

```
/* $OpenXM: OpenXM/src/asir-contrib/packages/src/tk_any2mpfr/-
solve_ode_by_r4d.h,v 1.1 2020/09/08 07:45:32 takayama Exp $ */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <mpfr.h>
#define N 2
#define PREC 64
#define MPFR_PREC PREC
// include proj.h
/* $OpenXM: OpenXM/src/asir-contrib/packages/src/tk_any2mpfr/proj.h,v
1.1 2020/09/08 07:45:32 takayama Exp $ */
void myerror(char *msg);
void output_eigen(double eigen_re[N], double eigen_im[N], double
eigen_vec_re[N][N], double eigen_vec_im[N][N]);
void pair_eigen(double eigen_re[],double eigen_im[],int pair[]);
void cp_mm(double m1[N][N],int row1,double m2[N][N],int row2);
void real_nf_basis(double eigen_vec_re[N][N], double eigen_vec_im[N]-
[N],int pair[N],double pseudo_eigen_vec[N][N]);
void to_normal_m(double mat[N][N],int row);
void cp_mv(double mat[N][N],int row, double vec[N]);
void cp_vm(double vec[N], double mat[N][N],int row);
void gs_step(double w[N],double orth_basis[N][N],int row);
void output_mat(double mat[N][N]);
void output_vec(double vec[],int n);

int gram_schmidt(double basis[N][N],int m,double orth_basis[N][N]);
int projection_mat(double Upper,double Lower,double eigen_re[N],double
eigen_im[N],double eigen_vec_re[N][N],double eigen_vec_im[N][N],double
proj_mat[N][N]);

int mat_get_d(double ans_d[N][N],mpfr_t ans[N][N]);
int mat_get_mdvec(double ans_d[N*N],mpfr_t ans[N][N]);
int mat_get_dvec(double ans_d[N*N],mpfr_t ans[N*N]);
int almost_eq(double a, double b);
int gram_schmidt(double basis[N][N],int m,double orth_basis[N][N]);

int projection_matrix_to_subspace_of(int n_prone, double data[],double
proj_mat[N][N]);

/* for check */
void mat_dup(double m1[N][N],double m2[N][N]);
int mat_transpose(double mat[N][N],double tmat[N][N]);
void check_projection(double upper,double lower,double
eigen_re[N],double eigen_im[N],double proj_mat[N][N],double
pseudo_eigen_vec[N][N],double basis[N][N],int m);
int mat_inverse(double mat[N][N],double inv_mat[N][N]);
int mat_prod(double m1[N][N],double m2[N][N],double m3[N][N]);
void mat_linear_transformation(double aa[N][N],double w[N],double
tw[N]);
void vec_scalar_multiplication(double s,double vec[N],double
vec_new[N]);
```

```
void cp_vmatm(double data[N*N],double mat[N][N]);
```

```
int test0(double data[]);
int test1();
int test2();
int main();
```

```
#define T_PROGNAME "tmp-test"
#define T_RANK 2
#define T_T0 0
#define T_T1 10.1
#define T_H 0.001
#define T_NOPROJ 0
#define T_STRAT 1
#define T_N_PRUNE 1
#define T_N_DEFUSE 5000
#define T_REF_VALUE_FILE "tmp_ref_value.txt"
#define T_F0 {0.355028053887817, -0.258819403792807}
```

```
/*code_mat_fac_mpfr, generated from
[[ 1/24*d1^2*d2*d0*h^4*t^2+(1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2)*t+(1/12*d1*d2+1/12*d1^2)*h^3+1
(1/12*d1^2*d0+1/12*d1^2*d2)*h^3*t+(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h ]
[ (1/12*d1^2*d0+1/12*d1^2*d2)*h^3*t^2+-
((1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h)*t+1/24*d1^2*d-
2*h^5+(1/6*d2+1/3*d1)*h^2
1/24*d1^2*d2*d0*h^4*t^2+(1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d-
1^2)*h^2)*t+1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1 ]
,[d0,1],[d1,1],[d2,1]] */
int mat_fac(mpfr_t ob_0 /* t */,mpfr_t ob_1 /* h */,int n,mpfr_t
*ans,int *argcv,mpfr_t *argv) {
```

```
int i,j;mpfr_t ob_85;
mpfr_t ob_84;
mpfr_t ob_83;
mpfr_t ob_82;
mpfr_t ob_81;
mpfr_t ob_80;
mpfr_t ob_79;
mpfr_t ob_78;
mpfr_t ob_77;
mpfr_t ob_76;
mpfr_t ob_75;
mpfr_t ob_74;
mpfr_t ob_73;
mpfr_t ob_72;
mpfr_t ob_71;
mpfr_t ob_70;
mpfr_t ob_69;
```

```
mpfr_t ob_68;  
mpfr_t ob_67;  
mpfr_t ob_66;  
mpfr_t ob_65;  
mpfr_t ob_64;  
mpfr_t ob_63;  
mpfr_t ob_62;  
mpfr_t ob_61;  
mpfr_t ob_60;  
mpfr_t ob_59;  
mpfr_t ob_58;  
mpfr_t ob_57;  
mpfr_t ob_56;  
mpfr_t ob_55;  
mpfr_t ob_54;  
mpfr_t ob_53;  
mpfr_t ob_52;  
mpfr_t ob_51;  
mpfr_t ob_50;  
mpfr_t ob_49;  
mpfr_t ob_48;  
mpfr_t ob_47;  
mpfr_t ob_46;  
mpfr_t ob_45;  
mpfr_t ob_44;  
mpfr_t ob_43;  
mpfr_t ob_42;  
mpfr_t ob_41;  
mpfr_t ob_40;  
mpfr_t ob_39;  
mpfr_t ob_38;  
mpfr_t ob_37;  
mpfr_t ob_36;  
mpfr_t ob_35;  
mpfr_t ob_34;  
mpfr_t ob_33;  
mpfr_t ob_32;  
mpfr_t ob_31;  
mpfr_t ob_30;  
mpfr_t ob_29;  
mpfr_t ob_28;  
mpfr_t ob_27;  
mpfr_t ob_26;  
mpfr_t ob_25;  
mpfr_t ob_24;  
mpfr_t ob_23;  
mpfr_t ob_22;  
mpfr_t ob_21;  
mpfr_t ob_20;  
mpfr_t ob_19;  
mpfr_t ob_18;  
mpfr_t ob_17;  
mpfr_t ob_16;
```

```
mpfr_t ob_15;
mpfr_t ob_14;
mpfr_t ob_13;
mpfr_t ob_12;
mpfr_t ob_11;
mpfr_t ob_10;
mpfr_t ob_9;
mpfr_t ob_8;
mpfr_t ob_7;
mpfr_t ob_6;
mpfr_t ob_5;
mpfr_t ob_4;
mpfr_t ob_3;
mpfr_t ob_2;
mpfr_t tmp_87;
mpfr_t tmp_86;
mpfr_t tmp_85;
mpfr_t tmp_84;
mpfr_t tmp_83;
mpfr_t tmp_82;
mpfr_t tmp_81;
mpfr_t tmp_80;
mpfr_t tmp_79;
mpfr_t tmp_78;
mpfr_t tmp_77;
mpfr_t tmp_76;
mpfr_t tmp_75;
mpfr_t tmp_74;
mpfr_t tmp_73;
mpfr_t tmp_72;
mpfr_t tmp_71;
mpfr_t tmp_70;
mpfr_t tmp_69;
mpfr_t tmp_68;
mpfr_t tmp_67;
mpfr_t tmp_66;
mpfr_t tmp_65;
mpfr_t tmp_64;
mpfr_t tmp_63;
mpfr_t tmp_62;
mpfr_t tmp_61;
mpfr_t tmp_60;
mpfr_t tmp_59;
mpfr_t tmp_58;
mpfr_t tmp_57;
mpfr_t tmp_56;
mpfr_t tmp_55;
mpfr_t tmp_54;
mpfr_t tmp_53;
mpfr_t tmp_52;
mpfr_t tmp_51;
mpfr_t tmp_50;
mpfr_t tmp_49;
```

```
mpfr_t tmp_48;
mpfr_t tmp_47;
mpfr_t tmp_46;
mpfr_t tmp_45;
mpfr_t tmp_44;
mpfr_t tmp_43;
mpfr_t tmp_42;
mpfr_t tmp_41;
mpfr_t tmp_40;
mpfr_t tmp_39;
mpfr_t tmp_38;
mpfr_t tmp_37;
mpfr_t tmp_36;
mpfr_t tmp_35;
mpfr_t tmp_34;
mpfr_t tmp_33;
mpfr_t tmp_32;
mpfr_t tmp_31;
mpfr_t tmp_30;
mpfr_t tmp_29;
mpfr_t tmp_28;
mpfr_t tmp_27;
mpfr_t tmp_26;
mpfr_t tmp_25;
mpfr_t tmp_24;
mpfr_t tmp_23;
mpfr_t tmp_22;
mpfr_t tmp_21;
mpfr_t tmp_20;
mpfr_t tmp_19;
mpfr_t tmp_18;
mpfr_t tmp_17;
mpfr_t tmp_16;
mpfr_t tmp_15;
mpfr_t tmp_14;
mpfr_t tmp_13;
mpfr_t tmp_12;
mpfr_t tmp_11;
mpfr_t tmp_10;
mpfr_t tmp_9;
mpfr_t tmp_8;
mpfr_t tmp_7;
mpfr_t tmp_6;
mpfr_t tmp_5;
mpfr_t tmp_4;
mpfr_t tmp_3;
mpfr_t tmp_2;
mpfr_t tmp_1;
mpfr_t tmp_0;
int first_4=1;
int first_3=1;
int first_2=1;
int first_1=1;
```

```
int first_0=1;
mpfr_init2(tmp_0,MPFR_PREC);
mpfr_init2(tmp_1,MPFR_PREC);
mpfr_init2(tmp_2,MPFR_PREC);
mpfr_init2(tmp_3,MPFR_PREC);
mpfr_init2(ob_2,MPFR_PREC); /* register d0 */
mpfr_init2(tmp_4,MPFR_PREC);
mpfr_init2(ob_3,MPFR_PREC); /* register 1 */
mpfr_strtofr(ob_3,"1",NULL,10,MPFR_RNDD);
mpfr_init2(ob_4,MPFR_PREC); /* register (1)<<0>> */
mpfr_init2(ob_5,MPFR_PREC); /* register d1 */
mpfr_init2(ob_6,MPFR_PREC); /* register d2 */
mpfr_init2(tmp_5,MPFR_PREC);
mpfr_init2(ob_7,MPFR_PREC); /* register 1/24 */
mpfr_init2(ob_8,MPFR_PREC); /* register 24 */
mpfr_strtofr(ob_8,"24",NULL,10,MPFR_RNDD);
mpfr_div(ob_7,ob_3,ob_8,MPFR_RNDD);
mpfr_init2(tmp_6,MPFR_PREC);
mpfr_init2(tmp_7,MPFR_PREC);
mpfr_init2(ob_9,MPFR_PREC); /* register d1^2 */
mpfr_init2(ob_10,MPFR_PREC); /* register 1/24*d1^2 */
mpfr_init2(tmp_8,MPFR_PREC);
mpfr_init2(tmp_9,MPFR_PREC);
mpfr_init2(ob_11,MPFR_PREC); /* register 1/24*d1^2*d2 */
mpfr_init2(tmp_10,MPFR_PREC);
mpfr_init2(tmp_11,MPFR_PREC);
mpfr_init2(ob_12,MPFR_PREC); /* register 1/24*d1^2*d2*d0 */
mpfr_init2(tmp_12,MPFR_PREC);
mpfr_init2(tmp_13,MPFR_PREC);
mpfr_init2(ob_13,MPFR_PREC); /* register h^4 */
mpfr_init2(ob_14,MPFR_PREC); /* register 1/24*d1^2*d2*d0*h^4 */
mpfr_init2(ob_15,MPFR_PREC); /* register 1/6 */
mpfr_init2(ob_16,MPFR_PREC); /* register 6 */
mpfr_strtofr(ob_16,"6",NULL,10,MPFR_RNDD);
mpfr_div(ob_15,ob_3,ob_16,MPFR_RNDD);
mpfr_init2(tmp_14,MPFR_PREC);
mpfr_init2(tmp_15,MPFR_PREC);
mpfr_init2(ob_17,MPFR_PREC); /* register 1/6*d1^2 */
mpfr_init2(tmp_16,MPFR_PREC);
mpfr_init2(tmp_17,MPFR_PREC);
mpfr_init2(ob_18,MPFR_PREC); /* register 1/6*d1 */
mpfr_init2(tmp_18,MPFR_PREC);
mpfr_init2(tmp_19,MPFR_PREC);
mpfr_init2(ob_19,MPFR_PREC); /* register 1/6*d1*d2+1/6*d1^2 */
mpfr_init2(tmp_20,MPFR_PREC);
mpfr_init2(tmp_21,MPFR_PREC);
mpfr_init2(ob_20,MPFR_PREC); /* register 1/6*d1*d0+1/6*d1*d2+1/6*d1^2
*/
mpfr_init2(ob_21,MPFR_PREC); /* register 1/48 */
mpfr_init2(ob_22,MPFR_PREC); /* register 48 */
mpfr_strtofr(ob_22,"48",NULL,10,MPFR_RNDD);
mpfr_div(ob_21,ob_3,ob_22,MPFR_RNDD);
mpfr_init2(tmp_22,MPFR_PREC);
```

```
mpfr_init2(tmp_23,MPFR_PREC);
mpfr_init2(ob_23,MPFR_PREC); /* register  $1/48*d1^2$  */
mpfr_init2(tmp_24,MPFR_PREC);
mpfr_init2(tmp_25,MPFR_PREC);
mpfr_init2(ob_24,MPFR_PREC); /* register  $1/48*d1^2*d2$  */
mpfr_init2(tmp_26,MPFR_PREC);
mpfr_init2(tmp_27,MPFR_PREC);
mpfr_init2(ob_25,MPFR_PREC); /* register  $1/48*d1^2*d2*d0$  */
mpfr_init2(tmp_28,MPFR_PREC);
mpfr_init2(tmp_29,MPFR_PREC);
mpfr_init2(ob_26,MPFR_PREC); /* register  $h^2$  */
mpfr_init2(ob_27,MPFR_PREC); /* register  $h^5$  */
mpfr_init2(ob_28,MPFR_PREC); /* register
 $1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2$  */
mpfr_init2(ob_29,MPFR_PREC); /* register  $1/12$  */
mpfr_init2(ob_30,MPFR_PREC); /* register  $12$  */
mpfr_strtofr(ob_30,"12",NULL,10,MPFR_RNDD);
mpfr_div(ob_29,ob_3,ob_30,MPFR_RNDD);
mpfr_init2(tmp_30,MPFR_PREC);
mpfr_init2(tmp_31,MPFR_PREC);
mpfr_init2(ob_31,MPFR_PREC); /* register  $1/12*d1^2$  */
mpfr_init2(tmp_32,MPFR_PREC);
mpfr_init2(tmp_33,MPFR_PREC);
mpfr_init2(ob_32,MPFR_PREC); /* register  $1/12*d1$  */
mpfr_init2(tmp_34,MPFR_PREC);
mpfr_init2(tmp_35,MPFR_PREC);
mpfr_init2(ob_33,MPFR_PREC); /* register  $1/12*d1*d2+1/12*d1^2$  */
mpfr_init2(tmp_36,MPFR_PREC);
mpfr_init2(tmp_37,MPFR_PREC);
mpfr_init2(ob_34,MPFR_PREC); /* register  $h^3$  */
mpfr_init2(ob_35,MPFR_PREC); /* register  $(1/12*d1*d2+1/12*d1^2)*h^3+1$ 
*/
mpfr_init2(tmp_38,MPFR_PREC);
mpfr_init2(tmp_39,MPFR_PREC);
mpfr_init2(ob_36,MPFR_PREC); /* register  $t^2$  */
mpfr_init2(ob_37,MPFR_PREC); /* register  $(1/24*d1^2*d2*d0*h^4)*\langle 2 \rangle$  */
mpfr_init2(ob_38,MPFR_PREC); /* register
 $(1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2)*\langle 1 \rangle$  */
mpfr_init2(ob_39,MPFR_PREC); /* register
 $((1/12*d1*d2+1/12*d1^2)*h^3+1)*\langle 0 \rangle$  */
mpfr_init2(ob_40,MPFR_PREC); /* register
 $1/24*d1^2*d2*d0*h^4*t^2+(1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2)*t+(1/12*d1*d2+1/12*d1^2)*h^3+1$  */
mpfr_init2(tmp_40,MPFR_PREC);
mpfr_init2(tmp_41,MPFR_PREC);
mpfr_init2(tmp_42,MPFR_PREC);
mpfr_init2(ob_41,MPFR_PREC); /* register  $1/12*d1^2*d2$  */
mpfr_init2(tmp_43,MPFR_PREC);
mpfr_init2(tmp_44,MPFR_PREC);
mpfr_init2(ob_42,MPFR_PREC); /* register  $1/12*d1^2*d0+1/12*d1^2*d2$  */
mpfr_init2(tmp_45,MPFR_PREC);
mpfr_init2(tmp_46,MPFR_PREC);
mpfr_init2(ob_43,MPFR_PREC); /* register
```

```
(1/12*d1^2*d0+1/12*d1^2*d2)*h^3 */
mpfr_init2(ob_44,MPFR_PREC); /* register 2/3 */
mpfr_init2(ob_45,MPFR_PREC); /* register 2 */
mpfr_strtofr(ob_45,"2",NULL,10,MPFR_RNDD);
mpfr_init2(ob_46,MPFR_PREC); /* register 3 */
mpfr_strtofr(ob_46,"3",NULL,10,MPFR_RNDD);
mpfr_div(ob_44,ob_45,ob_46,MPFR_RNDD);
mpfr_init2(tmp_47,MPFR_PREC);
mpfr_init2(tmp_48,MPFR_PREC);
mpfr_init2(ob_47,MPFR_PREC); /* register 2/3*d1 */
mpfr_init2(tmp_49,MPFR_PREC);
mpfr_init2(tmp_50,MPFR_PREC);
mpfr_init2(ob_48,MPFR_PREC); /* register 1/6*d2+2/3*d1 */
mpfr_init2(tmp_51,MPFR_PREC);
mpfr_init2(tmp_52,MPFR_PREC);
mpfr_init2(ob_49,MPFR_PREC); /* register 1/6*d0+1/6*d2+2/3*d1 */
mpfr_init2(tmp_53,MPFR_PREC);
mpfr_init2(tmp_54,MPFR_PREC);
mpfr_init2(ob_50,MPFR_PREC); /* register 1/24*d1^2*d0+1/24*d1^2*d2 */
mpfr_init2(tmp_55,MPFR_PREC);
mpfr_init2(tmp_56,MPFR_PREC);
mpfr_init2(ob_51,MPFR_PREC); /* register
(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
mpfr_init2(ob_52,MPFR_PREC); /* register
((1/12*d1^2*d0+1/12*d1^2*d2)*h^3)*<<1>> */
mpfr_init2(ob_53,MPFR_PREC); /* register
((1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h)*<<0>> */
mpfr_init2(ob_54,MPFR_PREC); /* register
(1/12*d1^2*d0+1/12*d1^2*d2)*h^3*t+(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*-
d0+1/6*d2+2/3*d1)*h */
mpfr_init2(tmp_57,MPFR_PREC);
mpfr_init2(ob_55,MPFR_PREC); /* register 1/8 */
mpfr_init2(ob_56,MPFR_PREC); /* register 8 */
mpfr_strtofr(ob_56,"8",NULL,10,MPFR_RNDD);
mpfr_div(ob_55,ob_3,ob_56,MPFR_RNDD);
mpfr_init2(tmp_58,MPFR_PREC);
mpfr_init2(tmp_59,MPFR_PREC);
mpfr_init2(ob_57,MPFR_PREC); /* register 1/8*d1^2 */
mpfr_init2(tmp_60,MPFR_PREC);
mpfr_init2(tmp_61,MPFR_PREC);
mpfr_init2(ob_58,MPFR_PREC); /* register 1/8*d1^2*d2 */
mpfr_init2(tmp_62,MPFR_PREC);
mpfr_init2(tmp_63,MPFR_PREC);
mpfr_init2(ob_59,MPFR_PREC); /* register 1/24*d1^2*d0+1/8*d1^2*d2 */
mpfr_init2(tmp_64,MPFR_PREC);
mpfr_init2(tmp_65,MPFR_PREC);
mpfr_init2(ob_60,MPFR_PREC); /* register
(1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
mpfr_init2(ob_61,MPFR_PREC); /* register 1/3 */
mpfr_div(ob_61,ob_3,ob_46,MPFR_RNDD);
mpfr_init2(tmp_66,MPFR_PREC);
mpfr_init2(tmp_67,MPFR_PREC);
mpfr_init2(ob_62,MPFR_PREC); /* register 1/3*d1 */
```



```
mpfr_init2(tmp_68,MPFR_PREC);
mpfr_init2(tmp_69,MPFR_PREC);
mpfr_init2(ob_63,MPFR_PREC); /* register  $1/6*d_2+1/3*d_1$  */
mpfr_init2(tmp_70,MPFR_PREC);
mpfr_init2(tmp_71,MPFR_PREC);
mpfr_init2(ob_64,MPFR_PREC); /* register
 $1/24*d_1^2*d_2*h^5+(1/6*d_2+1/3*d_1)*h^2$  */
mpfr_init2(ob_65,MPFR_PREC); /* register
 $((1/12*d_1^2*d_0+1/12*d_1^2*d_2)*h^3)*\langle\langle 2 \rangle\rangle$  */
mpfr_init2(ob_66,MPFR_PREC); /* register
 $((1/24*d_1^2*d_0+1/8*d_1^2*d_2)*h^4+(1/6*d_0+1/6*d_2+2/3*d_1)*h)*\langle\langle 1 \rangle\rangle$  */
mpfr_init2(ob_67,MPFR_PREC); /* register
 $(1/24*d_1^2*d_2*h^5+(1/6*d_2+1/3*d_1)*h^2)*\langle\langle 0 \rangle\rangle$  */
mpfr_init2(ob_68,MPFR_PREC); /* register
 $(1/12*d_1^2*d_0+1/12*d_1^2*d_2)*h^3*t^2+$ 
 $((1/24*d_1^2*d_0+1/8*d_1^2*d_2)*h^4+(1/6*d_0+1/6*d_2+2/3*d_1)*h)*t+1/24*d_1^2*d_2$ 
 $*h^5+(1/6*d_2+1/3*d_1)*h^2$  */
mpfr_init2(tmp_72,MPFR_PREC);
mpfr_init2(ob_69,MPFR_PREC); /* register  $1/16$  */
mpfr_init2(ob_70,MPFR_PREC); /* register  $16$  */
mpfr_strtofr(ob_70,"16",NULL,10,MPFR_RNDD);
mpfr_div(ob_69,ob_3,ob_70,MPFR_RNDD);
mpfr_init2(tmp_73,MPFR_PREC);
mpfr_init2(tmp_74,MPFR_PREC);
mpfr_init2(ob_71,MPFR_PREC); /* register  $1/16*d_1^2$  */
mpfr_init2(tmp_75,MPFR_PREC);
mpfr_init2(tmp_76,MPFR_PREC);
mpfr_init2(ob_72,MPFR_PREC); /* register  $1/16*d_1^2*d_2$  */
mpfr_init2(tmp_77,MPFR_PREC);
mpfr_init2(tmp_78,MPFR_PREC);
mpfr_init2(ob_73,MPFR_PREC); /* register  $1/16*d_1^2*d_2*d_0$  */
mpfr_init2(tmp_79,MPFR_PREC);
mpfr_init2(tmp_80,MPFR_PREC);
mpfr_init2(ob_74,MPFR_PREC); /* register
 $1/16*d_1^2*d_2*d_0*h^5+(1/6*d_1*d_0+1/6*d_1*d_2+1/6*d_1^2)*h^2$  */
mpfr_init2(tmp_81,MPFR_PREC);
mpfr_init2(tmp_82,MPFR_PREC);
mpfr_init2(ob_75,MPFR_PREC); /* register  $1/6*d_1*d_2+1/12*d_1^2$  */
mpfr_init2(tmp_83,MPFR_PREC);
mpfr_init2(tmp_84,MPFR_PREC);
mpfr_init2(ob_76,MPFR_PREC); /* register  $1/12*d_1*d_0+1/6*d_1*d_2+1/12*d_1^2$ 
*/
mpfr_init2(tmp_85,MPFR_PREC);
mpfr_init2(tmp_86,MPFR_PREC);
mpfr_init2(ob_77,MPFR_PREC); /* register  $h^6$  */
mpfr_init2(ob_78,MPFR_PREC); /* register
 $1/48*d_1^2*d_2*d_0*h^6+(1/12*d_1*d_0+1/6*d_1*d_2+1/12*d_1^2)*h^3+1$  */
mpfr_init2(ob_79,MPFR_PREC); /* register
 $(1/16*d_1^2*d_2*d_0*h^5+(1/6*d_1*d_0+1/6*d_1*d_2+1/6*d_1^2)*h^2)*\langle\langle 1 \rangle\rangle$  */
mpfr_init2(ob_80,MPFR_PREC); /* register
 $(1/48*d_1^2*d_2*d_0*h^6+(1/12*d_1*d_0+1/6*d_1*d_2+1/12*d_1^2)*h^3+1)*\langle\langle 0 \rangle\rangle$  */
mpfr_init2(ob_81,MPFR_PREC); /* register
 $1/24*d_1^2*d_2*d_0*h^4*t^2+(1/16*d_1^2*d_2*d_0*h^5+(1/6*d_1*d_0+1/6*d_1*d_2+1/6*d_2$ 
```

```

1^2)*h^2)*t+1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1
*/
mpfr_init2(tmp_87,MPFR_PREC);
mpfr_init2(ob_82,MPFR_PREC); /* register ob_40*tmp_0+ob_54*tmp_2 */
mpfr_init2(ob_83,MPFR_PREC); /* register ob_40*tmp_1+ob_54*tmp_3 */
mpfr_init2(ob_84,MPFR_PREC); /* register ob_68*tmp_0+ob_81*tmp_2 */
mpfr_init2(ob_85,MPFR_PREC); /* register ob_68*tmp_1+ob_81*tmp_3 */
    mpfr_strtofr(tmp_0,"1",NULL,10,MPFR_RNDD);
    /* Ob_mat[0][0] = tmp_0 */
    mpfr_set_si(tmp_1,(long) 0,MPFR_RNDD);
    /* Ob_mat[0][1] = tmp_1 */
    mpfr_set_si(tmp_2,(long) 0,MPFR_RNDD);
    /* Ob_mat[1][0] = tmp_2 */
    mpfr_strtofr(tmp_3,"1",NULL,10,MPFR_RNDD);
    /* Ob_mat[1][1] = tmp_3 */
    /* Matrix factorial is stored in P=[[tmp_0,tmp_1],[tmp_2,tmp_3]] */
    for (i=0; i<n; i++) { /* begin for */
        /* register d0 to ob_2 in init_code */
        /*begin of poly_to_mpfr(1,[t]); */
        mpfr_set_si(tmp_4,(long) 0,MPFR_RNDD);
        if (first_0) {
            /* register 1 to ob_3 in init_code */
            first_0=0;
        }

        /* register (1)*<<0>> to ob_4 in init_code */
        mpfr_mul(ob_4,ob_3,ob_3,MPFR_RNDD);
    mpfr_add(tmp_4,tmp_4,ob_4,MPFR_RNDD);

        /*Note: The summation tmp_ var tmp_4 is not used because obj 1
is already evaluated. */
        /*end of poly_to_mpfr(); */
        /* register d1 to ob_5 in init_code */
        /* register d2 to ob_6 in init_code */
        mpfr_set(ob_2,ob_3,MPFR_RNDD); /* set variable d0=1 */
    mpfr_set(ob_5,ob_3,MPFR_RNDD); /* set variable d1=1 */
    mpfr_set(ob_6,ob_3,MPFR_RNDD); /* set variable d2=1 */

        /*begin of
poly_to_mpfr(1/24*d1^2*d2*d0*h^4*t^2+(1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/-
6*d1*d2+1/6*d1^2)*h^2)*t+(1/12*d1*d2+1/12*d1^2)*h^3+1,[t]); */
        mpfr_set_si(tmp_5,(long) 0,MPFR_RNDD);
        if (first_1) {
            /* evaluate 1/24*d1^2*d2*d0*h^4 */
            /* evaluate 1/24*d1^2*d2*d0 */
            /* evaluate 1/24*d1^2*d2 */
            /* evaluate 1/24*d1^2 */
                /* register 1/24 to ob_7 in init_code */
                /* register 24 to ob_8 in init_code */
                mpfr_set_si(tmp_6,(long) 0,MPFR_RNDD);
                /* tmp_6 will be set to obj 1/24*d1^2 */
                /* register d1^2 to ob_9 in init_code */

```

```

mpfr_pow_ui(ob_9,ob_5,2,MPFR_RNDD);
mpfr_mul(tmp_7,ob_9,ob_7,MPFR_RNDD);/*
Tmp=d1^2*ob_7  coef of d1^2 */
/* Fr_F += Tmp */
init_code */
mpfr_add(tmp_6,tmp_6,tmp_7,MPFR_RNDD); /
/* register 1/24*d1^2 to ob_10 in
mpfr_set(ob_10,tmp_6,MPFR_RNDD);
/* Done of 1/24*d1^2 */
mpfr_set_si(tmp_8,(long) 0,MPFR_RNDD);
/* tmp_8 will be set to obj 1/24*d1^2*d2 */
mpfr_mul(tmp_9,ob_6,ob_10,MPFR_RNDD);/*
Tmp=d2^1*ob_10  coef of d2^1 */
mpfr_add(tmp_8,tmp_8,tmp_9,MPFR_RNDD); /*
Fr_F += Tmp */
/* register 1/24*d1^2*d2 to ob_11 in
init_code */
mpfr_set(ob_11,tmp_8,MPFR_RNDD);
/* Done of 1/24*d1^2*d2 */
mpfr_set_si(tmp_10,(long) 0,MPFR_RNDD);
/* tmp_10 will be set to obj 1/24*d1^2*d2*d0 */
mpfr_mul(tmp_11,ob_2,ob_11,MPFR_RNDD);/*
Tmp=d0^1*ob_11  coef of d0^1 */
mpfr_add(tmp_10,tmp_10,tmp_11,MPFR_RNDD); /*
Fr_F += Tmp */
/* register 1/24*d1^2*d2*d0 to ob_12 in
init_code */
mpfr_set(ob_12,tmp_10,MPFR_RNDD);
/* Done of 1/24*d1^2*d2*d0 */
mpfr_set_si(tmp_12,(long) 0,MPFR_RNDD);
/* tmp_12 will be set to obj 1/24*d1^2*d2*d0*h^4 */
/* register h^4 to ob_13 in init_code */
mpfr_pow_ui(ob_13,ob_1,4,MPFR_RNDD);
mpfr_mul(tmp_13,ob_13,ob_12,MPFR_RNDD);/*
Tmp=h^4*ob_12  coef of h^4 */
mpfr_add(tmp_12,tmp_12,tmp_13,MPFR_RNDD); /* Fr_F
+= Tmp */
/* register 1/24*d1^2*d2*d0*h^4 to ob_14 in
init_code */
mpfr_set(ob_14,tmp_12,MPFR_RNDD);
/* Done of 1/24*d1^2*d2*d0*h^4 */
/* evaluate
1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 */
/* evaluate 1/6*d1*d0+1/6*d1*d2+1/6*d1^2 */
/* evaluate 1/6*d1*d2+1/6*d1^2 */
/* evaluate 1/6*d1^2 */
/* register 1/6 to ob_15 in init_code */
/* register 6 to ob_16 in init_code */
mpfr_set_si(tmp_14,(long) 0,MPFR_RNDD);
/* tmp_14 will be set to obj 1/6*d1^2 */
mpfr_mul(tmp_15,ob_9,ob_15,MPFR_RNDD);/-
* Tmp=d1^2*ob_15  coef of d1^2 */

```

```

mpfr_add(tmp_14,tmp_14,tmp_15,MPFR_RNDD); /* Fr_F += Tmp */
/* register 1/6*d1^2 to ob_17 in
init_code */
    mpfr_set(ob_17,tmp_14,MPFR_RNDD);
/* Done of 1/6*d1^2 */
/* evaluate 1/6*d1 */
    mpfr_set_si(tmp_16,(long) 0,MPFR_RNDD);
/* tmp_16 will be set to obj 1/6*d1 */
    mpfr_mul(tmp_17,ob_5,ob_15,MPFR_RNDD);/-
* Tmp=d1^1*ob_15 coef of d1^1 */

mpfr_add(tmp_16,tmp_16,tmp_17,MPFR_RNDD); /* Fr_F += Tmp */
/* register 1/6*d1 to ob_18 in
init_code */
    mpfr_set(ob_18,tmp_16,MPFR_RNDD);
/* Done of 1/6*d1 */
    mpfr_set_si(tmp_18,(long) 0,MPFR_RNDD);
/* tmp_18 will be set to obj 1/6*d1*d2+1/6*d1^2
*/
    mpfr_mul(tmp_19,ob_3,ob_17,MPFR_RNDD);/*
Tmp=d2^0*ob_17 coef of d2^0 */
    mpfr_add(tmp_18,tmp_18,tmp_19,MPFR_RNDD); /
* Fr_F += Tmp */
    mpfr_mul(tmp_19,ob_6,ob_18,MPFR_RNDD);/*
Tmp=d2^1*ob_18 coef of d2^1 */
    mpfr_add(tmp_18,tmp_18,tmp_19,MPFR_RNDD); /
* Fr_F += Tmp */
/* register 1/6*d1*d2+1/6*d1^2 to ob_19 in
init_code */
    mpfr_set(ob_19,tmp_18,MPFR_RNDD);
/* Done of 1/6*d1*d2+1/6*d1^2 */
/* 1/6*d1 is already evaluated as ob_18 */
    mpfr_set_si(tmp_20,(long) 0,MPFR_RNDD);
/* tmp_20 will be set to obj
1/6*d1*d0+1/6*d1*d2+1/6*d1^2 */
    mpfr_mul(tmp_21,ob_3,ob_19,MPFR_RNDD);/*
Tmp=d0^0*ob_19 coef of d0^0 */
    mpfr_add(tmp_20,tmp_20,tmp_21,MPFR_RNDD); /*
Fr_F += Tmp */
    mpfr_mul(tmp_21,ob_2,ob_18,MPFR_RNDD);/*
Tmp=d0^1*ob_18 coef of d0^1 */
    mpfr_add(tmp_20,tmp_20,tmp_21,MPFR_RNDD); /*
Fr_F += Tmp */
/* register 1/6*d1*d0+1/6*d1*d2+1/6*d1^2 to
ob_20 in init_code */
    mpfr_set(ob_20,tmp_20,MPFR_RNDD);
/* Done of 1/6*d1*d0+1/6*d1*d2+1/6*d1^2 */
/* evaluate 1/48*d1^2*d2*d0 */
/* evaluate 1/48*d1^2*d2 */
/* evaluate 1/48*d1^2 */
/* register 1/48 to ob_21 in init_code
*/
/* register 48 to ob_22 in init_code */

```

```

        mpfr_set_si(tmp_22,(long) 0,MPFR_RNDD);
        /* tmp_22 will be set to obj 1/48*d1^2 */
        mpfr_mul(tmp_23,ob_9,ob_21,MPFR_RNDD);/-
* Tmp=d1^2*ob_21  coef of d1^2 */

mpfr_add(tmp_22,tmp_22,tmp_23,MPFR_RNDD); /* Fr_F += Tmp */
        /* register 1/48*d1^2 to ob_23 in
init_code */

        mpfr_set(ob_23,tmp_22,MPFR_RNDD);
        /* Done of 1/48*d1^2 */
        mpfr_set_si(tmp_24,(long) 0,MPFR_RNDD);
        /* tmp_24 will be set to obj 1/48*d1^2*d2 */
        mpfr_mul(tmp_25,ob_6,ob_23,MPFR_RNDD);/*
Tmp=d2^1*ob_23  coef of d2^1 */
        mpfr_add(tmp_24,tmp_24,tmp_25,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/48*d1^2*d2 to ob_24 in
init_code */

        mpfr_set(ob_24,tmp_24,MPFR_RNDD);
        /* Done of 1/48*d1^2*d2 */
        mpfr_set_si(tmp_26,(long) 0,MPFR_RNDD);
        /* tmp_26 will be set to obj 1/48*d1^2*d2*d0 */
        mpfr_mul(tmp_27,ob_2,ob_24,MPFR_RNDD);/*
Tmp=d0^1*ob_24  coef of d0^1 */
        mpfr_add(tmp_26,tmp_26,tmp_27,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/48*d1^2*d2*d0 to ob_25 in
init_code */

        mpfr_set(ob_25,tmp_26,MPFR_RNDD);
        /* Done of 1/48*d1^2*d2*d0 */
        mpfr_set_si(tmp_28,(long) 0,MPFR_RNDD);
        /* tmp_28 will be set to obj
1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 */
        /* register h^2 to ob_26 in init_code */
        mpfr_pow_ui(ob_26,ob_1,2,MPFR_RNDD);
        mpfr_mul(tmp_29,ob_26,ob_20,MPFR_RNDD);/*
Tmp=h^2*ob_20  coef of h^2 */
        mpfr_add(tmp_28,tmp_28,tmp_29,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register h^5 to ob_27 in init_code */
        mpfr_pow_ui(ob_27,ob_1,5,MPFR_RNDD);
        mpfr_mul(tmp_29,ob_27,ob_25,MPFR_RNDD);/*
Tmp=h^5*ob_25  coef of h^5 */
        mpfr_add(tmp_28,tmp_28,tmp_29,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register
1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 to ob_28 in
init_code */

        mpfr_set(ob_28,tmp_28,MPFR_RNDD);
        /* Done of
1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 */
        /* evaluate (1/12*d1*d2+1/12*d1^2)*h^3+1 */
        /* evaluate 1/12*d1*d2+1/12*d1^2 */

```

```

        /* evaluate 1/12*d1^2 */
        /* register 1/12 to ob_29 in init_code */
        /* register 12 to ob_30 in init_code */
        mpfr_set_si(tmp_30,(long) 0,MPFR_RNDD);
        /* tmp_30 will be set to obj 1/12*d1^2 */
        mpfr_mul(tmp_31,ob_9,ob_29,MPFR_RNDD);/*
Tmp=d1^2*ob_29  coef of d1^2 */
        mpfr_add(tmp_30,tmp_30,tmp_31,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/12*d1^2 to ob_31 in init_code
*/
        mpfr_set(ob_31,tmp_30,MPFR_RNDD);
        /* Done of 1/12*d1^2 */
        /* evaluate 1/12*d1 */
        mpfr_set_si(tmp_32,(long) 0,MPFR_RNDD);
        /* tmp_32 will be set to obj 1/12*d1 */
        mpfr_mul(tmp_33,ob_5,ob_29,MPFR_RNDD);/*
Tmp=d1^1*ob_29  coef of d1^1 */
        mpfr_add(tmp_32,tmp_32,tmp_33,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/12*d1 to ob_32 in init_code */
        mpfr_set(ob_32,tmp_32,MPFR_RNDD);
        /* Done of 1/12*d1 */
        mpfr_set_si(tmp_34,(long) 0,MPFR_RNDD);
        /* tmp_34 will be set to obj 1/12*d1*d2+1/12*d1^2 */
        mpfr_mul(tmp_35,ob_3,ob_31,MPFR_RNDD);/*
Tmp=d2^0*ob_31  coef of d2^0 */
        mpfr_add(tmp_34,tmp_34,tmp_35,MPFR_RNDD); /*
Fr_F += Tmp */
        mpfr_mul(tmp_35,ob_6,ob_32,MPFR_RNDD);/*
Tmp=d2^1*ob_32  coef of d2^1 */
        mpfr_add(tmp_34,tmp_34,tmp_35,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/12*d1*d2+1/12*d1^2 to ob_33 in
init_code */
        mpfr_set(ob_33,tmp_34,MPFR_RNDD);
        /* Done of 1/12*d1*d2+1/12*d1^2 */
        mpfr_set_si(tmp_36,(long) 0,MPFR_RNDD);
        /* tmp_36 will be set to obj
(1/12*d1*d2+1/12*d1^2)*h^3+1 */
        mpfr_mul(tmp_37,ob_3,ob_3,MPFR_RNDD);/*
Tmp=h^0*ob_3  coef of h^0 */
        mpfr_add(tmp_36,tmp_36,tmp_37,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register h^3 to ob_34 in init_code */
        mpfr_pow_ui(ob_34,ob_1,3,MPFR_RNDD);
        mpfr_mul(tmp_37,ob_34,ob_33,MPFR_RNDD);/*
Tmp=h^3*ob_33  coef of h^3 */
        mpfr_add(tmp_36,tmp_36,tmp_37,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register (1/12*d1*d2+1/12*d1^2)*h^3+1 to ob_35
in init_code */
        mpfr_set(ob_35,tmp_36,MPFR_RNDD);

```

```

        /* Done of  $(1/12*d1*d2+1/12*d1^2)*h^3+1$  */
        first_1=0;
    }

    /* evaluate  $t^2$  */
    mpfr_set_si(tmp_38,(long) 0,MPFR_RNDD);
    /* tmp_38 will be set to obj  $t^2$  */
    /* register  $t^2$  to ob_36 in init_code */
    mpfr_pow_ui(ob_36,ob_0,2,MPFR_RNDD);
    mpfr_mul(tmp_39,ob_36,ob_3,MPFR_RNDD); /*  $Tmp=t^2*ob_3$ 
coef of  $t^2$  */
    mpfr_add(tmp_38,tmp_38,tmp_39,MPFR_RNDD); /*  $Fr_F +=$ 
Tmp */

    /*Note: The summation tmp_ var tmp_38 is not used
because obj  $t^2$  is already evaluated. */
    /* Done of  $t^2$  */
    /*  $1/24*d1^2*d2*d0*h^4$  is already evaluated as ob_14 */
    /* register  $(1/24*d1^2*d2*d0*h^4)*\langle 2 \rangle$  to ob_37 in
init_code */
    mpfr_mul(ob_37,ob_14,ob_36,MPFR_RNDD);
    mpfr_add(tmp_5,tmp_5,ob_37,MPFR_RNDD);
    /* t is already evaluated as ob_0 */
    /*  $1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2$ 
is already evaluated as ob_28 */
    /* register
 $(1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2)*\langle 1 \rangle$  to ob_38
in init_code */
    mpfr_mul(ob_38,ob_28,ob_0,MPFR_RNDD);
    mpfr_add(tmp_5,tmp_5,ob_38,MPFR_RNDD);
    /*  $(1/12*d1*d2+1/12*d1^2)*h^3+1$  is already evaluated as
ob_35 */
    /* register  $((1/12*d1*d2+1/12*d1^2)*h^3+1)*\langle 0 \rangle$  to ob_39
in init_code */
    mpfr_mul(ob_39,ob_35,ob_3,MPFR_RNDD);
    mpfr_add(tmp_5,tmp_5,ob_39,MPFR_RNDD);

    /* register
 $1/24*d1^2*d2*d0*h^4*t^2+(1/48*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d-
1^2)*h^2)*t+(1/12*d1*d2+1/12*d1^2)*h^3+1$  to ob_40 in init_code */
    mpfr_set(ob_40,tmp_5,MPFR_RNDD);
    /*end of poly_to_mpfr(); */
    /*begin of
poly_to_mpfr( $(1/12*d1^2*d0+1/12*d1^2*d2)*h^3*t+(1/24*d1^2*d0+1/24*d1^2*-
d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h$ , [t]); */
    mpfr_set_si(tmp_40,(long) 0,MPFR_RNDD);
    if (first_2) {
        /* evaluate  $(1/12*d1^2*d0+1/12*d1^2*d2)*h^3$  */
        /* evaluate  $1/12*d1^2*d0+1/12*d1^2*d2$  */
        /* evaluate  $1/12*d1^2*d2$  */
        /*  $1/12*d1^2$  is already evaluated as ob_31
*/
        mpfr_set_si(tmp_41,(long) 0,MPFR_RNDD);

```

```

        /* tmp_41 will be set to obj 1/12*d1^2*d2 */
        mpfr_mul(tmp_42,ob_6,ob_31,MPFR_RNDD);/*
Tmp=d2^1*ob_31 coef of d2^1 */
        mpfr_add(tmp_41,tmp_41,tmp_42,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/12*d1^2*d2 to ob_41 in
init_code */
        mpfr_set(ob_41,tmp_41,MPFR_RNDD);
        /* Done of 1/12*d1^2*d2 */
        /* 1/12*d1^2 is already evaluated as ob_31 */
        mpfr_set_si(tmp_43,(long) 0,MPFR_RNDD);
        /* tmp_43 will be set to obj
1/12*d1^2*d0+1/12*d1^2*d2 */
        mpfr_mul(tmp_44,ob_3,ob_41,MPFR_RNDD);/*
Tmp=d0^0*ob_41 coef of d0^0 */
        mpfr_add(tmp_43,tmp_43,tmp_44,MPFR_RNDD); /*
Fr_F += Tmp */
        mpfr_mul(tmp_44,ob_2,ob_31,MPFR_RNDD);/*
Tmp=d0^1*ob_31 coef of d0^1 */
        mpfr_add(tmp_43,tmp_43,tmp_44,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/12*d1^2*d0+1/12*d1^2*d2 to ob_42
in init_code */
        mpfr_set(ob_42,tmp_43,MPFR_RNDD);
        /* Done of 1/12*d1^2*d0+1/12*d1^2*d2 */
        mpfr_set_si(tmp_45,(long) 0,MPFR_RNDD);
        /* tmp_45 will be set to obj
(1/12*d1^2*d0+1/12*d1^2*d2)*h^3 */
        mpfr_mul(tmp_46,ob_34,ob_42,MPFR_RNDD);/*
Tmp=h^3*ob_42 coef of h^3 */
        mpfr_add(tmp_45,tmp_45,tmp_46,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register (1/12*d1^2*d0+1/12*d1^2*d2)*h^3 to
ob_43 in init_code */
        mpfr_set(ob_43,tmp_45,MPFR_RNDD);
        /* Done of (1/12*d1^2*d0+1/12*d1^2*d2)*h^3 */
        /* evaluate
(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
        /* evaluate 1/6*d0+1/6*d2+2/3*d1 */
        /* evaluate 1/6*d2+2/3*d1 */
        /* evaluate 2/3*d1 */
        /* register 2/3 to ob_44 in init_code */
        /* register 2 to ob_45 in init_code */
        /* register 3 to ob_46 in init_code */
        mpfr_set_si(tmp_47,(long) 0,MPFR_RNDD);
        /* tmp_47 will be set to obj 2/3*d1 */
        mpfr_mul(tmp_48,ob_5,ob_44,MPFR_RNDD);/-
* Tmp=d1^1*ob_44 coef of d1^1 */
mpfr_add(tmp_47,tmp_47,tmp_48,MPFR_RNDD); /* Fr_F += Tmp */
        /* register 2/3*d1 to ob_47 in
init_code */
        mpfr_set(ob_47,tmp_47,MPFR_RNDD);

```



```

        /* Done of 2/3*d1 */
        mpfr_set_si(tmp_49,(long) 0,MPFR_RNDD);
        /* tmp_49 will be set to obj 1/6*d2+2/3*d1 */
        mpfr_mul(tmp_50,ob_3,ob_47,MPFR_RNDD);/*
Tmp=d2^0*ob_47 coef of d2^0 */
        mpfr_add(tmp_49,tmp_49,tmp_50,MPFR_RNDD); /
* Fr_F += Tmp */
        mpfr_mul(tmp_50,ob_6,ob_15,MPFR_RNDD);/*
Tmp=d2^1*ob_15 coef of d2^1 */
        mpfr_add(tmp_49,tmp_49,tmp_50,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/6*d2+2/3*d1 to ob_48 in
init_code */
        mpfr_set(ob_48,tmp_49,MPFR_RNDD);
        /* Done of 1/6*d2+2/3*d1 */
        mpfr_set_si(tmp_51,(long) 0,MPFR_RNDD);
        /* tmp_51 will be set to obj 1/6*d0+1/6*d2+2/3*d1 */
        mpfr_mul(tmp_52,ob_3,ob_48,MPFR_RNDD);/*
Tmp=d0^0*ob_48 coef of d0^0 */
        mpfr_add(tmp_51,tmp_51,tmp_52,MPFR_RNDD); /*
Fr_F += Tmp */
        mpfr_mul(tmp_52,ob_2,ob_15,MPFR_RNDD);/*
Tmp=d0^1*ob_15 coef of d0^1 */
        mpfr_add(tmp_51,tmp_51,tmp_52,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/6*d0+1/6*d2+2/3*d1 to ob_49 in
init_code */
        mpfr_set(ob_49,tmp_51,MPFR_RNDD);
        /* Done of 1/6*d0+1/6*d2+2/3*d1 */
        /* evaluate 1/24*d1^2*d0+1/24*d1^2*d2 */
        /* 1/24*d1^2*d2 is already evaluated as ob_11 */
        /* 1/24*d1^2 is already evaluated as ob_10 */
        mpfr_set_si(tmp_53,(long) 0,MPFR_RNDD);
        /* tmp_53 will be set to obj
1/24*d1^2*d0+1/24*d1^2*d2 */
        mpfr_mul(tmp_54,ob_3,ob_11,MPFR_RNDD);/*
Tmp=d0^0*ob_11 coef of d0^0 */
        mpfr_add(tmp_53,tmp_53,tmp_54,MPFR_RNDD); /*
Fr_F += Tmp */
        mpfr_mul(tmp_54,ob_2,ob_10,MPFR_RNDD);/*
Tmp=d0^1*ob_10 coef of d0^1 */
        mpfr_add(tmp_53,tmp_53,tmp_54,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/24*d1^2*d0+1/24*d1^2*d2 to ob_50
in init_code */
        mpfr_set(ob_50,tmp_53,MPFR_RNDD);
        /* Done of 1/24*d1^2*d0+1/24*d1^2*d2 */
        mpfr_set_si(tmp_55,(long) 0,MPFR_RNDD);
        /* tmp_55 will be set to obj
(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
        mpfr_mul(tmp_56,ob_1,ob_49,MPFR_RNDD);/*
Tmp=h^1*ob_49 coef of h^1 */
        mpfr_add(tmp_55,tmp_55,tmp_56,MPFR_RNDD); /* Fr_F

```

```

+= Tmp */
mpfr_mul(tmp_56,ob_13,ob_50,MPFR_RNDD);/*
Tmp=h^4*ob_50 coef of h^4 */
mpfr_add(tmp_55,tmp_55,tmp_56,MPFR_RNDD); /* Fr_F
+= Tmp */
/* register
(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h to ob_51 in
init_code */
mpfr_set(ob_51,tmp_55,MPFR_RNDD);
/* Done of
(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
first_2=0;
}

/* t is already evaluated as ob_0 */
/* (1/12*d1^2*d0+1/12*d1^2*d2)*h^3 is already evaluated as
ob_43 */
/* register ((1/12*d1^2*d0+1/12*d1^2*d2)*h^3)*<<1>> to
ob_52 in init_code */
mpfr_mul(ob_52,ob_43,ob_0,MPFR_RNDD);
mpfr_add(tmp_40,tmp_40,ob_52,MPFR_RNDD);
/* (1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h
is already evaluated as ob_51 */
/* register
((1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h)*<<0>> to
ob_53 in init_code */
mpfr_mul(ob_53,ob_51,ob_3,MPFR_RNDD);
mpfr_add(tmp_40,tmp_40,ob_53,MPFR_RNDD);

/* register
(1/12*d1^2*d0+1/12*d1^2*d2)*h^3*t+(1/24*d1^2*d0+1/24*d1^2*d2)*h^4+(1/6*
d0+1/6*d2+2/3*d1)*h to ob_54 in init_code */
mpfr_set(ob_54,tmp_40,MPFR_RNDD);
/*end of poly_to_mpfr(); */
/*begin of poly_to_mpfr((1/12*d1^2*d0+1/12*d1^2*d2)*h^3*t^2+-
((1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h)*t+1/24*d1^2*d-
2*h^5+(1/6*d2+1/3*d1)*h^2,[t]); */
mpfr_set_si(tmp_57,(long) 0,MPFR_RNDD);
if (first_3) {
/* (1/12*d1^2*d0+1/12*d1^2*d2)*h^3 is already evaluated
as ob_43 */
/* evaluate
(1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
/* 1/6*d0+1/6*d2+2/3*d1 is already evaluated as
ob_49 */
/* evaluate 1/24*d1^2*d0+1/8*d1^2*d2 */
/* evaluate 1/8*d1^2 */
/* register 1/8 to ob_55 in init_code */
/* register 8 to ob_56 in init_code */
mpfr_set_si(tmp_58,(long) 0,MPFR_RNDD);
/* tmp_58 will be set to obj 1/8*d1^2 */

```

```

mpfr_mul(tmp_59,ob_9,ob_55,MPFR_RNDD);/-
* Tmp=d1^2*ob_55 coef of d1^2 */
mpfr_add(tmp_58,tmp_58,tmp_59,MPFR_RNDD); /* Fr_F += Tmp */
/* register 1/8*d1^2 to ob_57 in
init_code */
mpfr_set(ob_57,tmp_58,MPFR_RNDD);
/* Done of 1/8*d1^2 */
mpfr_set_si(tmp_60,(long) 0,MPFR_RNDD);
/* tmp_60 will be set to obj 1/8*d1^2*d2 */
mpfr_mul(tmp_61,ob_6,ob_57,MPFR_RNDD);/*
Tmp=d2^1*ob_57 coef of d2^1 */
mpfr_add(tmp_60,tmp_60,tmp_61,MPFR_RNDD); /
* Fr_F += Tmp */
/* register 1/8*d1^2*d2 to ob_58 in
init_code */
mpfr_set(ob_58,tmp_60,MPFR_RNDD);
/* Done of 1/8*d1^2*d2 */
/* 1/24*d1^2 is already evaluated as ob_10 */
mpfr_set_si(tmp_62,(long) 0,MPFR_RNDD);
/* tmp_62 will be set to obj
1/24*d1^2*d0+1/8*d1^2*d2 */
mpfr_mul(tmp_63,ob_3,ob_58,MPFR_RNDD);/*
Tmp=d0^0*ob_58 coef of d0^0 */
mpfr_add(tmp_62,tmp_62,tmp_63,MPFR_RNDD); /*
Fr_F += Tmp */
mpfr_mul(tmp_63,ob_2,ob_10,MPFR_RNDD);/*
Tmp=d0^1*ob_10 coef of d0^1 */
mpfr_add(tmp_62,tmp_62,tmp_63,MPFR_RNDD); /*
Fr_F += Tmp */
/* register 1/24*d1^2*d0+1/8*d1^2*d2 to ob_59
in init_code */
mpfr_set(ob_59,tmp_62,MPFR_RNDD);
/* Done of 1/24*d1^2*d0+1/8*d1^2*d2 */
mpfr_set_si(tmp_64,(long) 0,MPFR_RNDD);
/* tmp_64 will be set to obj
(1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
mpfr_mul(tmp_65,ob_1,ob_49,MPFR_RNDD);/*
Tmp=h^1*ob_49 coef of h^1 */
mpfr_add(tmp_64,tmp_64,tmp_65,MPFR_RNDD); /* Fr_F
+= Tmp */
mpfr_mul(tmp_65,ob_13,ob_59,MPFR_RNDD);/*
Tmp=h^4*ob_59 coef of h^4 */
mpfr_add(tmp_64,tmp_64,tmp_65,MPFR_RNDD); /* Fr_F
+= Tmp */
/* register
(1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h to ob_60 in
init_code */
mpfr_set(ob_60,tmp_64,MPFR_RNDD);
/* Done of
(1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h */
/* evaluate 1/24*d1^2*d2*h^5+(1/6*d2+1/3*d1)*h^2 */
/* evaluate 1/6*d2+1/3*d1 */

```

```

        /* evaluate 1/3*d1 */
        /* register 1/3 to ob_61 in init_code */
        mpfr_set_si(tmp_66,(long) 0,MPFR_RNDD);
        /* tmp_66 will be set to obj 1/3*d1 */
        mpfr_mul(tmp_67,ob_5,ob_61,MPFR_RNDD);/*
Tmp=d1^1*ob_61 coef of d1^1 */
        mpfr_add(tmp_66,tmp_66,tmp_67,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/3*d1 to ob_62 in init_code */
        mpfr_set(ob_62,tmp_66,MPFR_RNDD);
        /* Done of 1/3*d1 */
        mpfr_set_si(tmp_68,(long) 0,MPFR_RNDD);
        /* tmp_68 will be set to obj 1/6*d2+1/3*d1 */
        mpfr_mul(tmp_69,ob_3,ob_62,MPFR_RNDD);/*
Tmp=d2^0*ob_62 coef of d2^0 */
        mpfr_add(tmp_68,tmp_68,tmp_69,MPFR_RNDD); /*
Fr_F += Tmp */
        mpfr_mul(tmp_69,ob_6,ob_15,MPFR_RNDD);/*
Tmp=d2^1*ob_15 coef of d2^1 */
        mpfr_add(tmp_68,tmp_68,tmp_69,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/6*d2+1/3*d1 to ob_63 in init_code
*/
        mpfr_set(ob_63,tmp_68,MPFR_RNDD);
        /* Done of 1/6*d2+1/3*d1 */
        /* 1/24*d1^2*d2 is already evaluated as ob_11 */
        mpfr_set_si(tmp_70,(long) 0,MPFR_RNDD);
        /* tmp_70 will be set to obj
1/24*d1^2*d2*h^5+(1/6*d2+1/3*d1)*h^2 */
        mpfr_mul(tmp_71,ob_26,ob_63,MPFR_RNDD);/*
Tmp=h^2*ob_63 coef of h^2 */
        mpfr_add(tmp_70,tmp_70,tmp_71,MPFR_RNDD); /* Fr_F
+= Tmp */
        mpfr_mul(tmp_71,ob_27,ob_11,MPFR_RNDD);/*
Tmp=h^5*ob_11 coef of h^5 */
        mpfr_add(tmp_70,tmp_70,tmp_71,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register 1/24*d1^2*d2*h^5+(1/6*d2+1/3*d1)*h^2 to
ob_64 in init_code */
        mpfr_set(ob_64,tmp_70,MPFR_RNDD);
        /* Done of 1/24*d1^2*d2*h^5+(1/6*d2+1/3*d1)*h^2 */
        first_3=0;
    }

    /* t^2 is already evaluated as ob_36 */
    /* (1/12*d1^2*d0+1/12*d1^2*d2)*h^3 is already evaluated as
ob_43 */
    /* register ((1/12*d1^2*d0+1/12*d1^2*d2)*h^3)*<<2>> to
ob_65 in init_code */
    mpfr_mul(ob_65,ob_43,ob_36,MPFR_RNDD);
    mpfr_add(tmp_57,tmp_57,ob_65,MPFR_RNDD);
    /* t is already evaluated as ob_0 */
    /* (1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h

```

```

is already evaluated as ob_60 */
    /* register
((1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h)*<<1>> to
ob_66 in init_code */
    mpfr_mul(ob_66,ob_60,ob_0,MPFR_RNDD);
mpfr_add(tmp_57,tmp_57,ob_66,MPFR_RNDD);
    /* 1/24*d1^2*d2*h^5+(1/6*d2+1/3*d1)*h^2 is already
evaluated as ob_64 */
    /* register (1/24*d1^2*d2*h^5+(1/6*d2+1/3*d1)*h^2)*<<0>> to
ob_67 in init_code */
    mpfr_mul(ob_67,ob_64,ob_3,MPFR_RNDD);
mpfr_add(tmp_57,tmp_57,ob_67,MPFR_RNDD);

    /* register (1/12*d1^2*d0+1/12*d1^2*d2)*h^3*t^2+-
((1/24*d1^2*d0+1/8*d1^2*d2)*h^4+(1/6*d0+1/6*d2+2/3*d1)*h)*t+1/24*d1^2*d-
2*h^5+(1/6*d2+1/3*d1)*h^2 to ob_68 in init_code */
    mpfr_set(ob_68,tmp_57,MPFR_RNDD);
    /*end of poly_to_mpfr(); */
    /*begin of
poly_to_mpfr(1/24*d1^2*d2*d0*h^4*t^2+(1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/-
6*d1*d2+1/6*d1^2)*h^2)*t+1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12-
*d1^2)*h^3+1,[t]); */
    mpfr_set_si(tmp_72,(long) 0,MPFR_RNDD);
    if (first_4) {
        /* 1/24*d1^2*d2*d0*h^4 is already evaluated as ob_14 */
        /* evaluate
1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 */
        /* 1/6*d1*d0+1/6*d1*d2+1/6*d1^2 is already
evaluated as ob_20 */
        /* evaluate 1/16*d1^2*d2*d0 */
        /* evaluate 1/16*d1^2*d2 */
        /* evaluate 1/16*d1^2 */
        /* register 1/16 to ob_69 in init_code
*/
        /* register 16 to ob_70 in init_code */
        mpfr_set_si(tmp_73,(long) 0,MPFR_RNDD);
        /* tmp_73 will be set to obj 1/16*d1^2 */
        mpfr_mul(tmp_74,ob_9,ob_69,MPFR_RNDD);/-
* Tmp=d1^2*ob_69 coef of d1^2 */

mpfr_add(tmp_73,tmp_73,tmp_74,MPFR_RNDD); /* Fr_F += Tmp */
        /* register 1/16*d1^2 to ob_71 in
init_code */
        mpfr_set(ob_71,tmp_73,MPFR_RNDD);
        /* Done of 1/16*d1^2 */
        mpfr_set_si(tmp_75,(long) 0,MPFR_RNDD);
        /* tmp_75 will be set to obj 1/16*d1^2*d2 */
        mpfr_mul(tmp_76,ob_6,ob_71,MPFR_RNDD);/*
Tmp=d2^1*ob_71 coef of d2^1 */
        mpfr_add(tmp_75,tmp_75,tmp_76,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/16*d1^2*d2 to ob_72 in

```

```

init_code */
        mpfr_set(ob_72,tmp_75,MPFR_RNDD);
        /* Done of 1/16*d1^2*d2 */
        mpfr_set_si(tmp_77,(long) 0,MPFR_RNDD);
        /* tmp_77 will be set to obj 1/16*d1^2*d2*d0 */
        mpfr_mul(tmp_78,ob_2,ob_72,MPFR_RNDD);/*
Tmp=d0^1*ob_72 coef of d0^1 */
        mpfr_add(tmp_77,tmp_77,tmp_78,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/16*d1^2*d2*d0 to ob_73 in
init_code */
        mpfr_set(ob_73,tmp_77,MPFR_RNDD);
        /* Done of 1/16*d1^2*d2*d0 */
        mpfr_set_si(tmp_79,(long) 0,MPFR_RNDD);
        /* tmp_79 will be set to obj
1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 */
        mpfr_mul(tmp_80,ob_26,ob_20,MPFR_RNDD);/*
Tmp=h^2*ob_20 coef of h^2 */
        mpfr_add(tmp_79,tmp_79,tmp_80,MPFR_RNDD); /* Fr_F
+= Tmp */
        mpfr_mul(tmp_80,ob_27,ob_73,MPFR_RNDD);/*
Tmp=h^5*ob_73 coef of h^5 */
        mpfr_add(tmp_79,tmp_79,tmp_80,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register
1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 to ob_74 in
init_code */
        mpfr_set(ob_74,tmp_79,MPFR_RNDD);
        /* Done of
1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2 */
        /* evaluate
1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1 */
        /* evaluate 1/12*d1*d0+1/6*d1*d2+1/12*d1^2 */
        /* evaluate 1/6*d1*d2+1/12*d1^2 */
        /* 1/12*d1^2 is already evaluated as ob_31
*/
        /* 1/6*d1 is already evaluated as ob_18 */
        mpfr_set_si(tmp_81,(long) 0,MPFR_RNDD);
        /* tmp_81 will be set to obj
1/6*d1*d2+1/12*d1^2 */
        mpfr_mul(tmp_82,ob_3,ob_31,MPFR_RNDD);/*
Tmp=d2^0*ob_31 coef of d2^0 */
        mpfr_add(tmp_81,tmp_81,tmp_82,MPFR_RNDD); /
* Fr_F += Tmp */
        mpfr_mul(tmp_82,ob_6,ob_18,MPFR_RNDD);/*
Tmp=d2^1*ob_18 coef of d2^1 */
        mpfr_add(tmp_81,tmp_81,tmp_82,MPFR_RNDD); /
* Fr_F += Tmp */
        /* register 1/6*d1*d2+1/12*d1^2 to ob_75 in
init_code */
        mpfr_set(ob_75,tmp_81,MPFR_RNDD);
        /* Done of 1/6*d1*d2+1/12*d1^2 */
        /* 1/12*d1 is already evaluated as ob_32 */

```

```

        mpfr_set_si(tmp_83,(long) 0,MPFR_RNDD);
        /* tmp_83 will be set to obj
1/12*d1*d0+1/6*d1*d2+1/12*d1^2 */
        mpfr_mul(tmp_84,ob_3,ob_75,MPFR_RNDD);/*
Tmp=d0^0*ob_75 coef of d0^0 */
        mpfr_add(tmp_83,tmp_83,tmp_84,MPFR_RNDD); /*
Fr_F += Tmp */
        mpfr_mul(tmp_84,ob_2,ob_32,MPFR_RNDD);/*
Tmp=d0^1*ob_32 coef of d0^1 */
        mpfr_add(tmp_83,tmp_83,tmp_84,MPFR_RNDD); /*
Fr_F += Tmp */
        /* register 1/12*d1*d0+1/6*d1*d2+1/12*d1^2 to
ob_76 in init_code */
        mpfr_set(ob_76,tmp_83,MPFR_RNDD);
        /* Done of 1/12*d1*d0+1/6*d1*d2+1/12*d1^2 */
        /* 1/48*d1^2*d2*d0 is already evaluated as ob_25 */
        mpfr_set_si(tmp_85,(long) 0,MPFR_RNDD);
        /* tmp_85 will be set to obj
1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1 */
        mpfr_mul(tmp_86,ob_3,ob_3,MPFR_RNDD);/*
Tmp=h^0*ob_3 coef of h^0 */
        mpfr_add(tmp_85,tmp_85,tmp_86,MPFR_RNDD); /* Fr_F
+= Tmp */
        mpfr_mul(tmp_86,ob_34,ob_76,MPFR_RNDD);/*
Tmp=h^3*ob_76 coef of h^3 */
        mpfr_add(tmp_85,tmp_85,tmp_86,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register h^6 to ob_77 in init_code */
        mpfr_pow_ui(ob_77,ob_1,6,MPFR_RNDD);
        mpfr_mul(tmp_86,ob_77,ob_25,MPFR_RNDD);/*
Tmp=h^6*ob_25 coef of h^6 */
        mpfr_add(tmp_85,tmp_85,tmp_86,MPFR_RNDD); /* Fr_F
+= Tmp */
        /* register
1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1 to ob_78 in
init_code */
        mpfr_set(ob_78,tmp_85,MPFR_RNDD);
        /* Done of
1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1 */
        first_4=0;
    }

    /* t^2 is already evaluated as ob_36 */
    /* 1/24*d1^2*d2*d0*h^4 is already evaluated as ob_14 */
    mpfr_add(tmp_72,tmp_72,ob_37,MPFR_RNDD);
    /* t is already evaluated as ob_0 */
    /* 1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2
is already evaluated as ob_74 */
    /* register
(1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d1^2)*h^2)*<<1>> to ob_79
in init_code */
    mpfr_mul(ob_79,ob_74,ob_0,MPFR_RNDD);
    mpfr_add(tmp_72,tmp_72,ob_79,MPFR_RNDD);

```

```

        /*
1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1 is already
evaluated as ob_78 */
        /* register
(1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1)*<<0>> to
ob_80 in init_code */
        mpfr_mul(ob_80,ob_78,ob_3,MPFR_RNDD);
mpfr_add(tmp_72,tmp_72,ob_80,MPFR_RNDD);

        /* register
1/24*d1^2*d2*d0*h^4*t^2+(1/16*d1^2*d2*d0*h^5+(1/6*d1*d0+1/6*d1*d2+1/6*d-
1^2)*h^2)*t+1/48*d1^2*d2*d0*h^6+(1/12*d1*d0+1/6*d1*d2+1/12*d1^2)*h^3+1
to ob_81 in init_code */
        mpfr_set(ob_81,tmp_72,MPFR_RNDD);
        /*end of poly_to_mpfr(); */
        /* multiplier M(t)=[[ob_40,ob_54],[ob_68,ob_81]] */
        mpfr_set_si(tmp_87,(long) 0,MPFR_RNDD);
        /* M*P = [[ob_40*tmp_0+ob_54*tmp_2,ob_40*tmp_1+ob_54*tmp_3],-
[ob_68*tmp_0+ob_81*tmp_2,ob_68*tmp_1+ob_81*tmp_3]], tmp_87 is a work
area to multiply */
        /* Evaluate [0,0] term ob_40*tmp_0+ob_54*tmp_2 */
        /* register ob_40*tmp_0+ob_54*tmp_2 to ob_82 in init_code */
        mpfr_set_si(ob_82,(long) 0, MPFR_RNDD); /* Put 0 to sum up
*/

        mpfr_mul(tmp_87,ob_40,tmp_0,MPFR_RNDD);
        mpfr_add(ob_82,ob_82,tmp_87,MPFR_RNDD);
        mpfr_mul(tmp_87,ob_54,tmp_2,MPFR_RNDD);
        mpfr_add(ob_82,ob_82,tmp_87,MPFR_RNDD);
        /* Evaluate [0,1] term ob_40*tmp_1+ob_54*tmp_3 */
        /* register ob_40*tmp_1+ob_54*tmp_3 to ob_83 in init_code */
        mpfr_set_si(ob_83,(long) 0, MPFR_RNDD); /* Put 0 to sum up
*/

        mpfr_mul(tmp_87,ob_40,tmp_1,MPFR_RNDD);
        mpfr_add(ob_83,ob_83,tmp_87,MPFR_RNDD);
        mpfr_mul(tmp_87,ob_54,tmp_3,MPFR_RNDD);
        mpfr_add(ob_83,ob_83,tmp_87,MPFR_RNDD);
        /* Evaluate [1,0] term ob_68*tmp_0+ob_81*tmp_2 */
        /* register ob_68*tmp_0+ob_81*tmp_2 to ob_84 in init_code */
        mpfr_set_si(ob_84,(long) 0, MPFR_RNDD); /* Put 0 to sum up
*/

        mpfr_mul(tmp_87,ob_68,tmp_0,MPFR_RNDD);
        mpfr_add(ob_84,ob_84,tmp_87,MPFR_RNDD);
        mpfr_mul(tmp_87,ob_81,tmp_2,MPFR_RNDD);
        mpfr_add(ob_84,ob_84,tmp_87,MPFR_RNDD);
        /* Evaluate [1,1] term ob_68*tmp_1+ob_81*tmp_3 */
        /* register ob_68*tmp_1+ob_81*tmp_3 to ob_85 in init_code */
        mpfr_set_si(ob_85,(long) 0, MPFR_RNDD); /* Put 0 to sum up
*/

        mpfr_mul(tmp_87,ob_68,tmp_1,MPFR_RNDD);
        mpfr_add(ob_85,ob_85,tmp_87,MPFR_RNDD);
        mpfr_mul(tmp_87,ob_81,tmp_3,MPFR_RNDD);
        mpfr_add(ob_85,ob_85,tmp_87,MPFR_RNDD);

```



```

        /*set_mat_mpfr_mpfr*/
mpfr_set(tmp_0,ob_82,MPFR_RNDD);mpfr_set(tmp_1,ob_83,MPFR_RNDD);mpfr_set
t(tmp_2,ob_84,MPFR_RNDD);mpfr_set(tmp_3,ob_85,MPFR_RNDD);
        mpfr_add(ob_0,ob_0,ob_1,MPFR_RNDD); /* t<-t+h */
    } /* end for */
    mpfr_set(ans[0*N+0],tmp_0,MPFR_RNDD);
    mpfr_set(ans[0*N+1],tmp_1,MPFR_RNDD);
    mpfr_set(ans[1*N+0],tmp_2,MPFR_RNDD);
    mpfr_set(ans[1*N+1],tmp_3,MPFR_RNDD);
    mpfr_set(argv[0],ob_0,MPFR_RNDD); /* argv[0]<- t_last */

    mpfr_set(argv[1],ob_1,MPFR_RNDD); /* argv[1]<- h */

    *argcv = 2;

    return(0);
}

/* $OpenXM: OpenXM/src/asir-contrib/packages/src/tk_any2mpfr/-
solve_ode_by_r4d.c,v 1.2 2021/03/17 11:22:32 takayama Exp $ */
int T_verbose=0;
void usage() {
    printf("%s [--rank r --t0 t0 --init f1 ... fr --t1
t1]\n",T_PROGNAME);
    printf("        [--h h --n_defuse n_defuse]\n");
    printf("        [--n_prune n_prune --strategy strat --t_noproj t_noproj]-
\n");
    printf("        [--ref_value_file ref_file]\n");
    printf("        [--verbose --go]\n");
}
void usage_example(int type,double t0,double f0[],double t1,double
h,double t_noproj,int r, int strat,int n_prune,int n_defuse,char
*ref_file) {
    int i;
    if (type==1) printf("Arguments: ");
    else printf("Example: ");
    printf(" %s --rank %d --t0 %g --init ",T_PROGNAME,r,t0);
    for (i=0; i<r; i++) {
        printf("%g ",f0[i]);
    }
    printf(" --t1 %g [ --h %g --n_defuse %d --n_prune %d --t_noproj %g --
strategy %d --ref_value_file %s --verbose ]
\n",t1,h,n_defuse,n_prune,t_noproj,strat,ref_file);
    printf("If you want to run this program with values above, type in
%s --go\n",T_PROGNAME);
}
void usage_draw_graph() {
    printf("--- To draw a graph of the solution, do\n");
    printf("This_program | grep '^gnuplot' | awk '{print $2, $3}'
>t.txt\n");
    printf("In the gnuplot, plot 't.txt' w lp\n");
}

```

```
void output_tf(double t,double f[]) {
    int i;
    printf("t=%g, ",t);
    for (i=0; i<N; i++) printf("f[%d]=%g, ",i,f[i]); putchar('\n');
    printf("gnuplot %g %g\n",t,f[0]);
}
void outout_mpfr_vec(mpfr_t m_ans[],int size) {
    int i;
    int m;
    m = sqrt(size);
    for (i=0; i<size; i++) {
        mpfr_out_str(stdout,10,0,m_ans[i],MPFR_RNDD);putchar(',');
        if (i % m == m-1) putchar('\n');
    }
}
int main(int argc,char *argv[]) {
    double t0=T_T0; // 初期時刻
    double f0[N]=T_F0; // 初期条件 gauge.rr より
    double t1=T_T1; // 終了時刻
    double h=T_H; // 微小ステップ
    double t_noproj=T_NOPROJ; // この時刻まではprojection しない
    int r=N; // ODE の rank
    int strat=T_STRAT; // projection の strategy
    int n_prune=T_N_PRUNE; /* 最大の Re eigen_val を除く */
    int n_defuse=T_N_DEFUSE; // matrix factorial の長さ.
    char ref_value_file[4096]=T_REF_VALUE_FILE; // 補間すべき正しい値

    int show_help=0;

    mpfr_t m_t; double t;
    mpfr_t m_h;
    mpfr_t m_ans[N*N];
    int m_argc=2;
    mpfr_t m_argv[2];
    int i,j;
    double f[N];
    double f_new[N];
    double data[N*N]; // 変換行列
    double proj_mat[N][N];
    double mat_step[N][N];
    if (argc <2) { show_help=1; }
    for (i=1; i<argc; i++) {
        if (strcmp(argv[i],"--rank")==0) {i++; sscanf(argv[i],"%d",&r);if
(r != N) myerror("rank != N"); continue;}
        if (strcmp(argv[i],"--t0")==0) {i++;
sscanf(argv[i],"%lg",&t0);continue;}
        if (strcmp(argv[i],"--init")==0) {
            for (j=0; j<r; j++) {
                i++;
                sscanf(argv[i],"%lg",&(f0[j]));
            }
            continue;
        }
    }
}
```

```

    }
    if (strcmp(argv[i],"--t1")==0) {i++;
sscanf(argv[i],"%lg",&t1);continue;}
    if (strcmp(argv[i],"--h")==0) {i++;
sscanf(argv[i],"%lg",&h);continue;}
    if (strcmp(argv[i],"--n_defuse")==0) {i++;
sscanf(argv[i],"%d",&n_defuse);continue;}
    if (strcmp(argv[i],"--n_prune")==0) {i++;
sscanf(argv[i],"%d",&n_prune);continue;}
    if (strcmp(argv[i],"--strategy")==0) {i++;
sscanf(argv[i],"%d",&strat);continue;}
    if (strcmp(argv[i],"--t_noproj")==0) {i++;
sscanf(argv[i],"%lg",&t_noproj);continue;}
    if (strcmp(argv[i],"--verbose")==0) {T_verbose=1; continue;}
    if (strcmp(argv[i],"--help")==0) { show_help=1; continue; }
    if (strcmp(argv[i],"--go")==0) { continue; }
    printf("Error: Unknown option %s.\n",argv[i]); usage(); return(-1);
}
if (show_help) {
    usage();
    usage_draw_graph(); printf("\n");
    usage_example(0, t0, f0, t1, h, t_noproj, r, strat, n_prune,
n_defuse,ref_value_file);
    return 0;
}
if (T_verbose) {
    usage_example(1, t0, f0, t1, h, t_noproj, r, strat, n_prune,
n_defuse,ref_value_file);
}

mpfr_init2(m_t,MPFR_PREC);
mpfr_init2(m_h,MPFR_PREC);
t=t0;
mpfr_set_d(m_t,t,MPFR_RNDD);
mpfr_set_d(m_h,h,MPFR_RNDD);
for (i=0; i<N*N; i++) mpfr_init2(m_ans[i],MPFR_PREC);
for (i=0; i<2; i++) mpfr_init2(m_argv[i],MPFR_PREC);

for (i=0; i<N; i++) f[i]=f0[i];
while (t < t1) {
    output_tf(t,f);
    mat_fac(m_t,m_h,n_defuse,m_ans,&m_argc,m_argv); // matrix factorial
    if (T_verbose) outout_mpfr_vec(m_ans,N*N);
    mat_get_dvec(data,m_ans); // data に変換行列
    cp_vmatm(data,mat_step); // mat_step として保存。
    printf("matrix factorial=\n"); output_mat(mat_step);
    t = mpfr_get_d(m_argv[0],MPFR_RNDD); // 次の t.

    if (t<=t_noproj) for (i=0; i<N; i++) f_new[i]=f[i]; // projection 無
    し。
    else {
        projection_matrix_to_subspace_of(n_prune,data,proj_mat);
        // Warning. data は変更される

```

```
    mat_linear_transformation(proj_mat,f,f_new);
    printf("f_init="); output_vec(f,N);
    printf("\nf_new_init="); output_vec(f_new,N); printf("\n");
    if (strat == 1) {
調整      vec_scalar_multiplication(f[0]/f_new[0],f_new,f_new); // 初期値の
\n");      printf("f_new_init="); output_vec(f_new,N); printf(" (scaled)-
\n");
    }else if (strat == 2) {
        printf("strategy 2 has not yet been implemented.\n");
        /* 答えを与えて初期値を調整する 補完 strategy はまだ.
        Todo, the initial value should be modified in the eigen-
space so that
        it attains the target value.
        Data in the file ref_value_file will be used.
        */
    }
}

    mat_linear_transformation(mat_step,f_new,f); // 新しい t での解.
    mpfr_set_d(m_t,t,MPFR_RNDD);
}
if (T_verbose) {
    usage_draw_graph();
}
return 0;
}
```