POMA による計算例

○計算例 1（HSQC 位相サイクル付き）

INPUT FILE

```
<<Poma2.m

nucleus[1]="I"
nucleus[2]="S"

tau=1/(4 j[1,2])

p1= { x,-x, x,-x, x,-x, x,-x}
p2= { x, x,-x,-x, x, x,-x,-x}
p3= { y, y, y, y,-y,-y,-y,-y}
rec={ x,-x,-x, x, x,-x,-x, x}
coupl={{1,2}}

spin[1,z] //
pulse[90,x,{1}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y,{1}] // pulse[90,p1,{2}] //
show["After INEPT"] //
   delay[t1/2,coupl] //
pulse[180,p3,{1}] //
   delay[t1/2,coupl] //
pulse[90,x,{1}] // pulse[90,p2,{2}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
receiver[rec] //
show["Final"] //
observable
```

OUTPUT FILE

```
-------------------------------------------------------
POMA 2.0 - Product Operator Formalism in Mathematica

Copyright (c) 2002 Peter Guntert
All rights reserved.
-------------------------------------------------------

pulse[90,x,{1}].....
   (1 term, 0.002107 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (4 terms, 0.002731 s CPU time)
pulse[180,x,allspins].....
   (4 terms, 0.007108 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
```

```
   (1 term, 0.002238 s CPU time)
pulse[90,y,{1}].....
   (1 term, 0.001856 s CPU time)
pulse[90,{x, -x, x, -x, x, -x, x, -x},{2}].....
   (8 terms, 0.009806 s CPU time)

==============================After INEPT==============================
{-2 I1z S2y, 2 I1z S2y, -2 I1z S2y, 2 I1z S2y, -2 I1z S2y, 2 I1z S2y,

>     -2 I1z S2y, 2 I1z S2y}

delay[t1/2,{{1, 2}},allspins].....
   (32 terms, 0.013406 s CPU time)
pulse[180,{y, y, y, y, -y, -y, -y, -y},{1}].....
   (32 terms, 0.038518 s CPU time)
delay[t1/2,{{1, 2}},allspins].....
   (16 terms, 0.017667 s CPU time)
pulse[90,x,{1}].....
   (16 terms, 0.016639 s CPU time)
pulse[90,{x, x, -x, -x, x, x, -x, -x},{2}].....
   (16 terms, 0.017458 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....
   (64 terms, 0.046483 s CPU time)
pulse[180,x,allspins].....
   (64 terms, 0.124125 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....
   (16 terms, 0.033393 s CPU time)
receiver[{x, -x, -x, x, x, -x, -x, x},allspins].....
   (1 term, 0.006609 s CPU time)

================================Final================================
Cos[t1 w2] I1x

observable[allspins]..
   (1 term, 0.000527 s CPU time)

Out[8]= Cos[t1 w2] I1x
```

○計算例 2（HSQC 位相サイクル付き、States-TPPI 2 スキャン目）

INPUT FILE

```
<<Poma2.m

nucleus[1]="I"
nucleus[2]="S"

tau=1/(4 j[1,2])

p1= { y,-y, y,-y, y,-y, y,-y}
p2= { x, x,-x,-x, x, x,-x,-x}
p3= { y, y, y, y,-y,-y,-y,-y}
rec={ x,-x,-x, x, x,-x,-x, x}
coupl={{1,2}}
```

```
spin[1,z] //
pulse[90,x,{1}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y,{1}] // pulse[90,p1,{2}] //
show["After INEPT"] //
   delay[t1/2,coupl] //
pulse[180,p3,{1}] //
   delay[t1/2,coupl] //
pulse[90,x,{1}] // pulse[90,p2,{2}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
receiver[rec] //
show["Final"] //
observable
```

OUTPUT FILE

```
-------------------------------------------------------
POMA 2.0 - Product Operator Formalism in Mathematica

Copyright (c) 2002 Peter Guntert
All rights reserved.
-------------------------------------------------------

pulse[90,x,{1}]......
   (1 term, 0.002035 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (4 terms, 0.002926 s CPU time)
pulse[180,x,allspins]......
   (4 terms, 0.007114 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (1 term, 0.002481 s CPU time)
pulse[90,y,{1}]......
   (1 term, 0.001859 s CPU time)
pulse[90,{y, -y, y, -y, y, -y, y, -y},{2}].....................................
   (8 terms, 0.010219 s CPU time)

================================ After INEPT ================================
{2 I1z S2x, -2 I1z S2x, 2 I1z S2x, -2 I1z S2x, 2 I1z S2x, -2 I1z S2x,

>     2 I1z S2x, -2 I1z S2x}

delay[t1/2,{{1, 2}},allspins].....................
   (32 terms, 0.013021 s CPU time)
pulse[180,{y, y, y, y, -y, -y, -y, -y},{1}].....................................................
   (32 terms, 0.038614 s CPU time)
delay[t1/2,{{1, 2}},allspins].....................
   (16 terms, 0.017954 s CPU time)
pulse[90,x,{1}].................................................
   (16 terms, 0.01636 s CPU time)
pulse[90,{x, x, -x, -x, x, x, -x, -x},{2}].....................................................
```

```
   (16 terms, 0.017541 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....
   (64 terms, 0.046555 s CPU time)
pulse[180,x,allspins].....
   (64 terms, 0.123711 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....
   (16 terms, 0.033292 s CPU time)
receiver[{x, -x, -x, x, x, -x, -x, x},allspins].....
   (1 term, 0.006604 s CPU time)

================================= Final ================================
-(Sin[t1 w2] I1x)

observable[allspins]..
   (1 term, 0.000529 s CPU time)

Out[10]= -(Sin[t1 w2] I1x)
```

○計算例 3-1 (HSQC gradient coherence selection 1 スキャン目)

INPUT FILE

```
<<Poma2.m

nucleus[1]="I"
nucleus[2]="S"
tau=1/(4 j[1,2])

p1= { x,-x, x,-x, x,-x, x,-x}
p2= { x, x,-x,-x, x, x,-x,-x}
p3= { y, y, y, y,-y,-y,-y,-y}
rec={-x, x, x,-x,-x, x, x,-x}
coupl={{1,2}}

spin[1,z] //
pulse[90,x,{1}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y,{1}] // pulse[90,p1,{2}] //
show["After INEPT"] //
   delay[t1/2,coupl] //
pulse[180,p3,{1}] //
   delay[t1/2,coupl] //
   gradient[G1] //
   delay[d1,coupl] //
pulse[180,x,{2}] //
   delay[d1,coupl] //
pulse[90,x,{1}] // pulse[90,p2,{2}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
   gradient[-G1 g[2]/g[1]] //
receiver[rec] //
```

```
dephase //
show["Final"] //
observable
```

OUTPUT FILE

```
-----------------------------------------------------
POMA 2.0 - Product Operator Formalism in Mathematica

Copyright (c) 2002 Peter Guntert
All rights reserved.
-----------------------------------------------------

pulse[90,x,{1}].....
   (1 term, 0.002525 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (4 terms, 0.002866 s CPU time)
pulse[180,x,allspins].....
   (4 terms, 0.007135 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (1 term, 0.002289 s CPU time)
pulse[90,y,{1}].....
   (1 term, 0.001842 s CPU time)
pulse[90,{x, -x, x, -x, x, -x, x, -x},{2}].....................................
   (8 terms, 0.009805 s CPU time)

================================ After INEPT ================================
{-2 I1z S2y, 2 I1z S2y, -2 I1z S2y, 2 I1z S2y, -2 I1z S2y, 2 I1z S2y,

>     -2 I1z S2y, 2 I1z S2y}

delay[t1/2,{{1, 2}},allspins].....................
   (32 terms, 0.013288 s CPU time)
pulse[180,{y, y, y, y, -y, -y, -y, -y},{1}].....................................................
   (32 terms, 0.038476 s CPU time)
delay[t1/2,{{1, 2}},allspins].....................
   (16 terms, 0.017735 s CPU time)
gradient[G1].....................................................
   (16 terms, 0.022108 s CPU time)
delay[d1,{{1, 2}},allspins].....................
   (32 terms, 0.019595 s CPU time)
pulse[180,x,{2}].....................................................
   (32 terms, 0.035757 s CPU time)
delay[d1,{{1, 2}},allspins].....................
   (16 terms, 0.023597 s CPU time)
pulse[90,x,{1}].....................................................
   (16 terms, 0.018536 s CPU time)
pulse[90,{x, x, -x, -x, x, x, -x, -x},{2}].....................................................
   (16 terms, 0.020655 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....................
   (64 terms, 0.050523 s CPU time)
pulse[180,x,allspins].....................................................
   (64 terms, 0.136625 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....................
   (16 terms, 0.035692 s CPU time)
```

```
gradient[-((G1*g[2])/g[1])].....................................................
   (48 terms, 0.05555 s CPU time)
receiver[{-x, x, x, -x, -x, x, x, -x},allspins].....................
   (2 terms, 0.029087 s CPU time)
dephase....
   (2 terms, 0.003768 s CPU time)

================================== Final ==================================
Cos[t1 w2] I1x     Sin[t1 w2] I1y
--------------------- + ---------------------
         2                        2

observable[allspins]..
   (2 terms, 0.00065 s CPU time)

               Cos[t1 w2] I1x     Sin[t1 w2] I1y
Out[11]= --------------------- + ---------------------
                     2                        2
```

○計算例 3-2 (HSQC gradient coherence selection 2 スキャン目)

INPUT FILE

```
<<Poma2.m

nucleus[1]="I"
nucleus[2]="S"
tau=1/(4 j[1,2])

p1= { x,-x, x,-x, x,-x, x,-x}
p2= { x, x,-x,-x, x, x,-x,-x}
p3= { y, y, y, y,-y,-y,-y,-y}
rec={-x, x, x,-x,-x, x, x,-x}
coupl={{1,2}}

spin[1,z] //
pulse[90,x,{1}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y,{1}] // pulse[90,p1,{2}] //
show["After INEPT"] //
   delay[t1/2,coupl] //
pulse[180,p3,{1}] //
   delay[t1/2,coupl] //
   gradient[-G1] //
   delay[d1,coupl] //
pulse[180,x,{2}] //
   delay[d1,coupl] //
pulse[90,x,{1}] // pulse[90,p2,{2}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
   gradient[-G1 g[2]/g[1]] //
```

```
receiver[rec] //
dephase //
show["Final"] //
observable

OUTPUT FILE
-----------------------------------------------------
POMA 2.0 - Product Operator Formalism in Mathematica

Copyright (c) 2002 Peter Guntert
All rights reserved.
-----------------------------------------------------

pulse[90,x,{1}].....
   (1 term, 0.002474 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (4 terms, 0.002743 s CPU time)
pulse[180,x,allspins].....
   (4 terms, 0.007257 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (1 term, 0.002325 s CPU time)
pulse[90,y,{1}].....
   (1 term, 0.00183 s CPU time)
pulse[90,{x, -x, x, -x, x, -x, x, -x},{2}].....................................
   (8 terms, 0.010186 s CPU time)

================================ After INEPT ================================
{-2 I1z S2y, 2 I1z S2y, -2 I1z S2y, 2 I1z S2y, -2 I1z S2y, 2 I1z S2y,

>     -2 I1z S2y, 2 I1z S2y}

delay[t1/2,{{1, 2}},allspins].....................
   (32 terms, 0.012873 s CPU time)
pulse[180,{y, y, y, y, -y, -y, -y, -y},{1}].....................................................
   (32 terms, 0.038595 s CPU time)
delay[t1/2,{{1, 2}},allspins].....................
   (16 terms, 0.018109 s CPU time)
gradient[-G1].................................................
   (16 terms, 0.023496 s CPU time)
delay[d1,{{1, 2}},allspins].....................
   (32 terms, 0.023641 s CPU time)
pulse[180,x,{2}].................................................
   (32 terms, 0.042302 s CPU time)
delay[d1,{{1, 2}},allspins].....................
   (16 terms, 0.02386 s CPU time)
pulse[90,x,{1}].................................................
   (16 terms, 0.020842 s CPU time)
pulse[90,{x, x, -x, -x, x, x, -x, -x},{2}].....................................................
   (16 terms, 0.022544 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....................
   (64 terms, 0.052614 s CPU time)
pulse[180,x,allspins].....................................................
   (64 terms, 0.145458 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins].....................
```

```
   (16 terms, 0.036422 s CPU time)
gradient[-((G1*g[2])/g[1])].....................................................
   (48 terms, 0.060513 s CPU time)
receiver[{-x, x, x, -x, -x, x, x, -x},allspins]...................
   (2 terms, 0.031296 s CPU time)
dephase....
   (2 terms, 0.00403 s CPU time)

================================== Final ================================
Cos[t1 w2] I1x     Sin[t1 w2] I1y
---------------------- - ----------------------
         2                          2

observable[allspins]..
   (2 terms, 0.000652 s CPU time)

               Cos[t1 w2] I1x     Sin[t1 w2] I1y
Out[12]= ---------------------- - ----------------------
                     2                          2
```

○計算例 4-1 (gradient sensitivity enhanced HSQC 1 スキャン目)

INPUT FILE

```
<<Poma2.m

nucleus[1]="I"
nucleus[2]="S"
tau=1/(4 j[1,2])

coupl={{1,2}}

spin[1,z] //
pulse[90,x,{1}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y,{1}] // pulse[90,x,{2}] //
show["After INEPT"] //
   delay[t1/2,coupl] //
pulse[180,x,{1}] //
   delay[t1/2,coupl] //
   delay[d1,coupl] //
pulse[180,x,{2}] //
   gradient[G1] //
   delay[d1,coupl] //
pulse[90,x,{1}] // pulse[90,x,{2}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
```

```
pulse[90,x,{1}] //
   delay[d2,coupl] //
   gradient[G1 g[2]/g[1]] //
pulse[180,x,{1}] //
   delay[d2,coupl] //
receiver[-x] //
dephase //
show["Final"] //
observable
```

OUTPUT FILE

```
-----------------------------------------------------
POMA 2.0 - Product Operator Formalism in Mathematica

Copyright (c) 2002 Peter Guntert
All rights reserved.
-----------------------------------------------------

pulse[90,x,{1}].....
   (1 term, 0.001801 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (4 terms, 0.002909 s CPU time)
pulse[180,x,allspins].....
   (4 terms, 0.007466 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (1 term, 0.002283 s CPU time)
pulse[90,y,{1}].....
   (1 term, 0.001861 s CPU time)
pulse[90,x,{2}].....
   (1 term, 0.001816 s CPU time)

============================== After INEPT ==============================
-2 I1z S2y

delay[t1/2,{{1, 2}},allspins]...
   (4 terms, 0.001809 s CPU time)
pulse[180,x,{1}].....
   (4 terms, 0.005203 s CPU time)
delay[t1/2,{{1, 2}},allspins]...
   (2 terms, 0.002581 s CPU time)
delay[d1,{{1, 2}},allspins]...
   (4 terms, 0.002438 s CPU time)
pulse[180,x,{2}].....
   (4 terms, 0.004257 s CPU time)
gradient[G1].....
   (4 terms, 0.006851 s CPU time)
delay[d1,{{1, 2}},allspins]...
   (2 terms, 0.003127 s CPU time)
pulse[90,x,{1}].....
   (2 terms, 0.002958 s CPU time)
pulse[90,x,{2}].....
   (2 terms, 0.002959 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (8 terms, 0.006796 s CPU time)
```

```
pulse[180,x,allspins]......
   (8 terms, 0.019159 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (2 terms, 0.004756 s CPU time)
pulse[90,y,allspins]......
   (2 terms, 0.002802 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (5 terms, 0.004084 s CPU time)
pulse[180,x,allspins]......
   (5 terms, 0.011579 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (2 terms, 0.003125 s CPU time)
pulse[90,x,{1}]......
   (2 terms, 0.002756 s CPU time)
delay[d2,{{1, 2}},allspins]...
   (4 terms, 0.003025 s CPU time)
gradient[(G1*g[2])/g[1]]......
   (4 terms, 0.007589 s CPU time)
pulse[180,x,{1}]......
   (4 terms, 0.004253 s CPU time)
delay[d2,{{1, 2}},allspins]...
   (2 terms, 0.002643 s CPU time)
receiver[-x,allspins]....
   (2 terms, 0.001555 s CPU time)
dephase....
   (2 terms, 0.001204 s CPU time)

================================ Final ================================
Sin[t1 w2] I1x + Cos[t1 w2] I1y

observable[allspins]..
   (2 terms, 0.000626 s CPU time)

Out[13]= Sin[t1 w2] I1x + Cos[t1 w2] I1y
```

○計算例 4-2 (gradient sensitivity enhanced HSQC 2 スキャン目)

INPUT FILE

```
<<Poma2.m

nucleus[1]="I"
nucleus[2]="S"
tau=1/(4 j[1,2])

coupl={{1,2}}

spin[1,z] //
pulse[90,x,{1}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y,{1}] // pulse[90,x,{2}] //
show["After INEPT"] //
```

```
   delay[t1/2,coupl] //
pulse[180,x,{1}] //
   delay[t1/2,coupl] //
   delay[d1,coupl] //
pulse[180,x,{2}] //
   gradient[G1] //
   delay[d1,coupl] //
pulse[90,x,{1}] // pulse[90,-x,{2}] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,y] //
   delay[tau,coupl] //
pulse[180,x] //
   delay[tau,coupl] //
pulse[90,x,{1}] //
   delay[d2,coupl] //
   gradient[-G1 g[2]/g[1]] //
pulse[180,x,{1}] //
   delay[d2,coupl] //
receiver[-x] //
dephase //
show["Final"] //
observable
```

OUTPUT FILE

```
-----------------------------------------------------
POMA 2.0 - Product Operator Formalism in Mathematica

Copyright (c) 2002 Peter Guntert
All rights reserved.
-----------------------------------------------------

pulse[90,x,{1}].....
   (1 term, 0.00239 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (4 terms, 0.002711 s CPU time)
pulse[180,x,allspins].....
   (4 terms, 0.007285 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
   (1 term, 0.002279 s CPU time)
pulse[90,y,{1}].....
   (1 term, 0.001893 s CPU time)
pulse[90,x,{2}].....
   (1 term, 0.001841 s CPU time)

================================ After INEPT ================================
-2 I1z S2y

delay[t1/2,{{1, 2}},allspins]...
   (4 terms, 0.001825 s CPU time)
pulse[180,x,{1}].....
   (4 terms, 0.004905 s CPU time)
delay[t1/2,{{1, 2}},allspins]...
```

(2 terms, 0.002404 s CPU time)
delay[d1,{{1, 2}},allspins]...
(4 terms, 0.002839 s CPU time)
pulse[180,x,{2}].....
(4 terms, 0.004422 s CPU time)
gradient[G1].....
(4 terms, 0.006834 s CPU time)
delay[d1,{{1, 2}},allspins]...
(2 terms, 0.003194 s CPU time)
pulse[90,x,{1}].....
(2 terms, 0.002987 s CPU time)
pulse[90,-x,{2}].....
(2 terms, 0.002976 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
(8 terms, 0.006729 s CPU time)
pulse[180,x,allspins].....
(8 terms, 0.018959 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
(2 terms, 0.004745 s CPU time)
pulse[90,y,allspins].....
(2 terms, 0.002828 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
(5 terms, 0.004138 s CPU time)
pulse[180,x,allspins].....
(5 terms, 0.011357 s CPU time)
delay[1/(4*j[1, 2]),{{1, 2}},allspins]...
(2 terms, 0.003319 s CPU time)
pulse[90,x,{1}].....
(2 terms, 0.002668 s CPU time)
delay[d2,{{1, 2}},allspins]...
(4 terms, 0.002819 s CPU time)
gradient[-((G1*g[2])/g[1])].....
(4 terms, 0.007588 s CPU time)
pulse[180,x,{1}].....
(4 terms, 0.004775 s CPU time)
delay[d2,{{1, 2}},allspins]...
(2 terms, 0.0026 s CPU time)
receiver[-x,allspins]....
(2 terms, 0.001559 s CPU time)
dephase....
(2 terms, 0.001213 s CPU time)

================================== Final ==================================
Sin[t1 w2] I1x - Cos[t1 w2] I1y

observable[allspins]..
(2 terms, 0.000638 s CPU time)

Out[14]= Sin[t1 w2] I1x - Cos[t1 w2] I1y