

1. General Intuition of the Candidate Exchange Algorithm

The MATLAB function `candexch()` implements a classical DOE optimal design algorithm known as the **exchange algorithm**, derived from **Fedorov's** work. Its purpose is to select a subset of points from a list of candidates to optimize a criterion (usually **D-optimality**).

The D-optimal design problem is:

Find the n points from the design (within a potentially very large set) that maximize the determinant of the information matrix $X^T X$.

This ensures: - The most precise estimation of model parameters. - Minimization of the uncertainty ellipsoid volume. - A statistically "compact" design.

2. Why Not Test All Combinations?

The number of possible combinations is astronomical.

For example: - 200 candidate points - selecting 20 points

the number of combinations is:

$$\binom{200}{20} \approx 1.6 \times 10^{27}$$

Clearly impossible to test all.

3. How the Exchange Algorithm Works

The key idea is: 👉 **start with an initial design** (random or heuristic), 👉 **then iteratively improve it** by small changes that increase the D-optimal criterion.

Steps

1. **Define a set of candidates** – a grid, Latin hypercube, etc.
2. **Initialize a design** – a random selection of points from the candidates.
3. **Optimization loop**
4. For each point in the design:
 - Test replacing it with an unselected candidate point.

- Calculate the D-optimality gain:

$$\Delta = \log \det(X^T X)_{\text{new}} - \log \det(X^T X)_{\text{old}}$$

- If gain > 0 → **perform the exchange**

5. **Repeat until convergence** – stop when no exchange improves the design.

4. Why It Is Efficient

Because it reduces a huge combinatorial problem:

$$\binom{N}{n}$$

to a **local optimization problem**: - Remove a point - Add another point - Update $X^T X$ efficiently without full recalculation

The algorithm often converges to a local optimum near the global optimum, especially when restarted from multiple initial designs.

5. Key Teaching Points for DOE Students

A. Why Optimal Designs

- Useful when experimental space is constrained.
- Non-orthogonal models.
- Non-factorial designs (complex response surfaces).
- Maximizing Fisher's criterion: $|X^T X|$.

B. Candidate-Based Design

- Students should understand: design is **selected from a catalog of candidates**, not generated continuously.
- Very concrete and intuitive.

C. Intuition of the D-Optimal Criterion

- Minimizes the uncertainty volume.
- Reduces variance of model coefficients.
- Geometric interpretation: points well dispersed.

D. Principle of the Exchange Algorithm

1. **Initialization**: random design (acceptable but not optimal).
2. **Replacement iterations**: exchange a design point with a candidate point only if it improves D-optimality.
3. **Convergence**: stop when no improvement is possible.

Animations or simple 2D examples help a lot.

E. Advanced Concepts (Optional)

- Fast determinant updates (Sherman–Morrison, etc.)
 - Risk of local optima
 - Multiple restarts for better results
 - Comparisons with other algorithms: Coordinate Exchange, Fedorov Complete Exchange, KL-Exchange.
-

6. Summary Statement

The candidate exchange algorithm is an iterative method that starts from an initial design and improves it progressively by exchanging design points with candidate points to maximize an optimality criterion, typically the determinant of $X^T X$. It is efficient because it avoids exploring all possible combinations.