How to use programs counting fixed points in D_n under given permutation of input variables

Bartłomiej Pawelski

Programs use the external library JavaEWAH (GitHub) to work with downsets. Programs also use common helper classes: `Cycle.java` and `Poset.java` written by me.

1 Algorithm 1

1.1 Without printing MBFs

Compile file `Alg1a.java` using command `javac Alg1a.java`. To run compiled program use command `java Alg1a` with elements of cycle type as parameters. For example, if you want to find the number of fixed points of D_4 under permutation $\pi = (1\ 2)(3\ 4)$ use command `java Alg1a 2 2`. Moreover, you can use 1–cycles here: for example, to calculate the number of fixed points of D_6 under permutation $\pi = (1\ 2)(3\ 4)$, you can use the command `java Alg1b 2 2 1 1`.

1.2 With printing MBFs in the form of bitmaps

Compile file `Alg1b.java` using command `javac Alg1b.java`. To run compiled program use command `java Alg1b` with elements of cycle type as parameters. For example, if you want to generate the set of fixed points of D_4 under permutation $\pi = (1\ 2)(3\ 4)$ with consideration of printing MBFs in the form of bitmaps, use command `java Alg1b 2 2`.

1.3 With printing MBFs in the form of integers

Compile file `Alg1c.java` using command `javac Alg1c.java`. To run compiled program use command `java Alg1c` with elements of cycle type as parameters. For example, if you want to generate the set of fixed points of D_4 under permutation $\pi = (1\ 2)(3\ 4)$ with consideration of printing MBFs in the form of integers, use command `java Alg1b 2 2`.

2 Algorithm 2

2.1 k=1

Compile file `Alg2a.java` using command `javac Alg2a.java`. To run compiled program use command `java Alg2a` with elements of cycle type as parameters. For example, if you want to find number of fixed points of D_5 under permutation $\pi = (1\ 2)(3\ 4)$ use command `java Alg2a 2 2`

2.2 k=2

Compile file `Alg2b.java` using command `javac Alg2b.java`. To run compiled program use command `java Alg2b` with elements of cycle type as parameters. For example, if you want to find number of fixed points of D_6 under permutation $\pi = (1\ 2)(3\ 4)$ use command `java Alg2b 2 2`

2.3 k=3

The program is in the file `Alg2c.java`. To calculate the number of fixed points in D_{n+3} under the permutation π, edit the array `dn[]` to include the set of fixed points in D_{n+4} under the permutation π. By default, the set of fixed points in D_5 under $\pi = (1; 2; 3; 4)$ is provided. Then, compile the file `Alg2d.java` using the command `javac Alg2d.java`. To run the compiled program, use the command `java Alg2d` with elements of the cycle type as parameters.

2.4 k=4

The program is in the file `Alg2d.java`. To calculate the number of fixed points in D_{n+4} under the permutation π, edit the array `dn[]` to include the set of fixed points in D_{n+4} under the permutation π. By default, the set of fixed points in D_5 under $\pi = (1; 2; 3; 4)$ is provided. Then, compile the file `Alg2d.java` using the command `javac Alg2d.java`. To run the compiled program, use the command `java Alg2d` with elements of the cycle type as parameters.

3 Algorithm 3

3.1 Algorithm 3c2

Algorithm corresponds with algorithm from Section 3C (with additional 2-cycle). Compile file `Alg3c2.java` using command `javac Alg3c2.java`. To run compiled program use command `java Alg3a` with elements of cycle type as parameters. For example, if you want to find the number of fixed points in D_5 under permutation $\pi = (1\ 2\ 3)(4\ 5)$ use command `java Alg3c2 3`. If you want to find the number of fixed points in D_8 under permutation $\pi = (1\ 2\ 3)(4\ 5)$ use command `java Alg3c2 3 1 1 1`. All parameters has to be coprime to 2.

3.2 Algorithm 3c3

Algorithm corresponds with algorithm from Section 3C (with additional 3-cycle). Compile file `Alg3c3.java` using command `javac Alg3c3.java`. To run compiled program use command `java Alg3b` with elements of cycle type as parameters. For example, if you want to find the number of fixed points of D_7 under permutation $\pi = (1\ 2)(3\ 4)(5\ 6\ 7)$ use command `java Alg3c3 2 2`. If you want to find number of fixed points of D_8 under permutation $\pi = (1\ 2)(3\ 4)(5\ 6\ 7)$ use command `java Alg3c3 2 2 1`. All parameters has to be coprime to 3.

3.3 Algorithm 3d

Algorithm corresponds with algorithm from Section 3D. Compile file `Alg3d.java` using command `javac Alg3d.java`. To run compiled program use command `java Alg3d`. Result is number of fixed points of D_8 under permutation $\pi = (1\ 2)(3\ 4)(5\ 6)(7\ 8)$. We do not support calculating the number of fixed points in D_9 under permutation $\pi = (1\ 2)(3\ 4)(5\ 6)(7\ 8)$ because the about 20 GB R_7 set (with cardinalities of classes and interval sizes) is needed on input.

3.4 Algorithm 3e

Algorithm corresponds with algorithm from Section 3E. Compile file `Alg3e.java` using command `javac Alg3e.java`. To run compiled program use command `java Alg3e`. Result is number of fixed points of D_9 under permutation $\pi = (1\ 2\ 3)(4\ 5\ 6)(7\ 8\ 9)$.

4 Proofs

4.1 Proof P0

This is a computer-assisted proof of special case of Lemma 11 described by Lemma 13. Compile file `P0.java` using command `javac P0.java`. To run compiled program use command `java P0`.

4.2 Proof P1

This is a computer-assisted proof of special case of Lemma 14 described by Lemma 16. Compile file `P1.java` using command `javac P1.java`. To run compiled program use command `java P1`.

5 Result tables

In this section, we present all partial results in calculating r_n, where $2 \leq n \leq 9$.

5.1 Calculation of r_2

i	π_i	μ_i	$\phi_2(\pi_i)$
1	(1)	1	6
2	(12)	1	4

$$r_2 = \frac{1}{2!} \cdot \sum_{i=1}^{k} \mu_i \phi_2(\pi_i) = 5.$$

5.2 Calculation of r_3

i	π_i	μ_i	$\phi_3(\pi_i)$
1	(1)	1	20
2	(12)	3	10
3	(123)	2	5

$$r_3 = \frac{1}{3!} \cdot \sum_{i=1}^{k} \mu_i \phi_3(\pi_i) = 10.$$

5.3 Calculation of r_4

i	π_i	μ_i	$\phi_4(\pi_i)$
1	(1)	1	168
2	(12)	6	50
3	(123)	8	15
4	(1234)	6	8
5	(12)(34)	3	28

$$r_4 = \frac{1}{4!} \cdot \sum_{i=1}^{k} \mu_i \phi_4(\pi_i) = 30.$$

5.4 Calculation of r_5

i	π_i	μ_i	$\phi_5(\pi_i)$
1	(1)	1	7581
2	(12)	10	887
3	(123)	20	105
4	(1234)	30	35
5	(12345)	15	11
6	(12)(34)	24	309
7	(12)(345)	20	35

$$r_5 = \frac{1}{5!} \cdot \sum_{i=1}^{k} \mu_i \phi_5(\pi_i) = 210.$$

5.5 Calculation of r_6

i	π_i	μ_i	$\phi_6(\pi_i)$
1	(1)	1	7828354
2	(12)	15	160948
3	(123)	40	3490
4	(1234)	90	494
5	(12345)	144	64
6	(123456)	120	44
7	(12)(34)	45	24302
8	(12)(345)	120	490
9	(12)(3456)	90	324
10	(123)(456)	40	562
11	(12)(34)(56)	15	8600

$$r_6 = \frac{1}{6!} \cdot \sum_{i=1}^{k} \mu_i \phi_6(\pi_i) = 16353.$$

5.6 Calculation of r_7

i	π_i	μ_i	$\phi_7(\pi_i)$
1	(1)	1	2414682040998
2	(12)	21	2208001624
3	(123)	70	2068224
4	(1234)	210	60312
5	(12345)	504	1548
6	(123456)	840	766
7	(1234567)	720	101
8	(12)(34)	105	67922470
9	(12)(345)	420	59542
10	(12)(3456)	630	26878
11	(12)(34567)	504	264
12	(123)(456)	280	69264
13	(123)(4567)	420	294
14	(12)(34)(56)	105	12015832
15	(12)(34)(567)	210	10192

$$r_7 = \frac{1}{7!} \cdot \sum_{i=1}^{k} \mu_i \phi_7(\pi_i) = 490013148.$$

5.7 Calculation of r_8

i	π_i	μ_i	$\phi_8(\pi_i)$
1	(1)	1	5613043722868755790 7788
2	(12)	28	101627867809333596
3	(123)	112	262808891710
4	(1234)	420	424234996
5	(12345)	1344	531708
6	(123456)	3360	144320
7	(1234567)	5760	3858
8	(12345678)	5040	2364
9	(12)(34)	210	182755441509724
10	(12)(345)	1120	401622018
11	(12)(3456)	2520	93994196
12	(12)(34567)	4032	21216
13	(12)(345678)	3360	70096
14	(123)(456)	1120	535426780
15	(123)(4567)	3360	25168
16	(123)(45678)	2688	870
17	(1234)(5678)	1260	3211276
18	(12)(34)(56)	420	7377670895900
19	(12)(34)(567)	1680	16380370
20	(12)(34)(5678)	1260	37834164
21	(12)(345)(678)	1120	3607596
22	(12)(34)(56)(78)	105	2038188253420

$$r_8 = \frac{1}{8!} \cdot \sum_{i=1}^{k} \mu_i \phi_6(\pi_i) = 1392195548889993358.$$

5.8 Calculation of r_9

π_i	μ_i	$\phi_9(\pi_i)$
(1)	1	286386577668298411128469151667598498812366
(12)	36	16278282012194909428324143293364
(123)	168	868329572680304346696
(1234)	756	5293103318608452
(12345)	3024	26258306096
(123456)	10080	2279384919
(1234567)	25920	3268698
(12345678)	45360	1144094
(123456789)	40320	97830
(12)(34)	378	107622766375525877620879430
(12)(345)	2520	5166662396125146
(12)(3456)	7560	323787762940974
(12)(34567)	18144	70165054
(12)(345678)	30240	547120947
(12)(3456789)	25920	80720
(123)(456)	3360	7107360458115201
(123)(4567)	15120	92605092
(123)(45678)	24192	197576
(123)(456789)	20160	218542866
(123)(456)(789)	2240	221557843276152
(1234)(5678)	11340	503500313130
(1234)(56789)	18144	10182
(12)(34)(56)	1260	328719964864138799170044
(12)(34)(567)	7560	14037774553676
(12)(34)(5678)	11340	66031909836340
(12)(34)(56789)	9072	3710840
(12)(345)(678)	10080	866494196253
(12)(345)(6789)	15120	22062570
(12)(34)(56)(78)	945	17143334331688770356814
(12)(34)(56)(789)	2520	807900672006

$$r_9 = \frac{1}{9!} \cdot \sum_{i=1}^{k} \mu_i \phi_9(\pi_i) = 789204635842035040527740846300252680.$$