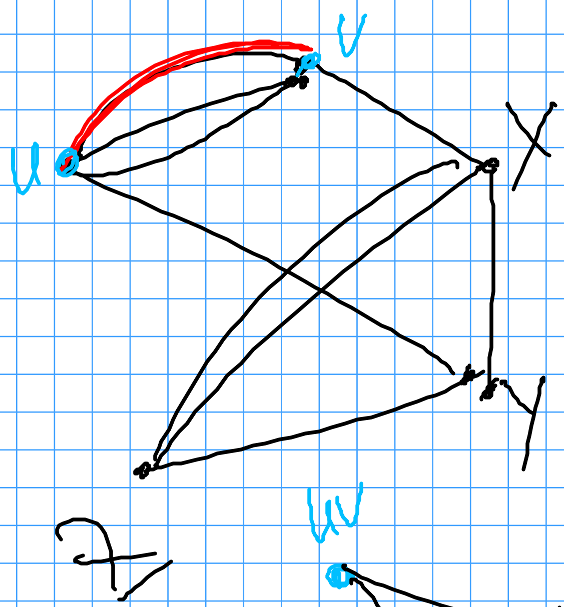
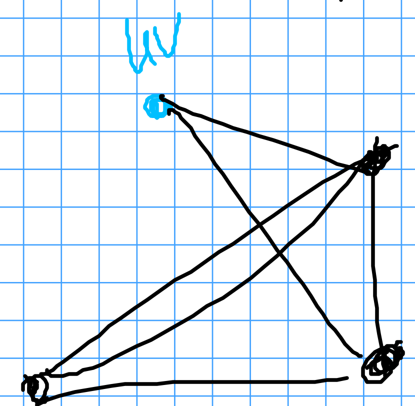
MIN CUT





edge contraction

O(n)

	u	v	x	y	z
u	0	3	0	1	0
v	3	0	1	0	0
x	0	1	0	1	2
y	1	0	1	0	1
z	0	0	2	1	0

adjacency matrix

	v_1	v_2	v_3	v_4	v_5		
v_1	0	0	1	4	2	7	7
v_2	0	0	5	1	1	7	14
v_3	1	5	0	0	3	9	23
v_4	4	1	0	0	1	6	
v_5	2	1	3	1	0	7	
	7	7	9	6	7	36	

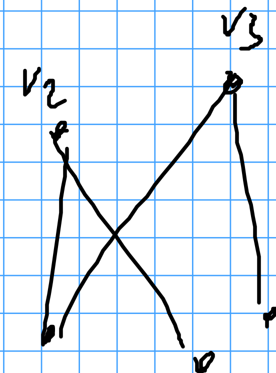
dice : 18

O(n) to select the entry

$\Downarrow$

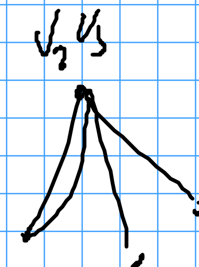
	V_1	V_2	V_3	V_4	V_5	
V_1	0	0	1	4	2	7
V_2	0	0	0	1	1	2
V_3	1	0	0	0	3	4
V_4	4	1	0	0	1	6
V_5	2	1	3	1	0	7
	7	2	4	6	7	26

$O(1)$



$\Downarrow$

	V_1	V_2V_3	V_4	V_5	
V_1	0	1	4	2	7
V_2V_3	1	0	1	4	6
V_4	4	1	0	1	6
V_5	2	4	1	0	7
	7	6	6	7	26



$O(n)$

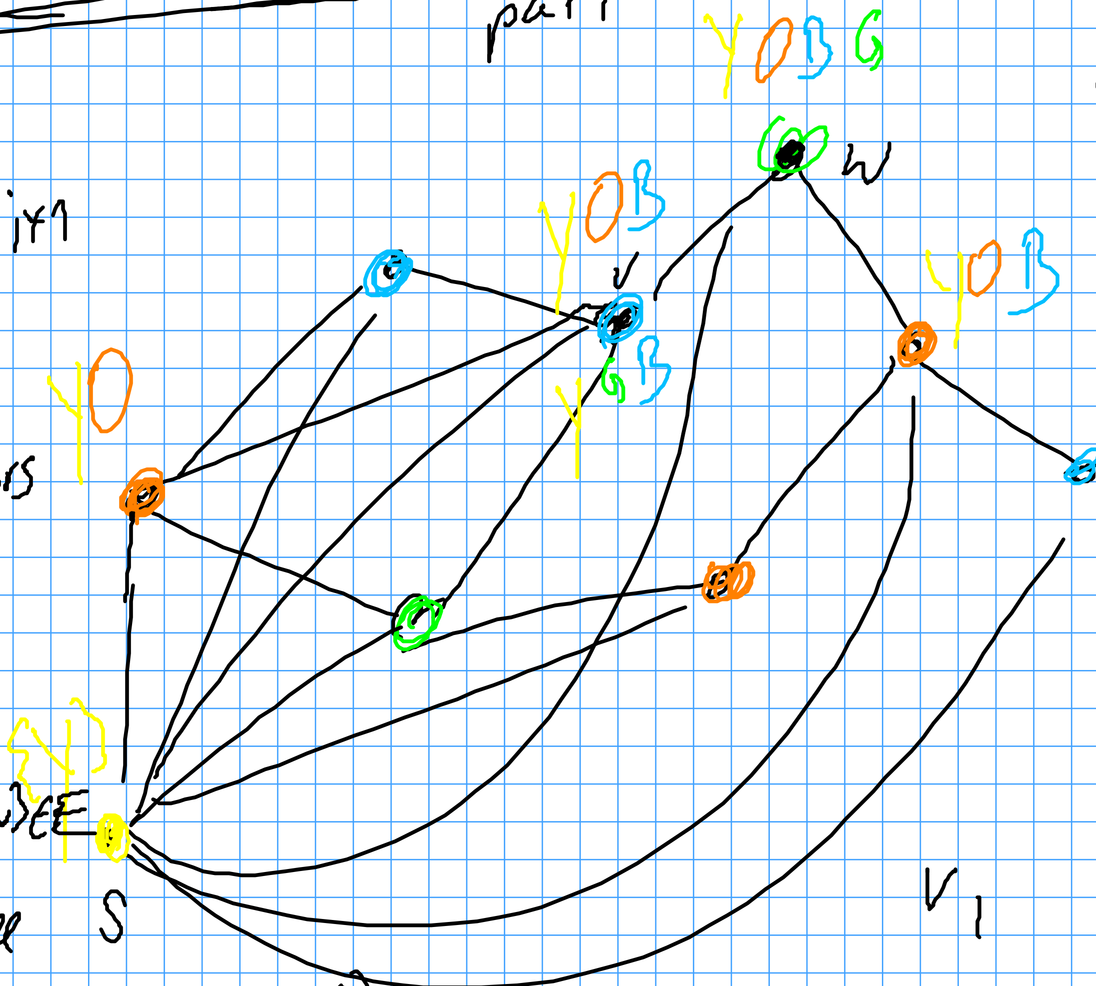
$O(2^k k m)$ for finding a colourful path

w @ i+1

$i \to i+1$

$i=2$

$i=3$



↳ look at all neighbors $O(m)$

↳ for each v with $\{v,w\} \in E$

↳ look at all colour sets of length i $\binom{k}{i}$

↳ look if color of w fits in, if yes create new label

i per color set

V_i $\quad i \in \{0, \dots, k\}$

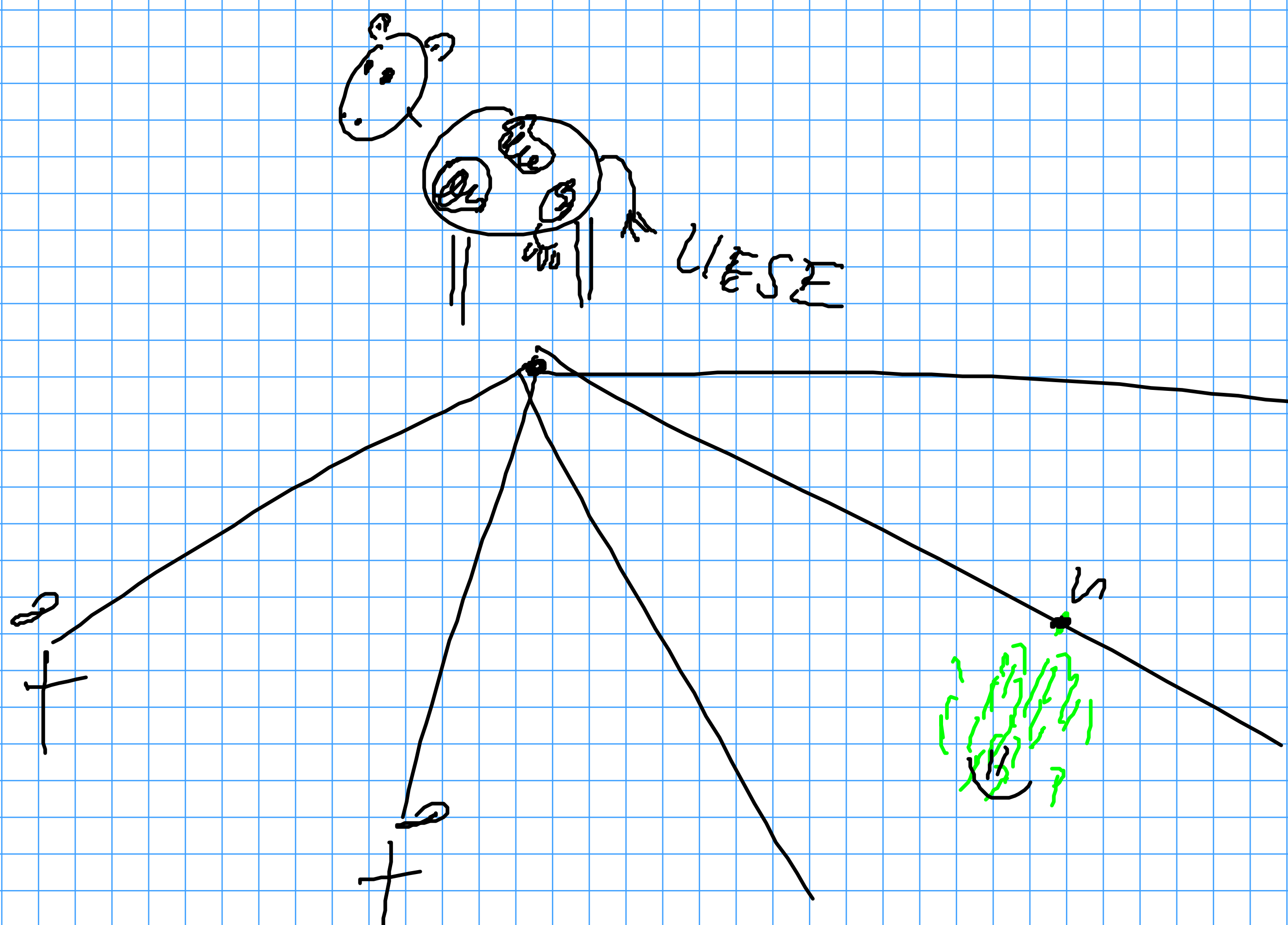
colour sets of v are color on a path of length i from S to v which could be a prefix of a colorful path

total cost:

$$\sum_{i=1}^{k} m \binom{k}{i} O(i) = O\left(m \sum_{i=1}^{k} k^i i\right)$$

$$= O(m 2^k k)$$

RANDOMIZATION IN GAMES & AI

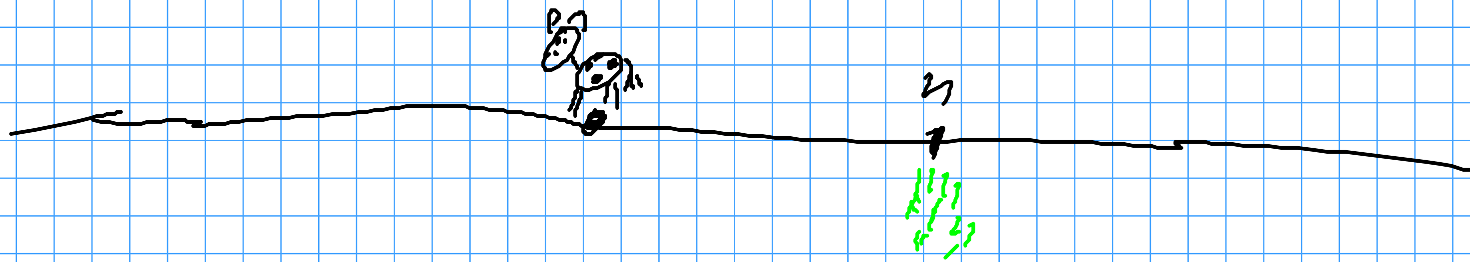


COW PATH PROBLEM

- Liese needs grass. Liese stands at a crossroad with w lanes. There is grass at only one lane at distance n. Liese only sees the grass if she

Stands right on it.

↳ Simplified version: $D = 2$



If Liese would know n, she should walk n in one direction, if there is no grass $2n$ in the other direction.

total walk distance $\leq 3n$

↳ at most a factor of 3 of walking overhead

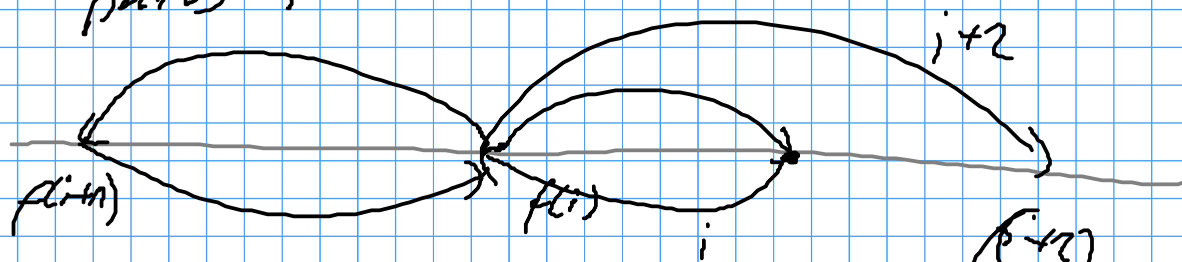
↳ expected walking distance if she chooses the initial direction u.a.r.

$$E(X) = \frac{1}{2} 3n + \frac{1}{2} n = 2n$$

Def: Competitive ratio c, tells how good an algorithm is compared to an optimal strategy with the whole environment known a priori.

Exp. comp. ratio . -
how good a rand. alg. is
expectedly ...

↳ Now: Liese is not aware of n.
Solution has to involve alternating the paths!



in round i Liese walks $f(i)$ steps in one direction, if she does not find grass she returns to the starting point and walks $f(i+1)$ steps in the next round in the other direction

$$\forall i \geq 3: \quad f(i) > f(i-2)$$

$$f(i) = i$$

$$\sum_{i=1}^{n} 2i = 2\,\frac{n(n-1)}{2} \in O(n^2)$$

$$f(i) = 2^i$$

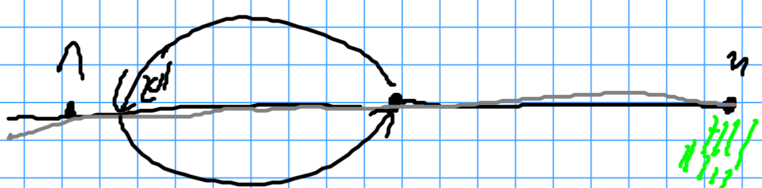
Lemma: For $f(i) = 2^i$ the competitive ratio of the algorithm is 9.

Proof: Let j be the integer such that
$$2^j < n \le 2^{j+1} \quad (j+1 = \lceil \log_2 n \rceil)$$

In the worst case Liese walks in the wrong direction in round $j+1$.

Therefore

$$\#\text{steps} \le \underline{\sum_{i=1}^{j+1} 2f(i) + n} = \sum_{i=1}^{j+1} 2\,2^i + n$$



$$= 2\left(2^{j+2} - 1\right) + n$$

$$= 2^{j+3} - 2 + n$$

$$= 2^3 \cdot 2^j - 2 + n$$

$$\le 8n - 2 + n < 9n$$

Comp. ratio of 9, as god cow needs to walk n. //

What about randomly choosing the initial direction?

Good case:

$$\# \text{steps} = \sum_{i=1}^{j} 2 \cdot 2^i + n = 2^{j+2} - 2 + n$$

$$\leq 4n - 2 + n < 5n$$

Expected walk distance $\leq \frac{1}{2} 9n + \frac{1}{2} 5n$
$= 7n$

Smart Cow Algorithm (general # of lines w)

1. choose a random permutation σ of the w lines, perm. of $\{0, \dots, w-1\}$

2. $\varepsilon \leftarrow$ u.a.r. $[0,1)$, some constant $r > 1$ (fixed)

3. $d \leftarrow r^{\varepsilon}$ (initial distance)

4. $p \leftarrow 0$

5 repeat
   - explore path $\sigma(p)$ up to distance d
   - if no grass was found, go

back to the origin

- $d \leftarrow d \cdot r$
- $p \leftarrow (p+1) \bmod w$

until Liese is happy

Theorem: For any fixed $r > 1$, the Smart Cow algorithm has a comp. ratio of $R(r, w)$

$$= 1 + \frac{2}{w} \cdot \frac{1 + r + r^2 + \ldots + r^{w-1}}{\ln r}$$

Remark: Smart Cow is optimal, because there is a matching lower bound

Exercise: Implement the naive alg. for w lines and Smart Cow. Compare results.

Overview of comp. ratios in dep. of w

for Smart Cow			opt det. ratio
$w=2$	4.59	($r=3.59$)	9
$w=3$	7.73	($r=2.01$)	14.5

Exercise: why?

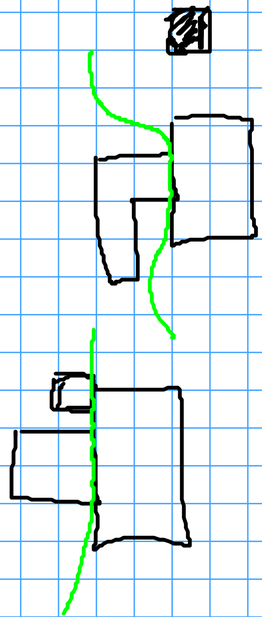
Ü 7   20.11.13 (v=11.28)   36.3

Robot in Unknown Geometric Terrain

Robot at source point $s \in \mathbb{R}^2$, known target $t \in \mathbb{R}^2$. But completely unknown environment. Goal: Find a strategy with a good competitive ratio compared to the optimal path when all obstacles are known a priori.

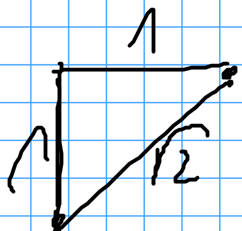
Assumptions:

- obstacles are opaque, impenetrable and non-overlapping
- no obstacle contains s or t or encloses it

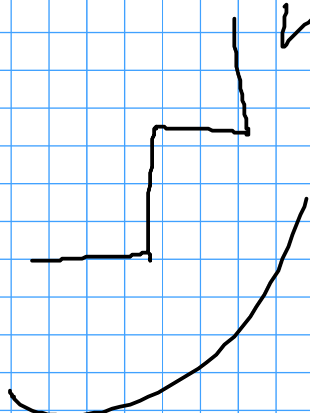


- the unit square can be inscribed in every obstacle
- if two obstacles touch, the robot can 'squeeze' between them, and obstacles are convex, then no non-convex obstacles can be build by combination

↳ also allow for the target to be a polygon or an infinite wall



- robot only can explore obstacles by bumping into them



n Euclidean distance between s and t, p length of the path taken by the robot. We would like to minimize $\frac{p}{n}$.

If we restrict robot movements to left/right and up/down one only looses a factor of $\sqrt{2}$.