Multi-Dimensional Bisection Method (MDBM) User's Guide

Dr. Daniel Bachrathy
Assistant Professor
Budapest University of Technology and Economics

February 2026

Contents

1 Introduction

The Multi-Dimensional Bisection Method (MDBM) is an efficient and robust root-finding algorithm for determining high-dimensional submanifolds of the roots of implicit non-linear equation systems. It is particularly powerful in cases where the number of unknowns surpasses the number of equations ($k \leq l$):

$$f_i(x_j) = 0, \quad i = 1 \ldots k, \quad j = 1 \ldots l \tag{1}$$

MDBM iteratively halves the parameter space and evaluates only those grid points where a root is likely to exist, significantly reducing the number of required function evaluations compared to brute-force methods.

2 Quick How-To-Use

To use MDBM, you need to define the parameter space, the vectorized function, and the number of iterations.

```
% 1. Define axes
ax(1).val = linspace(-3, 3, 10);
ax(2).val = linspace(-3, 3, 10);

% 2. Define vectorized function
foo = @(x) x(1,:).^2 + x(2,:).^2 - 4; % Unit circle

% 3. Run MDBM
Niteration = 4;
sol = mdbm(ax, foo, Niteration);

% 4. Plot
plot_mdbm(sol);
```

3 Detailed Options (mdbmset)

The behavior of the solver is controlled by the `mdbmset` structure. Each field is critical for fine-tuning the solver's performance and robustness.

Field	Related Demo	Description
`isconstrained`	Constrained Problems	If true, the last equation in the output is treated as a domain boundary. Solutions are only kept where this value is positive.
`interporder`	Interpolation Order	Order of root interpolation: 0 (raw grid points), 1 (linear), 2 (quadratic).
`bracketingdistance`	Case: Turning	Distance (in grid units) to look for sign changes. Helps recover branches in dense regions.
`checkneighbour`	Neighbor Check	Max consecutive neighbor check steps. Set to `Inf` for max robustness.
`refinetype`	Local Refinement	Refinement strategy: `'all'`, `'pos'` (range), `'object'` (near roots).

`zerotreshold`	Feature: Degenerate	Numerical value below which a function evaluation is considered exactly zero.
`connections`	Simplex Connections	If true, calculates the Delaunay connectivity needed for plotting surfaces and curves.
`timelimit`	Case: Mathieu	Max allowed time (seconds) for a single refinement step.
`funcallimit`	-	Max function evaluations allowed to prevent memory overflow.

4 Software Components (src)

The following files in the `src/` directory form the core of the MDBM ecosystem:

File	Description
`mdbm.p`	**Main Entry Point**. Orchestrates iterative bisection, calling refinement and interpolation.
`mdbmset.m`	**Configuration**. Utility to create the options structure with defaults.
`plot_mdbm.m`	**Visualization**. Intelligent plotting for 1D, 2D, and 3D submanifolds.
`extend_axis.m`	**Domain Extension**. Adds new parameter ranges to an existing solution.
`refine.p`	**Grid Management**. Implements the various refinement strategies.
`checkneighbour.p`	**Robustness Engine**. Scans adjacent grid cells to find missing branches.
`interpolating_cubes.p`	**Root Finder**. Performs root interpolation within a grid cell.
`DTconnect.p`	**Meshing**. Standard Delaunay triangulation for result visualization.
`interpplot.p`	**Smoothing**. Generates high-quality interpolation for plotting.
`plotthecomputedpoints.m`	**Debug Viz**. Visualizes all grid points evaluated by the solver.

5 History

The core idea of MDBM occurred during my MSc thesis at the University of Bristol (Erasmus program) in 2006. The motivation was the solution of the characteristic equation of time-delay systems (DDEs), which involve 2+1 parameters and 2 equations (real and imaginary parts). At that time, no direct method existed to handle these problems automatically in a robust way, especially for detecting closed manifolds of non-differentiable functions.

6 Disclaimer

The core solver files are provided as `.p` files to protect the development work spanning over a decade. All files necessary for customization and special needs remain as `.m` files.

Julia Version: A newer version is available in Julia. It includes sub-cube interpolation and a beta version of the highly anticipated error-based non-uniform refinement.

7 Known Alternatives

- **Mathematica (ContourPlot/Isosurface)**: Uses iterative grid refinement for visualization. Robust in 2D and 3D but usually limited to a single implicit equation.
- **Sequential Projections**: Solving equations sequentially and triangulating intermediate results. Complexity scales poorly with dimension and number of equations.
- **Interval Arithmetic**: Provides mathematically guaranteed results. Notable work includes **David P. Sanders** and the Julia ecosystem (`IntervalRootFinding.jl`).

8 Folder Descriptions and Gallery

8.1 Features (features/)

8.1.1 Simplex Connections

`connection_of_points_to_simplex.m` shows how points are connected (Figure 1).

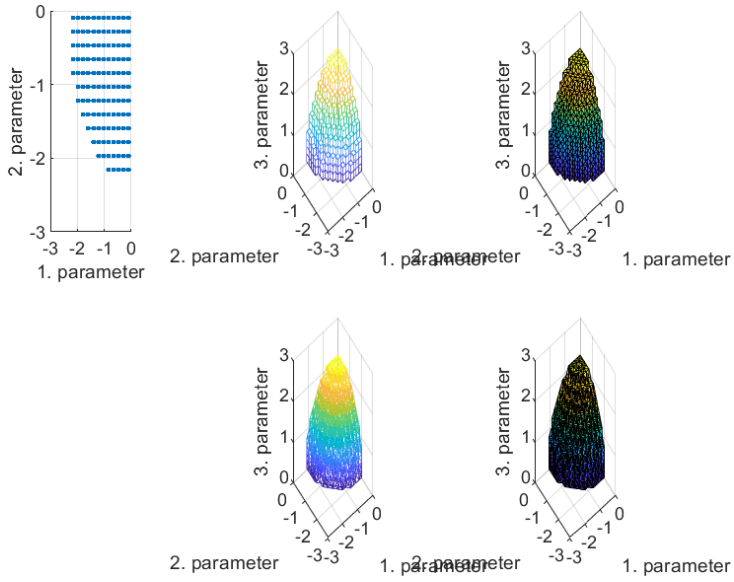


Figure 1: Connectivity of points in the parameter space.

8.1.2 Constrained Problems

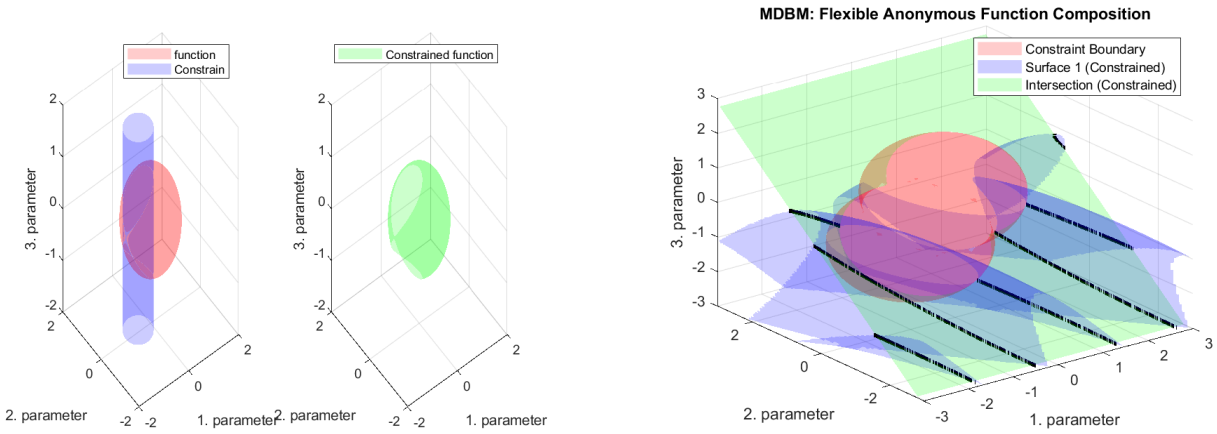
Handling domain boundaries (Figure 2a and 2b).

8.1.3 Interpolation Order

Comparison of 0th, 1st, and 2nd order (Figure 3).

8.1.4 Local Refinement

Zooming into regions (Figure 4).



(a) Simple constraint.

(b) Complex combined constraints.

Figure 2: Constraint handling in MDBM.

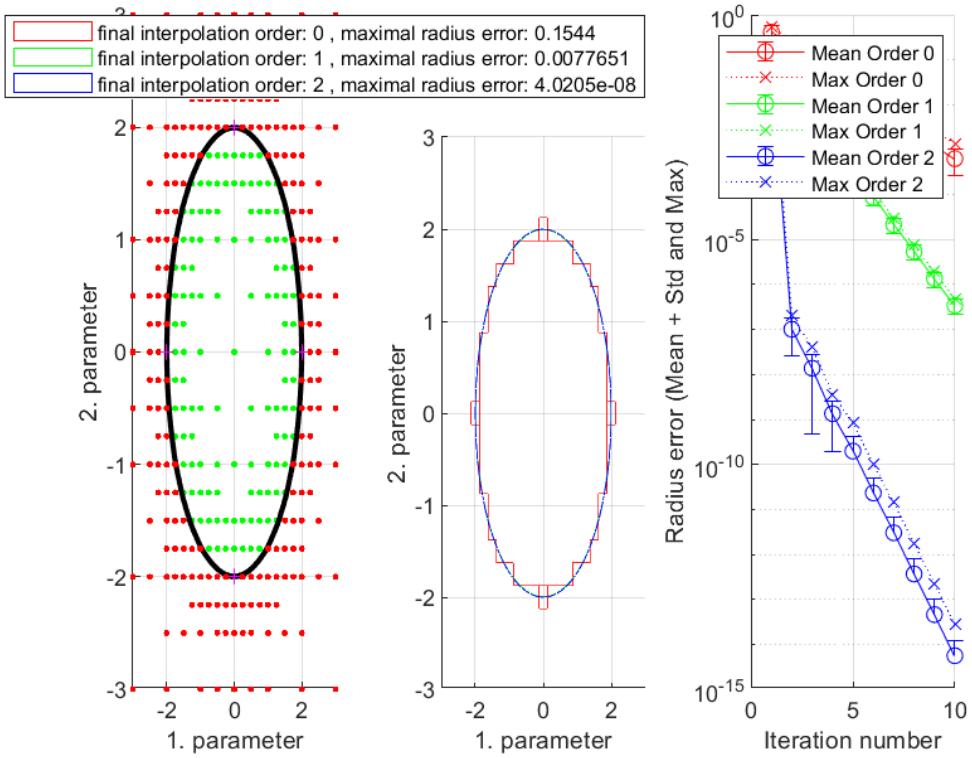


Figure 3: Effect of interpolation order on accuracy and convergence.

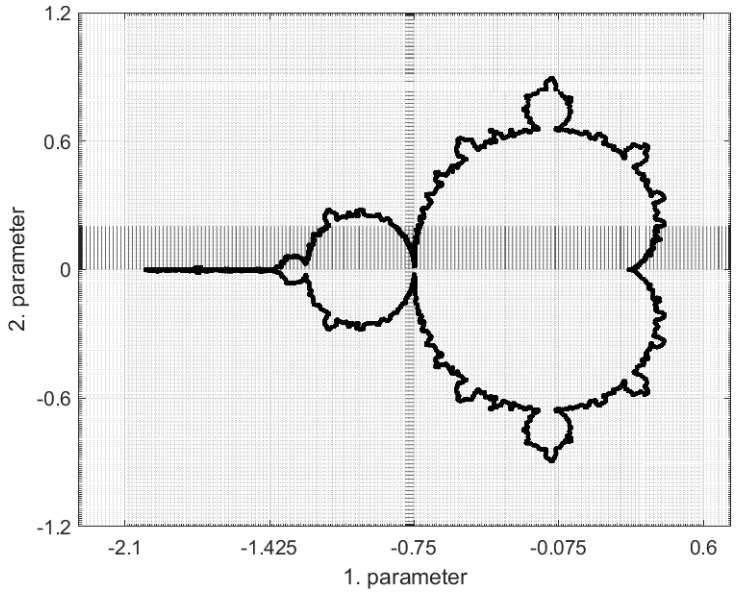


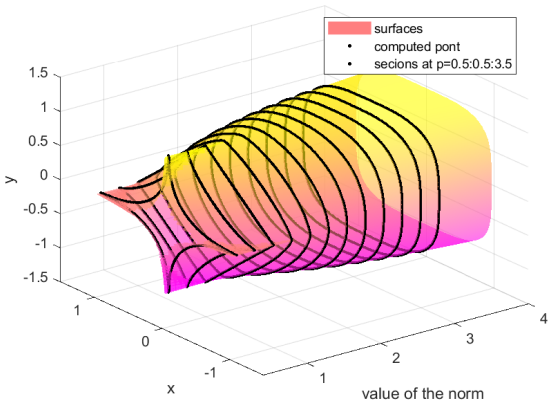
Figure 4: Local refinement of a non-smooth boundary.

8.1.5 Neighbor Check

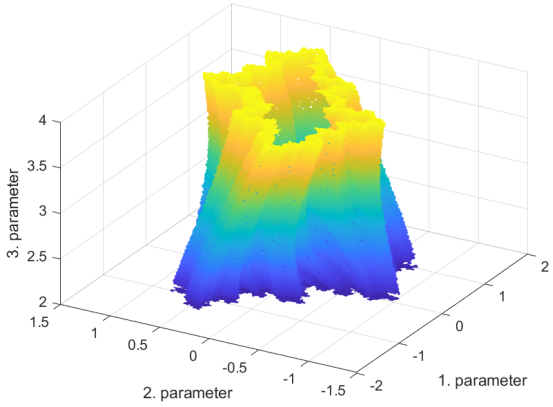
Recovering missing branches (Figure 5).

Figure 5: Neighbor check ensuring solution continuity.

8.2 Examples (examples/)



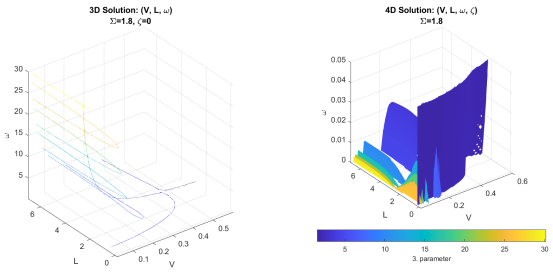
(a) Unit circles (L^p norms).



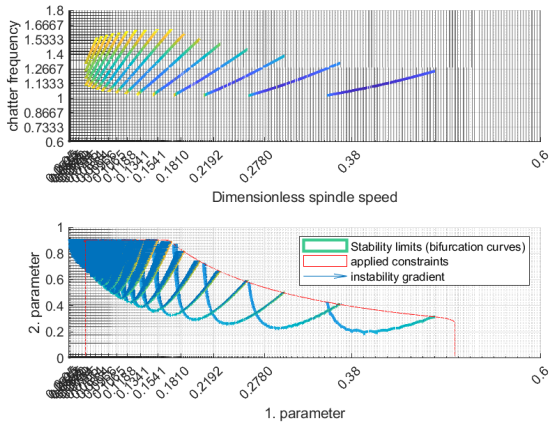
(b) 3D Julia set.

Figure 6: Geometric and fractal examples.

8.3 Case Studies (case_studies/)



(a) Shimmy vibration stability.



(b) Turning stability chart.

Figure 7: Engineering case studies.

9 Citing

If you use MDBM in your research, please cite: *Bachrathy, Daniel, and Gabor Stepan. "Bisection Method in Higher Dimensions and the Efficiency Number." Periodica Polytechnica Mechanical Engineering, 2012.*