MDDS (Mixed Directional Difference-Summation) package

Authors: G. Casciola, E. Franchini and L. Romani

This is a supplement to the paper

[1] G. Casciola, E. Franchini, L. Romani, *The mixed directional difference-summation algorithm for generating the Bézier net of a trivariate four-direction Box-spline*, Numerical Algorithms 43 (2006), pp. 75-98

The reader should consult that paper for more information.

Installation guide for MDDS package

MDDS package is available from Netlib at *www.netlib.org/numeralgo/na23*. This package can be used as a stand-alone package. A user needs only standard Matlab, version 7.X. No additional toolboxes are needed. No particular problems have been detected with Matlab version 6.5.

Step-by-step installation (Unix environment):

1. Download the gzipped tar archive

   na23.tgz

   in the directory you use for user's Matlab packages and toolboxes. Matlab provides a directory usually named

   usr/local/matlab/toolbox/

   but only an administrator user can write on it. Also the whole contents of this directory is erased when a new Matlab version is installed. Hence, our suggestion is to create a user's directory named (for instance)

   $HOME/matlab/toolbox

for installing additional packages.

2. Execute the commands

   cd $HOME/matlab/toolbox
   tar xvfz na23.tgz

   which will create the directory

   MDDS

   containing the package.

3. Remove (if you want) the archive:

   rm na23.tgz

4. Start the Matlab program.

5. Add the path to the package directory, e.g., by means of the Matlab menu File → Set Path ...

6. Make sure the path is placed at the bottom of Matlab's search path, e.g., by clicking the button "Move to Bottom".

7. Save the changed Matlab search path.

Alternatively, the directory MDDS can be saved everywhere and the programs can be run in the same directory.

Listing of files in MDDS package:

	manual.pdf	⟶	description of installation and utilization of the MDDS package
	Contents.m	⟶	description of the files contained in the MDDS package
⋆	Demonstration		
	demo_mdds2d.m	⟶	demo of Bézier net computation, exact evaluation and visualization of bivariate three-direction Box-splines
	demo_mdds3d.m	⟶	demo of Bézier net computation, exact evaluation and visualization of trivariate four-direction Box-splines

⋆ B-net computation

bnet_mdds.m	$\longrightarrow$	computation of the Bézier net coefficients of an arbitrary bivariate three-direction/ trivariate four-direction Box-spline
step1_mdds2d.m	$\longrightarrow$	computation of differences-summations along direction $e_1 = [1\ 0]'$
step2_mdds2d.m	$\longrightarrow$	computation of differences-summations along direction $e_2 = [0\ 1]'$
step3_mdds2d.m	$\longrightarrow$	computation of differences-summations along direction $e_{12} = [1\ 1]'$
step1_mdds3d.m	$\longrightarrow$	computation of differences-summations along direction $e_1 = [1\ 0\ 0]'$
step2_mdds3d.m	$\longrightarrow$	computation of differences-summations along direction $e_3 = [0\ 0\ 1]'$
step3_mdds3d.m	$\longrightarrow$	computation of differences-summations along direction $e_2 = [0\ 1\ 0]'$
step4_mdds3d.m	$\longrightarrow$	computation of differences-summations along direction $e_{123} = [1\ 1\ 1]'$
dupl_mdds.m	$\longrightarrow$	duplication of some coefficients of the main direction
integ_mdds.m	$\longrightarrow$	computation of antidifferences (summations) on the coefficients

⋆ Box-splines evaluation

pointeval_mdds2d.m	$\longrightarrow$	exact evaluation of bivariate three-direction Box-splines in an arbitrary set of 2D points
findsquare_mdds2d.m	$\longrightarrow$	determination of the unit square containing the 2D evaluation point chosen in the Box-spline domain
findtria_mdds2d.m	$\longrightarrow$	determination of the triangle type, down (“1”) or up (“2”), (see [1]) containing the 2D evaluation point

pointeval_mdds3d.m	$\longrightarrow$	exact evaluation of trivariate four-direction Box-splines in an arbitrary set of 3D points
findcube_mdds3d.m	$\longrightarrow$	determination of the unit cube containing the 3D evaluation point chosen in the Box-spline domain
findtetra_mdds3d.m	$\longrightarrow$	determination of the tetrahedron type, “1”,“2”,“3”,“4”,“5” or “6” (see [1]) containing the 3D evaluation point

⋆ Box-splines visualization

visual_mdds2d.m	$\longrightarrow$	visualization of an arbitrary bivariate three-direction Box-spline either through its surface graph or its B-net representation
computebez_mdds2d.m	$\longrightarrow$	evaluation of Bernstein-Bézier polynomials on a regular triangular grid
visual_mdds3d.m	$\longrightarrow$	visualization of an arbitrary trivariate four-direction Box-spline M_D either through the s-set ($s \in Imm(M_D)$) extraction of the Box-spline volume or through the contour lines of the regions obtained by intersecting the Box-spline volume with three families of planes respectively parallel to yz, xz, xy
computebez_mdds3d.m	$\longrightarrow$	evaluation of Bernstein-Bézier polynomials on a regular tetrahedral grid

How to use MDDS package

The user interface is provided by the following five main functions.

1. The function `bnet_mdds()` that allows to compute the Bézier net coefficients of either bivariate three-direction Box-splines or trivariate four-direction Box-splines of any degree.

   Algorithm: Starting from the Bézier net of the degree-1 Box-spline, the Bézier net of the required bivariate/trivariate Box-spline is generated by successively computing differences-summations of the Bézier coefficients along the three/four directions set. These procedures are implemented respectively in {`step1_mdds2d`, `step2_mdds2d`, `step3_mdds2d`}, {`step1_mdds3d`, `step2_mdds3d`, `step3_mdds3d`, `step4_mdds3d`}.

2. The function `pointeval_mdds2d()` that allows to exact evaluate any bivariate three-direction Box-spline in an arbitrary set of 2D points.

   Algorithm: For each evaluation point we determine the unit square of the Box-spline domain in which it is contained and we identify the triangle type, down ("1") or up ("2") (see [1]) containing the point. Then, after having computed the Bézier coefficients of the identified triangular patch, we exploit the de Casteljau algorithm to work out the Box-spline value in correspondence to that sample point.

3. The function `pointeval_mdds3d()` that allows to exact evaluate any trivariate four-direction Box-spline in an arbitrary set of 3D points.

   Algorithm: For each evaluation point we determine the unit cube of the Box-spline domain in which it is contained and we identify the tetrahedron type, ("1","2","3","4","5" or "6") (see [1]) containing the point. Then, after having computed the Bézier coefficients of the identified tetrahedral volume, we exploit the trivariate de Casteljau algorithm to work out the Box-spline value in correspondence to that sample point.

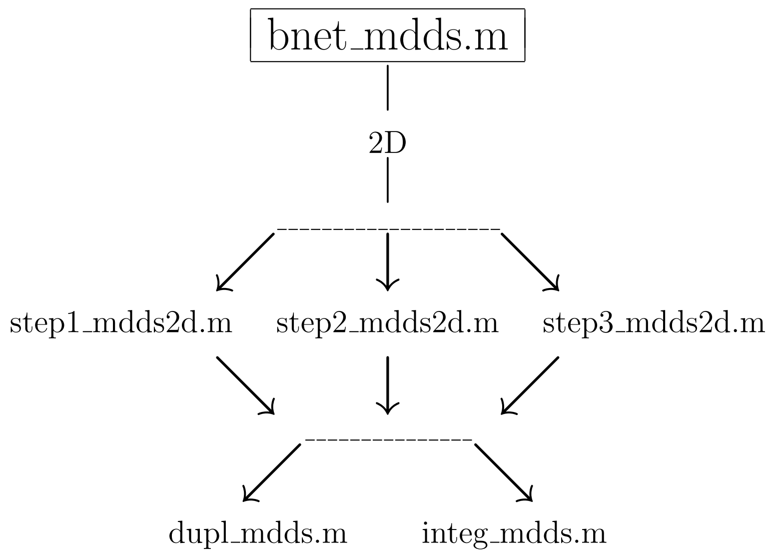
4. The function `visual_mdds2d()` that allows to visualize any bivariate three-direction Box-spline either through its surface graph or its B-net representation.

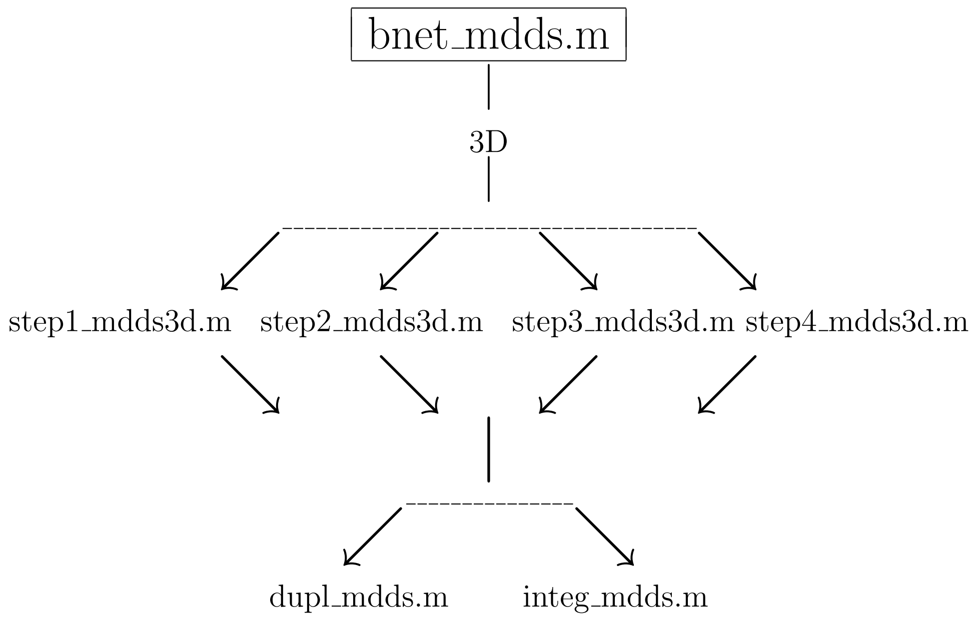
   Algorithm: First we compute the B-net coefficients of the bivariate three-direction Box-spline associated with the multiplicity vector logged in by the user. Then we evaluate the bivariate Bernstein-Bézier polynomials on a regular triangular grid and compute the values of the Box-spline surface over each one of the two triangles obtained by considering the tessellation of the unit square proposed in Fig.2 right (see [1]). Successively we exploit the Matlab functions `trisurf()` and `trimesh()` to visualize respectively the Box-spline surface and its Bézier net.

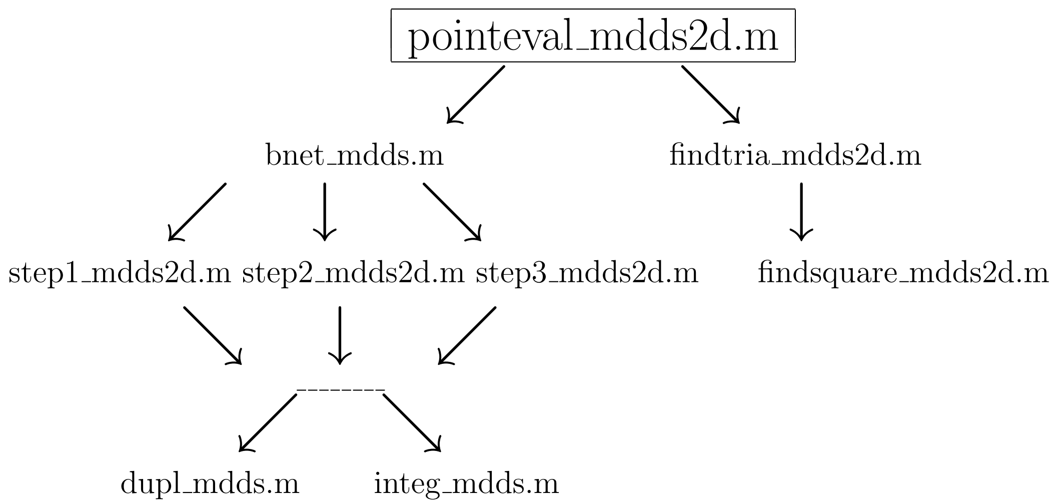
5. The function `visual_mdds3d()` that allows to visualize any trivariate four-direction Box-spline M_D either through the s-set ($s \in Imm(M_D)$) extraction of the Box-spline volume or through the contour lines of the regions obtained by intersecting the Box-spline volume with three families of planes respectively parallel to yz, xz, xy.

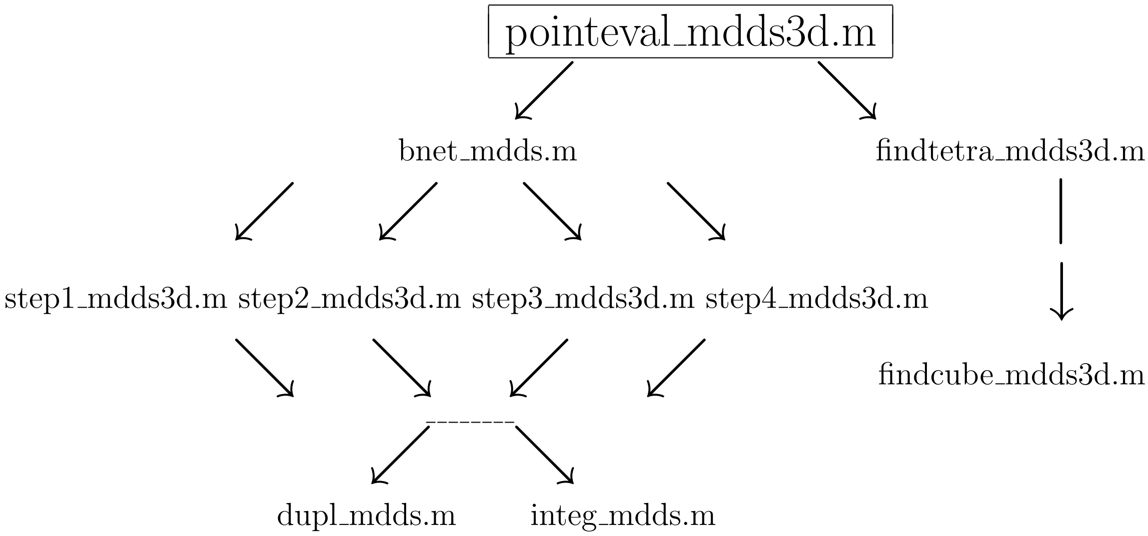
Algorithm: First we compute the B-net coefficients of the trivariate four-direction Box-spline associated with the multiplicity vector logged in by the user. Then we evaluate the trivariate Bernstein-Bézier polynomials on a regular tetrahedral grid and compute the values of the Box-spline volume over each one of the six tetrahedra obtained by considering the tessellation of the unit cube proposed in Fig.3 (see [1]). Successively we exploit the Matlab functions `isosurface()` and `contourslice()` to visualize respectively isosurfaces of the Box-spline volume and contour lines of the regions obtained by intersecting the Box-spline volume with three families of planes respectively parallel to yz, xz, xy.

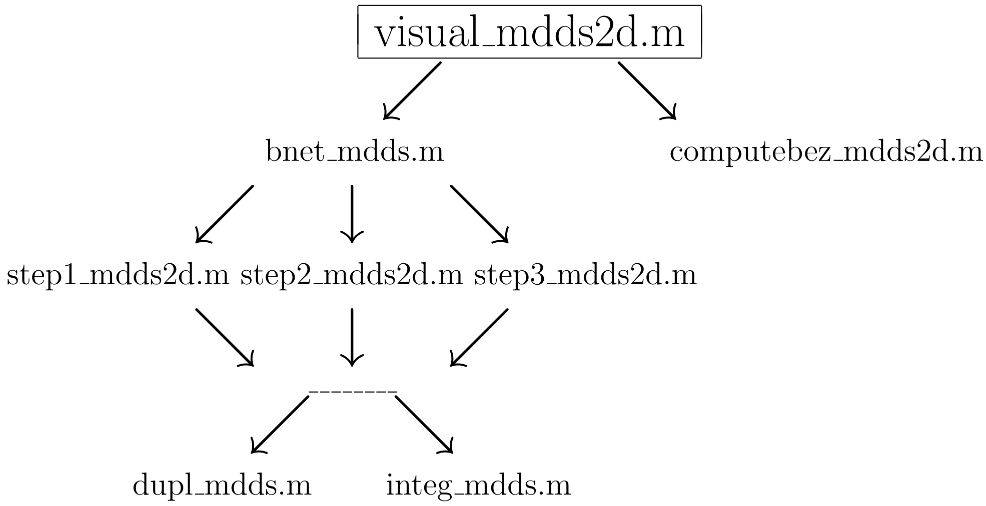
Hierarchy of files in the five main functions:

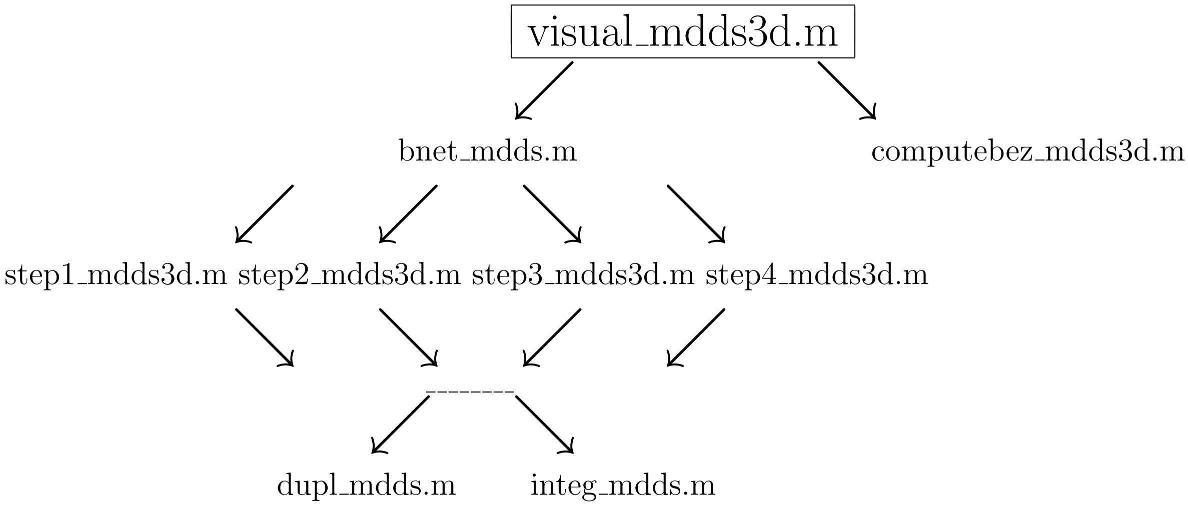



bnet_mdds.m
3D
step1_mdds3d.m
step2_mdds3d.m
step3_mdds3d.m
step4_mdds3d.m
dupl_mdds.m
integ_mdds.m


pointeval_mdds2d.m
bnet_mdds.m
findtria_mdds2d.m
step1_mdds2d.m
step2_mdds2d.m
step3_mdds2d.m
findsquare_mdds2d.m
dupl_mdds.m
integ_mdds.m


pointeval_mdds3d.m
bnet_mdds.m
findtetra_mdds3d.m
step1_mdds3d.m
step2_mdds3d.m
step3_mdds3d.m
step4_mdds3d.m
findcube_mdds3d.m
dupl_mdds.m
integ_mdds.m


visual_mdds2d.m
bnet_mdds.m
computebez_mdds2d.m
step1_mdds2d.m
step2_mdds2d.m
step3_mdds2d.m
dupl_mdds.m
integ_mdds.m



Demonstration files:

In order to give you a better understanding of how to use the MDDS package, we will provide two demonstration files also. These files will show you how to

- generate the Bézier coefficients matrix;
- exact evaluate a bivariate three-direction/trivariate four-direction Box-spline in an arbitrary point of the domain;
- visualize the Box-spline.

In the first M-file `demo_mdds3d.m` three examples of four-direction Box-splines on $\mathbb{R}^3$ are examined: the cubic Box-spline M^3_{2112}, the quintic Box-spline M^5_{2222} and the sextic Box-spline M^6_{3222}. Their Bézier coefficients matrix and their evaluation in an arbitrary point of the domain are provided. Furthermore a visualization through the s-set extraction ($s = 0$, $s = 0.01$ for the Box-spline volume M^3_{2112}; $s = 0$, $s = 0.005$ for M^5_{2222}; $s = 0$, $s = 0.001$ for M^6_{3222}) and the three families of contour lines is available.
In the second M-file `demo_mdds2d.m` three examples of three-direction Box-splines on $\mathbb{R}^2$ are examined: the quadratic Box-spline M^2_{211}, the quartic Box-spline M^4_{222} and the quintic Box-spline M^5_{322}. Their Bézier coefficients matrix and their evaluation in an arbitrary point of the domain are provided. Furthermore the visualization of the Box-spline surface and its B-net is available. For a complete explanation read the paper [1] that accompanies this package.

Remark:

Mail any questions, problems or suggestions to the authors at:
casciola@dm.unibo.it
elena.franchini@inwind.it
romani@dm.unibo.it

==

Prof. G. Casciola
Dept. of Mathematics - University of Bologna, Italy
casciola@dm.unibo.it

Dr. E. Franchini
Dept. of Pure and Applied Mathematics - University of Padova, Italy
elena.franchini@inwind.it

Dr. L. Romani
Dept. of Mathematics - University of Bologna, Italy
romani@dm.unibo.it

==