```
% AmirHosein Sadeghimanesh
% 2022 July
%
% This script contains computations for finding the PSS representation
% of degree 2 of the multistationarity region of the example 4.7
% of the paper, which is depicted in Figure 8b.
%
% This file uses the sampling representation found by the script
% "Matlab_S_system_example_sampling.m".
%
% PART 1
%
% Computing the target function for the optimization problem of PART 2.
%
n = 3;
syms x [1, n]
B = [3.0, 7.0; 1.0, 5.0; 10.0, 14.0];
d = 2;
[p, c] = multPoly(n, x, d);
f = intOverB(p, n, x, B);
%
% PART 2
%
% Computing the coefficients of the polynomial p of the PSS
% representation using optimization by YALMIP and SEDUMI having only 1 SOS
% constraints, the rest of constraints are linear inequalities.
%
K = [6.653503,3.529437,10.390162;
4.113993,3.187526,13.830027;
6.859554,1.630452,13.882371;
3.567545,2.687045,13.662942;
3.142847,4.396517,13.735973;
4.568908,3.621912,10.684747;
5.779314,2.268398,13.800888;
6.062067,4.180800,10.747490;
4.959058,2.782345,12.585252;
5.837459,4.018747,11.104100;
5.718811,3.620392,10.650447;
3.475991,2.993456,13.838976;
4.361543,3.341071,10.895248;
6.005068,2.020380,12.023828;
4.030033,4.362869,11.017129;
6.257139,1.974100,13.717054;
5.464179,2.893155,11.406638;
6.014916,2.521783,12.271287;
5.275295,2.877563,10.047608;
4.348491,1.648729,13.177138;
5.407928,2.051885,12.616316;
3.609512,4.303268,12.153370;
6.984539,1.312702,11.770713;
3.544274,4.477169,12.318818;
5.199441,1.579819,13.412124;
5.488221,2.403810,12.052998;
4.350878,4.600215,11.476987;
3.444811,4.121008,11.558955;
```

```
3.527893,4.768202,13.824538;
4.412634,4.284776,10.061614;
5.188036,2.185283,12.978771;
4.473938,3.502474,13.120910;
3.324503,4.717544,13.102851;
4.947167,2.743434,11.787135;
4.225398,3.034035,12.043086;
4.514438,4.246322,12.131302;
5.200625,3.489900,12.348179;
3.921953,4.377235,10.779057;
4.742795,2.244409,13.693519;
4.720830,1.739265,13.619524;
6.918994,2.755480,10.444477;
4.032259,2.634879,12.379584;
4.048847,3.411372,12.844863;
4.275113,2.696667,12.031433;
3.116881,4.715417,12.921323;
4.954436,3.314100,10.949134;
4.835395,4.852354,12.187223;
5.084543,1.926378,11.955591;
5.496240,3.716542,11.582061;
4.469747,4.951928,10.150955;
5.718912,1.546213,12.884910;
3.427047,3.615029,11.976696;
5.000090,2.919689,13.618889;
6.221958,3.306886,10.731690;
3.959728,4.546048,10.114697;
4.959606,1.671709,13.914723;
5.850778,3.001886,11.884353;
3.599462,3.638421,12.074380;
4.815191,2.729566,13.301255;
4.667198,3.627440,12.511893;
6.936255,1.668674,10.424865;
4.691342,3.191484,13.770948;
4.670976,4.932210,11.205820;
5.792422,3.666112,10.712530;
3.512058,4.996322,10.684484;
3.130403,3.244799,13.527466;
5.676701,1.761733,11.475666;
4.842904,4.926552,10.625620;
4.538476,3.331946,11.007224;
4.161763,3.468364,11.061124;
4.375508,3.336277,10.431076;
5.394095,2.883697,12.783797;
5.799551,3.554123,10.134415;
5.617783,2.630477,13.279925;
6.115209,2.693812,10.363293;
4.760341,3.108571,11.829697;
5.550836,4.830776,10.962828;
5.704489,2.156258,12.687233;
3.896160,3.671331,13.377569;
4.377850,4.122079,12.701328;
6.663965,1.004604,11.849797;
4.697396,2.843665,13.080639;
4.289887,4.138957,11.885429;
```

```
5.429557,1.766981,12.953707;
3.971398,4.669697,11.076246;
6.062000,1.754648,11.149993;
3.364454,3.304838,12.733453;
5.186372,2.702915,12.577771;
6.780696,1.835740,12.837127;
4.800551,2.834902,12.647779;
6.081142,2.400872,12.648038;
4.025764,3.453843,12.328997;
5.162957,4.479764,11.059116;
4.272296,1.476858,13.759318;
5.582207,2.917853,12.557268;
5.178864,3.589246,12.175544;
4.463265,4.054019,12.511586;
6.444559,2.939413,11.573825;
3.971141,2.769609,12.751184;
4.436913,3.945360,11.578830;
5.733663,3.816190,11.769222;
4.081082,1.788215,13.286885;
4.719686,4.551084,11.564732;
6.020308,2.509582,10.864076;
6.075417,1.669014,13.447922;
4.627819,3.994823,13.302335;
6.159852,2.274097,12.136257;
5.714609,2.980708,10.758842;
5.110720,2.918094,13.205390;
3.911372,2.992377,13.603410;
5.343948,1.986938,12.665665;
3.333931,3.503839,12.643778;
3.837620,3.209165,12.519534;
4.127467,3.154387,12.780652;
4.996464,3.143204,11.780733;
3.495729,2.961429,13.411993;
6.495710,2.081177,10.833845;
5.259918,3.561247,11.668116;
3.823902,4.791732,10.328285;
5.483835,3.294839,10.208312;
3.253618,4.441762,13.737620;
4.868273,3.592794,10.100913;
3.708430,3.651232,11.323316;
6.593945,1.472621,13.953672;
4.151397,2.658090,11.859360;
6.055828,4.272816,10.400886;
5.087543,2.343396,10.702676;
3.835787,4.620614,12.701565;
4.873873,4.648530,10.416046;
4.824231,1.406678,13.981559;
6.045704,3.524280,10.359567;
3.323450,4.108962,13.620539;
5.135088,1.436617,13.303235;
4.352391,2.175892,12.985254;
5.413872,3.104410,12.918838;
5.829014,4.125508,11.151908;
5.770128,3.226679,11.586083;
5.431464,3.965016,10.419253;
```

```
5.960129,1.939308,12.939830;
4.465746,2.476795,12.740114;
5.391767,4.157456,11.470612;
3.915813,3.567762,11.937921;
5.621992,1.439020,13.735039;
4.950415,4.075833,11.584027;
4.718258,2.806957,12.439429;
6.951739,1.681728,11.031169;
4.609553,4.931341,11.608736;
5.482688,1.617479,11.525381;
3.644536,4.032450,13.484444;
4.403107,3.742143,11.176595;
4.690583,2.438425,12.233277;
5.970181,2.697339,11.717423;
4.270082,3.614761,13.827744;
6.742923,2.831545,10.961914;
5.974753,1.423682,12.726242;
6.294298,1.700039,10.654280;
5.810809,1.614362,13.813828;
5.163536,3.718936,10.146252;
6.236815,3.994475,10.480748;
5.100181,2.303335,12.185798;
5.614800,4.730454,10.654049;
4.760142,2.030455,13.007786;
4.676193,2.563048,13.264560;
4.269711,4.258159,13.156294;
6.803578,2.775857,10.240075;
6.988013,1.896686,12.609804;
5.419963,2.548982,10.568749;
5.936919,3.284103,10.707420;
5.892694,2.389751,12.642467;
4.535474,3.509386,10.086599;
6.252451,2.533225,12.469117;
5.301979,3.120207,11.100279;
6.217798,4.944417,10.119968;
5.142657,1.348309,13.208366;
6.956580,1.267785,13.757593;
3.072710,3.735354,13.134946;
4.346797,3.649527,10.976661;
4.182029,3.720713,12.111387;
4.646374,3.410553,13.002080;
5.334133,3.207170,12.334282;
5.047280,1.330371,12.878281;
4.385795,4.546175,11.818779;
4.235658,3.904418,13.131488;
5.775150,1.039209,13.372853;
6.689328,4.083817,10.170639;
4.512745,3.817358,12.918052;
4.909969,3.494866,10.945780;
3.708495,4.318574,13.067687;
6.737913,1.431556,10.728910;
5.229181,4.089980,11.247760;
5.005132,2.726885,13.990241;
3.550186,2.560020,13.709425;
4.373152,4.744109,10.499096;
```

```
4.593129,3.999289,13.340882;
4.289842,3.209046,13.916517;
5.197234,2.321694,12.477886;
4.442546,4.026038,11.655603;
4.311020,4.351213,12.956289;
6.816698,1.127691,11.427476;
5.650615,2.126006,10.921532;
4.805362,1.963620,12.860180;
6.424729,2.126031,12.924203;
5.352838,2.464627,13.227038;
5.015123,2.958377,13.508195;
4.412567,2.797774,13.854121;
4.444088,3.481114,13.244604;
5.605398,1.924951,11.613965;
6.720163,2.596080,10.189606;
4.369494,3.943865,13.178729;
3.781907,3.880663,12.887013;
6.511196,3.329732,10.282737;
6.355878,2.733042,11.882499;
5.242854,2.076366,12.996074;
5.015551,3.587239,11.230982;
4.146482,4.203281,13.584445;
5.196632,3.913547,12.307033;
5.084812,2.489251,13.748539;
5.673857,1.827106,12.615402;
4.818967,1.986749,13.137692;
5.395472,1.595507,13.598854;
4.801574,1.822689,13.598604;
5.692904,3.657120,10.491260;
4.629274,2.101148,12.866679;
4.133538,4.584795,13.306316;
4.560106,2.991612,12.779221;
4.304169,2.825698,12.855182;
6.537620,3.883423,10.074451;
5.699106,2.754035,11.751281;
3.468147,4.258727,11.299422;
5.186215,3.247681,11.583289;
4.592524,3.061469,12.630122;
4.322728,2.720007,11.967225;
4.744740,4.306518,11.578139;
6.724447,1.763139,11.034329;
5.880658,2.387581,12.067962;
5.226779,1.625981,12.248224;
5.925548,2.440124,11.816849;
4.545560,4.102219,12.937084;
6.136930,3.822287,10.437337;
4.559723,3.363619,11.837520;
5.676170,3.000845,10.871975;
5.286463,1.488757,12.684665;
6.455473,1.390792,13.632209;
5.031395,3.342437,13.051548;
3.331851,3.646385,12.067916;
3.684192,4.754231,12.361933;
4.807783,4.358790,12.130494;
5.215548,3.720262,11.468760;
```

```
3.957162,3.315694,13.467548;
4.200738,2.605547,13.333454;
4.614515,2.560704,11.441796;
5.825629,1.974293,13.140280;
3.554596,3.395542,13.604232;
6.757519,1.884738,11.930686;
4.504044,3.095120,11.059490;
3.104428,4.818713,11.722386;
6.846234,4.049658,10.029395;
5.209239,1.872435,13.089465;
3.912113,2.483459,13.563715;
6.425508,2.609734,11.272076;
5.366378,2.330286,13.412255;
4.769592,4.617422,10.132718;
5.129706,3.865989,10.717207;
4.615427,3.194265,10.194954;
5.210929,2.099246,10.966007;
5.501021,3.172249,11.756149;
4.149709,3.006636,13.046185;
5.582138,1.492878,12.017591;
3.792679,3.689081,11.726045;
5.129132,2.117568,13.784921;
6.625773,2.570738,10.099421;
5.907658,2.495391,12.326328;
4.139295,3.379897,13.848644;
6.731592,2.562670,11.092867;
6.158919,1.941467,11.792079;
5.569261,1.885063,13.348226;
4.115502,3.986469,10.947722;
5.900729,1.919544,12.304214;
3.242425,3.902752,12.226223;
6.156116,2.271332,11.808830;
4.079535,3.098549,13.890604;
5.841635,2.247440,11.165829;
5.338874,4.792435,10.244116;
5.338565,2.140432,13.310929;
6.306296,3.707484,10.830412;
4.904319,4.632410,12.208700;
4.805499,2.530585,13.158575;
4.457147,3.129400,12.846627;
6.485906,2.314758,12.600472;
6.899345,1.303869,12.348077;
6.124581,1.783192,13.969436;
4.993414,4.235961,11.426036;
3.292974,3.363966,13.640751;
3.775064,2.729471,12.996639;
3.156738,4.785300,13.054693;
5.235282,1.735372,11.991795;
6.739916,2.917938,10.927166;
4.585161,3.820310,12.234236;
5.140268,4.855481,10.462504;
4.727922,3.170668,12.849659;
4.913898,2.027341,11.476367;
5.647060,1.678435,11.115136;
6.521351,2.884407,11.615877;
```

```
3.716926,4.875700,11.629823;
4.337318,3.386588,13.607963;
5.808266,2.509820,12.939824;
6.660104,3.570967,10.005676;
4.416466,4.121784,11.746627;
5.147151,4.048439,11.390269;
3.646290,3.862542,12.310956;
4.733196,4.536971,11.572207;
3.715901,3.533334,12.496002;
6.923913,1.508148,10.928961;
4.629838,4.536307,12.192531;
6.824785,1.496104,11.883053;
6.427585,1.173562,12.766501;
6.915942,2.133072,10.535122;
6.599931,1.773735,13.017698;
6.220450,1.268891,13.803161;
6.324518,1.626007,11.829235;
6.419704,1.139463,13.541680;
4.625493,2.544765,12.439207;
4.292745,4.078388,10.936472;
6.311910,2.173472,11.237477;
5.092119,2.301196,13.327370;
6.241179,3.227993,11.051856;
5.722265,1.934613,11.825701;
4.538268,3.154406,13.966815;
5.114236,1.205744,13.027502;
6.717938,2.638060,10.001366;
5.994036,3.173198,11.352529;
6.571333,2.426014,12.185608;
4.386727,3.491211,13.186499;
5.983500,1.502145,13.289576;
6.125496,2.469144,12.979471;
6.569069,1.970414,10.518388;
6.709952,1.205255,12.370667;
3.651596,4.353623,10.670244;
5.008802,4.997318,11.421629;
4.334673,1.918410,13.744481;
6.761347,1.023337,12.441228;
6.204303,1.931926,13.729875;
6.170327,2.316165,10.893848;
4.249545,3.338094,13.319657;
4.161850,2.610218,13.448229;
5.458958,4.964751,10.814796;
6.308836,3.703446,10.995798;
4.903143,2.596301,12.397753;
6.202091,1.420275,13.285769;
6.364345,2.418025,11.720278;
6.031537,2.556515,11.717210;
4.105381,3.489294,12.353447;
6.853874,1.343611,12.001996;
5.086359,1.360664,13.618666;
4.782625,4.375998,10.784820;
4.215406,2.933178,11.351248;
3.947519,3.808947,11.501887;
5.881373,2.936154,12.556124;
```

```
6.550547,1.794947,11.581465;
5.650064,3.093253,11.039577;
6.847975,3.160816,10.121081;
5.785258,3.078865,10.236122;
6.560145,2.320809,10.918805;
4.127281,4.520265,11.777321;
3.455723,4.914256,13.394387;
5.520820,1.921197,12.319540;
5.412625,3.399516,11.793712;
5.963621,3.027178,10.799702;
4.708774,1.674761,13.006778;
4.473403,4.767271,10.068690;
5.602031,3.906518,10.377954;
3.719660,4.705177,10.272722;
5.324373,3.548605,12.605077;
6.458488,1.223810,13.267421;
5.115690,3.777402,10.849620;
4.381231,4.787267,12.080761;
6.815252,1.294383,10.828128;
6.139418,2.083326,10.911243;
4.284093,4.318247,13.288729;
5.282731,3.287319,11.144073;
4.784862,2.862650,11.116157;
5.988788,2.042046,12.758551;
5.154102,2.128821,13.903830;
6.663063,1.541913,11.328472;
5.332532,3.793016,10.117329;
4.359945,4.386844,10.984278;
5.325965,4.750708,10.191149;
5.394514,1.456120,13.184981;
5.602122,3.065082,11.305553;
3.079059,4.857167,13.881492;
3.495442,2.869640,12.626776;
4.160742,4.018147,12.232472;
4.711170,2.068776,13.014944;
6.753263,2.021710,12.132653;
6.819019,2.070990,11.000338;
6.710691,1.274329,11.197602;
5.366334,1.813197,12.543533;
6.183820,1.933495,12.403356;
3.449850,3.063063,13.351363;
6.683160,2.992912,11.110444;
6.896766,1.789116,10.444740;
4.189417,2.585674,11.683023;
4.245901,3.775373,10.367487;
5.203243,4.483609,10.169014;
6.618888,1.523896,13.334916;
5.018929,2.619834,10.694289;
5.300734,3.424872,10.857783;
5.779493,2.645687,10.139107;
4.171326,4.205767,11.386008;
6.267645,1.757886,10.494774;
6.283986,3.551590,10.064479;
4.327957,4.358998,11.486893;
6.312861,1.706075,10.518079;
```

```
6.519534,1.176317,12.746880;
5.935092,2.748690,11.519356;
6.918630,2.595974,11.760748;
6.578003,1.988097,11.242716;
4.635476,3.832044,10.574550;
6.485288,1.332624,11.846951;
3.121556,4.012802,12.800171;
3.858049,3.719619,12.229171;
4.678071,2.432513,11.955950;
4.023849,4.716677,11.867027;
4.016031,2.724874,12.810120;
4.609319,1.727360,13.425004;
5.336805,2.494315,10.886778;
3.875976,3.088930,11.733691;
5.965216,1.281799,13.389334;
5.719521,1.546606,13.433609;
3.799336,3.429360,12.172180;
6.059150,2.684277,10.227253;
5.342988,1.696621,12.914443;
5.137164,2.012257,13.668227;
6.032778,4.548123,10.275193;
3.734113,3.948291,12.786859;
6.107972,3.007614,11.701987;
5.094368,2.195262,12.815876;
4.526444,3.270740,13.551444;
4.696339,2.364260,12.165416;
6.704676,2.193998,11.352339;
6.437920,2.361913,10.552482;
5.031196,4.426626,11.537254;
5.782764,3.511617,11.801553;
4.894472,4.798825,10.333990;
4.119316,2.788029,12.350285;
6.510537,2.876402,11.749674;
4.866048,2.992418,11.949723;
6.553564,1.932671,13.446383;
3.558757,2.575601,13.922251;
5.109736,3.899968,12.429663;
5.353466,2.733739,10.976692;
6.544940,1.987765,10.035660;
6.259681,1.561998,13.519466;
4.733476,1.559035,13.007718;
3.967147,3.601837,13.429496;
6.341618,4.342853,10.199430;
5.183545,4.772679,11.285892;
5.993960,3.271365,11.195857;
4.024439,4.546255,11.787203;
6.263949,1.393349,13.438374;
3.110516,4.596626,13.599742;
5.824430,4.325439,10.139335;
5.552975,2.372023,10.865886;
6.144803,3.892360,11.115356;
5.329726,2.684023,10.368275;
4.597053,3.790598,10.814706;
5.665306,2.772264,11.733181;
4.076042,3.238712,13.779136;
```

```
4.667739,2.520597,10.853082;
4.333491,4.903379,12.221774;
6.385215,2.632254,11.848073;
5.585384,2.518294,11.906299;
4.886272,3.171967,10.238764;
5.632121,4.558541,10.438530;
4.751092,2.120920,13.940985;
6.333395,1.814862,12.177671;
6.499771,1.484000,13.425403;
6.599104,1.871460,10.307916;
4.896861,4.340110,11.877574;
4.655080,3.010985,10.501760;
3.529141,4.481901,12.411801;
4.061207,4.459210,10.232438;
4.831016,3.888839,11.355997;
4.604879,3.107919,13.576943;
4.517479,4.458690,11.679841;
3.959510,3.390618,11.917618;
6.312382,3.029742,11.464650;
6.595759,2.414556,10.480712;
5.276444,4.500131,11.394300;
5.962106,2.826101,12.672997;
5.856970,2.231944,12.684656;
5.019242,2.951997,11.991353;
6.743902,2.557127,10.468581;
3.961694,3.739634,13.357014;
6.880579,1.860678,13.041373;
5.336411,2.611806,12.040161;
4.982577,3.605473,12.974825;
4.558348,4.101219,10.717476;
3.437445,4.620633,13.505404;
6.494049,1.280886,13.950155;
4.351378,3.882894,10.054622;
4.496241,4.690741,12.185919;
4.895547,2.986149,11.235884;
5.047498,3.207847,10.853140;
5.733232,3.434238,10.878624;
4.625120,3.519632,12.221230;
4.672650,2.953943,10.639475;
4.975527,4.450889,10.971505;
5.721407,2.515293,12.527819;
3.973078,3.285652,13.926915;
6.398728,2.133777,12.729831;
4.432498,4.947707,10.335945;
4.001224,4.245386,10.337692;
4.913990,3.349614,10.583223;
6.621322,3.560777,10.651758;
5.742749,2.864877,11.041272;
5.277073,1.995083,11.277206;
6.703239,1.715360,12.070165;
4.556771,3.960030,13.270539;
5.401379,1.339988,13.689432;
4.520572,4.251330,10.976384;
3.995678,2.011415,13.068974;
3.199448,3.741154,12.481112;
```

```
4.040822,4.509878,13.224384;
4.259134,3.906966,12.063091;
6.162580,1.817970,12.712424;
3.209942,4.204689,12.714275;
6.784036,1.366233,13.633753;
5.039812,3.459614,11.264285;
4.481943,3.489566,13.990208;
5.069377,4.962045,10.906138;
4.592021,3.786275,10.258563];
%
a = partSOSFitting(n, x, p, c, f, B, K);
disp(a)
% Saving the coefficient vector of the PSS polynomial in a txt file.
folder = 'C:\Home\PSS\Codes\Section 4-7'; % replace this directory to the directory of↙
the folder you are using.
baseFileName = "PSS_via_sampling_degree_" + d + "_output_vector_a.txt";
fullFileName = fullfile(folder, baseFileName);
a_vec_file = fopen(fullFileName, 'w');
fprintf(a_vec_file, 'The coefficient vector of the polynomial p of the PSS↙
respresentation.\n\n');
fprintf(a_vec_file, 'a: ');
for i = 1:length(a)
    fprintf(a_vec_file, '%f,', a(i));
end
fclose(a_vec_file);
%
% PART 3
%
% Plotting the 3d figure. The multistationary points and the surface that
% the PSS representation is surrended by it.
%
fig = figure;
fig.Units = 'pixels';
fig.Position(1:2) = [100, 100];
fig.Position(3:4) = [540, 460];
% The main plot.
fimplicit3(subs(p, c, a) - 1, [B(1, 1), B(1, 2), B(2, 1), B(2, 2), B(3, 1), B(3, 2)],↙
'EdgeColor','none', 'FaceColor', 'y')
axis([3 7 1 5 10 14])
xlabel('$k_1$', 'interpreter', 'latex', 'FontName', 'Times New Roman', 'FontSize', 18)
ylabel('$k_2$', 'interpreter', 'latex', 'FontName', 'Times New Roman', 'FontSize', 18)
zlabel('$k_3$', 'interpreter', 'latex', 'FontName', 'Times New Roman', 'FontSize', 18)
set(get(gca, 'ylabel'), 'rotation', 0)
set(get(gca, 'zlabel'), 'rotation', 0)
ax = gca;
ax.XRuler.Exponent = 0;
hold on
x_list = zeros(size(K, 1), 1);
y_list = zeros(size(K, 1), 1);
z_list = zeros(size(K, 1), 1);
for i = 1:size(K, 1)
    x_list(i) = K(i, 1);
    y_list(i) = K(i, 2);
    z_list(i) = K(i, 3);
end
```

```
scatter3(x_list, y_list, z_list, 40, [1, 0.5, 0], 'filled');
hold off
%
%%%%%%%%%%%%
% Functions %
%%%%%%%%%%%%
%
% Generating a matrix that each of its rows is an exponent vector of a
% monomial of degree at most d in n variables.
% The monomials are ordered with respect to the graded lexicographic
% monomial order.
%
function vMtx = allMonomials(n, d)
    % if isinteger(n)==0 || isinteger(d)==0 || n<=0 || d<0
    %     error("The input arguments are not valid. The first argument must be a↙
positive integer and the second argument must be a nonnegative integer.");
    % end
    vMtx = zeros(nchoosek(d+n, n), n);
    for idx = 1:1:nchoosek(d+n, n)-1
        vMtx(idx+1, :) = nxtMonomial(vMtx(idx, :));
    end
end
%
% Generating the next expnent vector of the next monomial in the graded
% lexicographic order.
%
function v = nxtMonomial(v, n)
    % if nargin>1
    %     if isinteger(n)==0 || n<1 || n~=length(v)
    %         error("The second input argument must be a positive integer equal to the↙
length of the first input argument.");
    %     end
    % else
    %     n=length(v);
    % end
    if nargin == 1
        n = length(v);
    end
    if n == 1
        v(1) = v(1)+1;
        return
    end
    idx1 = 0;
    for idx2 = n:-1:1
        if v(idx2) ~= 0
            idx1 = idx2;
            break;
        end
    end
    if idx1 == 0
        v(1) = 1;
        return
    end % so there is a first nonzero index.
    if idx1 ~= n
        v(idx1) = v(idx1)-1;
```

```
        v(idx1+1) = v(idx1+1)+1;
        return
    end % here we know idx1=n, so no point in keeping idx1 for saving a fixed known↙
number which is already saved at some variable.
    idx1 = 0;
    for idx2 = n-1:-1:1
        if v(idx2) ~= 0
            idx1 = idx2;
            break;
        end
    end
    if idx1 == 0
        v(1) = v(n)+1;
        v(n) = 0;
        return
    end % so there is a second nonzero index.
    tmpMem = v(n);
    v(n) = 0;
    v(idx1) = v(idx1)-1;
    v(idx1+1) = v(idx1+1)+tmpMem+1;
end
%
% The following function receives an integer, a symbol and another integer,
% respectively n, x and d. Then returns a polynomial in n variables of
% total degree d with all monomials. It also returns a second output which
% is a vector of coefficients of this polynomial.
%
function [p, c] = multPoly(n, x, d)
    %syms x [1,n] % remove x from the input arguments and ncomment this
    %line if you want the function generate the variables as x1 ... xn
    %itself.
    Mtx = allMonomials(n, d);
    c = str2sym(arrayfun(@(idx) "c" + join("_" + Mtx(idx,:), ""), 1:size(Mtx, 1)));
    p = sum(arrayfun(@(idx) c(idx).*prod(x.^Mtx(idx,:)), 1:size(Mtx, 1)));
end
%
% The following function receives a polynomial, an integer, a symbol and a
% hyperrectangle, respectively called p, n, x and B. Then it returns the
% n-dimensional integral of p in x over B.
%
function f = intOverB(p, n, x, B)
    f = p;
    for idx = n:-1:1
        f = int(f, x(idx), B(idx, :));
    end
end
%
% The following function is the SOS + linear optimization formulating the
% PSS polynomial with the help of YALMIP and SeDuMi.
%
function PSSCoeffs = partSOSFitting(n, x, p, c, f, B, K) %#ok<STOUT>
    % The following string should not be produced after the spdvar xi's,
    % otherwise you may get a strange string with x61 or x82 etc.
    tmpStr = "Goal=[sos(p";
    tmpStr = tmpStr + join(arrayfun(@(idx) "-s" + idx + "B*(" + string(x(idx)) + "-("↙
```

```
+ B(idx, 1) + "))*((" + B(idx ,2)+ ")-" + string(x(idx)) + ")", 1:n), "") + "),\n";
    tmpStr = tmpStr + join(arrayfun(@(idx) "sos(s" + idx + "B)", 1:n), ",\n") + ",\n";
    tmpStr = tmpStr + join(arrayfun(@(idx) ...
        string(subs(p, x, K(idx, :))) + ">=1", 1:size(K, 1)), ",\n") + "];";
    % back to the expected order of the lines.
    for idx = 1:n
        eval("sdpvar " + string(x(idx)));
    end
    for idx = 1:length(c)
        eval("sdpvar " + string(c(idx)));
    end
    eval("p=" + string(p));
    eval("F=" + string(f));
    for idx1 = 1:n
        eval("[s" + idx1 + "B,c" + idx1 + "B]=polynomial([" + join(arrayfun(@(idx2)↙
string(x(idx2)), 1:n)) + "],0)");
    end
    eval(sprintf(tmpStr));
    eval("solvesos(Goal, F, [], ["+ ...
        join(arrayfun(@(idx) "c" + idx + "B;", 1:n), "") + ...
        join(arrayfun(@(idx) string(c(idx)), 1:length(c)), ";") + "])");
    eval("PSSCoeffs=[" + join(arrayfun(@(idx) "value(" + string(c(idx)) + ")", 1:↙
length(c)), ",") + "]");
end
%
% End of the file.
```