Problem 1. **Blocking Sets**

France 1 — ITYM 2010

Abstract

We study this problem from a combinatorial point of view, associating with a given configuration the corresponding permutation induced in the vertical ordering by the points $(i, \sigma(i))$.

We solve the problem of finding minimal blocking sets in two steps. First, we show an example of blocking sets, with 8 points in question 1 for dimension 2, (see question 5 for dimensions 3 and greater than 4). Then, starting from simple remarks, we will progressively include all restrictions of the situation to deduce that this example is indeed minimal. It is interesting to see that for sufficiently high dimensions, only the "simple remarks" are effective, and they imply the lower bound of 6 points for any minimal blocking set.

The variant of the problem with different restrictions for admissible paths is more easily handled: points "behind" the current one do not affect anymore the situation. This suffices to show the non-existence of blocking sets in this case, but also opens the possibility of calculating the number of paths. We'll establish a recurrence relation for their mean, and then estimate it from below to show that it grows without bound as the number of points increase, which shows that not only there will be a paths, but indeed many of them if we consider sufficiently many points.

Finally, the most involved part is estimating the number of paths in the original problem. We can prove a recurrence relation for paths given only their numerical invariants, that is, the number of points, and the number of different types of moves we make. We use this recurrence to calculate some explicit values, which support the conjecture that the mean number of paths goes to 0.

1st Question

We change notations to give integer coordinates for the points (a_i, b_i) of the set. This is very useful for our combinatorial point of view.

We can choose the numbering so that $a_i = i$ and $b_i = \sigma(i)$, where σ is a permutation of $\{0, \ldots, n-1\}$. Note that $\sigma(0) = 0$ and $\sigma(n-1) = n-1$ from the conditions of the problem.

We have a blocking set for $n = 8$ points in dimension 2:

$$(\sigma(0), \sigma(1), \ldots, \sigma(7)) = (0, 4, 2, 6, 1, 5, 3, 7).$$

We will show that there is no blocking set with $n < 8$.

Let n be the smallest integer such that there exists a blocking set B of n points; $n \geq 3$ because if there are only two points in B there is a path $(0,0) \longrightarrow (1,1)$.

Next, note that there is always one path or two paths starting from $(0,0)$; we have only one path if and only if $\sigma(1) = 1$. If B contains the point $(1,1)$ (i.e $\sigma(1) = 1$), then a blocking set with $n-1$ points exists: just "forget" that we started from $(0,0)$ and count from $(1,1)$. Explicitly, take τ a permutation of $\{0, \ldots, n-2\}$ such that

$$(\tau(0), \tau(1), \ldots, \tau(n-2)) = (\sigma(1) - 1, \sigma(2) - 2, \ldots, \sigma(n-1) - 1).$$

Therefore, as B is minimal, it has two paths starting from $(0,0)$. Symmetrically, there are two points arriving at the last point $(n-1, n-1)$.

So $n \geq 6$, because the two points bound to $(n-1, n-1)$ and the two bound to $(0,0)$ are different, otherwise there is a diagonal path in B.

It remains to show that there is no blocking set for $n = 6$ and $n = 7$.

If $n = 6$

There are two groups of points: the first one is $(0,0)$, $(1,a)$ and $(b,1)$; and the second one is $(5,5)$, $(4,c)$, $(d,4)$.

We have $2 \leq a, b, c, d \leq 3$, so the possible sets are such that

$$(\sigma(0), \sigma(1), \ldots, \sigma(5)) = (0, a, 1, 4, c, 5)$$

or

$$(\sigma(0), \sigma(1), \ldots, \sigma(5)) = (0, a, 4, 1, c, 5).$$

In the first case, there is a path $(0,0) \longrightarrow (2,1) \longrightarrow (3,4) \longrightarrow (5,5)$. In the second case, we have a path $(0,0) \longrightarrow (3,1) \longrightarrow (4,c) \longrightarrow (5,5)$ because $1 < c < 5$.

So there does not exist a blocking set with $n = 6$ points.

If $n = 7$

We have $\sigma(1) \neq 5$ and $\sigma(5) \neq 1$, otherwise it is clear that there is a diagonal path going to $(1,5)$ or $(5,1)$ then to $(6,6)$.

Recall also that $\sigma(1) \neq 1$ and $\sigma(5) \neq 5$, therefore the (i,j) such that $(\sigma(i),\sigma(j)) = (1,5)$ verifies $i,j \in \{2,3,4\}$. So we have $3 \times 2 = 6$ cases to study for the choice of (i,j). In each case, we verify that there is a diagonal path.

$$
\begin{aligned}
(i,j) = (2,3) &\implies (0,0) \to (2,1) \to (3,5) \to (6,6)\\
(i,j) = (2,4) &\implies (0,0) \to (2,1) \to (3,\sigma(3)) \to (4,5) \to (6,6)\\
(i,j) = (3,2) &\implies (0,0) \to (1,\sigma(1)) \to (2,5) \to (6,6)\\
(i,j) = (3,4) &\implies (0,0) \to (3,1) \to (4,5) \to (6,6)\\
(i,j) = (4,2) &\implies (0,0) \to (1,\sigma(1)) \to (2,5) \to (6,6)\\
(i,j) = (4,3) &\implies (0,0) \to (4,1) \to (5,\sigma(5)) \to (6,6)
\end{aligned}
$$

This shows that the blocking set B can't have 7 points. **Therefore, the minimum number of elements that a blocking set can have is 8.**

3rd Question

We'll establish a recurrence relation among the number of paths in sets of n points. As there are many quantities involved, we start fixing notation.

$D(n,x,y,z)$ is the number of paths in $n+2$-point sets (we have only n points to choose) that have x steps "to the right", y steps "above" and z steps "in diagonal". Here, "to the right" means we go to the point whose ordinate is exactly one bigger, but is farther away in the abscissa. For instance, $(0,0) \to (4,1)$ is a step to the right. Analogously, "above" is the reciprocal situation. A diagonal move is $(a,b) \to (a+1,b+1)$.

Next, start counting. A path in $D(n)$ certainly starts with one of the above types. If it is a diagonal move, then this contributes with $D(n-1)$. In the refined notation, $D(n-1,x,y,z-1)$ contributes to $D(n,x,y,z)$.

Now comes the hard part: the combinatorics of the blocking. Say that a path starts $(0,0) \to (i,1)$ with $2 \leq i \leq n$. How many do that? Look at the remaining path from $(i,1)$ to $(n+1,n+1)$, and consider its "reduced" representation as a path in $D(n-i)$. It is more or less a path in $D(n-i,x-1,y,z)$, except from the following fact:

If we go to abscissa i, then we have $i-1$ points to the left. These points fit into horizontal strips, determined by the ordinates of the points to the right of i. It could happen that, in the reduced representation, we have a diagonal step that is no more a diagonal move when we consider all $i-1$ left points. We say that this step was "broken", by a point in the strip of the move.

This does not invalidate the path. But if we place a point in the strip of a horizontal move, then it is no more true that considerint the points to the left we have a diagonal path, because in fact the arrow of the move becomes invalid in the whole setting.

This way, we see that a path in $D(n-i,x-1,y,z)$ has:

- z fragile steps and strips, which can be broken;
- $x-1$ unbreakable steps and strips, otherwise this invalidates the path for $D(n)$;

- the remaining strips, $n - i + 1 - (x - 1) - z$, which do not alter the verticality.

We must place $i-1$ points in all those strips. So choose $k \geq 0$ among the z for breaking. Then, choose $j \geq 0$ more to place in those k, this can be done in $\binom{j+k-1}{j}$. We have $i-1-j-k$ left to put in the remaining strips; this goes in $\binom{n-i+1-(x-1)-z+i-1-j-k-1}{i-1-j-k} = \binom{n-(x+z+k+j)}{i-1-j-k}$ ways.

But in so doing, we have contributed a path not to $D(n, x, y, z)$, but to $D(n, x, y + k, z - k)$! So to get the formula right, we must shift z and y by k.

This gives for $D_i(n, x, y, z)$, the paths starting to $(i, 1)$:

$$\sum_{k=0}^{z} \binom{z+k}{k} \left(\sum_{j=0}^{i-1-k} \binom{j+k-1}{j} \binom{n-(x+z+k+j)}{i-1-j-k} \right) D(n-i, x-1, y-k, z+k).$$

Analogously, replacing x by y, we have $D^i(n, x, y, z)$, the paths starting to $(1, i)$:

$$\sum_{k=0}^{z} \binom{z+k}{k} \left(\sum_{j=0}^{i-1-k} \binom{j+k-1}{j} \binom{n-(y+z+k+j)}{i-1-j-k} \right) D(n-i, x-k, y-1, z+k).$$

Therefore, as clearly

$$D(n, x, y, z) = D(n-1, x, y, z-1) + \sum_{i=0}^{n} D_i(n, x, y, z) + D^i(n, x, y, z)$$

we have established a recurrence.

Calculating the first values for this, we obtain for $8 < n < 120$ a decreasing sequence for $\frac{D(n)}{n!}$, going under 0.5 after topping at 2.281771 for $n = 8$, which means 10 points in the set S. This, added to the intuitive fact that moves to the right allow for many invalidatings, maks us conjecture that the limit of the mean number of paths is 0.

4$^{\text{th}}$ Question

We will answer questions 1 and 3 adapted for the different setting in subparts 4.1 and 4.3 respectively.

4.1)

There is always a diagonal path with the conditions of question 4.

Indeed, if we take a point A in the graph, we consider the points whose two coordinates are greater than one of those of A. Then, we take among these points one which has one coordinate successive to the same coordinate of A, and we can draw an arrow from A to this point.

Therefore, from any point A of the graph we can draw an arrow to an other point, and consequently we always have a diagonal path.

In particular, **the probability of having a blocking set is 0.**

4.3)

Let U_n be the number of diagonal paths over all non-equivalent n-point acceptable sets. We will establish an induction relation for U_n.

We distinguish three kinds of diagonal paths according to the nature of its first step: The ones beginning by $(0,0) \longrightarrow (i,1)$ with $2 \leq i \leq n-2$ (type H for horizontal), the ones beginning by $(0,0) \longrightarrow (1,i)$ with $2 \leq i \leq n-2$ (type V for vertical) and the ones beginning by $(0,0) \longrightarrow (1,1)$ (type D for diagonal).

Then write h_n, v_n and d_n as the respective number of diagonal paths of these 3 kinds of paths over all non-equivalent n-point acceptable sets. We have $U_n = h_n + v_n + d_n$.

We remark that $h_n = v_n$. Indeed, if we have a diagonal path of type H in a graph G, say $(0,0) \longrightarrow (i,1) \longrightarrow \cdots \longrightarrow (n-1,n-1)$, then the graph G' obtained by the reflection through the line of equation $y = x$ contains a diagonal path of type V: $(0,0) \longrightarrow (1,i) \longrightarrow \cdots \longrightarrow (n-1,n-1)$. Therefore we have $v_n \geq h_n$ and similarly $h_n \geq v_n$, and then equality.

Also, we remark that $d_n = U_{n-1}$: like in the first question, we have a bijection considering the permutation τ).

So let's calculate h_n.

We fix $i \in \{2,\ldots,n-2\}$. We will first find the number $d(n,i)$ of diagonal paths beginning by $(0,0) \longrightarrow (1,i)$ over all non-equivalent n-point acceptable sets such that $\sigma(i) = 1$.

We have $i-1$ points whose first coordinate is in $\{1,\ldots,i-1\}$. There are $n-3$ possible second coordinates for these points, because the second coordinate can't be 0, 1 or $n-1$. Therefore, we have $(n-3)(n-3-1)\cdots(n-3-(i-1)+1) = \frac{(n-3)!}{(n-i-2)!}$ possibles choices for these $i-1$ points.

When the $i-1$ first points have been chosen, the number of diagonal paths beginning by $(0,0) \longrightarrow (1,i)$ over all the $n-i$ possible last points (that is, whose first coordinate is in $\{i,\ldots,n-1\}$) is U_{n-i}.

Indeed, if we consider the relative vertical order of the $n-i$ last points, we have a bijection between the acceptable sets with $n-i$ points and these $n-i$ last points; and according to the condition ii) we see that the $i-1$ first points won't have any influence on the diagonal paths we are counting.

Therefore $d(n,i) = \frac{(n-3)!}{(n-i-2)!}U_{n-i}$ and

$$h_n = \sum_{i=2}^{n-2} d(n,i) = \sum_{i=2}^{n-2} \frac{(n-3)!}{(n-i-2)!}U_{n-i}$$

Using $U_n = 2h_n + U_{n-1}$,

$$U_n = 2\left(\sum_{i=2}^{n-2} \frac{(n-3)!}{(n-i-2)!}U_{n-i}\right) + U_{n-1}$$

And if D_n is the arithmetic mean of the number of diagonal paths over all non-equivalent n-point acceptable sets, we have

$$D_n = \frac{U_n}{(n-2)!}$$

because we have $(n-2)!$ permutation σ of $\{0,\dots,n-1\}$ with $\sigma(0)=0$ and $\sigma(n-1)=n-1$. This simplifies to (for $n>2$)

$$D_n = \frac{2(\sum_{i=2}^{n-2} D_{n-i}) + D_{n-1}}{n-2}.$$

So start the recurrence: we have clearly $D_2 = 1$. The formula gives $D_3 = 1$, $D_4 = \frac{3}{2}$, $D_5 = \frac{11}{6}$, $D_6 = \frac{53}{24}$ and $D_7 = \frac{103}{40}$. We verify that $D_i \geq \ln(i)$ for all $7 \geq n \geq 4$ and that $D_2 + D_3 = 2 \geq \ln(2) + \ln(3)$.

Let's show the following by induction on $n \geq 4$:

$$D_n \geq \ln(n).$$

Suppose the property holds for all $4 \leq k \leq n+1$ (where $n \geq 5$); we have

$$D_{n+2} = \frac{2(\sum_{i=2}^{n} D_i) + D_{n+1}}{n} \geq \frac{2(\sum_{i=1}^{n} \ln(i)) + \ln(n+1)}{n}$$

So we need to evaluate $\sum_{i=1}^{n} \ln(i)$.

Let $f(t) = \ln(t)$ for $t \in [1,+\infty[$. As f is increasing on $[1,+\infty[$, for $k \geq 1$

$$\int_k^{k+1} f(t)\,dt \leq \int_k^{k+1} f(k+1)\,dt = f(k+1).$$

and when we sum these inequalities for $k = 1 \dots n-1$, we get

$$\sum_{k=1}^{n} \ln(k) \geq \int_1^n \ln(t)\,dt = n\ln(n) - n + 1.$$

Consequently,

$$D_{n+2} \geq \frac{2(n\ln(n) - n + 1) + \ln(n+1)}{n}$$

Remark that this is bigger than $2\ln(n) - 1$, and we want to show that it is bigger than $\ln(n+2)$. That is equivalent to

$$\frac{n^2}{n+2} \geq e,$$

and as the function on the left is increasing for $n > 0$ (its reciprocal is clearly decreasing!), it suffices to show that it is true for $n = 5$. But $25/7 > 3 > e$, so we get $D_{n+2} \geq \log(n+2)$ and the induction step is proven.

So $D_n \geq \ln(n)$ and then the limit of D_n when n tends to $+\infty$ is $+\infty$.

Question 5

In this question, we will study the minimum number of points that has a blocking set in dimension N, where $N \geq 3$.

First, for dimension 3, it is clear that 8 points are enough: just take the blocking set for dimension 1 and duplicate one coordinate. The fact that there is no blocking set with

7 points was done by computer. The two programs, one generating all permutations of 5 points, the other reading two permutations and deciding whether the resulting set has a diagonal path. We present them both in the appendix.

We remark that in dimension 4, we have a blocking set for $n = 6$ points:

$$((a_1, b_1, c_1, d_1), \ldots, (a_6, b_6, c_6, d_6)) = \\ ((0,0,0,0), (1,1,1,3), (2,4,4,2), (3,3,1,1), (4,2,2,4), (5,5,5,5)).$$

And *a fortiori* we have an example of blocking set with 6 points in dimension $N \geq 4$: we just have to take the other coordinates equal to the first one.

In fact, like in question 1, we see that in any dimension $N \geq 2$ there is no blocking set with less than 5 points: in a minimal blocking set, there is no point $(1, 1, \ldots, 1)$ or $(5, 5, \ldots, 5)$. Therefore, there are at least 2 paths starting from $(0, 0, \ldots, 0)$ and 2 arriving at $(1, 1, \ldots, 1)$, and as in dimension 2, these 6 points must be distinct.

To conclude, **the minimum number of elements that a blocking set in dimension $N \geq 4$ can have is 6.**

Programs

generator.c

```c
/* Generates all permutations of n elements for the graph problem */

/**************************************************************************
 *  This program is free software; you can redistribute it and/or modify   *
 *  it under the terms of the GNU General Public License as published by   *
 *  the Free Software Foundation; either version 2 of the License, or      *
 *  (at your option) any later version.                                    *
 *                                                                         *
 *  This program is distributed in the hope that it will be useful,        *
 *  but WITHOUT ANY WARRANTY; without even the implied warranty of         *
 *  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the          *
 *  GNU General Public License for more details.                           *
 *                                                                         *
 *  You should have received a copy of the GNU General Public License      *
 *  along with this program; if not, write to the Free Software            *
 *  Foundation, Inc.,                                                      *
 *  59 Temple Place, Suite 330, Boston, MA  02111-1307  USA                *
 **************************************************************************/

#include <stdio.h>
#include <stdlib.h>

#define N 1024

struct node {
  struct node * next;
  int i;
};

void build_permut(int max, struct node * fn);
void print_permut(void);
```

```c
int permut[N];
int n;
int ifile=1;

int main()
{
int i;
struct node freenodes[N];

  scanf("%d", &n);
  if ( n > N )
  {
    printf("Recompile with N > %d, now N = %d.\n\n\
          Running this program is at best 0(n!*n), so one must have *lots* of
              time.\n", n, N);
    exit(1);
  }

  for(i=0;i<n;i++)
  {
    freenodes[i].i = i;
    freenodes[i].next = freenodes + (i+1);
  }
  freenodes[n-1].next = 0;

  build_permut(n, freenodes);

  return 0;
}

void build_permut(int max, struct node * fn)
{
struct node * pt, * tmp;

  permut[fn->i] = max-1;
  if(max > 1)
  {
    build_permut(max-1, fn->next);
  }
  else
  {
    print_permut();
  }

  for(pt=fn; pt->next; pt = pt->next)
  {
    tmp = pt->next;
    permut[tmp->i] = max-1;
    pt->next = tmp->next;
    build_permut(max-1, fn);
    pt->next = tmp;
  }
}

void print_permut(void)
{
int i;
FILE * fpt;
char fname[128];

  sprintf(fname, "%09d-%09d.in", n, ifile++);
  fpt = fopen(fname, "w");
```

```
  for(i=0;i<n;i++)
    fprintf(fpt, "%d\n", 1+permut[i]);

  fclose(fpt);
}
```

3Dblocking.c

```
/* Decides if a set with NPTS points has a diagonal path */

/************************************************************************
 *  This program is free software; you can redistribute it and/or modify  *
 *  it under the terms of the GNU General Public License as published by  *
 *  the Free Software Foundation; either version 2 of the License, or     *
 *  (at your option) any later version.                                   *
 *                                                                        *
 *  This program is distributed in the hope that it will be useful,       *
 *  but WITHOUT ANY WARRANTY; without even the implied warranty of        *
 *  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the         *
 *  GNU General Public License for more details.                          *
 *                                                                        *
 *  You should have received a copy of the GNU General Public License     *
 *  along with this program; if not, write to the Free Software           *
 *  Foundation, Inc.,                                                     *
 *  59 Temple Place, Suite 330, Boston, MA  02111-1307  USA               *
 ************************************************************************/

#include <stdio.h>
#include <stdlib.h>

#define NPTS 7


int main()
{
int i;
int abscisse[NPTS], ordonnee[NPTS], /* The x and y induced permutations by the z-
    ordering */
    rec_abscisse[NPTS], rec_ordonnee[NPTS], /* and their reciprocal */
    horizontal[NPTS], vertical[NPTS], profondeur[NPTS], /* Valid paths in each
        direction */
    somme[NPTS]; /* Number of paths to point i */

  /* Fixed values */
  abscisse[0] = 0;
  ordonnee[0] = 0;
  abscisse[NPTS-1] = NPTS-1;
  ordonnee[NPTS-1] = NPTS-1;

  /* Read permutations from stdin */
  for(i=0; i < NPTS - 2; i++)
    scanf("%d", &abscisse[1+i]);
  for(i=0; i < NPTS - 2; i++)
    scanf("%d", &ordonnee[1+i]);

/*  for(i=0; i < NPTS; i++){
    printf("%d %d %d\n", i, abscisse[i], ordonnee[i]);
  } */
  /* Build the inverse permutations, and initialize the vector of admissible moves
      */
  for(i=0; i<NPTS ; i++)
```

```
{
  rec_abscisse[abscisse[i]] = i;
  rec_ordonnee[ordonnee[i]] = i;
  horizontal[i]=0;
  vertical[i]=0;
  profondeur[i]=0;
  somme[i]=0;
}

/* Calculates the admissibles moves, filling three vectors for each case */
for(i=0; i < NPTS - 1 ; i++)
{
  /* Move in z-direction: x and y must increase also */
  if(abscisse[i+1]>abscisse[i])
  {
    if(ordonnee[i+1]>ordonnee[i])
    {
      profondeur[i]=i+1;
    }
  }

  /* Move in x-direction */
  if(rec_abscisse[abscisse[i]+1]>i)
  {
    if(ordonnee[rec_abscisse[abscisse[i]+1]]>ordonnee[i])
    {
      if(rec_abscisse[abscisse[i]+1] != i+1)
      {
        horizontal[i]=rec_abscisse[abscisse[i]+1];
      }
    }
  }

  /* Move in y-direction */
  if(rec_ordonnee[ordonnee[i]+1]>i)
  {
    if(abscisse[rec_ordonnee[ordonnee[i]+1]] > abscisse[i])
    {
      if(rec_ordonnee[ordonnee[i]+1] != i+1)
      {
        if(rec_ordonnee[ordonnee[i]+1] != rec_abscisse[abscisse[i]+1])
        {
          vertical[i]=rec_ordonnee[ordonnee[i]+1];
        }
      }
    }
  }
}


/* Calculates number of paths */
somme[0] = 1;   /* (0,0) is always accessible */
for(i=0; i < NPTS - 1; i++)
{
  if (profondeur[i] > 0) somme[profondeur[i]] += somme[i];
  if (   vertical[i] > 0) somme[vertical[i]]   += somme[i];
  if (horizontal[i] > 0) somme[horizontal[i]] += somme[i];
}

/* Displays blocking sets */
if(somme[6] == 0)
{
```

```
    printf("One blocking set: \n");
    for(i=0 ; i<NPTS ; i++)
    {
      printf("%3d,%3d,%3d\n", i, abscisse[i], ordonnee[i]);
    }
  }

  return 0;
}
```