saint-Étienne

Outline

- Résumé des travaux (en français)

Outline

- Résumé des travaux (en français)
- Introduction to composite laminate optimization

Outline

- Résumé des travaux (en français)
- Introduction to composite laminate optimization
- Introduction to Estimation of Distribution Algorithms (EDA)

Outline

- Résumé des travaux (en français)
- Introduction to composite laminate optimization
- Introduction to Estimation of Distribution Algorithms (EDA)
  - Principles

Outline

- Résumé des travaux (en français)
- Introduction to composite laminate optimization
- Introduction to Estimation of Distribution Algorithms (EDA)
  - Principles
  - Importance of the statistical model

Outline

- Résumé des travaux (en français)
- Introduction to composite laminate optimization
- Introduction to Estimation of Distribution Algorithms (EDA)
  - Principles
  - Importance of the statistical model
- Application of a simple EDA to laminate optimization

Outline

- Résumé des travaux (en français)
- Introduction to composite laminate optimization
- Introduction to Estimation of Distribution Algorithms (EDA)
  - Principles
  - Importance of the statistical model
- Application of a simple EDA to laminate optimization
- Introduction of variable dependencies through auxiliary variables and the Double-Distribution Optimization Algorithm

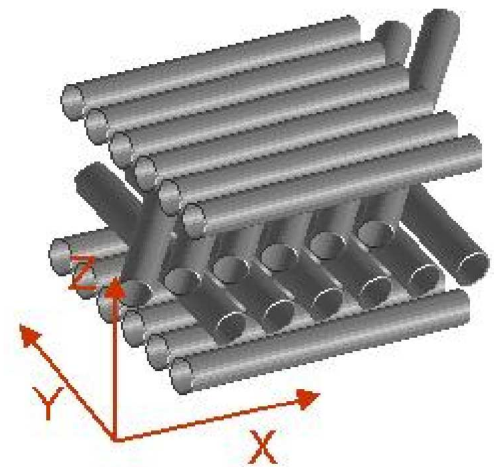
Outline

- Résumé des travaux (en français)
- Introduction to composite laminate optimization
- Introduction to Estimation of Distribution Algorithms (EDA)
  - Principles
  - Importance of the statistical model
- Application of a simple EDA to laminate optimization
- Introduction of variable dependencies through auxiliary variables and the Double-Distribution Optimization Algorithm
- Conclusions

Résumé des Travaux

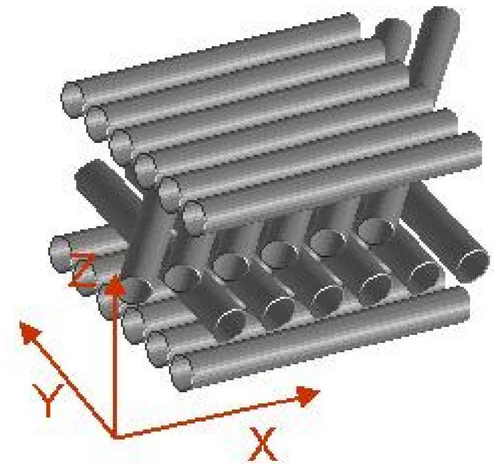
Problématique:
Optimisation de stratifiés composites

- Stratifié composite : empilement de couches de renforts (fibres) noyées dans la résine (polymère)



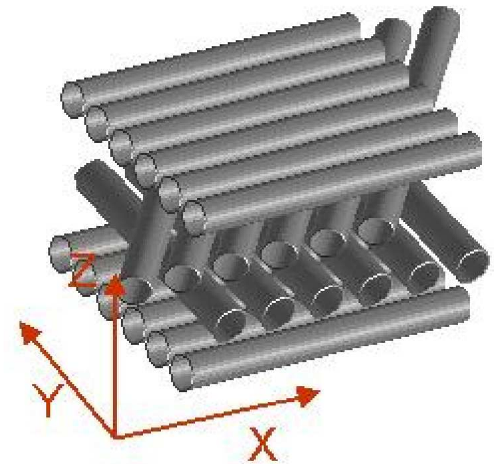
Problématique:
Optimisation de stratifiés composites

- Stratifié composite : empilement de couches de renforts (fibres) noyées dans la résine (polymère)
- Exemples : aéronautique (ex. Boeing 7E7), construction navale, équipements sportifs,. . .



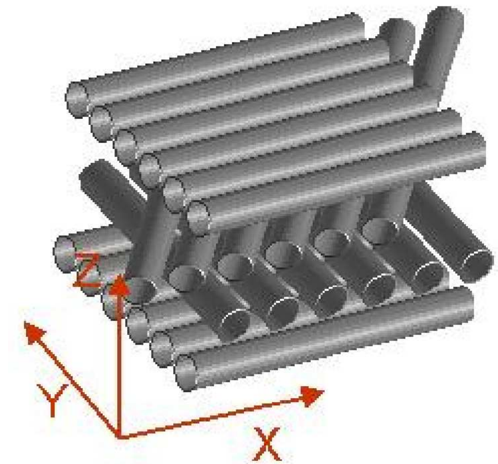
Problématique:
Optimisation de stratifiés composites

- Stratifié composite : empilement de couches de renforts (fibres) noyées dans la résine (polymère)
- Exemples : aéronautique (ex. Boeing 7E7), construction navale, équipements sportifs,. . .
- La réponse dépend de l'orientation des fibres



Problématique:
Optimisation de stratifiés composites

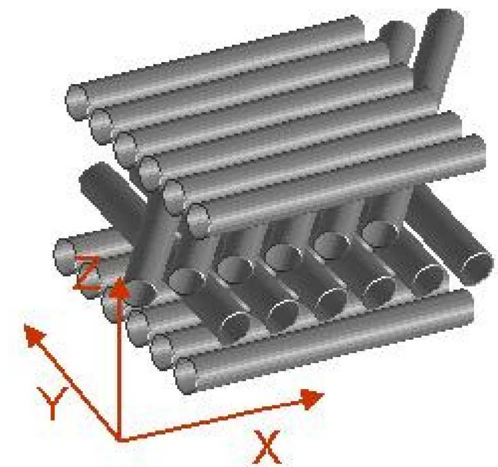
- Stratifié composite : empilement de couches de renforts (fibres) noyées dans la résine (polymère)
- Exemples : aéronautique (ex. Boeing 7E7), construction navale, équipements sportifs,. . .
- La réponse dépend de l'orientation des fibres



- But de l'optimisation = déterminer l'orientation optimale de toutes les couches pour une application particulière :

Problématique:
Optimisation de stratifiés composites

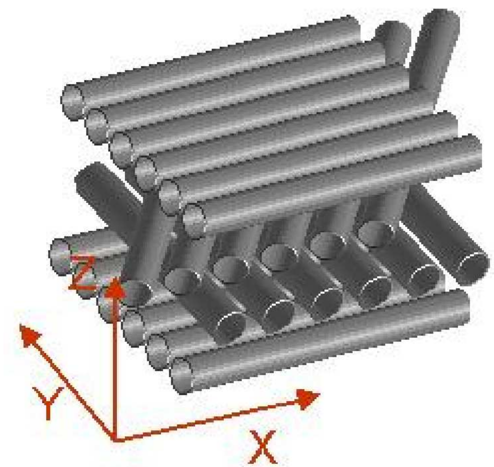
- Stratifié composite : empilement de couches de renforts (fibres) noyées dans la résine (polymère)
- Exemples : aéronautique (ex. Boeing 7E7), construction navale, équipements sportifs,. . .
- La réponse dépend de l'orientation des fibres



- But de l'optimisation = déterminer l'orientation optimale de toutes les couches pour une application particulière :
  - maximiser la résistance d'un élément de structure d'un avion

Problématique:
Optimisation de stratifiés composites

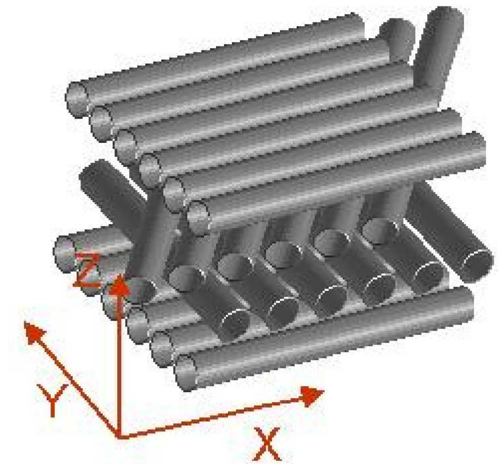
- Stratifié composite : empilement de couches de renforts (fibres) noyées dans la résine (polymère)
- Exemples : aéronautique (ex. Boeing 7E7), construction navale, équipements sportifs,. . .
- La réponse dépend de l'orientation des fibres



- But de l'optimisation = déterminer l'orientation optimale de toutes les couches pour une application particulière :
  - maximiser la résistance d'un élément de structure d'un avion
  - minimiser le poids

Problématique:
Optimisation de stratifiés composites

- Stratifié composite : empilement de couches de renforts (fibres) noyées dans la résine (polymère)
- Exemples : aéronautique (ex. Boeing 7E7), construction navale, équipements sportifs,. . .
- La réponse dépend de l'orientation des fibres



- But de l'optimisation = déterminer l'orientation optimale de toutes les couches pour une application particulière :
  - maximiser la résistance d'un élément de structure d'un avion
  - minimiser le poids
  - déterminer la séquence d'empilement de la coque d'une voiture de course de manière à minimiser les vibrations

État de l'art, objectifs de la thèse

- Méthodes traditionnelles basées sur les gradients mais problèmes continus uniquement et recherche locale

État de l’art, objectifs de la thèse

- Méthodes traditionnelles basées sur les gradients mais problèmes continus uniquement et recherche locale
- Optimisation stochastique pour globalité (algorithmes génétiques, années 90)

État de l'art, objectifs de la thèse

- Méthodes traditionnelles basées sur les gradients mais problèmes continus uniquement et recherche locale
- Optimisation stochastique pour globalité (algorithmes génétiques, années 90)
- La thèse fait suite à des travaux du groupe de Prof. Haftka:

État de l'art, objectifs de la thèse

- Méthodes traditionnelles basées sur les gradients mais problèmes continus uniquement et recherche locale
- Optimisation stochastique pour globalité (algorithmes génétiques, années 90)
- La thèse fait suite à des travaux du groupe de Prof. Haftka:
  - R. Le Riche : application d'algorithmes évolutionnaires à l'optimisation de structures composites (1994)

État de l'art, objectifs de la thèse

- Méthodes traditionnelles basées sur les gradients mais problèmes continus uniquement et recherche locale
- Optimisation stochastique pour globalité (algorithmes génétiques, années 90)
- La thèse fait suite à des travaux du groupe de Prof. Haftka:
  - R. Le Riche : application d'algorithmes évolutionnaires à l'optimisation de structures composites (1994)
  - B. Liu: optimisation globale de structures composites par algorithmes multi-niveaux (2001)

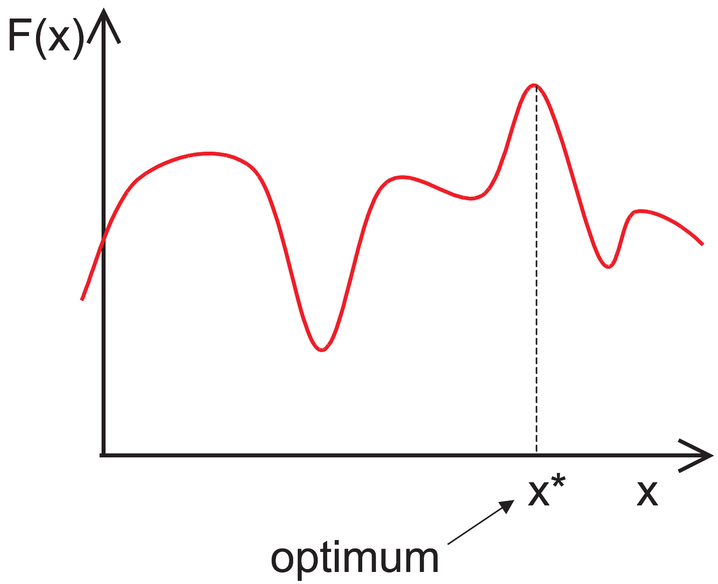
État de l'art, objectifs de la thèse

- Méthodes traditionnelles basées sur les gradients mais problèmes continus uniquement et recherche locale
- Optimisation stochastique pour globalité (algorithmes génétiques, années 90)
- La thèse fait suite à des travaux du groupe de Prof. Haftka:
  - R. Le Riche : application d'algorithmes évolutionnaires à l'optimisation de structures composites (1994)
  - B. Liu: optimisation globale de structures composites par algorithmes multi-niveaux (2001)

Objectifs de la thèse :

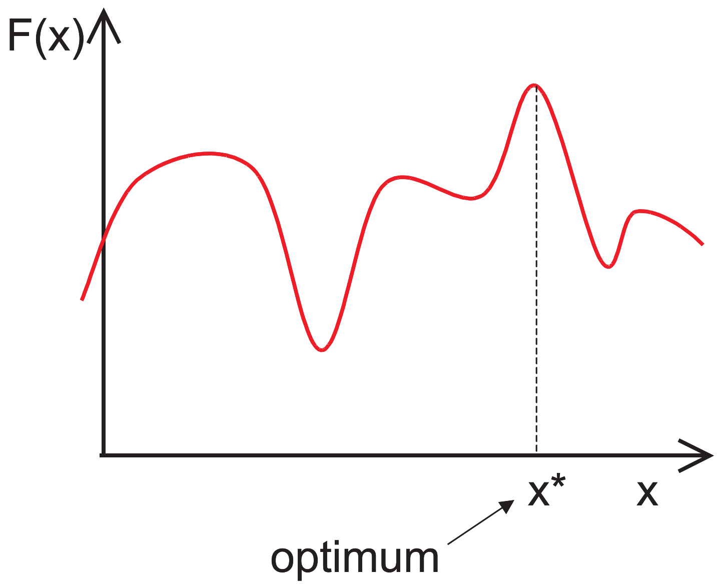
- Maturation des méthodes évolutionnaires depuis 20 ans (ES, EDA)
- Transférer ces nouvelles méthodes à l'optimisation de composites :
  - utilisation plus simple
  - méthodes plus efficaces

Algorithmes à estimation de distribution



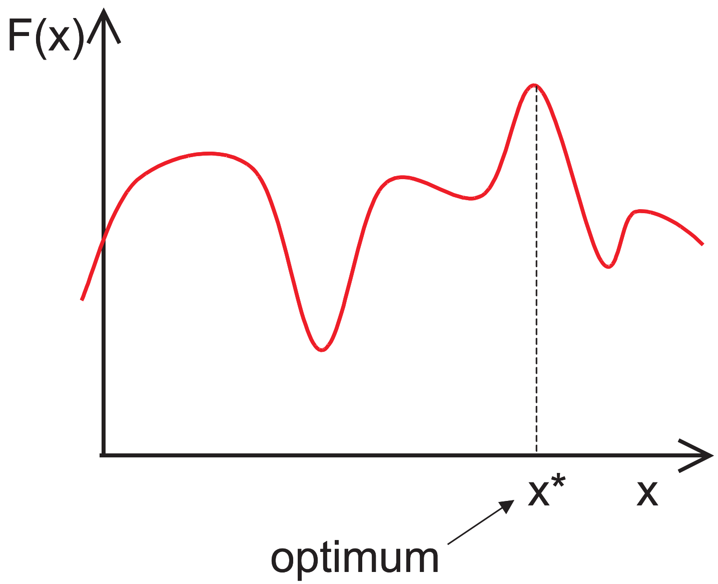
- But : trouver le point de plus élevé d'une "fonction coût" F sur un domaine $\mathcal{D}$

Algorithmes à estimation de distribution



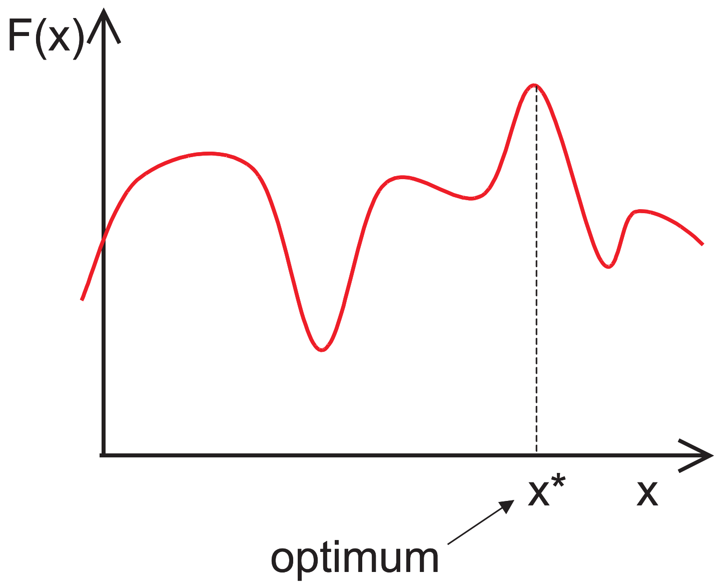
- But : trouver le point de plus élevé d'une "fonction coût" F sur un domaine $\mathcal{D}$
- Difficulté : on ne dispose que d'un budget limité de N évaluations de la fonction F (on ne peut pas calculer toutes les combinaisons pour choisir la meilleure!)

Algorithmes à estimation de distribution



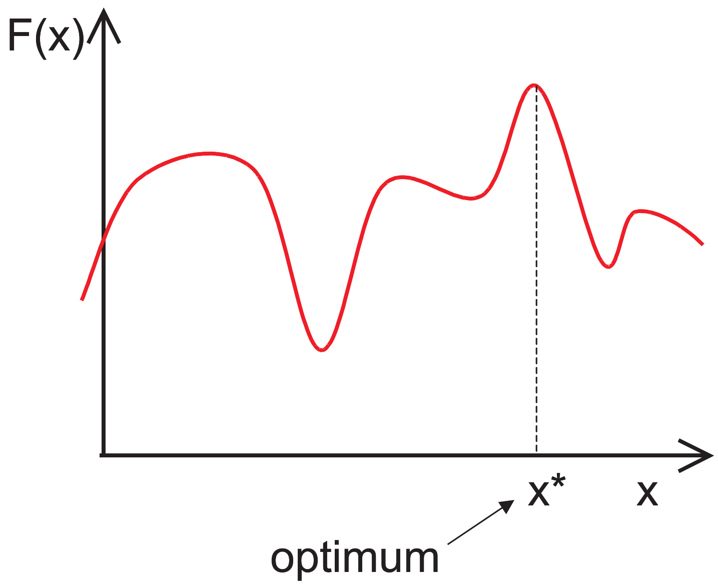
- But : trouver le point de plus élevé d'une "fonction coût" F sur un domaine $\mathcal{D}$
- Difficulté : on ne dispose que d'un budget limité de N évaluations de la fonction F (on ne peut pas calculer toutes les combinaisons pour choisir la meilleure!)
  - On représente notre croyance que l'optimum est dans une région de $\mathcal{D}$ par une densité de probabilité $p(\mathbf{x})$

Algorithmes à estimation de distribution



- But : trouver le point de plus élevé d'une “fonction coût” F sur un domaine $\mathcal{D}$
- Difficulté : on ne dispose que d'un budget limité de N évaluations de la fonction F (on ne peut pas calculer toutes les combinaisons pour choisir la meilleure!)
  - On représente notre croyance que l'optimum est dans une région de $\mathcal{D}$ par une densité de probabilité $p(\mathbf{x})$
  - $p(\mathbf{x})$ est utilisée pour créer de nouveaux points

Algorithmes à estimation de distribution



- But : trouver le point de plus élevé d'une "fonction coût" F sur un domaine $\mathcal{D}$
- Difficulté : on ne dispose que d'un budget limité de N évaluations de la fonction F (on ne peut pas calculer toutes les combinaisons pour choisir la meilleure!)
  - On représente notre croyance que l'optimum est dans une région de $\mathcal{D}$ par une densité de probabilité $p(\mathbf{x})$
  - $p(\mathbf{x})$ est utilisée pour créer de nouveaux points
  - les nouveaux points sont utilisés pour mettre à jour $p(\mathbf{x})$

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)
- Représentation de la distribution $p(\mathbf{x})$: mauvais choix de la forme de $p(\mathbf{x})$ ⇨ gaspillage d'évaluations dans des régions médiocres

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)
- Représentation de la distribution $p(\mathbf{x})$: mauvais choix de la forme de $p(\mathbf{x})$ ⇨ gaspillage d'évaluations dans des régions médiocres

Solutions proposées

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)
- Représentation de la distribution $p(\mathbf{x})$: mauvais choix de la forme de $p(\mathbf{x})$ ⇨ gaspillage d'évaluations dans des régions médiocres

Solutions proposées

- Amélioration du modèle statistique par injection de connaissance sur la structure du problème

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)
- Représentation de la distribution $p(\mathbf{x})$: mauvais choix de la forme de $p(\mathbf{x})$ ⇨ gaspillage d'évaluations dans des régions médiocres

Solutions proposées

- Amélioration du modèle statistique par injection de connaissance sur la structure du problème
- Mécanismes de contrôle de l'exploration

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)
- Représentation de la distribution $p(\mathbf{x})$: mauvais choix de la forme de $p(\mathbf{x})$ ⇨ gaspillage d'évaluations dans des régions médiocres

Solutions proposées

- Amélioration du modèle statistique par injection de connaissance sur la structure du problème
- Mécanismes de contrôle de l'exploration

Résultats

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)
- Représentation de la distribution $p(\mathbf{x})$: mauvais choix de la forme de $p(\mathbf{x})$ ⇨ gaspillage d'évaluations dans des régions médiocres

Solutions proposées

- Amélioration du modèle statistique par injection de connaissance sur la structure du problème
- Mécanismes de contrôle de l'exploration

Résultats

- Les EDAs sont facilement applicables à l'optimisation de stratifiés

Enjeux, améliorations proposées et résultats

Éléments critiques de l'algorithme

- Adaptation aux problèmes de composites (alphabet non binaire, nombre de variables réduit, évaluations coûteuses)
- Représentation de la distribution $p(\mathbf{x})$: mauvais choix de la forme de $p(\mathbf{x})$ ⇨ gaspillage d'évaluations dans des régions médiocres

Solutions proposées

- Amélioration du modèle statistique par injection de connaissance sur la structure du problème
- Mécanismes de contrôle de l'exploration

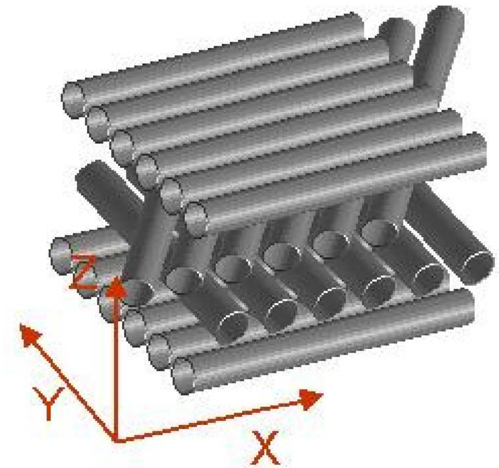
Résultats

- Les EDAs sont facilement applicables à l'optimisation de stratifiés
- La stratégie proposée conduit à une amélioration de l'efficacité sur les problèmes testés

Introduction to Composite Laminate Optimization

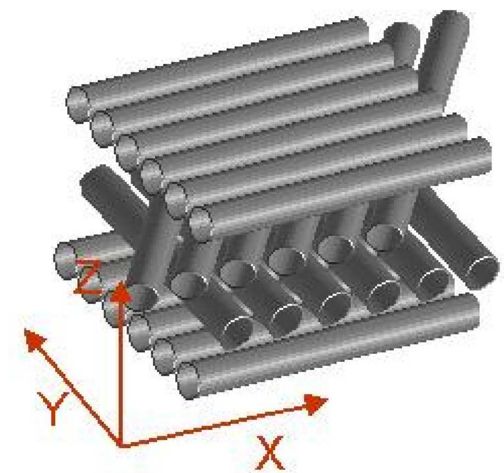
Composite Laminates

- Composite laminate: structure made of layers (plies) of fibrous material embedded in a matrix



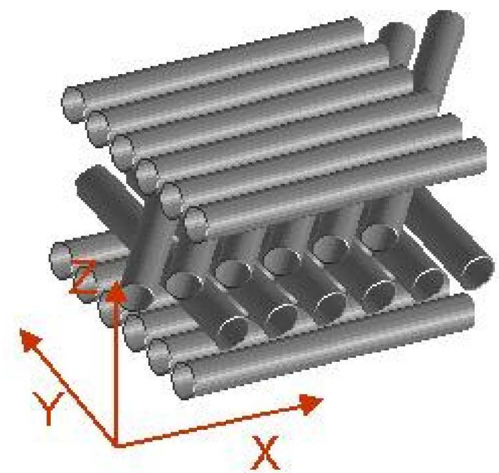
Composite Laminates

- Composite laminate: structure made of layers (plies) of fibrous material embedded in a matrix
- The fibers (graphite, glass,. . .) provide the mechanical properties, the matrix (polymer) hold the fibers together



Composite Laminates

- Composite laminate: structure made of layers (plies) of fibrous material embedded in a matrix
- The fibers (graphite, glass,. . .) provide the mechanical properties, the matrix (polymer) hold the fibers together
- Applications:
  - aerospace industry (rudder, wing box, flying control surfaces, helicopter blades,. . .)
  - sporting goods (skis, sailing, tennis)
  - wind turbines
  - motorsports and performance cars



Optimization of Composite Laminates

- Each layer is orthotropic: its mechanical properties depend on the direction

Optimization of Composite Laminates

- Each layer is orthotropic: its mechanical properties depend on the direction
- Overall response F (stiffness, strength, natural frequency, buckling load,...) of the whole laminate depends on the stacking sequence $[\theta_1, \theta_2, \ldots, \theta_n]$:

$$F = F(\theta_1, \theta_2, \ldots, \theta_n)$$

Optimization of Composite Laminates

- Each layer is orthotropic: its mechanical properties depend on the direction
- Overall response F (stiffness, strength, natural frequency, buckling load,...) of the whole laminate depends on the stacking sequence $[\theta_1, \theta_2, \ldots, \theta_n]$:

$$F = F(\theta_1, \theta_2, \ldots, \theta_n)$$

- Goal of the optimization: maximize F

Optimization of Composite Laminates

- Each layer is orthotropic: its mechanical properties depend on the direction
- Overall response F (stiffness, strength, natural frequency, buckling load,...) of the whole laminate depends on the stacking sequence $[\theta_1, \theta_2, \ldots, \theta_n]$:

$$F = F(\theta_1, \theta_2, \ldots, \theta_n)$$

- Goal of the optimization: maximize F
- Difficulty:

Optimization of Composite Laminates

- Each layer is orthotropic: its mechanical properties depend on the direction
- Overall response F (stiffness, strength, natural frequency, buckling load,...) of the whole laminate depends on the stacking sequence $[\theta_1, \theta_2, \ldots, \theta_n]$:

$$F = F(\theta_1, \theta_2, \ldots, \theta_n)$$

- Goal of the optimization: maximize F
- Difficulty:
    - θ_k discrete (e.g. $\theta_k \in \{0^\circ, 45^\circ, 90^\circ\}$) ⇨ combinatorial problems

Optimization of Composite Laminates

- Each layer is orthotropic: its mechanical properties depend on the direction
- Overall response F (stiffness, strength, natural frequency, buckling load,...) of the whole laminate depends on the stacking sequence $[\theta_1, \theta_2, \ldots, \theta_n]$:

$$F = F(\theta_1, \theta_2, \ldots, \theta_n)$$

- Goal of the optimization: maximize F
- Difficulty:
  - θ_k discrete (e.g. $\theta_k \in \{0°, 45°, 90°\}$) ⇨ combinatorial problems
  - F non-convex in general ⇨ many local optima

Optimization of Composite Laminates

- Each layer is orthotropic: its mechanical properties depend on the direction
- Overall response F (stiffness, strength, natural frequency, buckling load,...) of the whole laminate depends on the stacking sequence $[\theta_1, \theta_2, \ldots, \theta_n]$:

$$F = F(\theta_1, \theta_2, \ldots, \theta_n)$$

- Goal of the optimization: maximize F
- Difficulty:
  - θ_k discrete (e.g. $\theta_k \in \{0°, 45°, 90°\}$) ⇨ combinatorial problems
  - F non-convex in general ⇨ many local optima

⇨ Require specific optimization methods

Laminate Optimization Methods

- Gradient-based continuous optimization using ply thicknesses as design variables (Schmit and Farshi, 1977) or a penalty approach to force discreteness of the ply angles (Shin et al., 1990)

Laminate Optimization Methods

- Gradient-based continuous optimization using ply thicknesses as design variables (Schmit and Farshi, 1977) or a penalty approach to force discreteness of the ply angles (Shin et al., 1990)
  - ⇨ can use well-established methods, but no guaranty to find the global optimum

Laminate Optimization Methods

- Gradient-based continuous optimization using ply thicknesses as design variables (Schmit and Farshi, 1977) or a penalty approach to force discreteness of the ply angles (Shin et al., 1990)
  - ⇨ can use well-established methods, but no guaranty to find the global optimum
- Stochastic optimization: Genetic Algorithms (Hajela and Lin, 1992; Le Riche and Haftka, 1993)

Laminate Optimization Methods

- Gradient-based continuous optimization using ply thicknesses as design variables (Schmit and Farshi, 1977) or a penalty approach to force discreteness of the ply angles (Shin et al., 1990)
  - ➩ can use well-established methods, but no guaranty to find the global optimum
- Stochastic optimization: Genetic Algorithms (Hajela and Lin, 1992; Le Riche and Haftka, 1993)
  - ➩ stochastic search + discrete variables: can solve the problem directly, but

Laminate Optimization Methods

- Gradient-based continuous optimization using ply thicknesses as design variables (Schmit and Farshi, 1977) or a penalty approach to force discreteness of the ply angles (Shin et al., 1990)
  - can use well-established methods, but no guaranty to find the global optimum
- Stochastic optimization: Genetic Algorithms (Hajela and Lin, 1992; Le Riche and Haftka, 1993)
  - stochastic search + discrete variables: can solve the problem directly, but

    can be difficult to use (many problem-dependent parameters to tune),

Laminate Optimization Methods

- Gradient-based continuous optimization using ply thicknesses as design variables (Schmit and Farshi, 1977) or a penalty approach to force discreteness of the ply angles (Shin et al., 1990)
  - ➪ can use well-established methods, but no guaranty to find the global optimum
- Stochastic optimization: Genetic Algorithms (Hajela and Lin, 1992; Le Riche and Haftka, 1993)
  - ➪ stochastic search + discrete variables: can solve the problem directly, but

    can be difficult to use (many problem-dependent parameters to tune),

    and computationally expensive

Purpose of this work

Our goal is to

1. investigate the application of a new class of algorithms called "Estimation of Distribution Algorithms" (EDAs) to the field of laminate optimization;

Purpose of this work

Our goal is to

1. investigate the application of a new class of algorithms called "Estimation of Distribution Algorithms" (EDAs) to the field of laminate optimization;
2. propose improvements to EDAs in the context of laminate optimization:
   use physics-based knowledge to improve efficiency

Purpose of this work

Our goal is to

1. investigate the application of a new class of algorithms called "Estimation of Distribution Algorithms" (EDAs) to the field of laminate optimization;
2. propose improvements to EDAs in the context of laminate optimization:
   use physics-based knowledge to improve efficiency
3. study the general behavior of EDAs and propose improvements to EDAs that can be used for other fields.

Introduction to Estimation of Distribution Algorithms

Recall: Genetic Algorithms (GA)

- Inspiration: Darwin’s theory of Evolution (“survival of the fittest”)

Recall: Genetic Algorithms (GA)

- Inspiration: Darwin's theory of Evolution ("survival of the fittest")

  A species can adapt to an environment because those individuals that possess features that give them a competitive advantage over other individuals are more likely to have offspring, and therefore to pass on their advantageous traits to the next generation.

Recall: Genetic Algorithms (GA)

- Inspiration: Darwin's theory of Evolution ("survival of the fittest")

  A species can adapt to an environment because those individuals that possess features that give them a competitive advantage over other individuals are more likely to have offspring, and therefore to pass on their advantageous traits to the next generation.

- Application to function optimization:
  Idea: let a population of candidate solutions evolve to adapt to a task to perform

Recall: Genetic Algorithms (GA)

- Inspiration: Darwin's theory of Evolution ("survival of the fittest")

  A species can adapt to an environment because those individuals that possess features that give them a competitive advantage over other individuals are more likely to have offspring, and therefore to pass on their advantageous traits to the next generation.

- Application to function optimization:
  Idea: let a population of candidate solutions evolve to adapt to a task to perform
  - Environment $\rightarrow$ Function $F(\mathbf{x})$ to maximize ("fitness")

Recall: Genetic Algorithms (GA)

- Inspiration: Darwin's theory of Evolution ("survival of the fittest")

  A species can adapt to an environment because those individuals that possess features that give them a competitive advantage over other individuals are more likely to have offspring, and therefore to pass on their advantageous traits to the next generation.

- Application to function optimization:
  Idea: let a population of candidate solutions evolve to adapt to a task to perform
  - Environment $\rightarrow$ Function $F(\mathbf{x})$ to maximize ("fitness")
  - Population of individuals $\rightarrow$ population of points $\mathbf{x}_i$, $i = 1, \ldots, \lambda$

Recall: Genetic Algorithms (GA)

- Inspiration: Darwin's theory of Evolution ("survival of the fittest")

  A species can adapt to an environment because those individuals that possess features that give them a competitive advantage over other individuals are more likely to have offspring, and therefore to pass on their advantageous traits to the next generation.

- Application to function optimization:
  Idea: let a population of candidate solutions evolve to adapt to a task to perform
  - Environment $\rightarrow$ Function $F(\mathbf{x})$ to maximize ("fitness")
  - Population of individuals $\rightarrow$ population of points $\mathbf{x}_i$, $i = 1, \ldots, \lambda$
  - Natural selection $\rightarrow$ Selection based on F

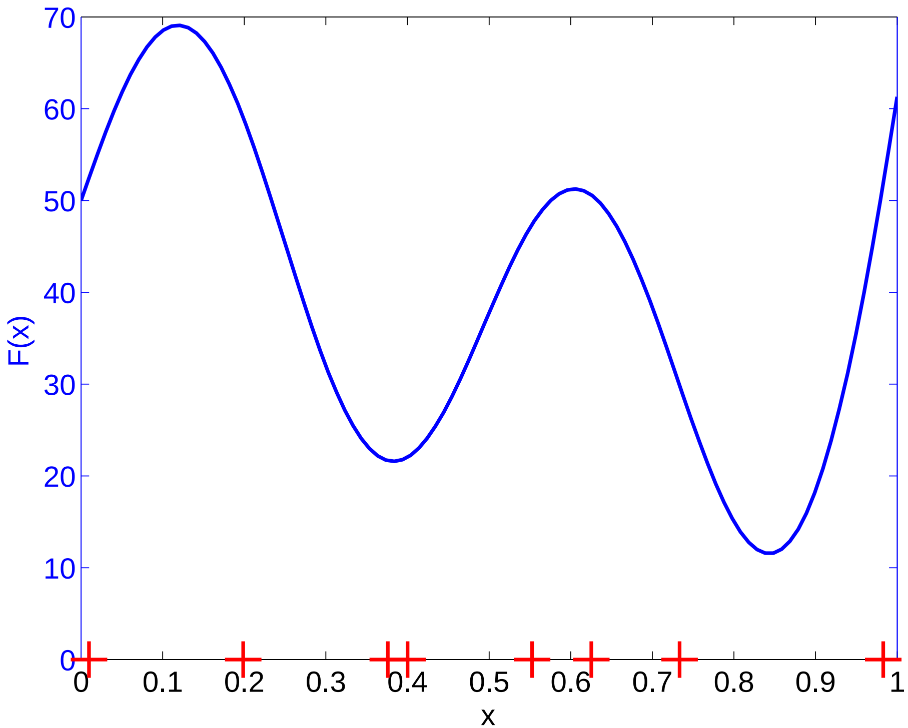
Recall: Genetic Algorithms (GA)

- Inspiration: Darwin's theory of Evolution ("survival of the fittest")

  A species can adapt to an environment because those individuals that possess features that give them a competitive advantage over other individuals are more likely to have offspring, and therefore to pass on their advantageous traits to the next generation.

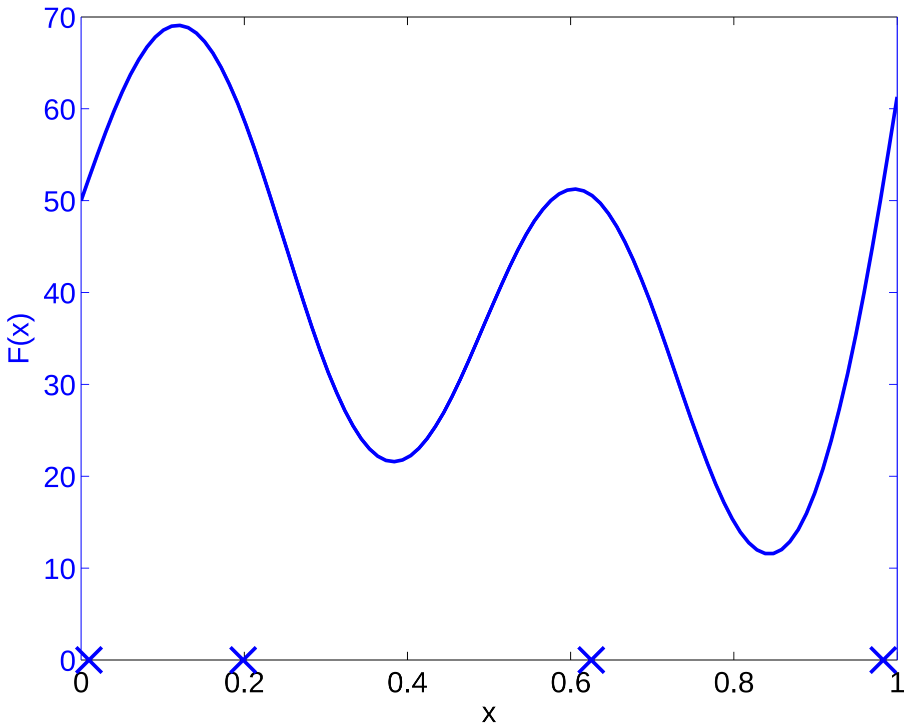
- Application to function optimization:
  Idea: let a population of candidate solutions evolve to adapt to a task to perform
  - Environment $\rightarrow$ Function $F(\mathbf{x})$ to maximize ("fitness")
  - Population of individuals $\rightarrow$ population of points $\mathbf{x}_i$, $i = 1, \ldots, \lambda$
  - Natural selection $\rightarrow$ Selection based on F
  - Reproduction $\rightarrow$ recombination (crossover) and perturbation (mutation) operators

GA Mechanism



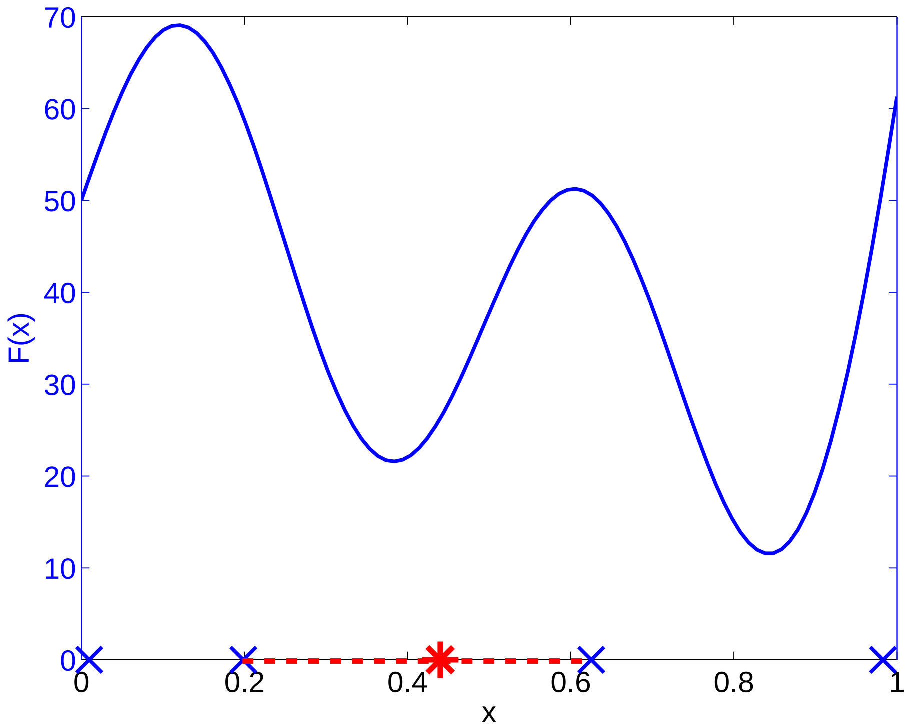
- Initial population: uniform

GA Mechanism



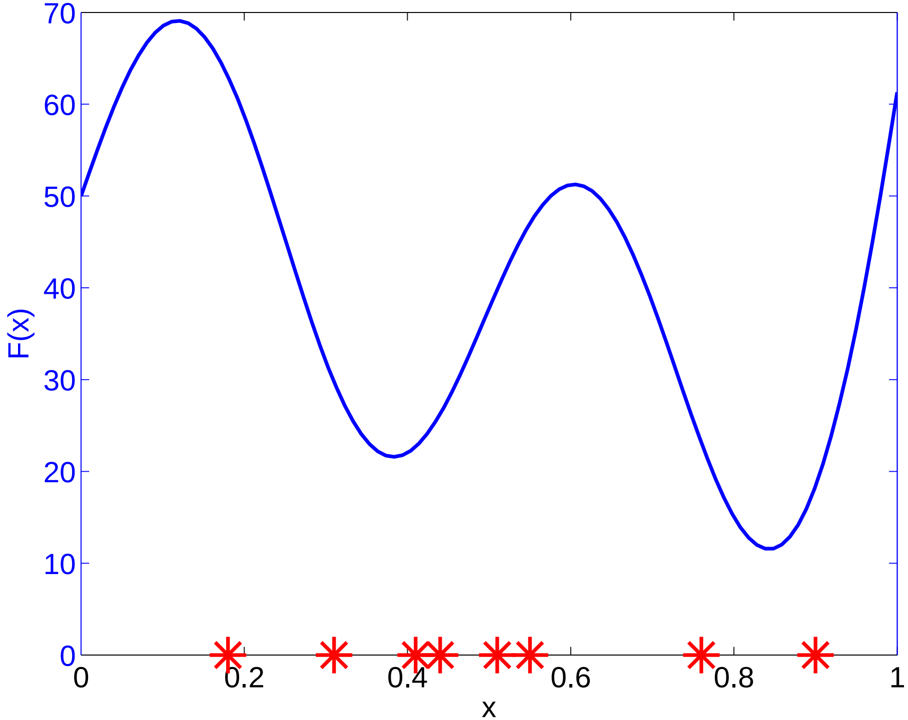
- Initial population: uniform
- Points obtained by selection ⇨ parents

GA Mechanism



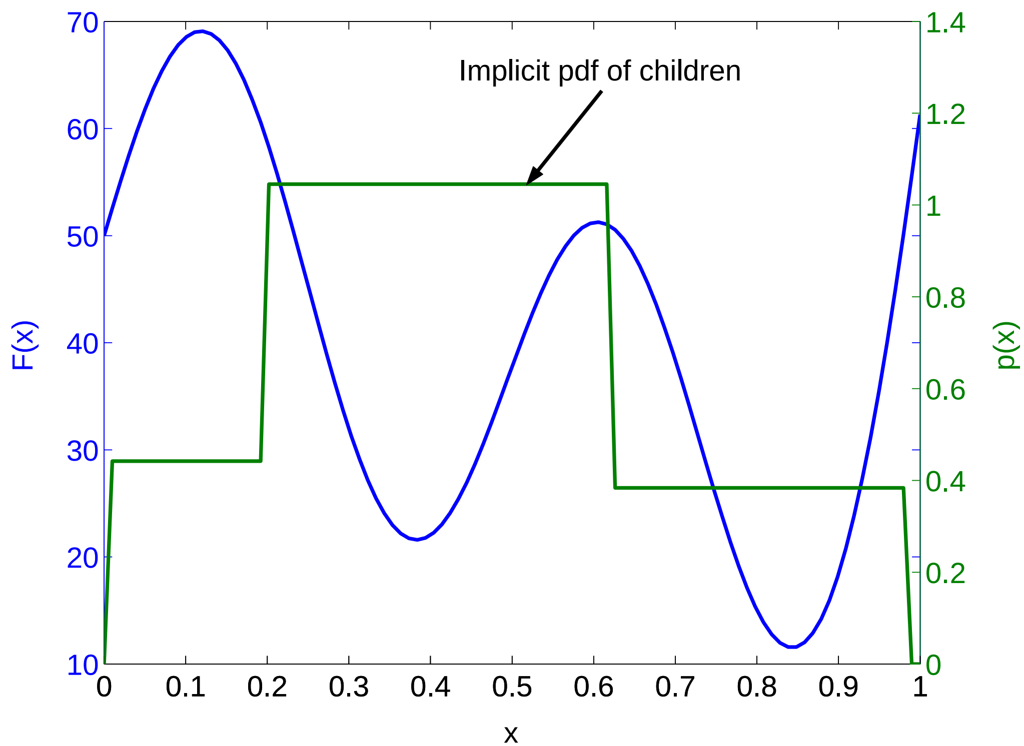
- Initial population: uniform
- Points obtained by selection ⇨ parents
- Creation of new points by recombination (e.g. weighted average in $\mathbb{R}^n$) ⇨ children

GA Mechanism



- Initial population: uniform
- Points obtained by selection ➪ parents
- Creation of new points by recombination (e.g. weighted average in $\mathbb{R}^n$) ➪ children
- Resulting new population

GA Mechanism



- Initial population: uniform
- Points obtained by selection ⇨ parents
- Creation of new points by recombination (e.g. weighted average in $\mathbb{R}^n$) ⇨ children
- Resulting new population

	fitness function F		implicit
+	selection procedure	}	⇨ probability distribution $p(\mathbf{x})$
+	variation operators		over the design space

Formalization of GAs ⇨ Estimation of Distribution Algorithms

- Goal: control the way the search distribution $p(\mathbf{x})$ is constructed so that it learns information about “promising regions”

Formalization of GAs ⇨ Estimation of Distribution Algorithms

- Goal: control the way the search distribution $p(\mathbf{x})$ is constructed so that it learns information about "promising regions"

Principle: express explicitly $p(\mathbf{x})$ and use it to create new points in high-fitness areas

Formalization of GAs ⇨ Estimation of Distribution Algorithms

- Goal: control the way the search distribution $p(\mathbf{x})$ is constructed so that it learns information about "promising regions"

Principle: express explicitly $p(\mathbf{x})$ and use it to create new points in high-fitness areas

- Potential advantages:

Formalization of GAs ⇨ Estimation of Distribution Algorithms

- Goal: control the way the search distribution $p(\mathbf{x})$ is constructed so that it learns information about "promising regions"

Principle: express explicitly $p(\mathbf{x})$ and use it to create new points in high-fitness areas

- Potential advantages:
  - Better understanding of the algorithm (based on statistical principles, not nature imitation)
    ⇨ improve efficiency

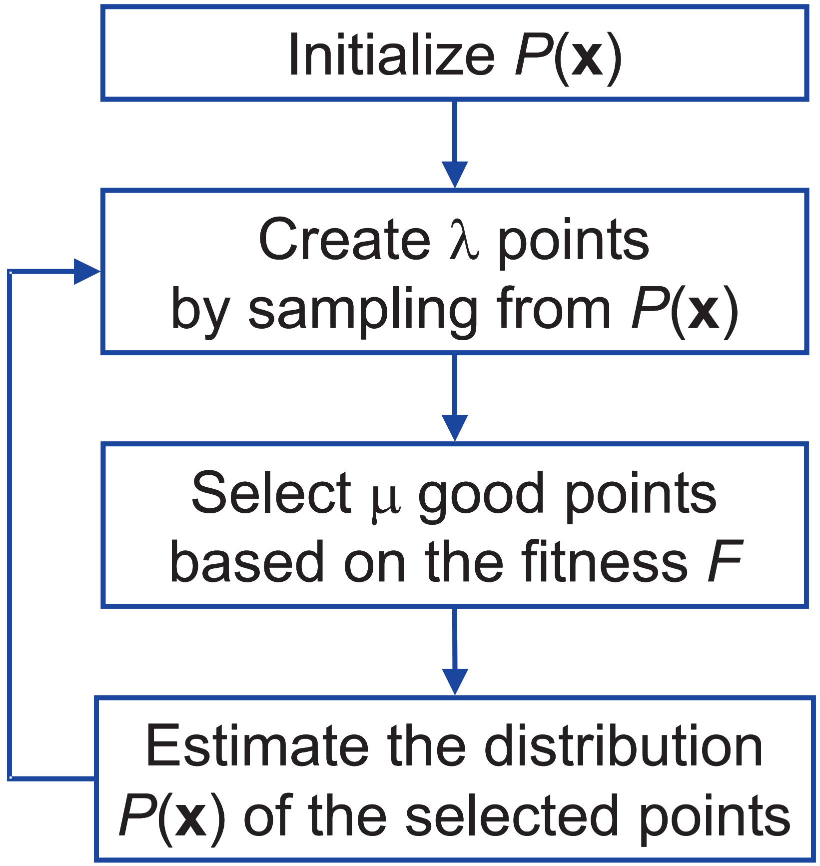
Formalization of GAs ➪ Estimation of Distribution Algorithms

- Goal: control the way the search distribution $p(\mathbf{x})$ is constructed so that it learns information about "promising regions"

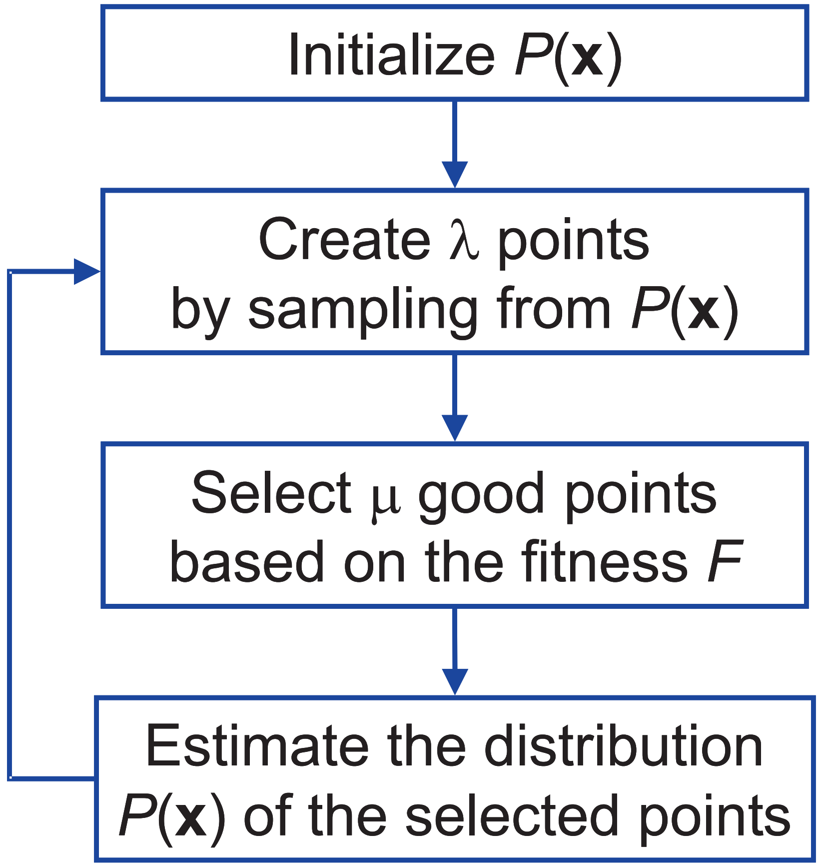
> Principle: express explicitly $p(\mathbf{x})$ and use it to create new points in high-fitness areas

- Potential advantages:
  - Better understanding of the algorithm (based on statistical principles, not nature imitation)
    ➪ improve efficiency
  - Potentially fewer parameters (avoid many ad hoc operators)
    ➪ algorithm easier to use

The Estimation of Distribution Algorithm (EDA)

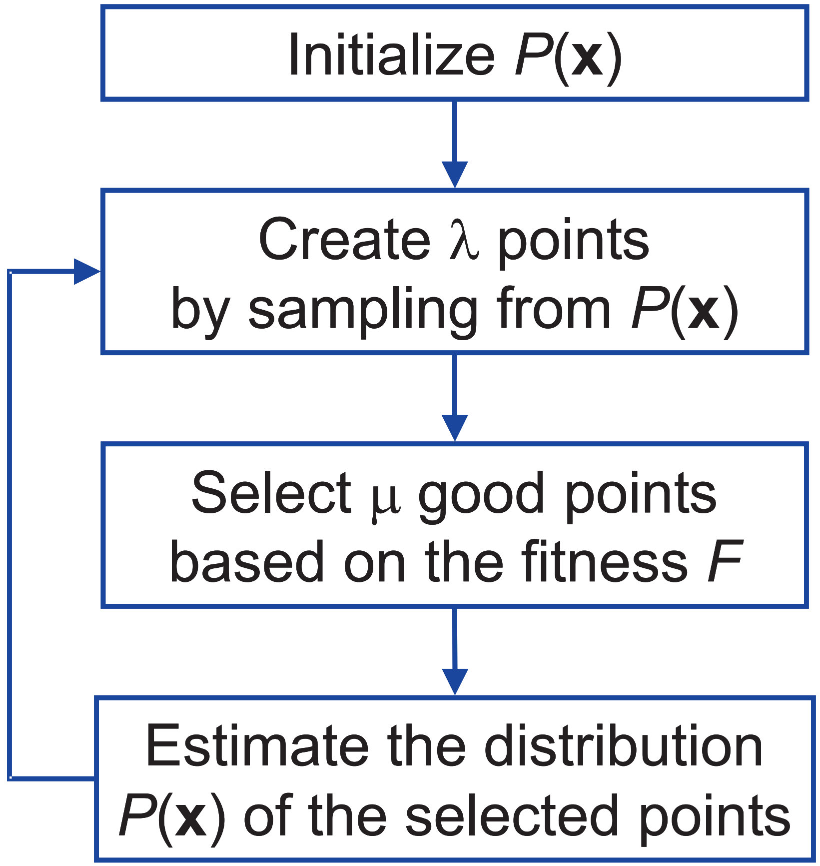


The Estimation of Distribution Algorithm (EDA)



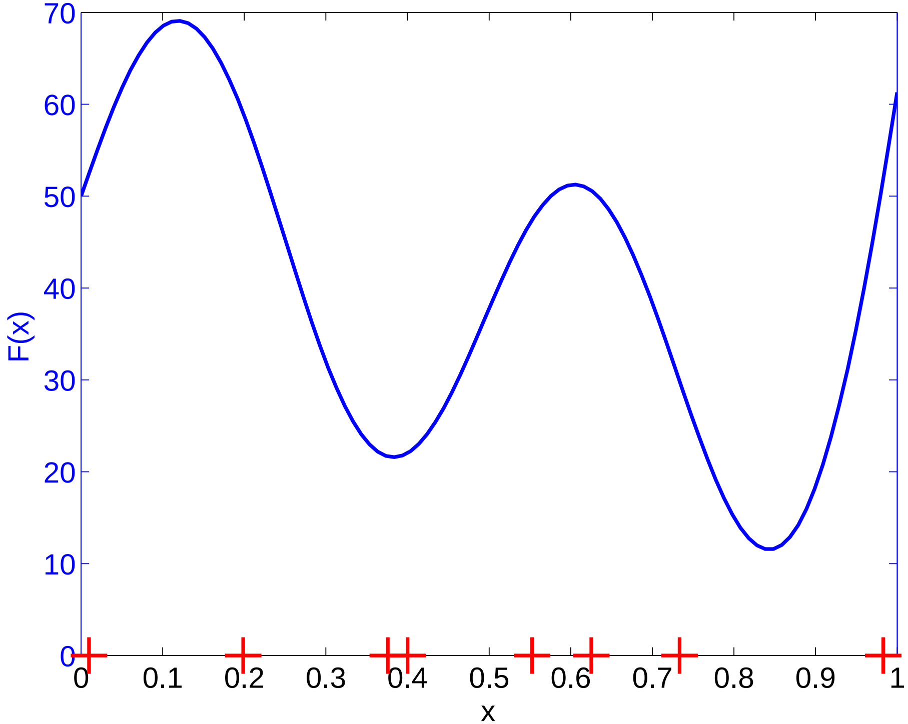
1. Use created points to infer statistical information about good regions ➩ $p(\mathbf{x})$

The Estimation of Distribution Algorithm (EDA)



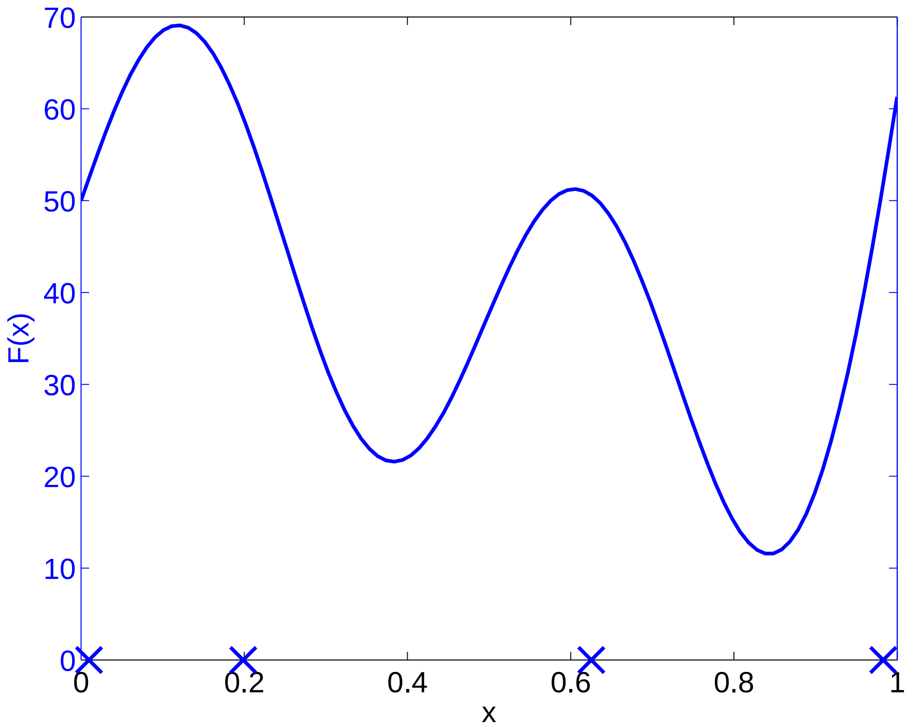
1. Use created points to infer statistical information about good regions ⇨ $p(\mathbf{x})$
2. Use $p(\mathbf{x})$ to create new points in promising regions

Estimation of Distribution Algorithm: Illustration



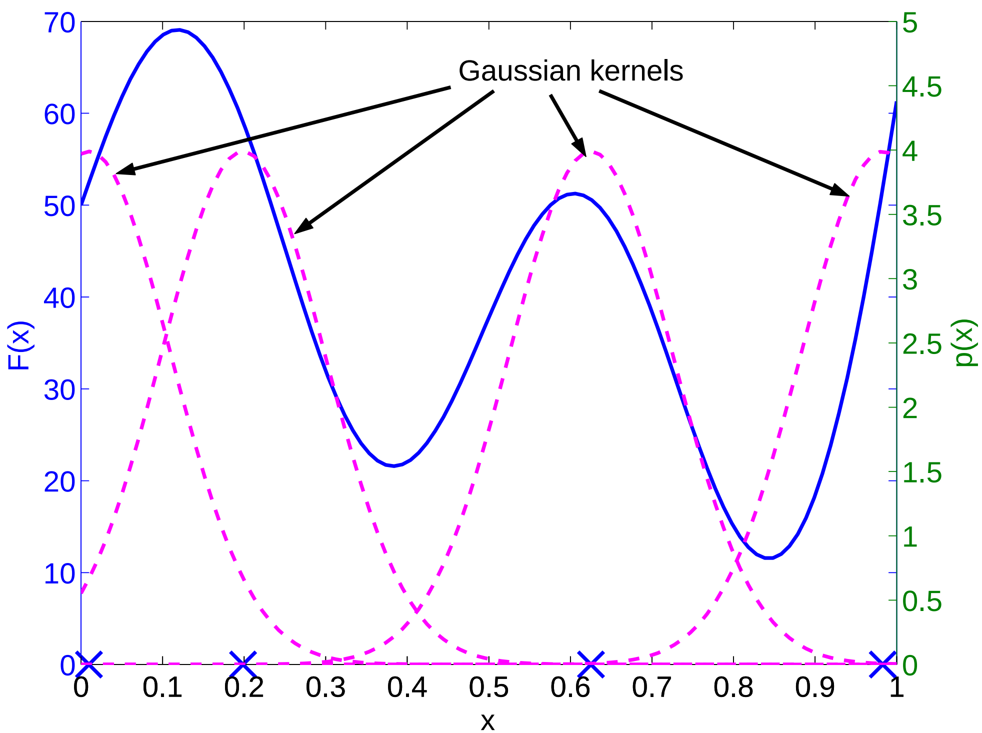
- Initial population: uniform $p(\mathbf{x})$

Estimation of Distribution Algorithm: Illustration



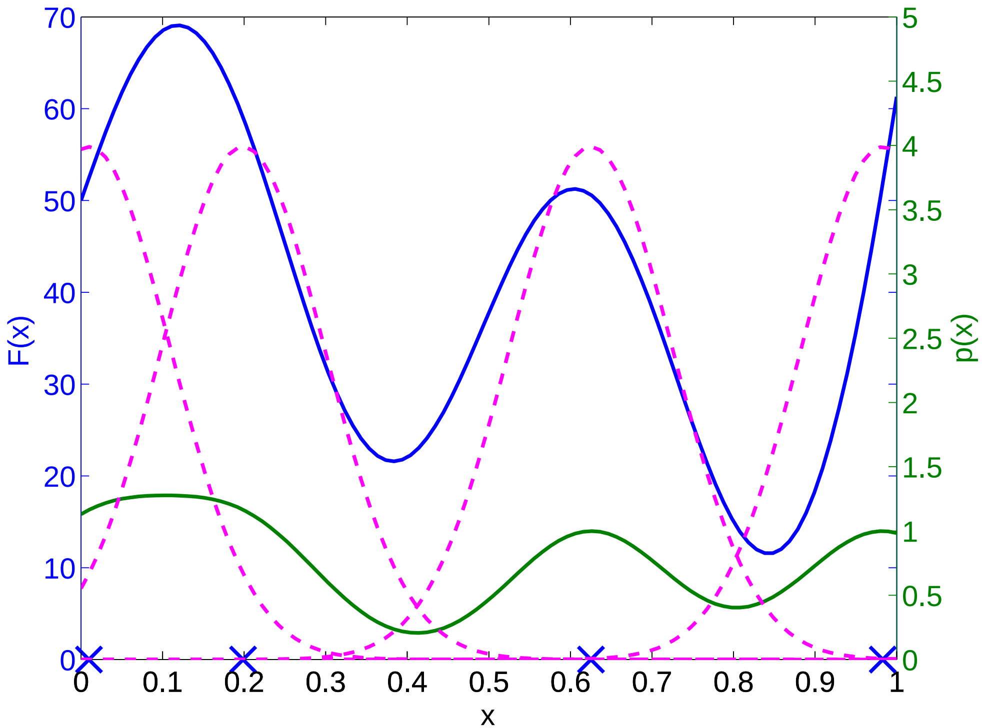
- Initial population: uniform $p(\mathbf{x})$
- Points obtained by selection

Estimation of Distribution Algorithm: Illustration



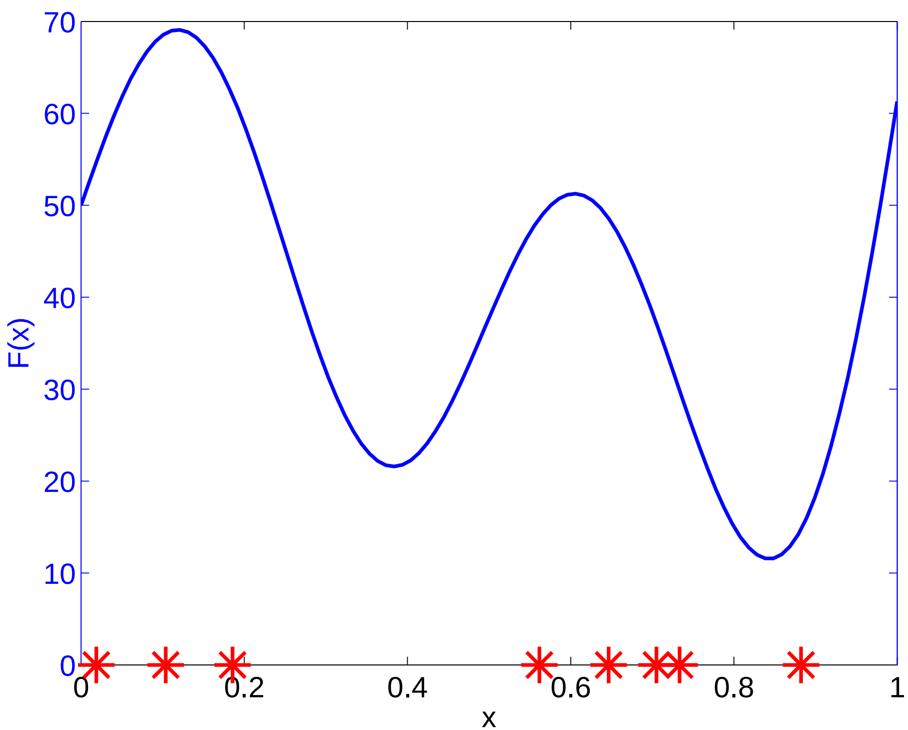
- Initial population: uniform $p(\mathbf{x})$
- Points obtained by selection
- Estimation of the distribution of good points $p(x)$ (continuous case, kernels, $p(x) = \frac{\text{const}}{N} \sum_{i=1}^{N} \exp(-\frac{(x-x_i)^2}{2\sigma^2}))$

Estimation of Distribution Algorithm: Illustration



- Initial population: uniform $p(\mathbf{x})$
- Points obtained by selection
- Estimation of the distribution of good points $p(x)$ (continuous case, kernels, $p(x) = \frac{\text{const}}{N} \sum_{i=1}^{N} \exp(-\frac{(x-x_i)^2}{2\sigma^2})$)
- Estimated distribution of good points $p(x)$

Estimation of Distribution Algorithm: Illustration



- Initial population: uniform $p(\mathbf{x})$
- Points obtained by selection
- Estimation of the distribution of good points $p(x)$ (continuous case, kernels, $p(x) = \frac{\text{const}}{N} \sum_{i=1}^{N} \exp(-\frac{(x-x_i)^2}{2\sigma^2})$)
- Estimated distribution of good points $p(x)$
- New population obtained by sampling from $p(x)$

Estimating the distribution of good points

- Task: given a sample of μ selected points, infer distribution $p(\mathbf{x})$ of all the points of comparable fitness ("promising regions")

Estimating the distribution of good points

- Task: given a sample of μ selected points, infer distribution $p(\mathbf{x})$ of all the points of comparable fitness (“promising regions”)
- Procedure:

Estimating the distribution of good points

- Task: given a sample of μ selected points, infer distribution $p(\mathbf{x})$ of all the points of comparable fitness (“promising regions”)
- Procedure:
  1. choose a model $\hat{p}(\mathbf{x}; m_1, \ldots, m_r)$,

Estimating the distribution of good points

- Task: given a sample of μ selected points, infer distribution $p(\mathbf{x})$ of all the points of comparable fitness ("promising regions")
- Procedure:
  1. choose a model $\hat{p}(\mathbf{x}; m_1, \ldots, m_r)$,
  2. find the value of the model parameters m_j that maximizes the likelihood of the good observed points.

Estimating the distribution of good points

- Task: given a sample of μ selected points, infer distribution $p(\mathbf{x})$ of all the points of comparable fitness ("promising regions")
- Procedure:
  1. choose a model $\hat{p}(\mathbf{x}; m_1, \ldots, m_r)$,
  2. find the value of the model parameters m_j that maximizes the likelihood of the good observed points.
- The choice of the model is a compromise between

Estimating the distribution of good points

- Task: given a sample of μ selected points, infer distribution $p(\mathbf{x})$ of all the points of comparable fitness ("promising regions")
- Procedure:
  1. choose a model $\hat{p}(\mathbf{x}; m_1, \ldots, m_r)$,
  2. find the value of the model parameters m_j that maximizes the likelihood of the good observed points.
- The choice of the model is a compromise between
  1. the exploitation or accuracy of p, i.e. its ability to learn the distribution of selected points, and

Estimating the distribution of good points

- Task: given a sample of μ selected points, infer distribution $p(\mathbf{x})$ of all the points of comparable fitness ("promising regions")
- Procedure:
  1. choose a model $\hat{p}(\mathbf{x}; m_1, \ldots, m_r)$,
  2. find the value of the model parameters m_j that maximizes the likelihood of the good observed points.
- The choice of the model is a compromise between
  1. the exploitation or accuracy of p, i.e. its ability to learn the distribution of selected points, and
  2. the generalization stability of p in unvisited regions, in particular its ability to explore them.

Application of a Simple EDA to Laminate Optimization

Simple Model: Independent Variables

- General principle in machine learning: in the absence of information about the distribution, use simple models (cf. “Occam’s razor”)

Simple Model: Independent Variables

- General principle in machine learning: in the absence of information about the distribution, use simple models (cf. “Occam’s razor”)
- Simplest statistical model of selected points: independent variables:

$$p(x_1, x_2, \ldots, x_n) = \prod_{k=1}^{n} p(x_k)$$

Simple Model: Independent Variables

- General principle in machine learning: in the absence of information about the distribution, use simple models (cf. “Occam’s razor”)
- Simplest statistical model of selected points: independent variables:

$$p(x_1, x_2, \ldots, x_n) = \prod_{k=1}^{n} p(x_k)$$

Only the marginal distributions of the variables appear in the model: discrete case = frequency of all possible values of each variable

Simple Model: Independent Variables

- General principle in machine learning: in the absence of information about the distribution, use simple models (cf. "Occam's razor")
- Simplest statistical model of selected points: independent variables:

$$p(x_1, x_2, \ldots, x_n) = \prod_{k=1}^{n} p(x_k)$$

  Only the marginal distributions of the variables appear in the model: discrete case = frequency of all possible values of each variable
- Proposed by S. Baluja (PBIL, 1994) and H. Mühlenbein (1996) in the Univariate Marginal Distribution Algorithm (UMDA)

Simple Model: Independent Variables

- General principle in machine learning: in the absence of information about the distribution, use simple models (cf. “Occam’s razor”)
- Simplest statistical model of selected points: independent variables:

$$p(x_1, x_2, \ldots, x_n) = \prod_{k=1}^{n} p(x_k)$$

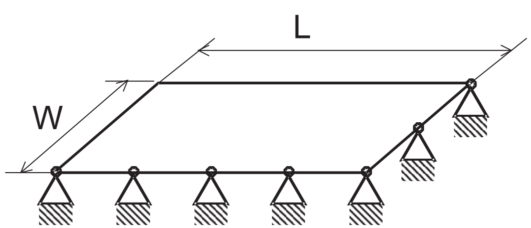
  Only the marginal distributions of the variables appear in the model: discrete case = frequency of all possible values of each variable
- Proposed by S. Baluja (PBIL, 1994) and H. Mühlenbein (1996) in the Univariate Marginal Distribution Algorithm (UMDA)
- Changes: non-binary alphabet, mutation to compensate for estimation error

Application to Laminates: Frequency problem

- Constrained maximization of the first natural frequency of a simply-supported rectangular laminated plate:

maximize $f_1(\theta_1, \ldots, \theta_{15})$
such that $\nu_{\mathrm{l}} \leq \nu_{\mathrm{eff}} \leq \nu_{\mathrm{u}}$

Poisson's ratio = deformation observed in the transverse direction when a unit deformation is applied in the longitudinal direction

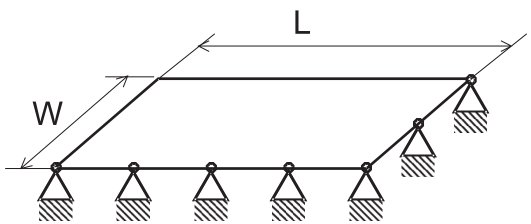


Application to Laminates: Frequency problem

- Constrained maximization of the first natural frequency of a simply-supported rectangular laminated plate:

$$\textbf{maximize} \quad f_1(\theta_1, \ldots, \theta_{15})$$
$$\textbf{such that} \quad \nu_{\mathrm{l}} \leq \nu_{\mathrm{eff}} \leq \nu_{\mathrm{u}}$$

Poisson’s ratio = deformation observed in the transverse direction when a unit deformation is applied in the longitudinal direction



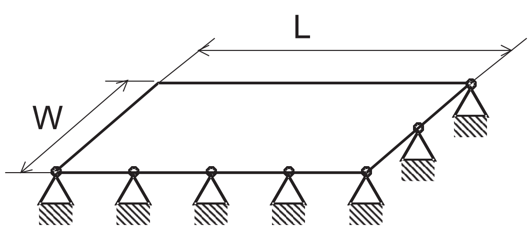
- $\theta_k \in \{0°, 15°, 30°, 45°, 60°, 75°, 90°\}$

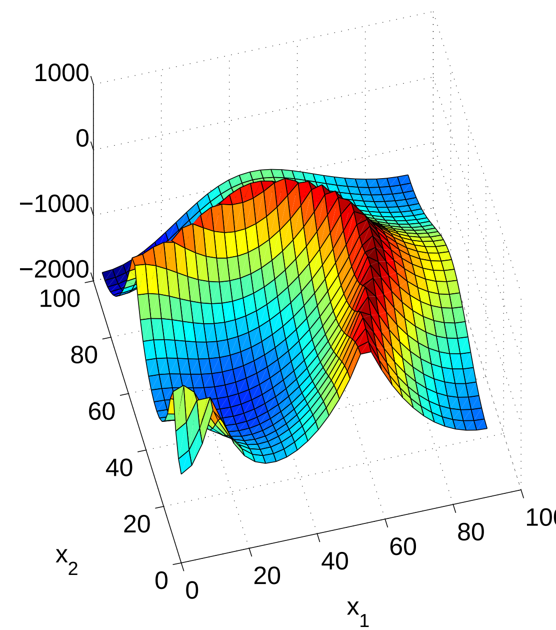
Application to Laminates: Frequency problem

- Constrained maximization of the first natural frequency of a simply-supported rectangular laminated plate:

$$\textbf{maximize} \quad f_1(\theta_1, \ldots, \theta_{15})$$
$$\textbf{such that} \quad \nu_\mathrm{l} \le \nu_\mathrm{eff} \le \nu_\mathrm{u}$$

Poisson's ratio = deformation observed in the transverse direction when a unit deformation is applied in the longitudinal direction





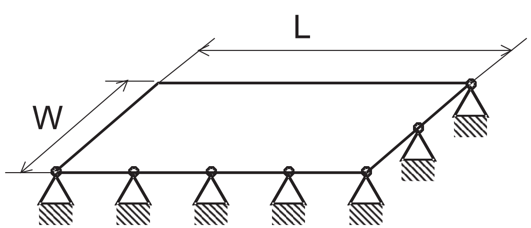
- $\theta_k \in \{0^\circ, 15^\circ, 30^\circ, 45^\circ, 60^\circ, 75^\circ, 90^\circ\}$
- The constraints are enforced through a penalty approach

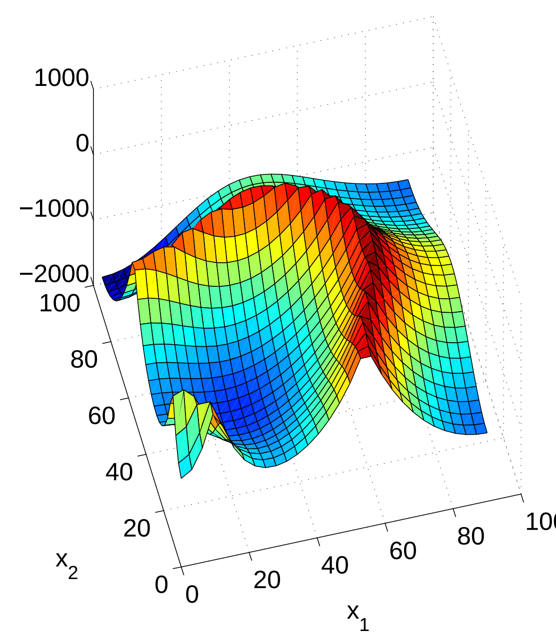
Application to Laminates: Frequency problem

- Constrained maximization of the first natural frequency of a simply-supported rectangular laminated plate:

$$\begin{array}{ll} \textbf{maximize} & f_1(\theta_1, \ldots, \theta_{15}) \\ \textbf{such that} & \nu_{\mathrm{l}} \leq \nu_{\mathrm{eff}} \leq \nu_{\mathrm{u}} \end{array}$$

> Poisson's ratio = deformation observed in the transverse direction when a unit deformation is applied in the longitudinal direction





- $\theta_k \in \{0°, 15°, 30°, 45°, 60°, 75°, 90°\}$
- The constraints are enforced through a penalty approach
- The constraints create a narrow ridge in the design space

Results: reliability of the optimization

- Optimum:

$$[90_4/\pm 75/\pm 60_2/\pm 45_5/\pm 30_5]_s$$

Results: reliability of the optimization

- Optimum:

$$[90_4/\pm 75/\pm 60_2/\pm 45_5/\pm 30_5]_s$$

- Compare UMDA to a GA and a hill-climbing algorithm (SHC)

Results: reliability of the optimization

- Optimum:

$$[90_4/\pm 75/\pm 60_2/\pm 45_5/\pm 30_5]_s$$

- Compare UMDA to a GA and a hill-climbing algorithm (SHC)
- UMDA and SHC: optimized parameters, GA: same setting as UMDA

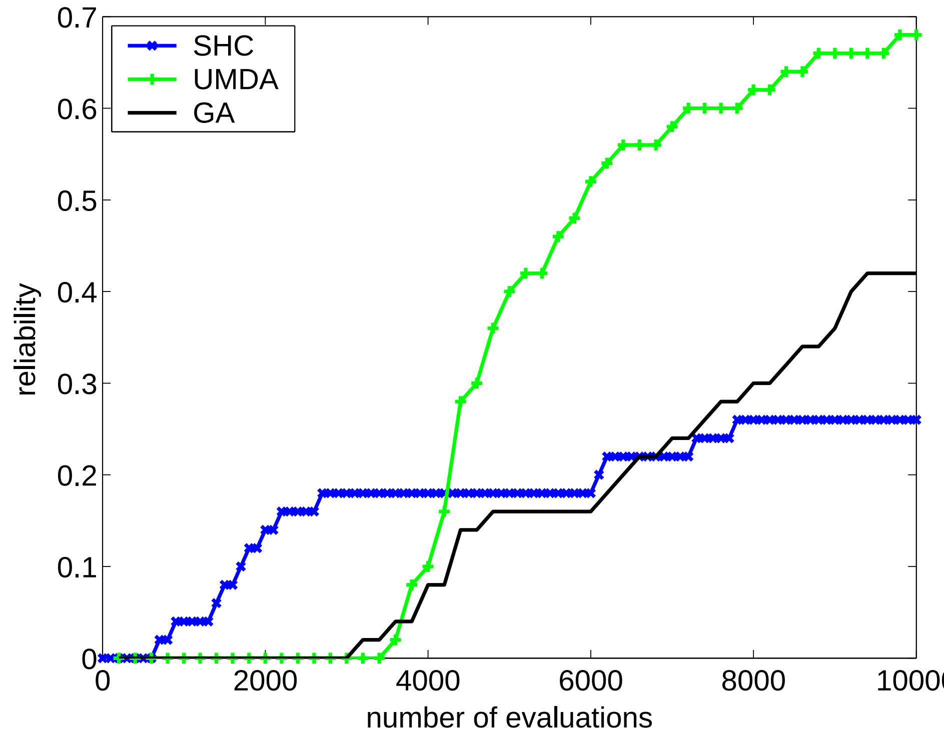
Results: reliability of the optimization

- Optimum:

$$[90_4/\pm75/\pm60_2/\pm45_5/\pm30_5]_s$$

- Compare UMDA to a GA and a hill-climbing algorithm (SHC)
- UMDA and SHC: optimized parameters, GA: same setting as UMDA

Reliability R: probability of finding the optimum in a given number of function evaluations.



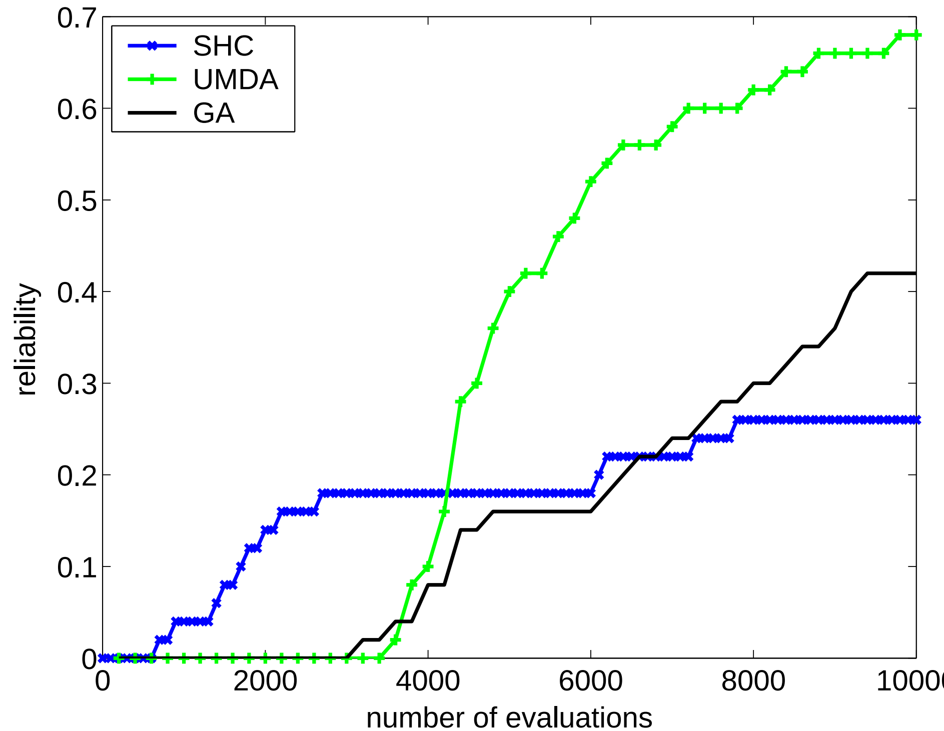
Results: reliability of the optimization

- Optimum:

$$[90_4/\pm75/\pm60_2/\pm45_5/\pm30_5]_s$$

- Compare UMDA to a GA and a hill-climbing algorithm (SHC)
- UMDA and SHC: optimized parameters, GA: same setting as UMDA

Reliability R: probability of finding the optimum in a given number of function evaluations.



- ➪ UMDA outperforms SHC because its distribution approach (global) allows it to handle narrow search spaces
- ➪ the performance of UMDA is substantially higher than that of GA for this problem

Improvement of the Statistical Model of Selected Points through Auxiliary Variables: the Double-Distribution Optimization Algorithm

Limitations of simple models

- Usual assumption: independent variables

$$p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2)\ldots p(x_n)$$

Limitations of simple models

- Usual assumption: independent variables

$$p(x_1, x_2, \dots, x_n) = p(x_1)p(x_2) \dots p(x_n)$$

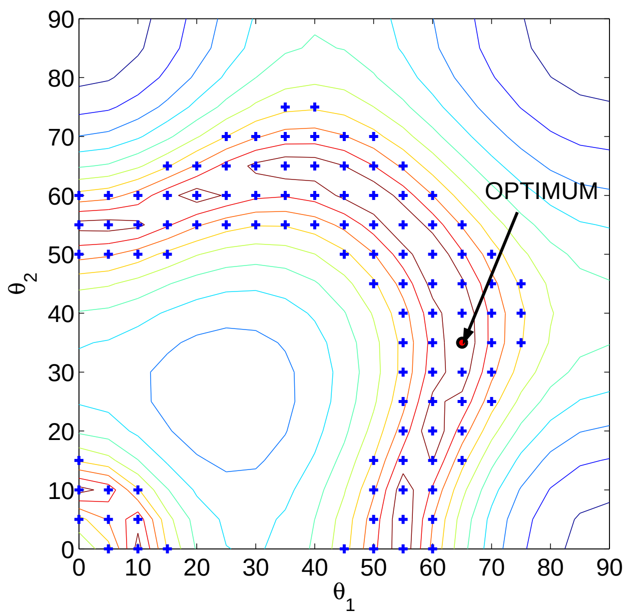
- Problem: does not work for problems with strong variable interactions

Limitations of simple models

- Usual assumption: independent variables

$$p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2)\ldots p(x_n)$$

- Problem: does not work for problems with strong variable interactions

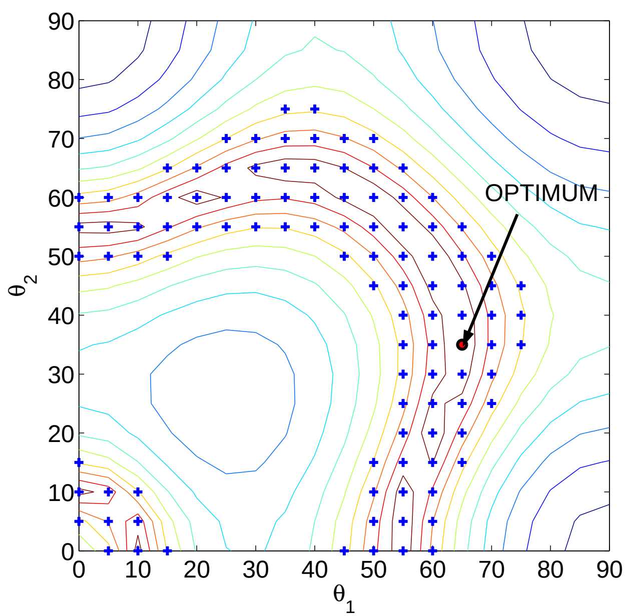


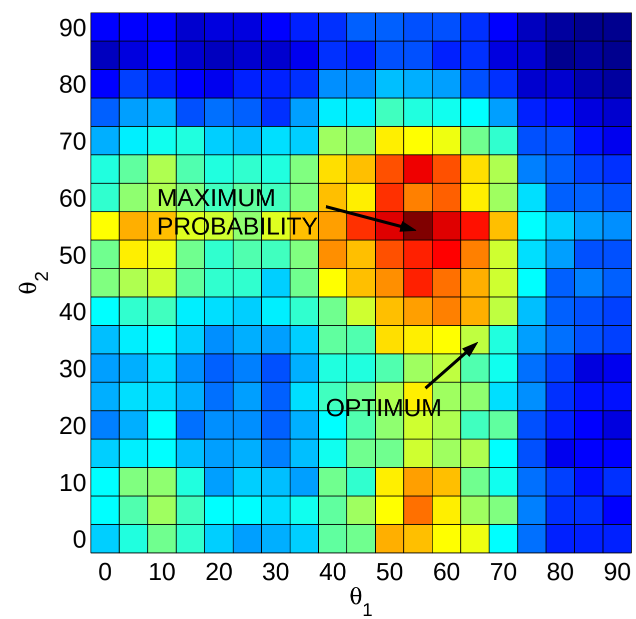
Limitations of simple models

- Usual assumption: independent variables

$$p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2)\ldots p(x_n)$$

- Problem: does not work for problems with strong variable interactions



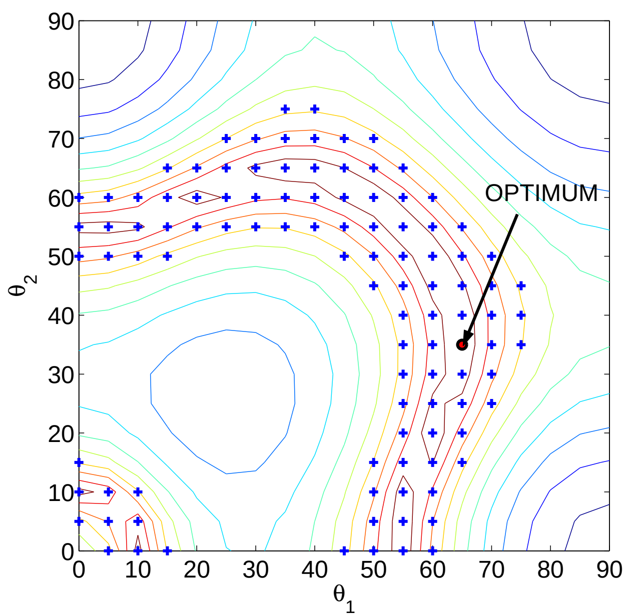


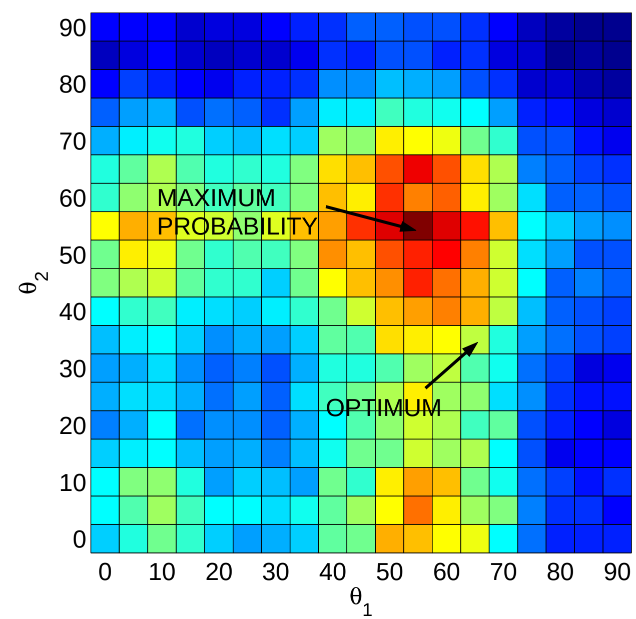
Limitations of simple models

- Usual assumption: independent variables

$$p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2)\ldots p(x_n)$$

- Problem: does not work for problems with strong variable interactions





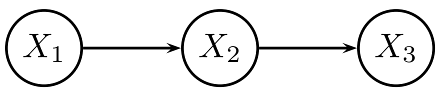
- High-probability areas do not coincide with high-fitness regions

Representation of variable dependencies

- Complex statistical models have been tried: chain models, tree models, full Bayesian networks (*Pelikan, 1999*)

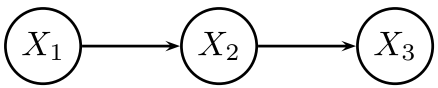
Representation of variable dependencies

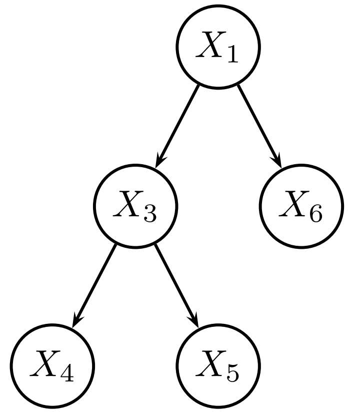
- Complex statistical models have been tried: chain models, tree models, full Bayesian networks (*Pelikan, 1999*)



Representation of variable dependencies

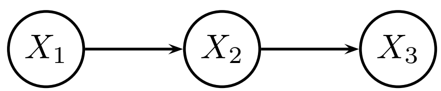
- Complex statistical models have been tried: chain models, tree models, full Bayesian networks (*Pelikan, 1999*)

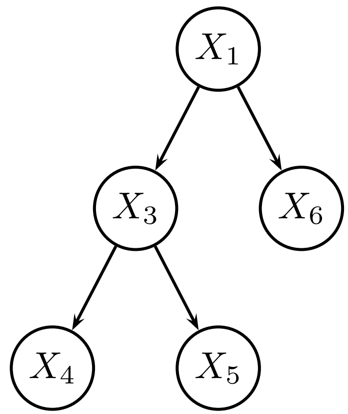


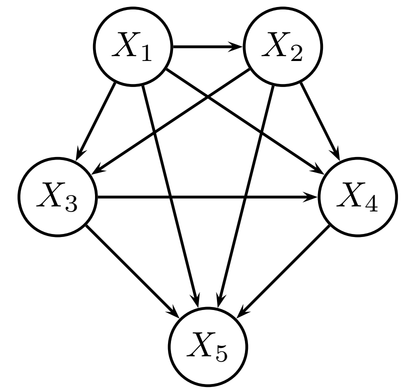


Representation of variable dependencies

- Complex statistical models have been tried: chain models, tree models, full Bayesian networks (*Pelikan, 1999*)

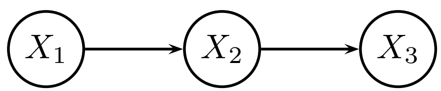


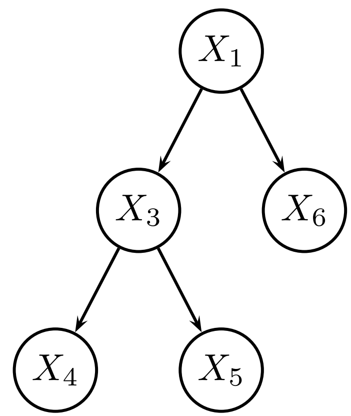


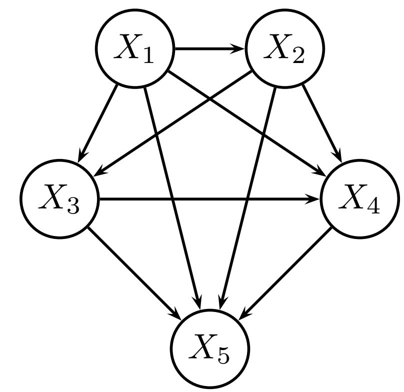


Representation of variable dependencies

- Complex statistical models have been tried: chain models, tree models, full Bayesian networks (*Pelikan, 1999*)



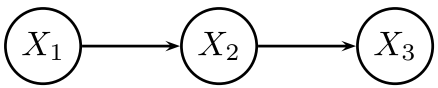


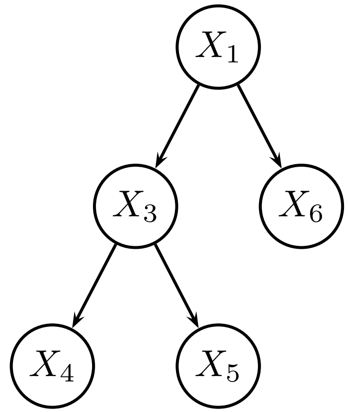


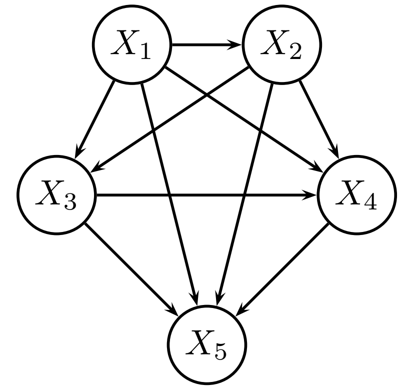
- Disadvantages:

Representation of variable dependencies

- Complex statistical models have been tried: chain models, tree models, full Bayesian networks (*Pelikan, 1999*)



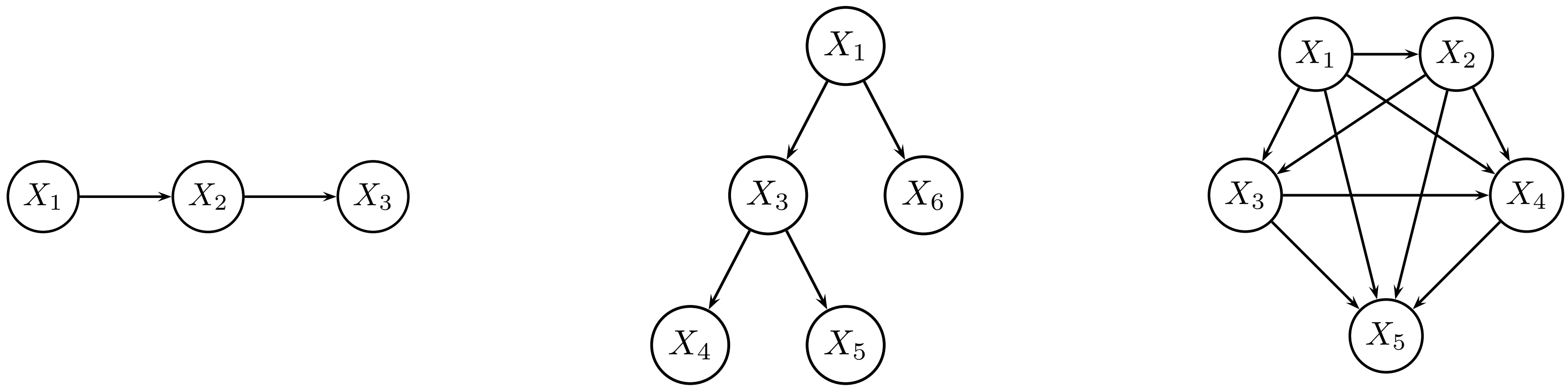




- Disadvantages:
  - The number of parameters m_j to estimate increases rapidly with the model complexity

Representation of variable dependencies

- Complex statistical models have been tried: chain models, tree models, full Bayesian networks (*Pelikan, 1999*)



- Disadvantages:
  - The number of parameters m_j to estimate increases rapidly with the model complexity
  - ➪ the population size needed to estimate these parameters ensure (with a constant confidence) increases with the model complexity because flexible models do not generalize well the information contained in the sample to other regions

Representation of joint actions of the variables via auxiliary variables

- **Observation :** in many situations, a small number of high order variables $\mathbf{V} = (V_1, V_2, \ldots, V_m)$ partially determine the objective function

Representation of joint actions of the variables via auxiliary variables

- **Observation :** in many situations, a small number of high order variables $\mathbf{V} = (V_1, V_2, \ldots, V_m)$ partially determine the objective function
- **Examples**:

Representation of joint actions of the variables via auxiliary variables

- **Observation :** in many situations, a small number of high order variables $\mathbf{V} = (V_1, V_2, \ldots, V_m)$ partially determine the objective function
- **Examples**:
  - the dimensions of a beam determine its flexural behavior through the moment of inertia I,

Representation of joint actions of the variables via auxiliary variables

- **Observation :** in many situations, a small number of high order variables $\mathbf{V} = (V_1, V_2, \ldots, V_m)$ partially determine the objective function
- **Examples**:
  - the dimensions of a beam determine its flexural behavior through the moment of inertia I,
  - the geometry of a vehicle influences its aerodynamic behavior via the drag coefficient C_V,

Representation of joint actions of the variables via auxiliary variables

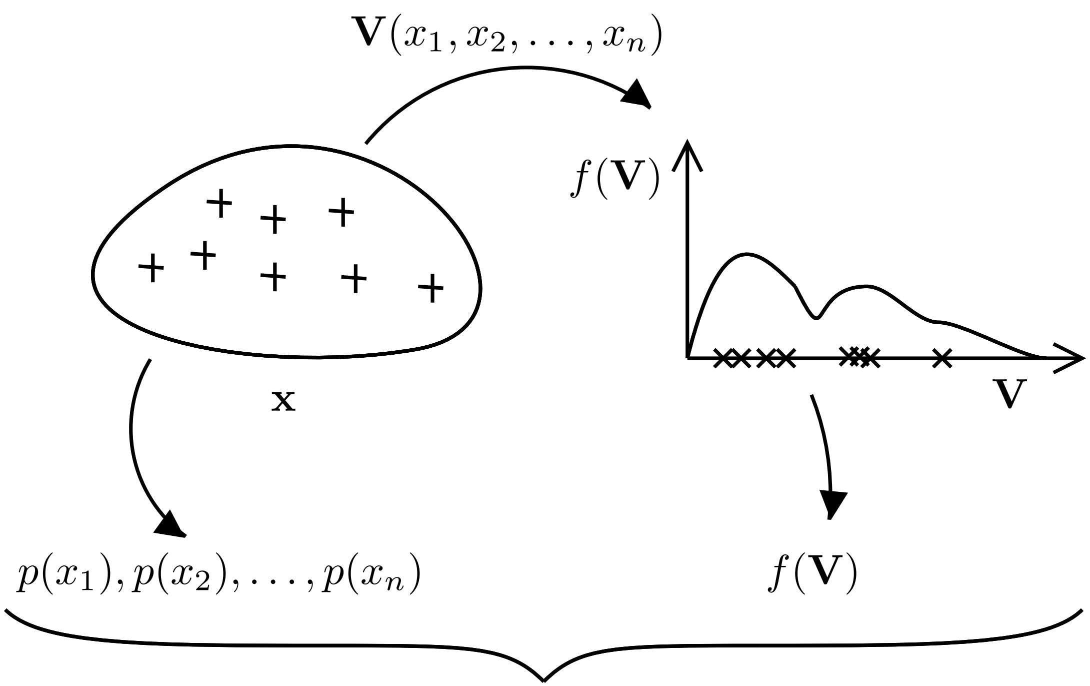
- **Observation :** in many situations, a small number of high order variables $\mathbf{V} = (V_1, V_2, \ldots, V_m)$ partially determine the objective function
- **Examples**:
  - the dimensions of a beam determine its flexural behavior through the moment of inertia I,
  - the geometry of a vehicle influences its aerodynamic behavior via the drag coefficient C_V,
  - the locations of the holes/particles in a porous media affect the flow through the permeability.

Representation of joint actions of the variables via auxiliary variables

- **Observation :** in many situations, a small number of high order variables $\mathbf{V} = (V_1, V_2, \ldots, V_m)$ partially determine the objective function
- **Examples**:
  - the dimensions of a beam determine its flexural behavior through the moment of inertia I,
  - the geometry of a vehicle influences its aerodynamic behavior via the drag coefficient C_V,
  - the locations of the holes/particles in a porous media affect the flow through the permeability.
- These meaningful quantities $\mathbf{V}$ reflect joint actions of the variables $\mathbf{x}$ and can be used as auxiliary variables to capture such interactions

Algorithm Principle

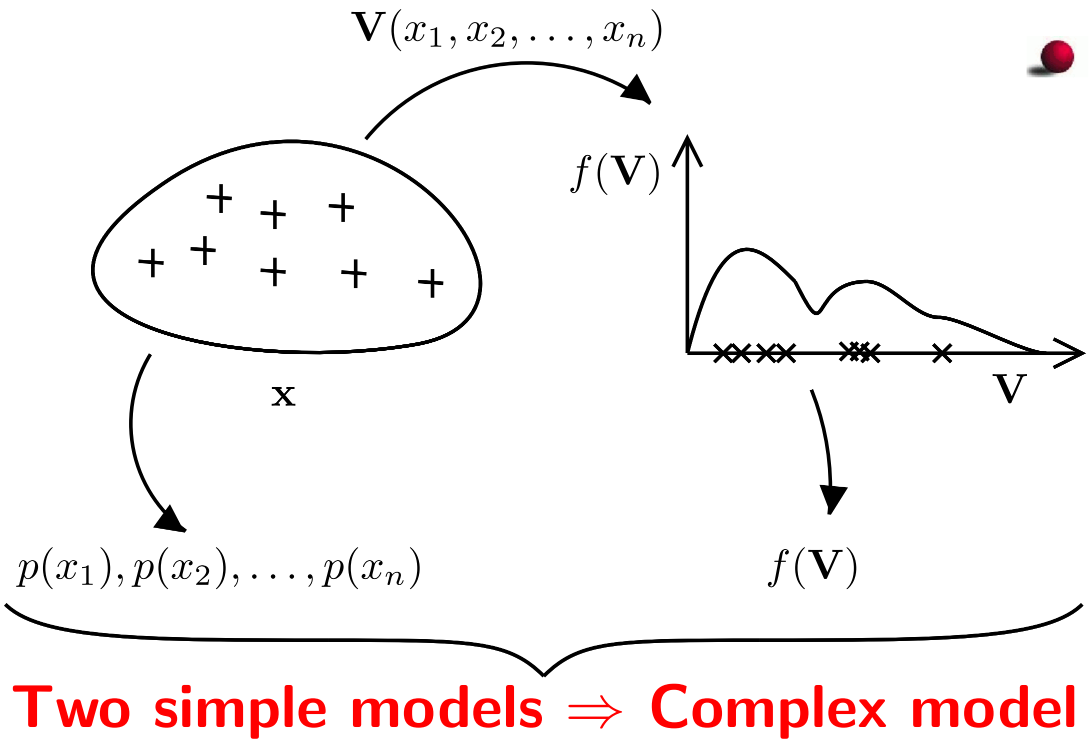
- The distribution $p(\mathbf{x})$ is represented by a simple model (UMDA)
- A simple distribution $f(\mathbf{V})$ is used to introduce variable dependencies



Two simple models $\Rightarrow$ Complex model

Algorithm Principle

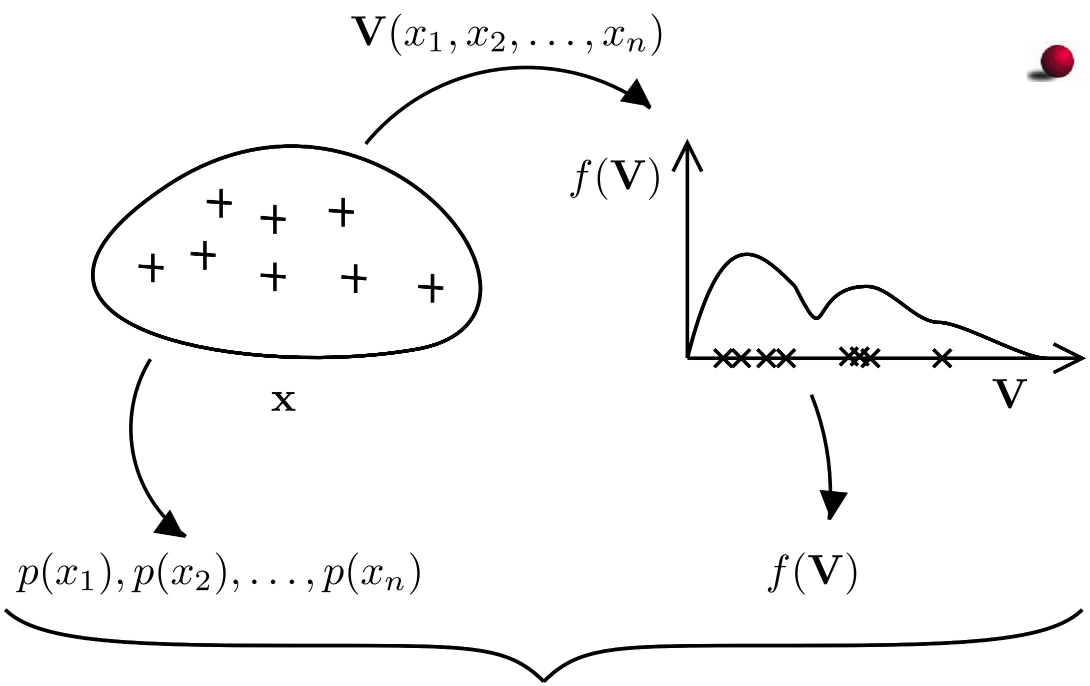
- The distribution $p(\mathbf{x})$ is represented by a simple model (UMDA)
- A simple distribution $f(\mathbf{V})$ is used to introduce variable dependencies



- Required attributes of the V's:

Algorithm Principle

- The distribution $p(\mathbf{x})$ is represented by a simple model (UMDA)
- A simple distribution $f(\mathbf{V})$ is used to introduce variable dependencies

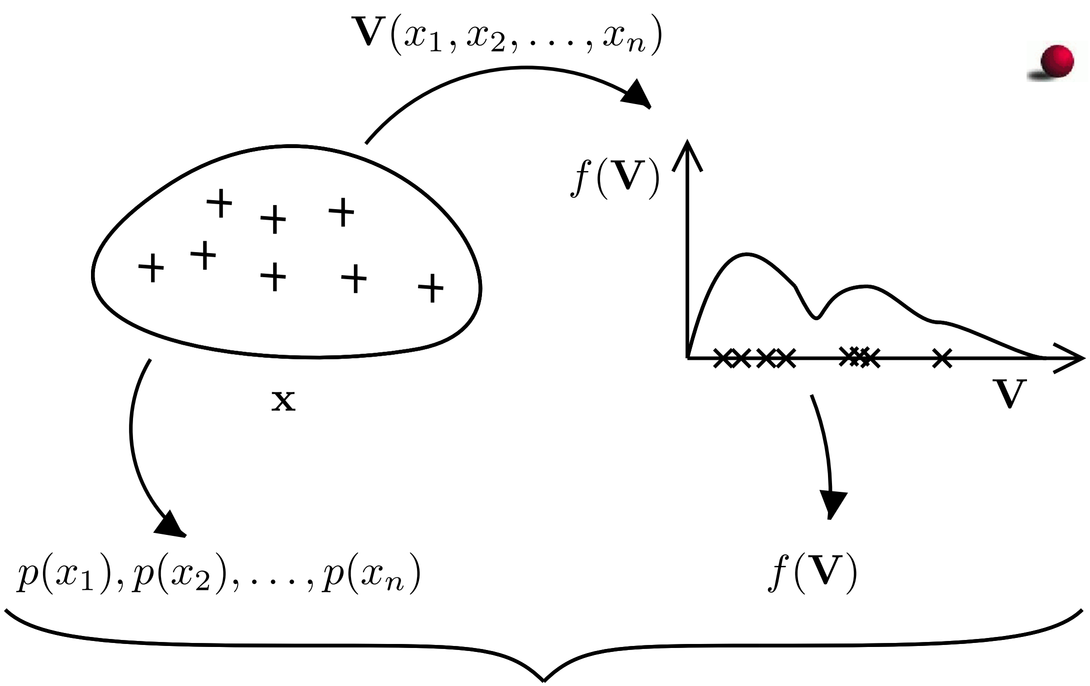


Two simple models $\Rightarrow$ Complex model

- Required attributes of the V's:
  1. $m < n$

Algorithm Principle

- The distribution $p(\mathbf{x})$ is represented by a simple model (UMDA)
- A simple distribution $f(\mathbf{V})$ is used to introduce variable dependencies

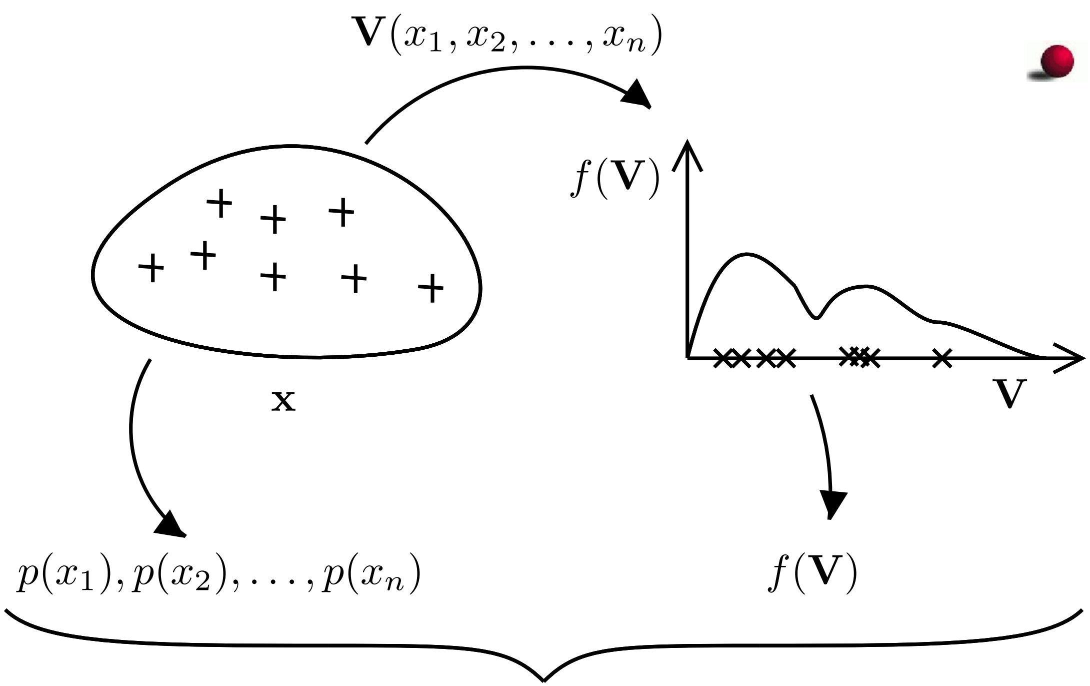


Two simple models $\Rightarrow$ Complex model

- Required attributes of the V's:
  1. $m < n$
  2. m does not grow with n

Algorithm Principle

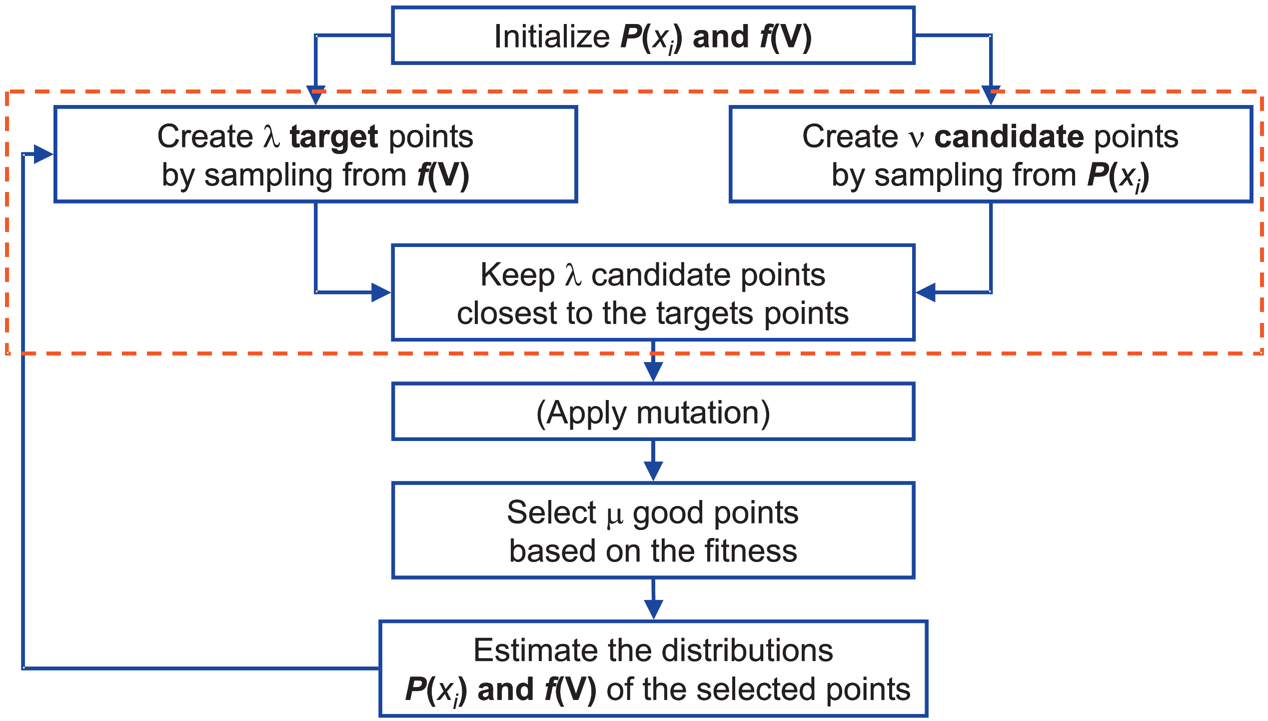
- The distribution $p(\mathbf{x})$ is represented by a simple model (UMDA)
- A simple distribution $f(\mathbf{V})$ is used to introduce variable dependencies



Two simple models $\Rightarrow$ Complex model

- Required attributes of the V's:
  1. $m < n$
  2. m does not grow with n
  3. inexpensive to compute

The Double-Distribution Optimization Algorithm



Goal: create points whose distribution that reflects both $p(\mathbf{x})$ and $f(\mathbf{V})$:

- $p(x_i)$ provides a pool of points that have correct marginal distributions.
- $f(\mathbf{V})$ is used as a filter that favors promising regions.
- The relative influence of the 2 distributions is adjusted through ν/λ

Application to Laminate Optimization Problems

Case of composite laminates: Auxiliary variables = “Lamination Parameters”

- $\mathbf{V} \equiv$ Lamination parameters = geometric contribution of the plies to the stiffness.

Case of composite laminates: Auxiliary variables = "Lamination Parameters"

- $\mathbf{V} \equiv$ Lamination parameters = geometric contribution of the plies to the stiffness.
- E.g. in-plane problem:

$$\begin{Bmatrix} A_{11} \\ A_{22} \\ A_{12} \\ A_{66} \end{Bmatrix} = h \begin{bmatrix} U_1 & U_2 & U_3 \\ U_1 & -U_2 & U_3 \\ U_5 & 0 & -U_3 \\ U_4 & 0 & -U_3 \end{bmatrix} \begin{Bmatrix} 1 \\ V_1^* \\ V_3^* \end{Bmatrix}$$

h: total laminate thickness, U_i's: material invariants

Case of composite laminates:
Auxiliary variables = "Lamination Parameters"

- $\mathbf{V} \equiv$ Lamination parameters = geometric contribution of the plies to the stiffness.
- E.g. in-plane problem:

$$\begin{Bmatrix} A_{11} \\ A_{22} \\ A_{12} \\ A_{66} \end{Bmatrix} = h \begin{bmatrix} U_1 & U_2 & U_3 \\ U_1 & -U_2 & U_3 \\ U_5 & 0 & -U_3 \\ U_4 & 0 & -U_3 \end{bmatrix} \begin{Bmatrix} 1 \\ V_1^* \\ V_3^* \end{Bmatrix}$$

h: total laminate thickness, U_i's: material invariants

- Symmetric balanced laminates $[\pm\theta_1, \pm\theta_2, \ldots, \pm\theta_n]_s$:

$$V^*_{\{1,3\}} = \frac{2}{h}\int_0^{h/2} \{\cos 2\theta, \cos 4\theta\} dz = \frac{1}{n}\sum_{k=1}^{n} \{\cos 2\theta_k, \cos 4\theta_k\}$$

Representation of the probability distributions

Distribution in the θ-domain

Representation of the probability distributions

Distribution in the θ-domain

- $\mathbf{x} \equiv \boldsymbol{\theta}$

Representation of the probability distributions

Distribution in the θ-domain

- $\mathbf{x} \equiv \boldsymbol{\theta}$
- Discrete univariate model ⇨ estimate marginal frequencies

Representation of the probability distributions

Distribution in the θ-domain

- $\mathbf{x} \equiv \boldsymbol{\theta}$
- Discrete univariate model ➩ estimate marginal frequencies

Distribution in the $\mathbf{V}$-domain

Representation of the probability distributions

Distribution in the θ-domain

- $\mathbf{x} \equiv \boldsymbol{\theta}$
- Discrete univariate model
  ➪ estimate marginal frequencies

Distribution in the $\mathbf{V}$-domain

- Continuous variables

Representation of the probability distributions

Distribution in the θ-domain

- $\mathbf{x} \equiv \boldsymbol{\theta}$
- Discrete univariate model ⇨ estimate marginal frequencies

Distribution in the $\mathbf{V}$-domain

- Continuous variables
- Use kernel density estimate:

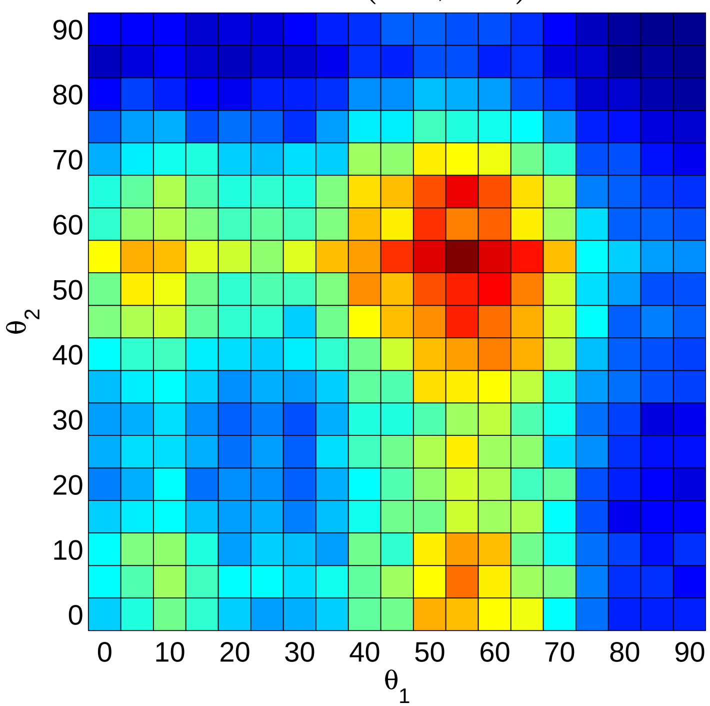
$$f(\mathbf{V}) = \frac{1}{\mu} \sum_{i=1}^{\mu} K\left(\mathbf{V} - \mathbf{V}_i\right)$$

In this work, we used Gaussian kernels:

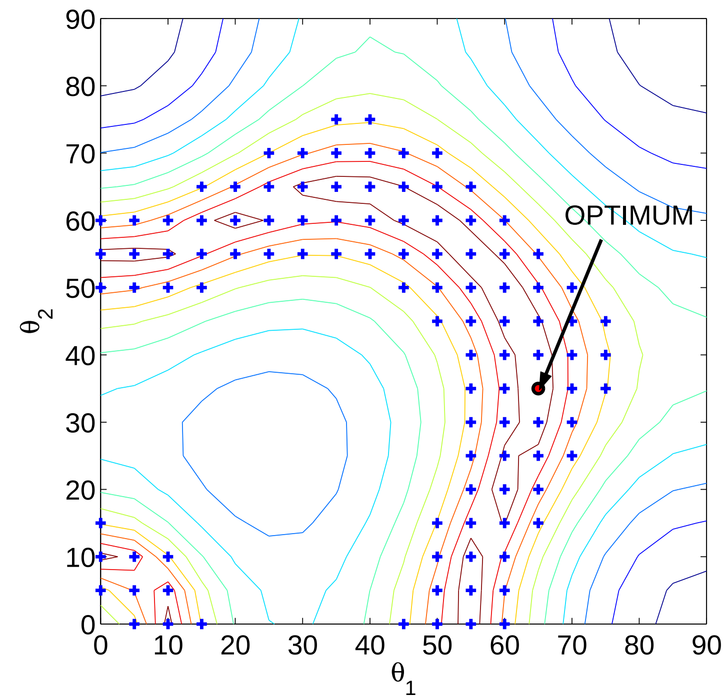
$$K(\mathbf{u}) = \frac{1}{(2\pi)^{d/2}\sigma^d} \exp\left(-\frac{\mathbf{u}^T\mathbf{u}}{\sigma^2}\right)$$

Estimated Distributions: UMDA and DDOA

UMDA: probability concentrated around $(55, 55)$

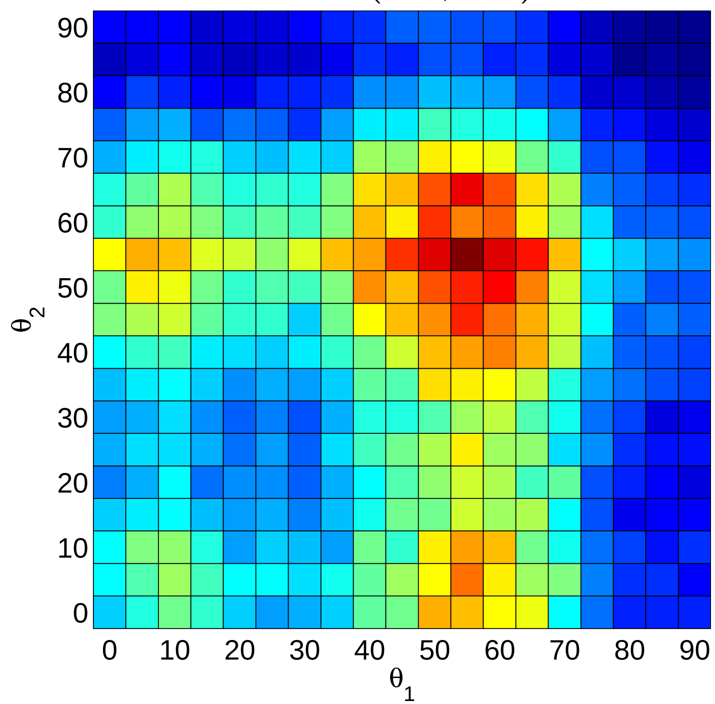


Fitness function and selected points

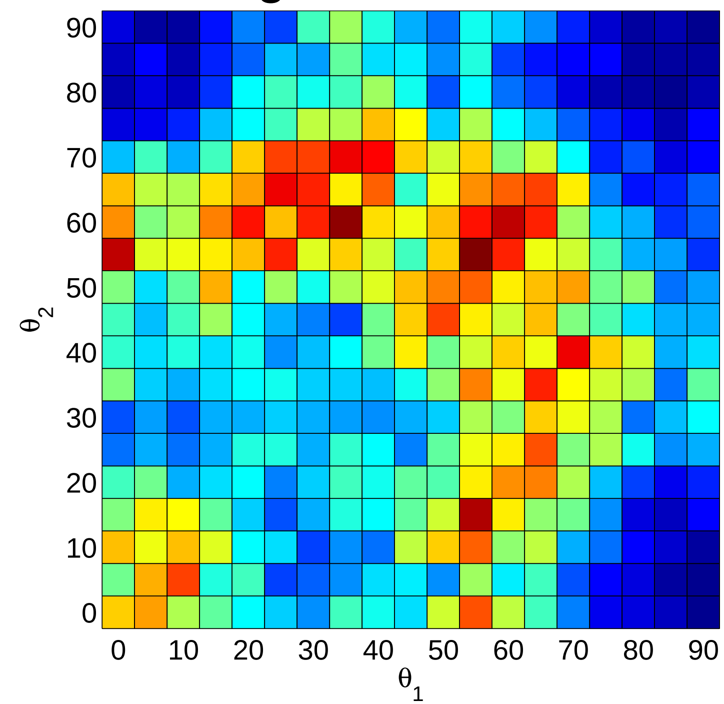


Estimated Distributions: UMDA and DDOA

UMDA: probability concentrated around $(55, 55)$

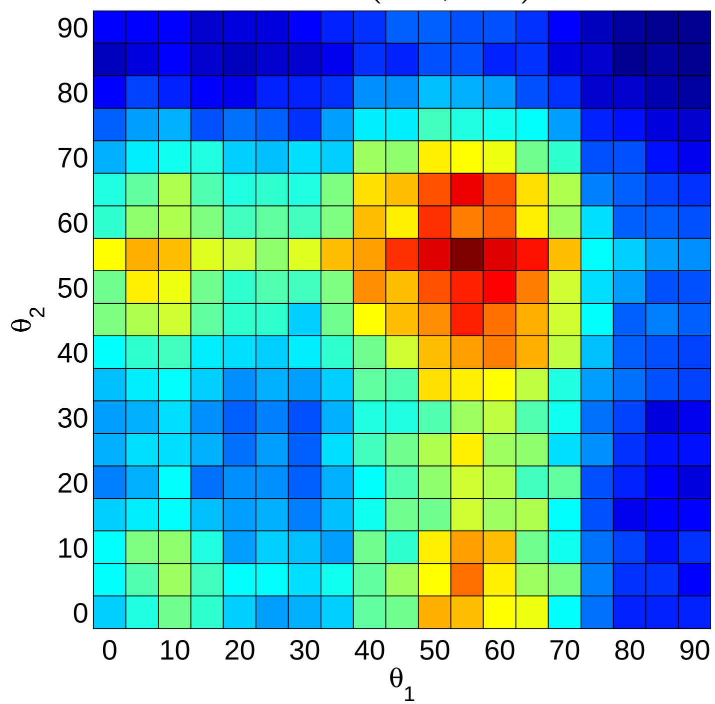


DDOA: the mass is distributed along the “tunnel”

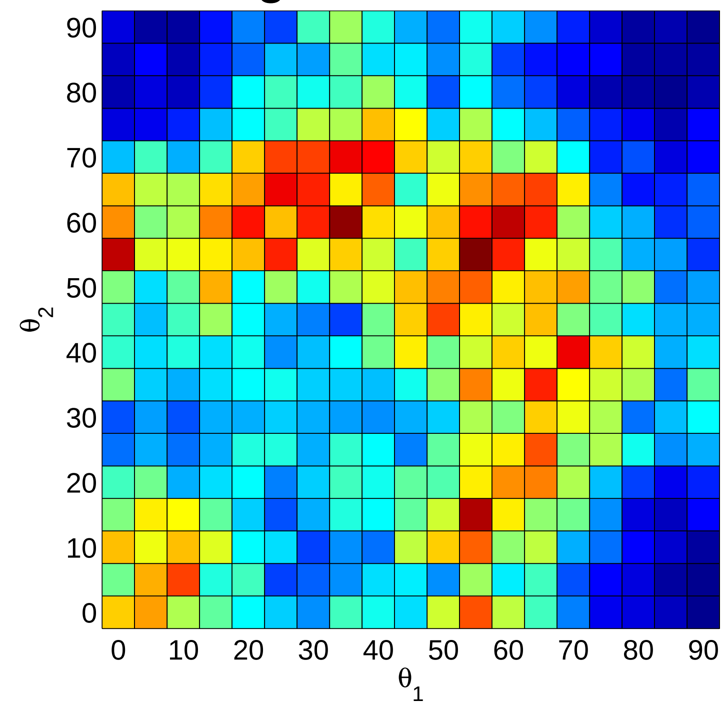


Estimated Distributions: UMDA and DDOA

UMDA: probability concentrated around $(55, 55)$



DDOA: the mass is distributed along the "tunnel"



- Sampling from $p(\theta_1), \ldots, p(\theta_n)$ leads to an inaccurate distribution
- $\mathbf{V}$-based selection improves the distribution by introducing variable dependencies

Application I: Design of “zero-CTE” laminates

- Objective: minimize the longitudinal coefficient of thermal expansion (CTE) subject to a constraint on the first natural vibration frequency, for a 50 in $\times$ 15 in plate

minimize $|\overline{\alpha}_x|$
such that $f_1 \geq f_{\text{min}}$

Application I: Design of “zero-CTE” laminates

- Objective: minimize the longitudinal coefficient of thermal expansion (CTE) subject to a constraint on the first natural vibration frequency, for a 50 in $\times$ 15 in plate

minimize $|\overline{\alpha}_x|$
such that $f_1 \geq f_{\text{min}}$

- Constraint enforced through a penalty approach

Application I: Design of “zero-CTE” laminates

- Objective: minimize the longitudinal coefficient of thermal expansion (CTE) subject to a constraint on the first natural vibration frequency, for a 50 in $\times$ 15 in plate

minimize $|\overline{\alpha}_x|$
such that $f_1 \geq f_{\text{min}}$

- Constraint enforced through a penalty approach
- Possible values of the angles: $\theta_k \in \{0°, 22.5°, 45°, 67.5°, 90°\}$

Application I: Design of “zero-CTE” laminates

- Objective: minimize the longitudinal coefficient of thermal expansion (CTE) subject to a constraint on the first natural vibration frequency, for a 50 in $\times$ 15 in plate

minimize $|\overline{\alpha}_x|$
such that $f_1 \geq f_{\text{min}}$

- Constraint enforced through a penalty approach
- Possible values of the angles: $\theta_k \in \{0°, 22.5°, 45°, 67.5°, 90°\}$
- Case $n = 12$: optimum = $[90_4/\pm 67.5/0_6/\pm 22.5_6]_s$

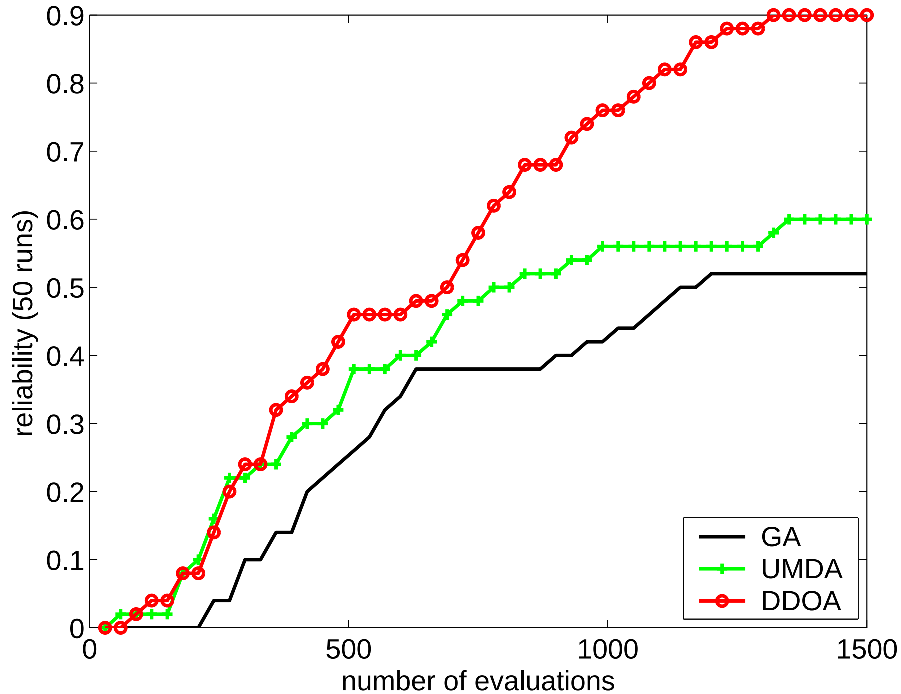
Application I: Design of “zero-CTE” laminates

- Objective: minimize the longitudinal coefficient of thermal expansion (CTE) subject to a constraint on the first natural vibration frequency, for a 50 in $\times$ 15 in plate

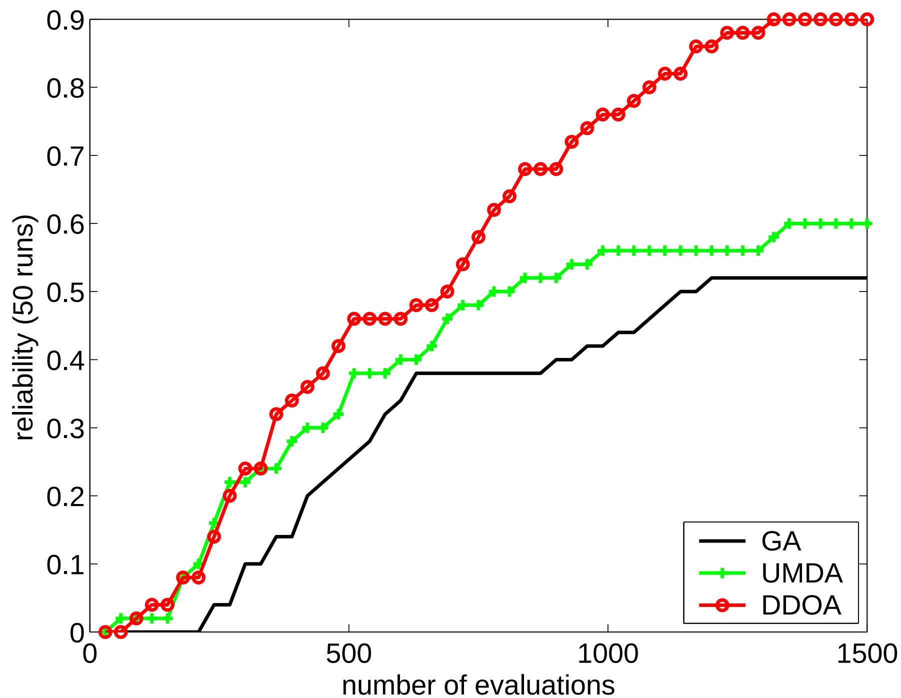
$$\textbf{minimize } |\overline{\alpha}_x|$$
$$\textbf{such that } f_1 \geq f_{\text{min}}$$

- Constraint enforced through a penalty approach
- Possible values of the angles: $\theta_k \in \{0°, 22.5°, 45°, 67.5°, 90°\}$
- Case $n = 12$: optimum = $[90_4/\pm 67.5/0_6/\pm 22.5_6]_s$
- Particularity: response is a function of the lamination parameters only
  ➾ V’s provide reliable information about the optimum

Comparison with UMDA and GA

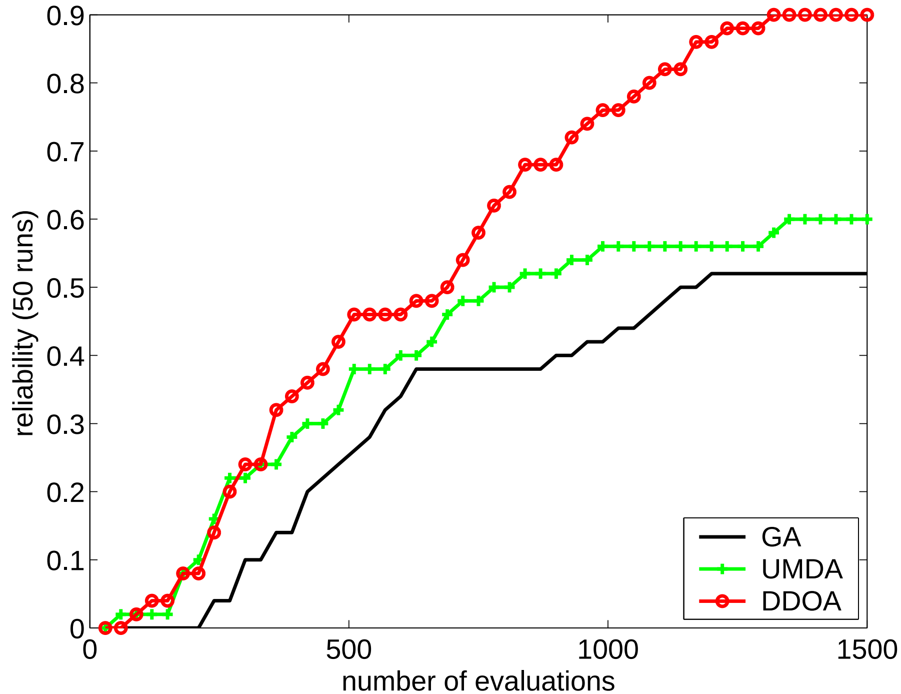


Comparison with UMDA and GA



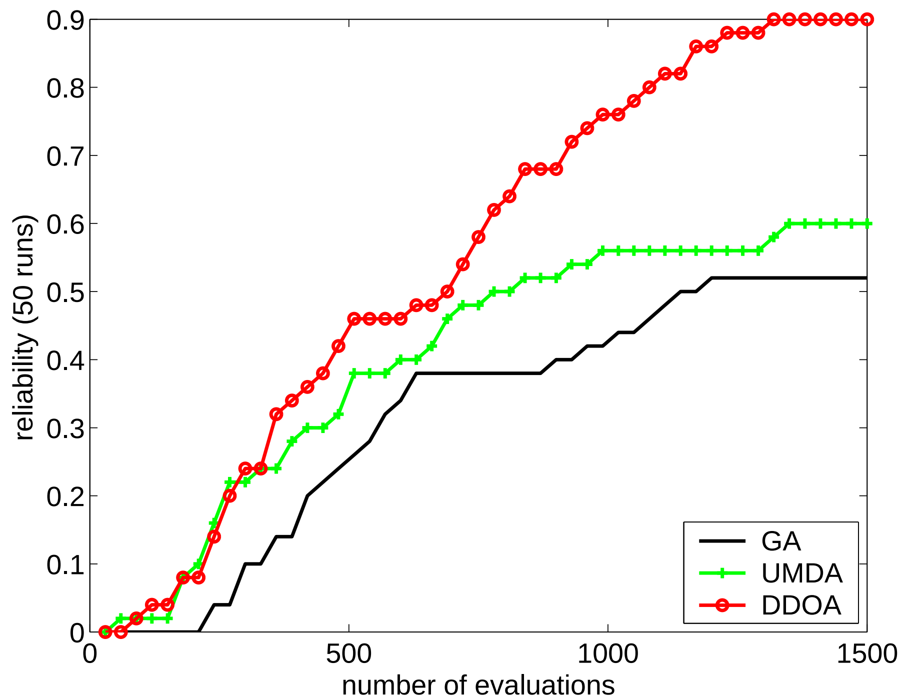
- GA and UMDA’s progress falls off after 600 analyses

Comparison with UMDA and GA



- GA and UMDA's progress falls off after 600 analyses
- DDOA reaches high a reliability (90%)

Comparison with UMDA and GA

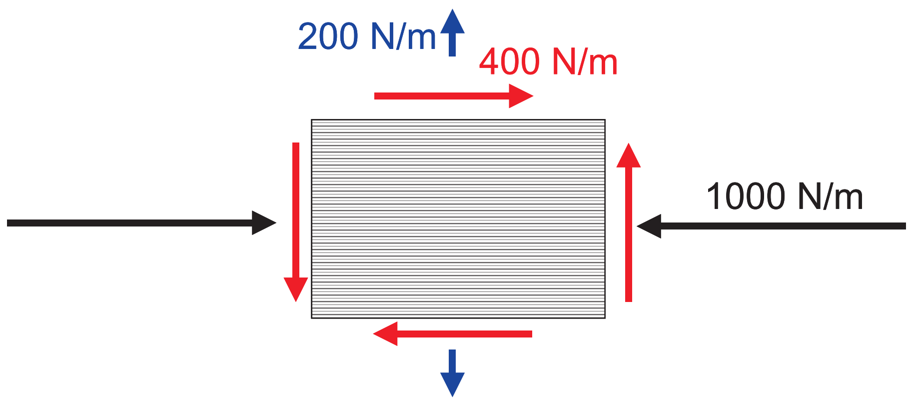


- GA and UMDA's progress falls off after 600 analyses
- DDOA reaches high a reliability (90%)
- For this problem, DDOA benefits from the use of auxiliary variables (incorporation of physics-based information improves accuracy of $p(\mathbf{x})$)

Application II: Strength Maximization

- Strength maximization problem:

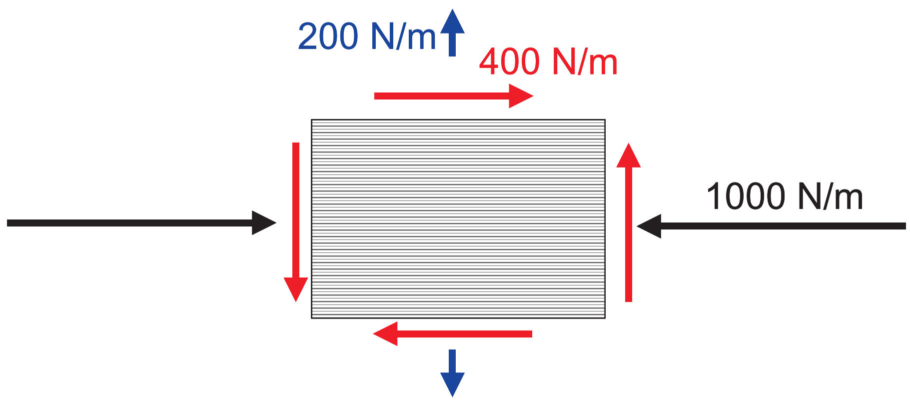
$$\textbf{maximize } \lambda_s = \min_{k=1}^{n}\left(\min\left(\frac{\epsilon_1^{\text{ult}}}{\epsilon_1(k)}, \frac{\epsilon_2^{\text{ult}}}{\epsilon_2(k)}, \frac{\gamma_{12}^{\text{ult}}}{\gamma_{12}(k)}\right)\right)$$



Application II: Strength Maximization

- Strength maximization problem:

$$\textbf{maximize}\ \lambda_s = \min_{k=1}^{n}\left(\min\left(\frac{\epsilon_1^{\text{ult}}}{\epsilon_1(k)}, \frac{\epsilon_2^{\text{ult}}}{\epsilon_2(k)}, \frac{\gamma_{12}^{\text{ult}}}{\gamma_{12}(k)}\right)\right)$$

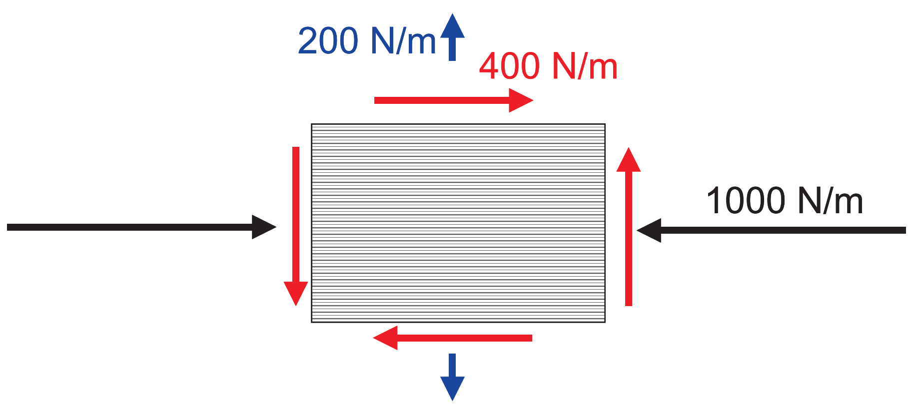


- Characteristics of this problem:

Application II: Strength Maximization

- Strength maximization problem:

$$\textbf{maximize } \lambda_s = \min_{k=1}^{n} \left(\min \left(\frac{\epsilon_1^{\text{ult}}}{\epsilon_1(k)}, \frac{\epsilon_2^{\text{ult}}}{\epsilon_2(k)}, \frac{\gamma_{12}^{\text{ult}}}{\gamma_{12}(k)} \right) \right)$$

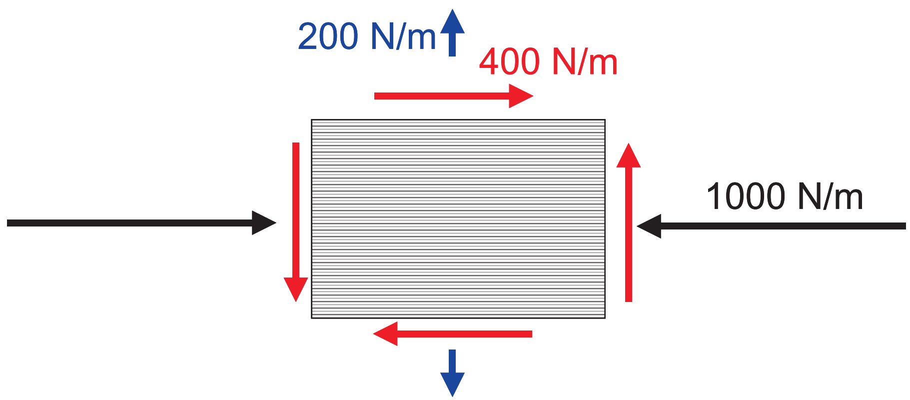


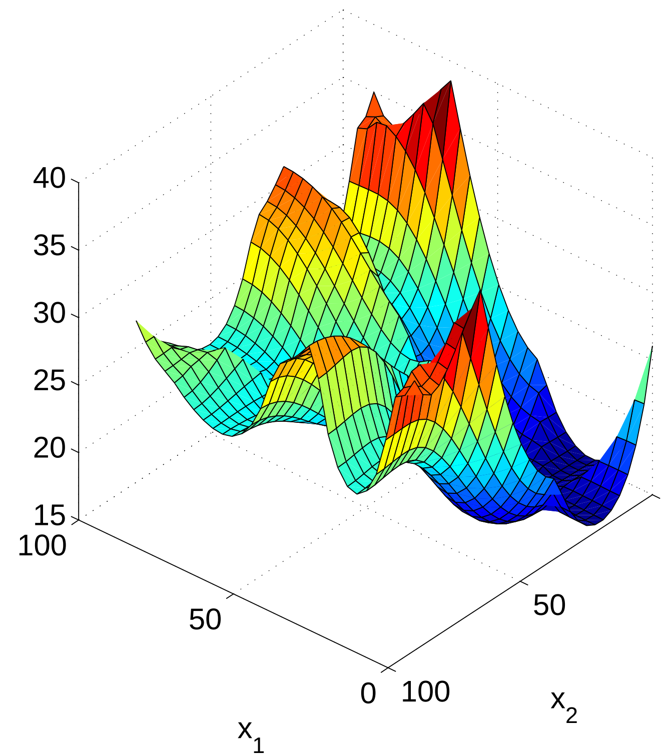
- Characteristics of this problem:
  - the V's do not capture all the response: $\lambda_s = \lambda_s(V_1, V_3, \theta_1, \ldots, \theta_n)$

Application II: Strength Maximization

- Strength maximization problem:

$$\textbf{maximize } \lambda_s = \min_{k=1}^{n}\left(\min\left(\frac{\epsilon_1^{\text{ult}}}{\epsilon_1(k)}, \frac{\epsilon_2^{\text{ult}}}{\epsilon_2(k)}, \frac{\gamma_{12}^{\text{ult}}}{\gamma_{12}(k)}\right)\right)$$





- Characteristics of this problem:
  - the V's do not capture all the response: $\lambda_s = \lambda_s(V_1, V_3, \theta_1, \ldots, \theta_n)$
  - many local optima

Comparison with UMDA and GA

- Case $n = 12$

Comparison with UMDA and GA

- Case $n = 12$
- $\theta_k \in \{0°, 22.5°, 45°, 67.5°,$
  $90°$

Comparison with UMDA and GA

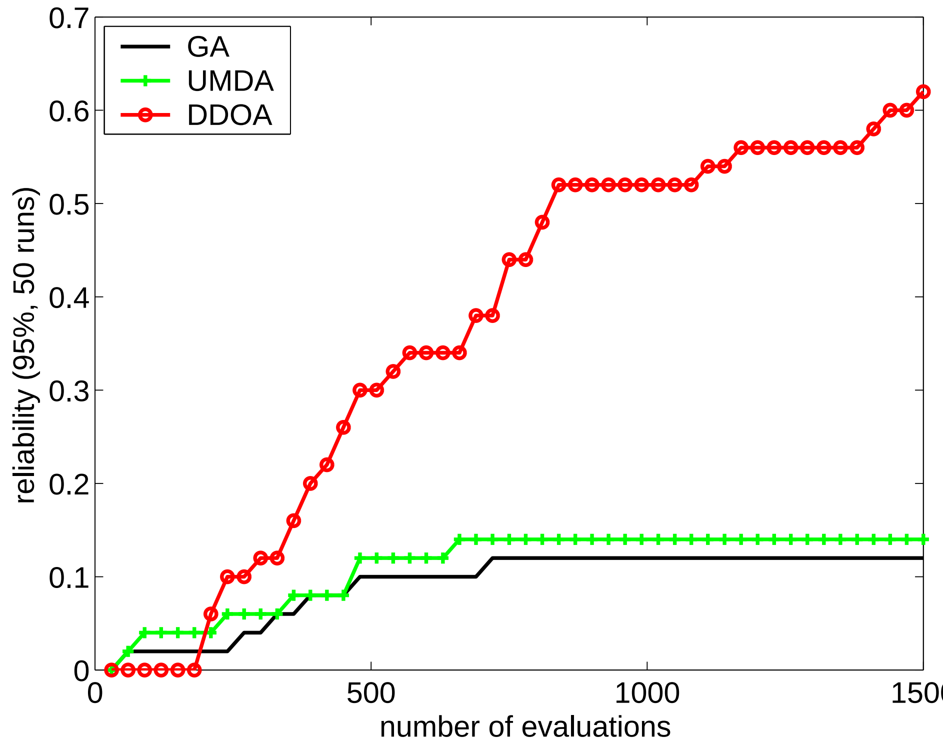
- Case $n = 12$
- $\theta_k \in \{0°, 22.5°, 45°, 67.5°,$ $90°$
- Global optimum: $[0_{14}/\pm 67.5_5]_s$, $\lambda_s = 4.74$

Comparison with UMDA and GA

- Case $n = 12$
- $\theta_k \in \{0°, 22.5°, 45°, 67.5°, 90°$
- Global optimum: $[0_{14}/\pm 67.5_5]_s$, $\lambda_s = 4.74$
- Parameters: $\lambda = \mu = 30$, linear ranking selection, $p_m = 0.02$

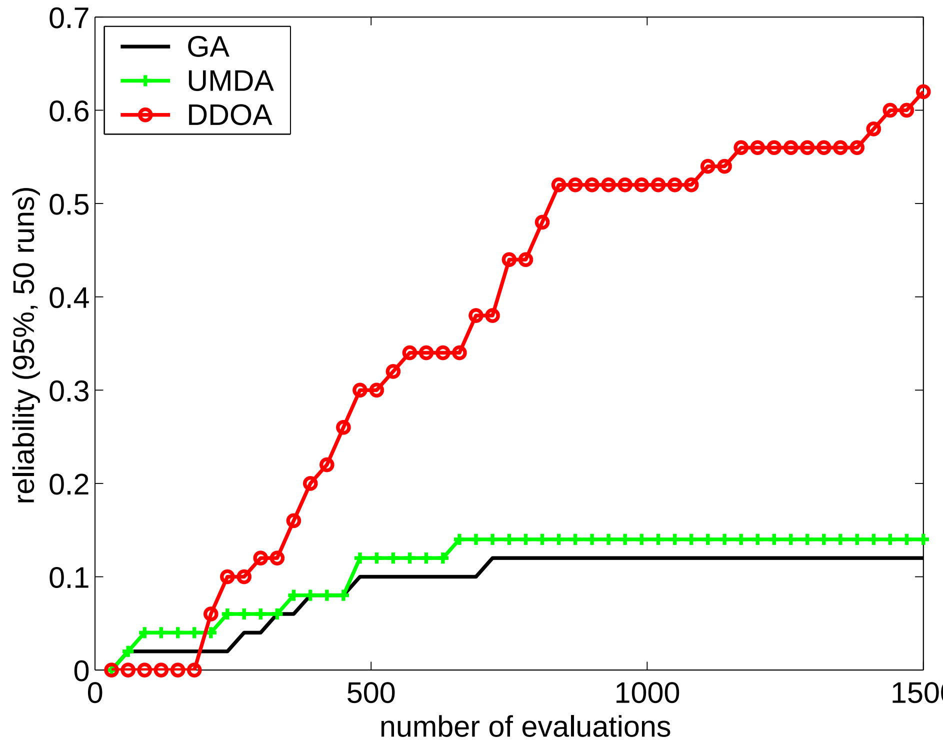
Comparison with UMDA and GA

- Case $n = 12$
- $\theta_k \in \{0°, 22.5°, 45°, 67.5°, 90°$
- Global optimum: $[0_{14}/\pm 67.5_5]_s$, $\lambda_s = 4.74$
- Parameters: $\lambda = \mu = 30$, linear ranking selection, $p_m = 0.02$



Comparison with UMDA and GA

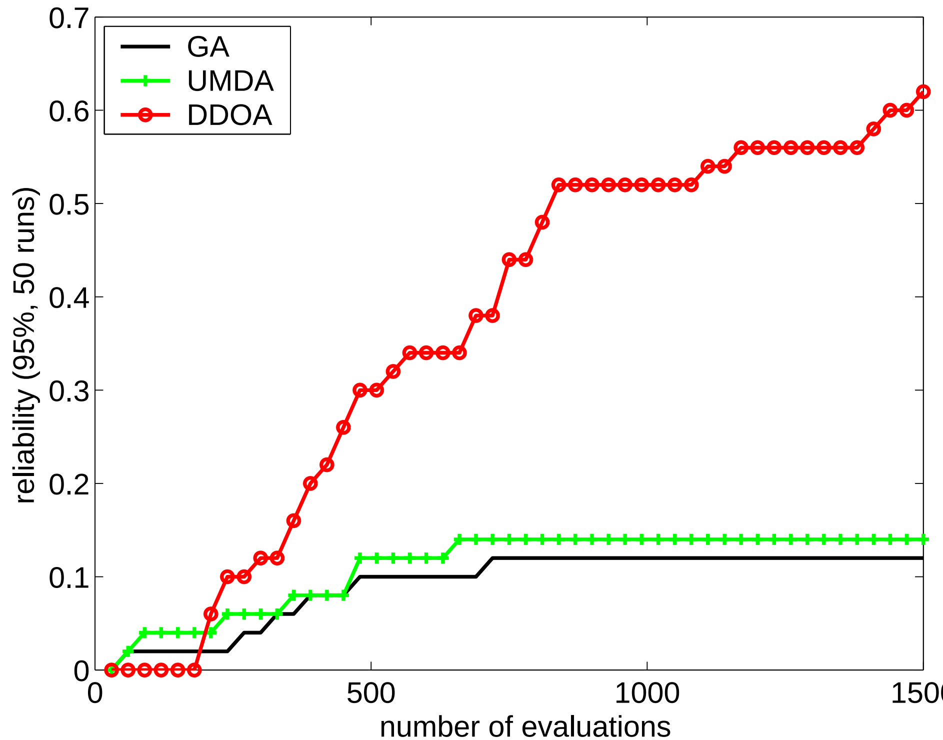
- Case $n = 12$
- $\theta_k \in \{0°, 22.5°, 45°, 67.5°, 90°$
- Global optimum: $[0_{14}/\pm 67.5_5]_s$, $\lambda_s = 4.74$
- Parameters: $\lambda = \mu = 30$, linear ranking selection, $p_m = 0.02$



➩ UMDA and GA locally improve the initial candidate solutions, but fail to converge to high-fitness solutions

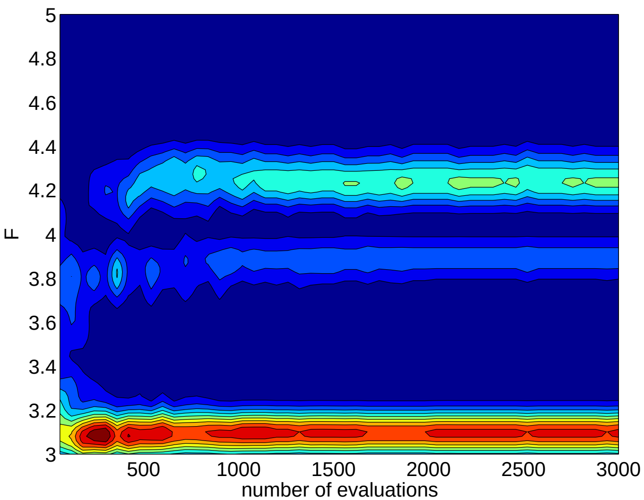
Comparison with UMDA and GA

- Case $n = 12$
- $\theta_k \in \{0°, 22.5°, 45°, 67.5°, 90°$
- Global optimum: $[0_{14}/\pm 67.5_5]_s$, $\lambda_s = 4.74$
- Parameters: $\lambda = \mu = 30$, linear ranking selection, $p_m = 0.02$

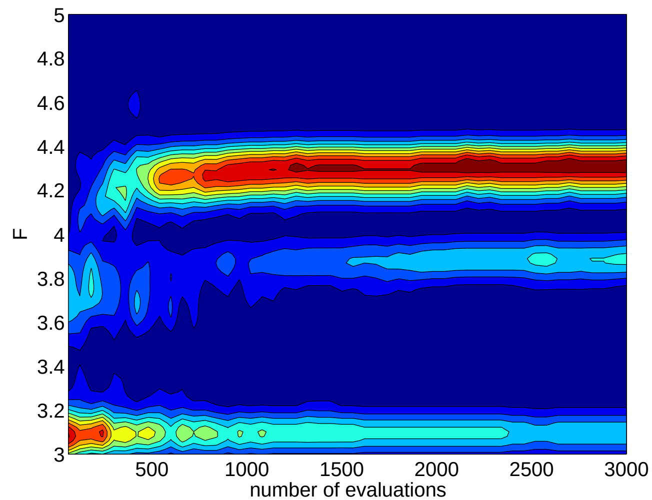


- UMDA and GA locally improve the initial candidate solutions, but fail to converge to high-fitness solutions
- DDOA reliably finds the optimum

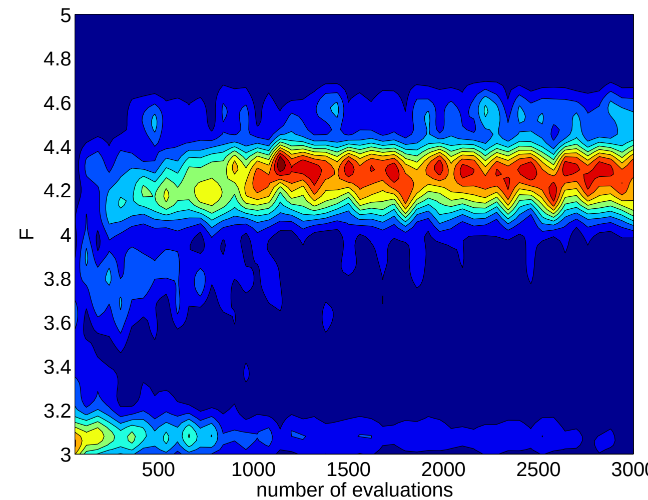
Distribution of the solutions found



UMDA

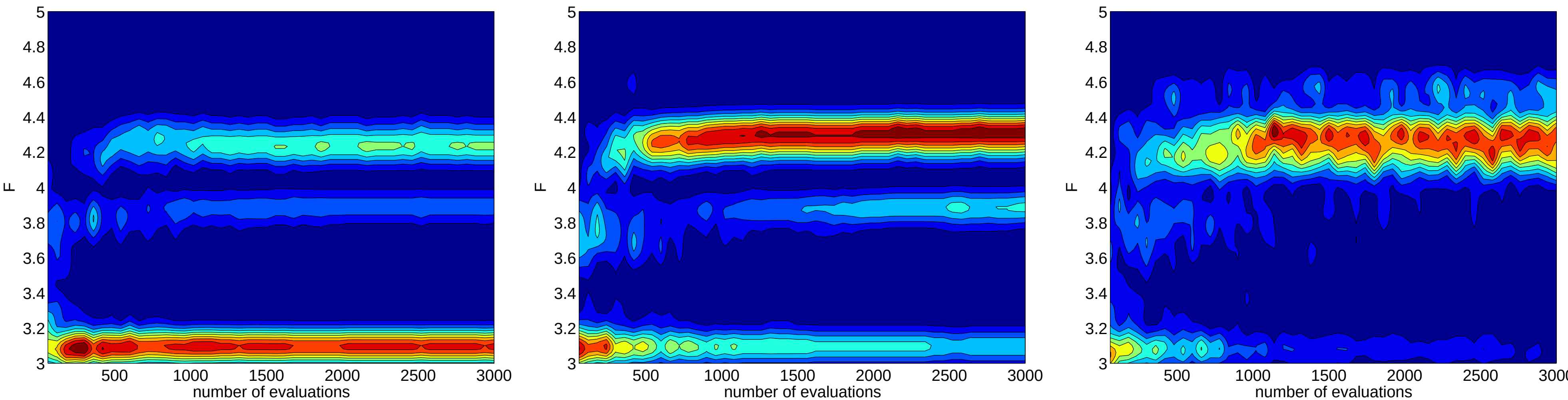


GA



DDOA

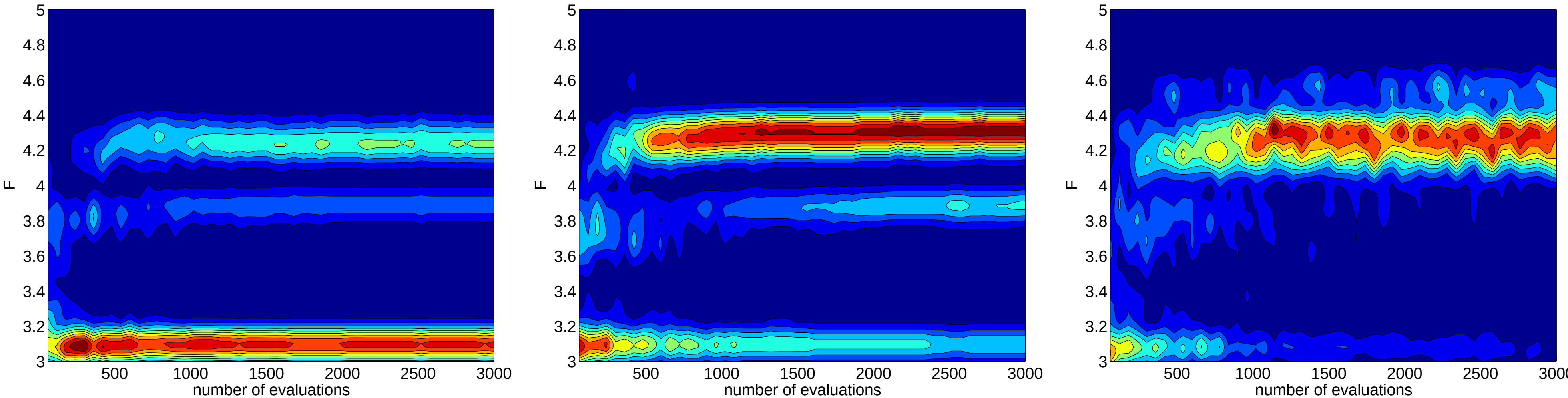
Distribution of the solutions found



UMDA GA DDOA

- UMDA converges to poor solutions easy to find (e.g. $[0_4/\pm 22.5_3/\pm 45_3/\pm 67.5/90_6]_s$): the univariate model cannot identify good regions

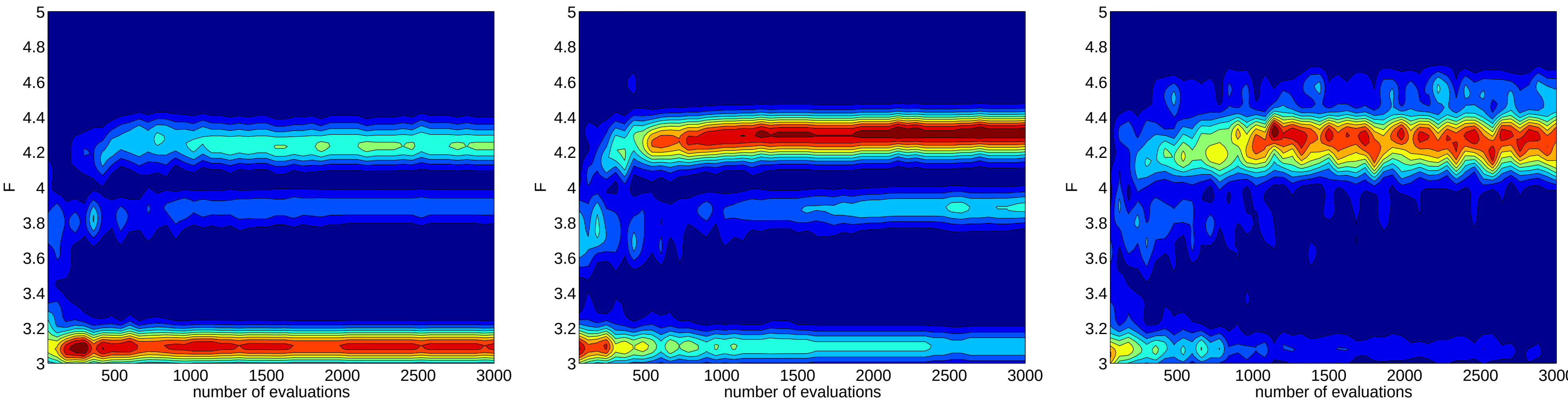
Distribution of the solutions found



UMDA GA DDOA

- UMDA converges to poor solutions easy to find (e.g. $[0_4/\pm 22.5_3/\pm 45_3/\pm 67.5/90_6]_s$): the univariate model cannot identify good regions
- GA correctly converges to the neighborhood of the global optimum, but does not explore it (small variance)

Distribution of the solutions found



UMDA GA DDOA

- UMDA converges to poor solutions easy to find (e.g. $[0_4/\pm 22.5_3/\pm 45_3/\pm 67.5/90_6]_s$): the univariate model cannot identify good regions
- GA correctly converges to the neighborhood of the global optimum, but does not explore it (small variance)
- DDOA focuses the search on high-fitness regions. The two lower basins of attraction present in GA and UMDA bearly appear

Diversity preservation mechanisms

- Theoretical EDA: infinite population λ

Diversity preservation mechanisms

- Theoretical EDA: infinite population λ
- Implementation: finite (small) population ➩ estimation error on $p(\mathbf{x})$

Diversity preservation mechanisms

- Theoretical EDA: infinite population λ
- Implementation: finite (small) population ⇨ estimation error on $p(\mathbf{x})$
- Observation: tendency to underestimate p in unexplored regions (loss of variable values = “premature convergence”)

Diversity preservation mechanisms

- Theoretical EDA: infinite population λ
- Implementation: finite (small) population ⇨ estimation error on $p(\mathbf{x})$
- Observation: tendency to underestimate p in unexplored regions (loss of variable values = “premature convergence”)

⇨ Diversity preservation mechanisms must be implemented to compensate for lost points

Diversity preservation mechanisms

- Theoretical EDA: infinite population λ
- Implementation: finite (small) population ⇨ estimation error on $p(\mathbf{x})$
- Observation: tendency to underestimate p in unexplored regions (loss of variable values = “premature convergence”)

⇨ Diversity preservation mechanisms must be implemented to compensate for lost points

- Two mechanisms:

Diversity preservation mechanisms

- Theoretical EDA: infinite population λ
- Implementation: finite (small) population ⇨ estimation error on $p(\mathbf{x})$
- Observation: tendency to underestimate p in unexplored regions (loss of variable values = "premature convergence")

⇨ Diversity preservation mechanisms must be implemented to compensate for lost points

- Two mechanisms:
  - mutation: a perturbation is applied with probability p_m to each variable θ_k of each of the λ created points

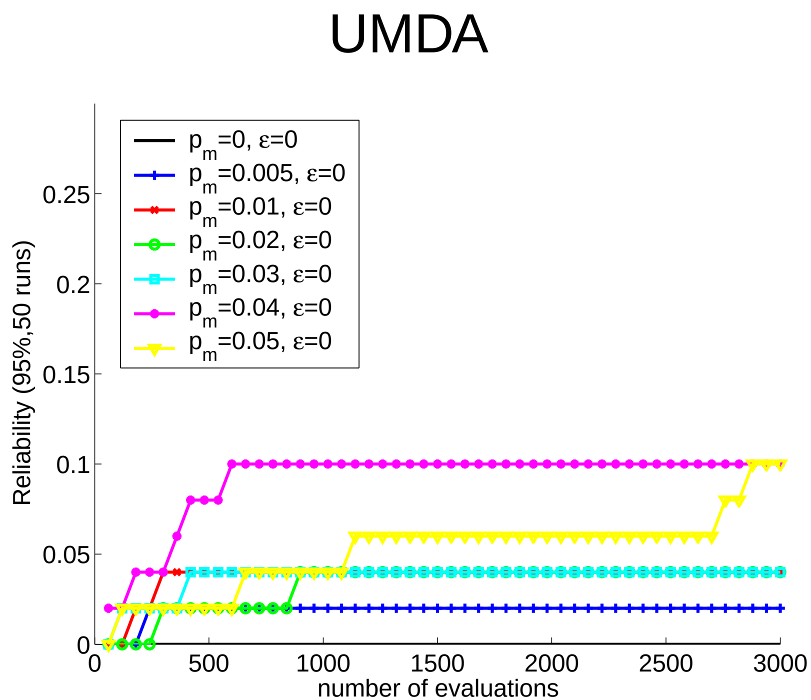
Diversity preservation mechanisms

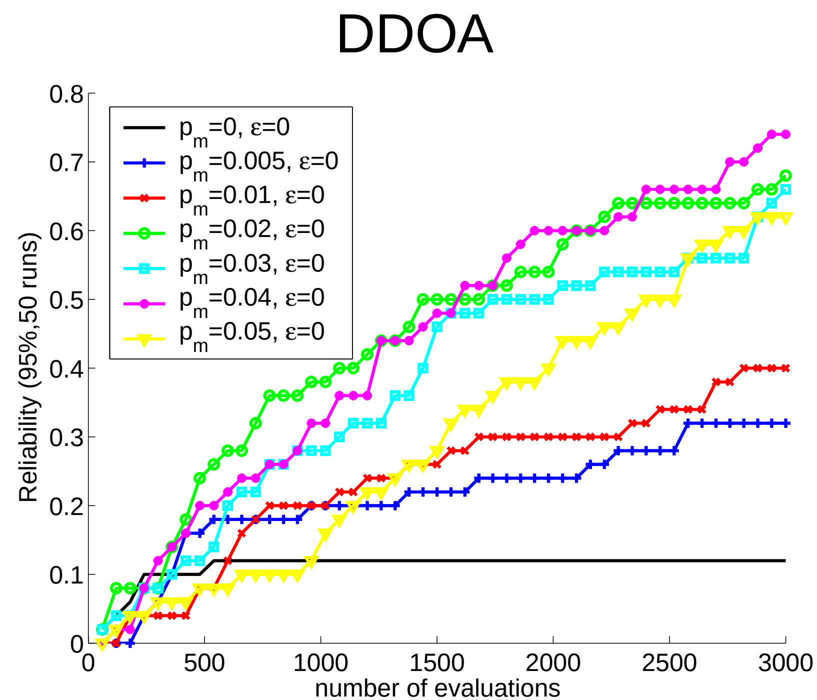
- Theoretical EDA: infinite population λ
- Implementation: finite (small) population ⇨ estimation error on $p(\mathbf{x})$
- Observation: tendency to underestimate p in unexplored regions (loss of variable values = "premature convergence")

⇨ Diversity preservation mechanisms must be implemented to compensate for lost points

- Two mechanisms:
  - mutation: a perturbation is applied with probability p_m to each variable θ_k of each of the λ created points
  - lower bound on marginal probabilities: the probability $p(\theta_k = c_l)$ is not allowed to fall below a threshold ϵ

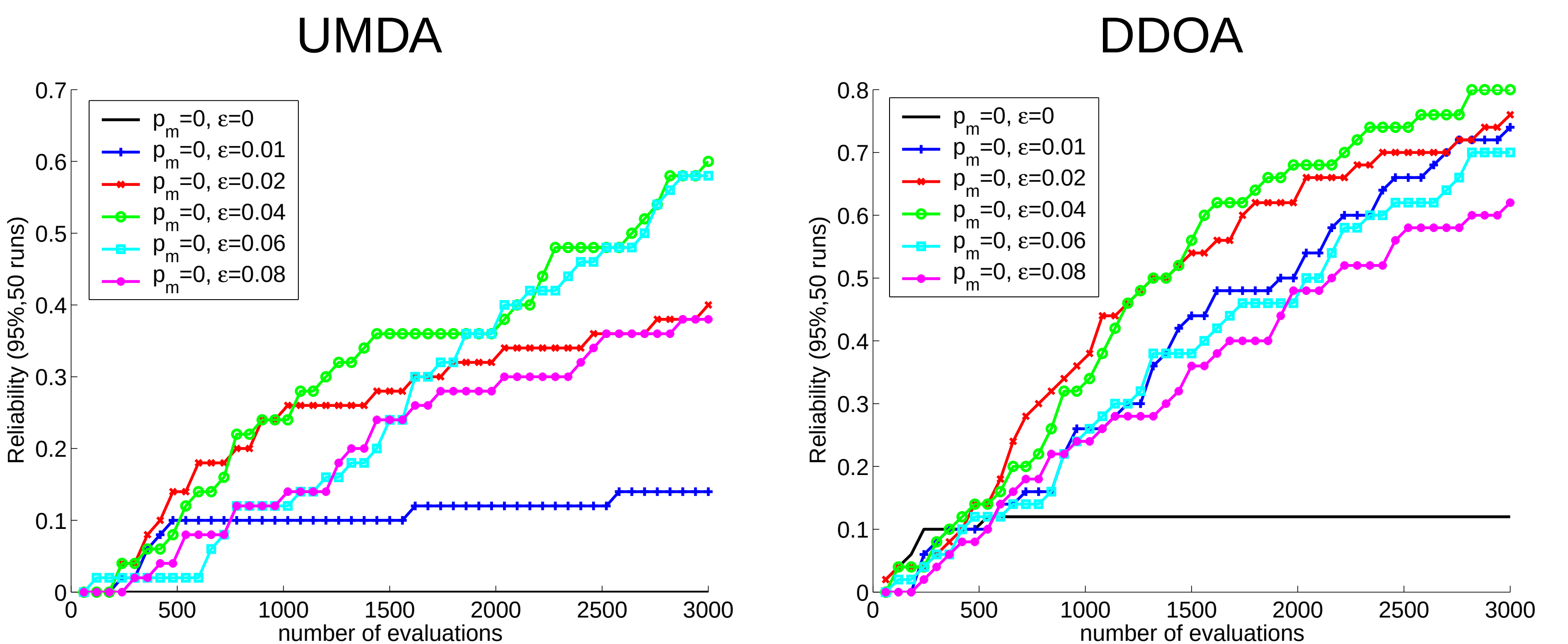
Effect of mutation





- even with a large mutation rate, UMDA does not reliably find the optimum (probability of obtaining a good point by chance very low with $n = 12$)
- mutation greatly improve DDOA's performance: the auxiliary variable scheme filters out poor candidates created by mutation

Effect of the lower bound on marginal distributions



- preventing probabilities to vanish improves UMDA's reliability but it remains inferior to DDOA.
- DDOA greatly benefits from bounding the probabilities. The performance is not sensitive to the value of ϵ
  same explanation as for mutation

Performance with optimized parameters for the strength problem

- Parameter study for GA, UMDA, and DDOA

Performance with optimized parameters for the strength problem

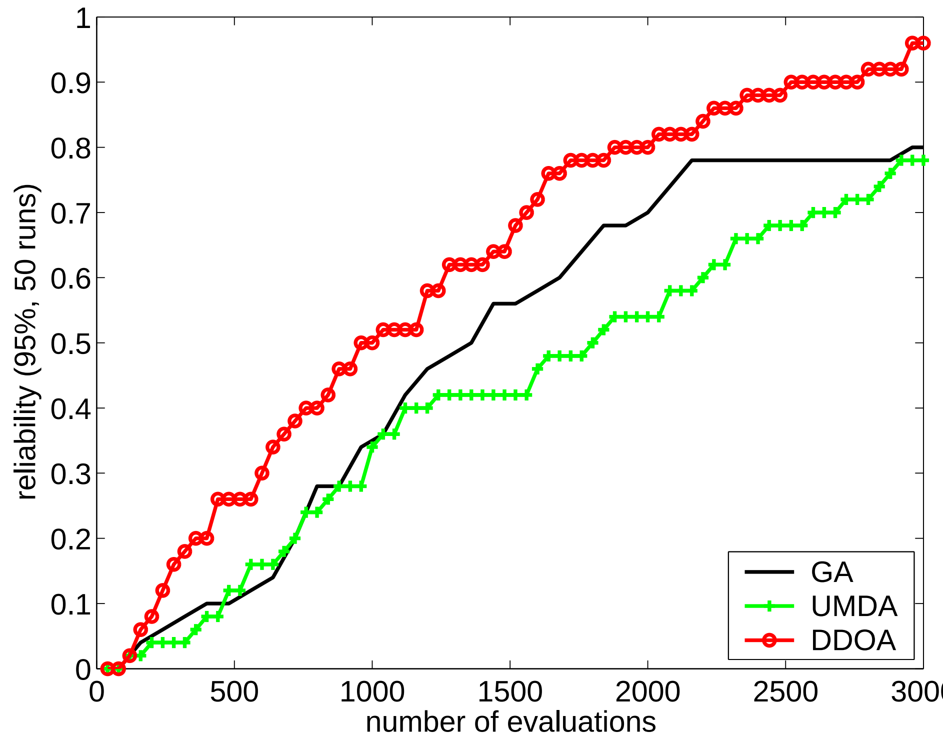
- Parameter study for GA, UMDA, and DDOA
- Let λ, ν, p_m, ϵ vary

Performance with optimized parameters for the strength problem

- Parameter study for GA, UMDA, and DDOA
- Let λ, ν, p_m, ϵ vary
- Best variants:
  GA: $\lambda = 80$, $p_m = 0.02$
  UMDA: $\lambda = 40$, $\epsilon = 0.06$
  DDOA: $\lambda = 40$, $\nu = 200$, $\epsilon = 0.06$

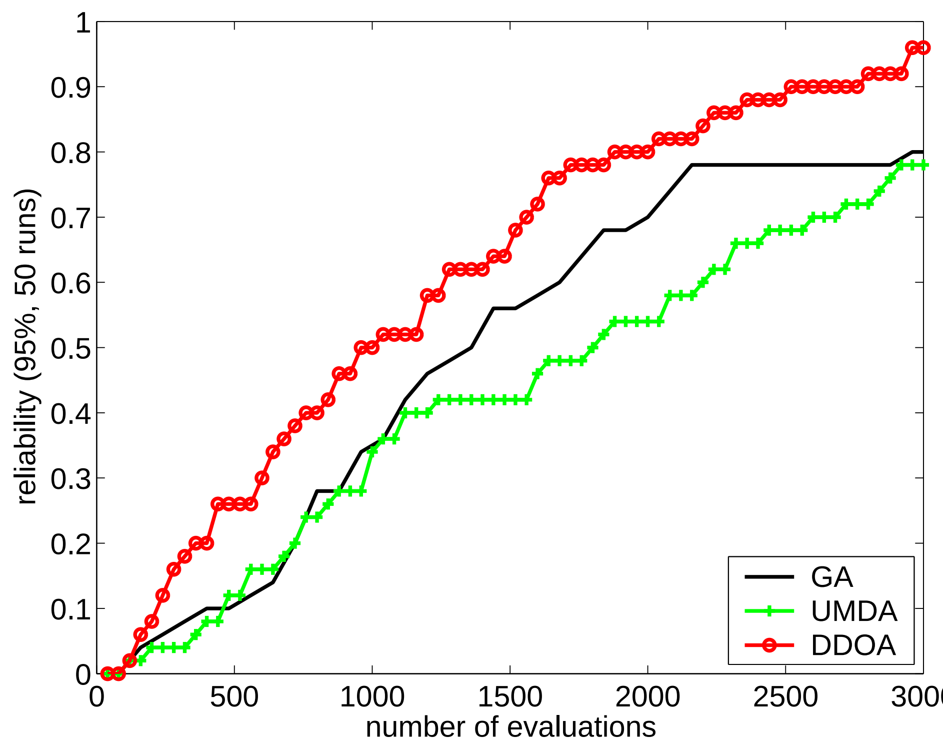
Performance with optimized parameters for the strength problem

- Parameter study for GA, UMDA, and DDOA
- Let λ, ν, p_m, ϵ vary
- Best variants:
  GA: $\lambda = 80$, $p_m = 0.02$
  UMDA: $\lambda = 40$, $\epsilon = 0.06$
  DDOA: $\lambda = 40$, $\nu = 200$, $\epsilon = 0.06$



Performance with optimized parameters for the strength problem

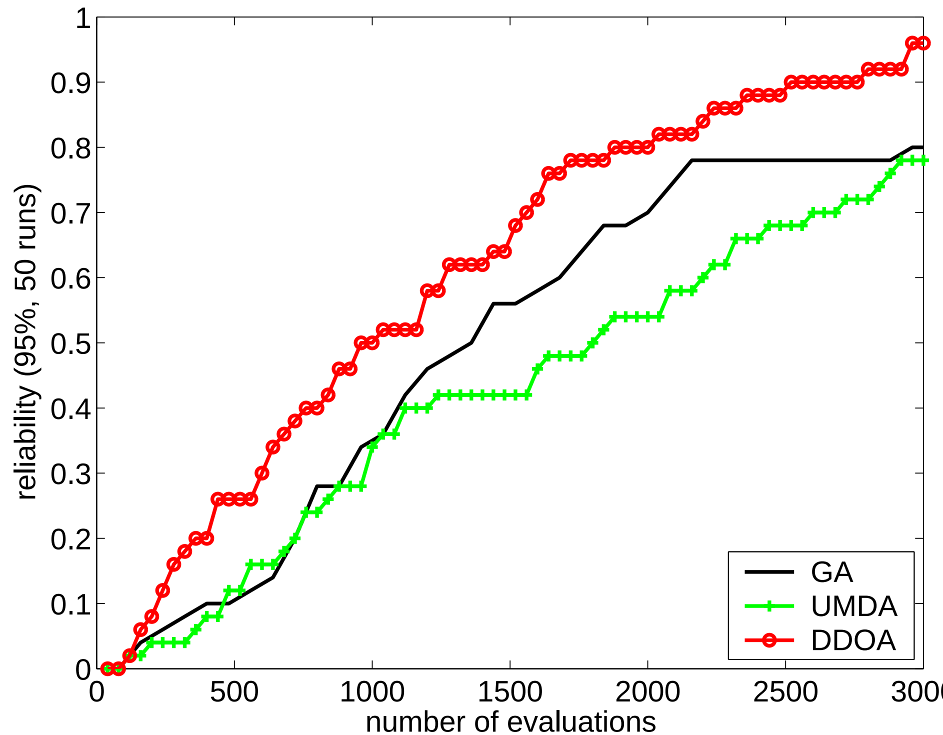
- Parameter study for GA, UMDA, and DDOA
- Let λ, ν, p_m, ϵ vary
- Best variants:
  GA: $\lambda = 80$, $p_m = 0.02$
  UMDA: $\lambda = 40$, $\epsilon = 0.06$
  DDOA: $\lambda = 40$, $\nu = 200$, $\epsilon = 0.06$



➪ Significant improvement over UMDA (variable dependencies)

Performance with optimized parameters for the strength problem

- Parameter study for GA, UMDA, and DDOA
- Let λ, ν, p_m, ϵ vary
- Best variants:
  GA: $\lambda = 80$, $p_m = 0.02$
  UMDA: $\lambda = 40$, $\epsilon = 0.06$
  DDOA: $\lambda = 40$, $\nu = 200$, $\epsilon = 0.06$



- ➩ Significant improvement over UMDA (variable dependencies)
- ➩ DDOA more efficient than GA even without mutation

General Conclusions

Concluding Remarks

- Estimation of Distribution Algorithms are a formalization of evolutionary algorithms: provide a sound statistical framework

Concluding Remarks

- Estimation of Distribution Algorithms are a formalization of evolutionary algorithms: provide a sound statistical framework
- Applicability to laminate optimization was demonstrated: UMDA displayed comparable to higher performance than other stochastic algorithms (SHC and GA)

Concluding Remarks

- Estimation of Distribution Algorithms are a formalization of evolutionary algorithms: provide a sound statistical framework
- Applicability to laminate optimization was demonstrated: UMDA displayed comparable to higher performance than other stochastic algorithms (SHC and GA)
- A strategy for improving the statistical model of promising regions through auxiliary variables was proposed: the Double-Distribution Optimization Algorithm

Concluding Remarks

- Estimation of Distribution Algorithms are a formalization of evolutionary algorithms: provide a sound statistical framework
- Applicability to laminate optimization was demonstrated:
UMDA displayed comparable to higher performance than other stochastic algorithms (SHC and GA)
- A strategy for improving the statistical model of promising regions through auxiliary variables was proposed: the Double-Distribution Optimization Algorithm
- Application to laminate optimization showed the efficiency of the approach for problems with strong variable dependencies
+ greater stability to the value of the algorithlm parameters

Concluding Remarks

- Estimation of Distribution Algorithms are a formalization of evolutionary algorithms: provide a sound statistical framework
- Applicability to laminate optimization was demonstrated: UMDA displayed comparable to higher performance than other stochastic algorithms (SHC and GA)
- A strategy for improving the statistical model of promising regions through auxiliary variables was proposed: the Double-Distribution Optimization Algorithm
- Application to laminate optimization showed the efficiency of the approach for problems with strong variable dependencies + greater stability to the value of the algorithlm parameters
- A study of the diversity was conducted: proposed direct control of the distribution diversity

Future work

- Extension of DDOA to continuous problems (constraints as auxiliary variables): preliminary results are promising

Future work

- Extension of DDOA to continuous problems (constraints as auxiliary variables): preliminary results are promising
- Application to other fields where auxiliary variables are available

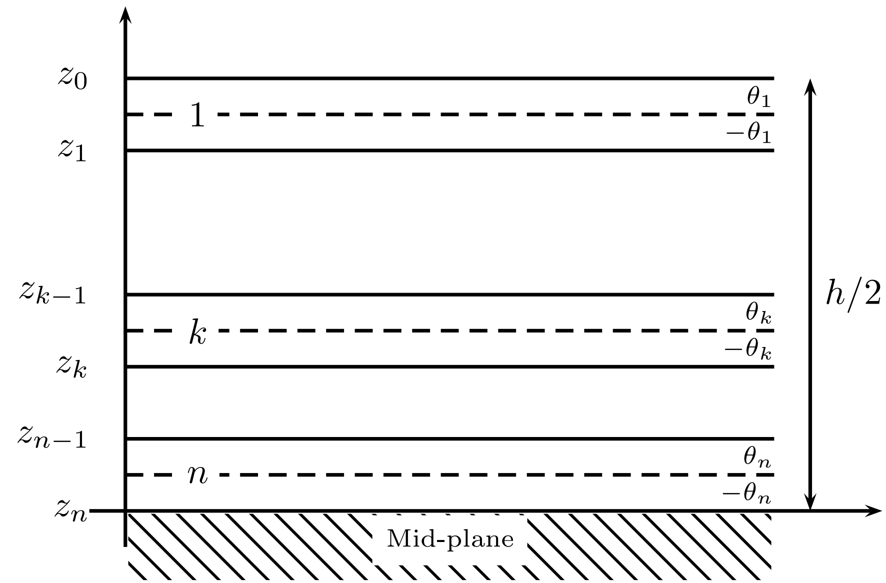
Future work

- Extension of DDOA to continuous problems (constraints as auxiliary variables): preliminary results are promising
- Application to other fields where auxiliary variables are available
- Detection of failure situations: how to check the validity of $p(\mathbf{x})$ and what to do when failure is detected?

Backup Slides

Balanced symmetric laminate

- Particular case: balanced symmetric laminates $[\pm\theta_1, \pm\theta_2, \ldots, \pm\theta_n]_s$



Constraint enforcement through a penalty approach

- Consider the following optimization problem:

$$\begin{array}{ll}\textbf{maximize} & F(\mathbf{x}) \\ \textbf{such that} & g_j(\mathbf{x}) \geq 0,\ j = 1, \ldots, m\end{array}$$

Constraint enforcement through a penalty approach

- Consider the following optimization problem:

$$\begin{array}{ll} \textbf{maximize} & F(\mathbf{x}) \\ \textbf{such that} & g_j(\mathbf{x}) \geq 0,\ j = 1, \ldots, m \end{array}$$

- The penalty approach transforms this constrained problem into an unconstrained problem by decreasing the objective function proportionally to the constraint violation:

$$F_p(\mathbf{x}) = \begin{cases} F(\mathbf{x}) & \text{if } g_j(\mathbf{x}) \geq 0,\ j = 1, \ldots, m \\ F(\mathbf{x}) + p \, \min_{j=1}^{m} \left(g_j(\mathbf{x}) \right) & \text{if } \exists\, k \in \{1, \ldots, m\} \text{ s.t. } g_k(\mathbf{x}) < 0 \end{cases}$$

Kernel density estimate: principle

- $f(V)$ is continuous, low dimensional (2D or 4D in our problems) ➩A kernel density estimation approach is adopted:

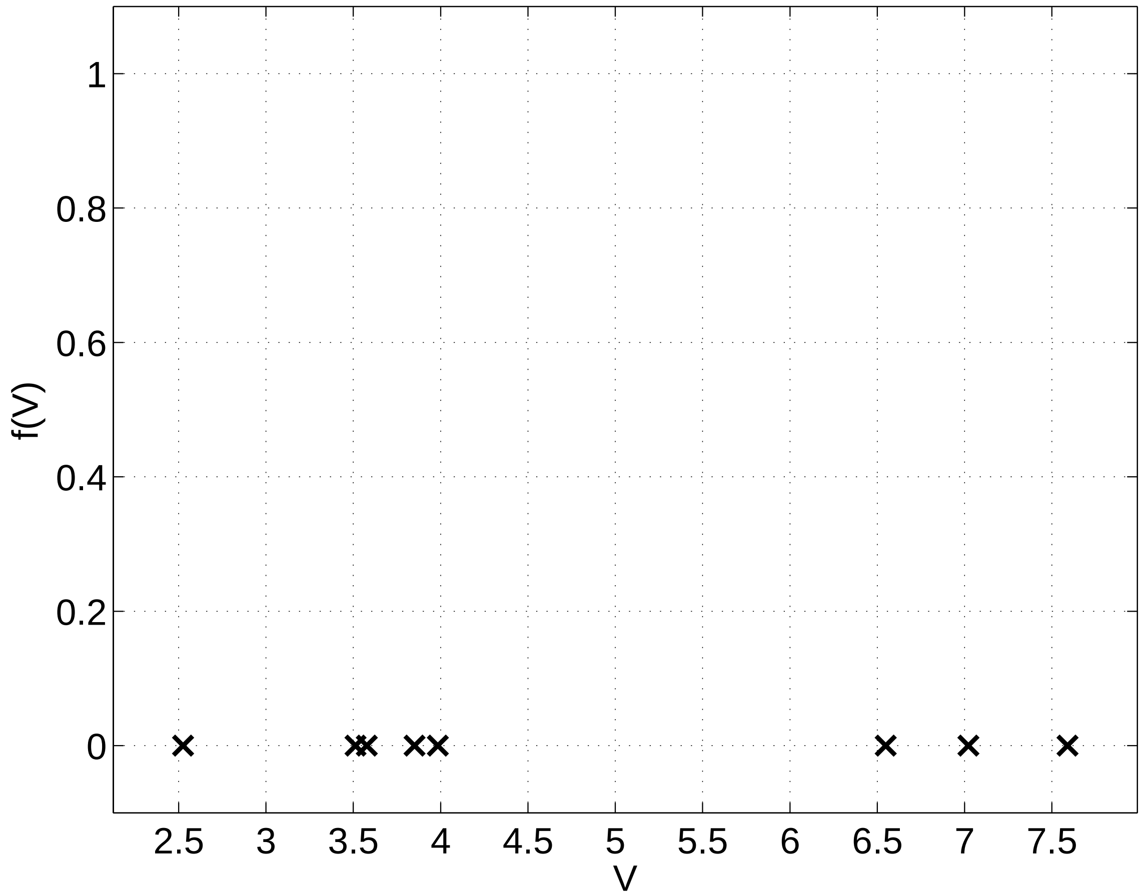
$$f(\mathbf{V}) = \frac{1}{\mu} \sum_{i=1}^{\mu} K\left(\mathbf{V} - \mathbf{V}_i\right)$$

In this work, we used Gaussian kernels:

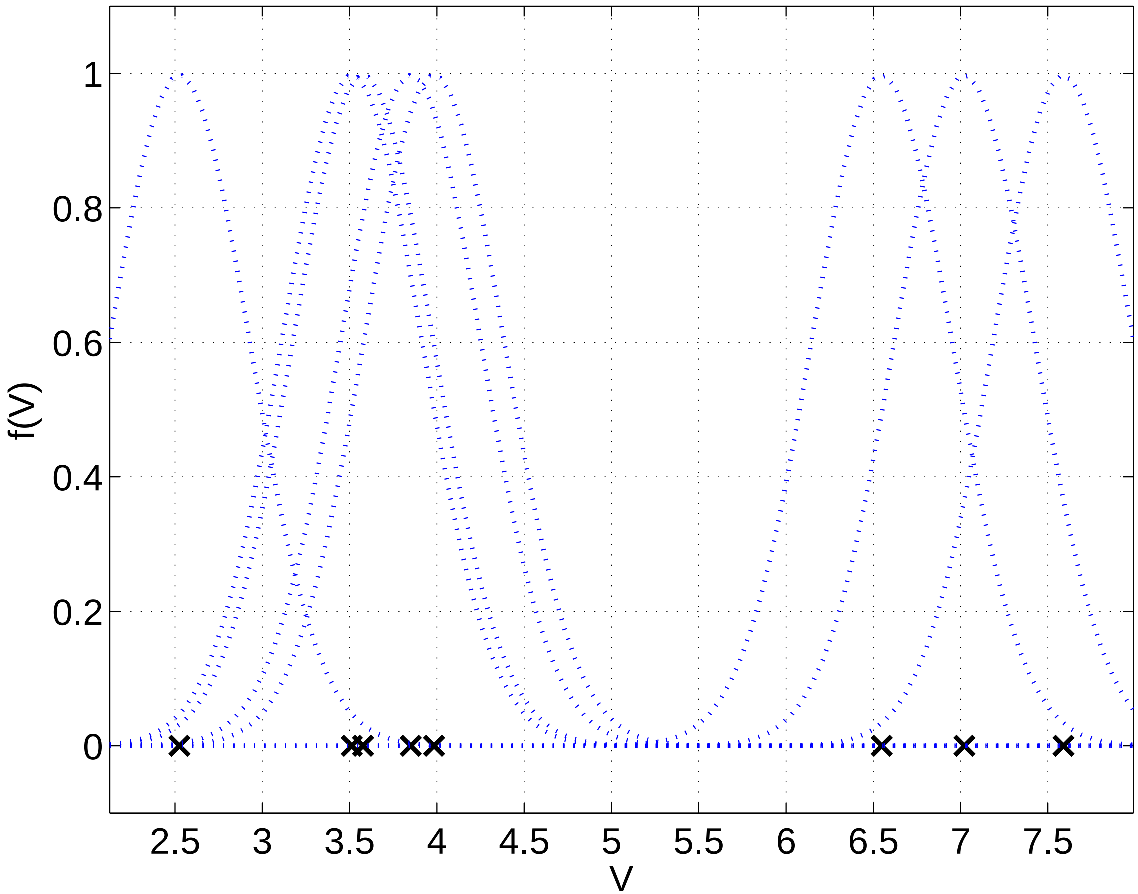
$$K(\mathbf{u}) = \frac{1}{(2\pi)^{d/2}\sigma^d} \exp\left(-\frac{\mathbf{u}^T\mathbf{u}}{\sigma^2}\right)$$

where σ determines the smoothness of the estimate

Kernel density estimate: illustration



Kernel density estimate: illustration



Kernel density estimate: illustration