

Video Lectures On Artificial Intelligence

Lecture 13 Tabu Search

Prof. Deepak Khemani

Department of Computer Science and Engineering
IIT Madras



· Hill Climbing

Escaping local

Hall Chaining

$C = \text{current}$

$n = \text{next}$



NPTEL

Hill Climbing $C = \text{current}$
 $n = \text{next}$

$n = \text{Best}(\text{Move}(C))$



Hill Climbing $C = \text{current}$
 $n = \text{next}$

if $n = \text{Best}(\text{Neighbors}(C))$ is better than C
 $n \leftarrow C$

Hill Climbing $c = \text{current}$
 $n = \text{next}$

Escape

\rightarrow If $n = \text{Best}(\text{Neighbors}(c))$ is better than c
 $n \leftarrow c$



NPTEL

Hill Climbing $C = \text{current}$
 $n = \text{next}$

Escaping local

$\text{if } \text{Next}(\text{Move}(C)) \text{ is better than } C$

$\leftarrow C$

EXPLOITATION of gradient

Hill Climbing $C = \text{current}$
 $n = \text{next}$

Escape

if $n = \text{Best}(\text{Neigh}(C))$ is better than C
 $n \leftarrow C$

EXPLOITATION of gradient



Hill Climbing $c = \text{current}$
 $n = \text{next}$

Escaping local

if $n = \text{Best}(\text{Neigh}(c))$ is better than c

$n \leftarrow c$

EXPLOITATION of gradient

Hill Climbing $c = \text{current}$
 $n = \text{next}$

Best (Moveben(c)) is better than c

$n \leftarrow c$

EXPLOITATION of gradient

$m \leftarrow \text{Best}$

Escaping local Optima

EXPLORATION



NPTEL

Hill Climbing $c = \text{current}$
 $n = \text{next}$

→ If $n = \text{Best}(\text{MoveGen}(c))$ is better than c

$n \leftarrow c$

EXPLOITATION of gradient

$n \leftarrow \text{Best}(\text{Allowed}(\text{MoveGen}(c)))$

Escaping local Optima

EXPLORATION

Hill Climbing $c = \text{current}$
 $n = \text{next}$

Escaping

→ If $n = \text{Best}(\text{MoveGen}(c))$ is better than c

$n \leftarrow c$

EXPLOITATION of gradient

→ $n \leftarrow \text{Best}(\text{Allowed}(\text{MoveGen}(c)))$



ing $C = \text{current}$
 $n = \text{next}$

Escaping local Optima


Best (Moreben(C)) is better than C

$n \leftarrow C$

EXPLOITATION of gradient

EXPLORATION

$n \leftarrow \text{Best (Allowed (M$



NPTEL

$n \leftarrow \text{Best}(\text{Allowed}(\text{MoveGen}(c)))$

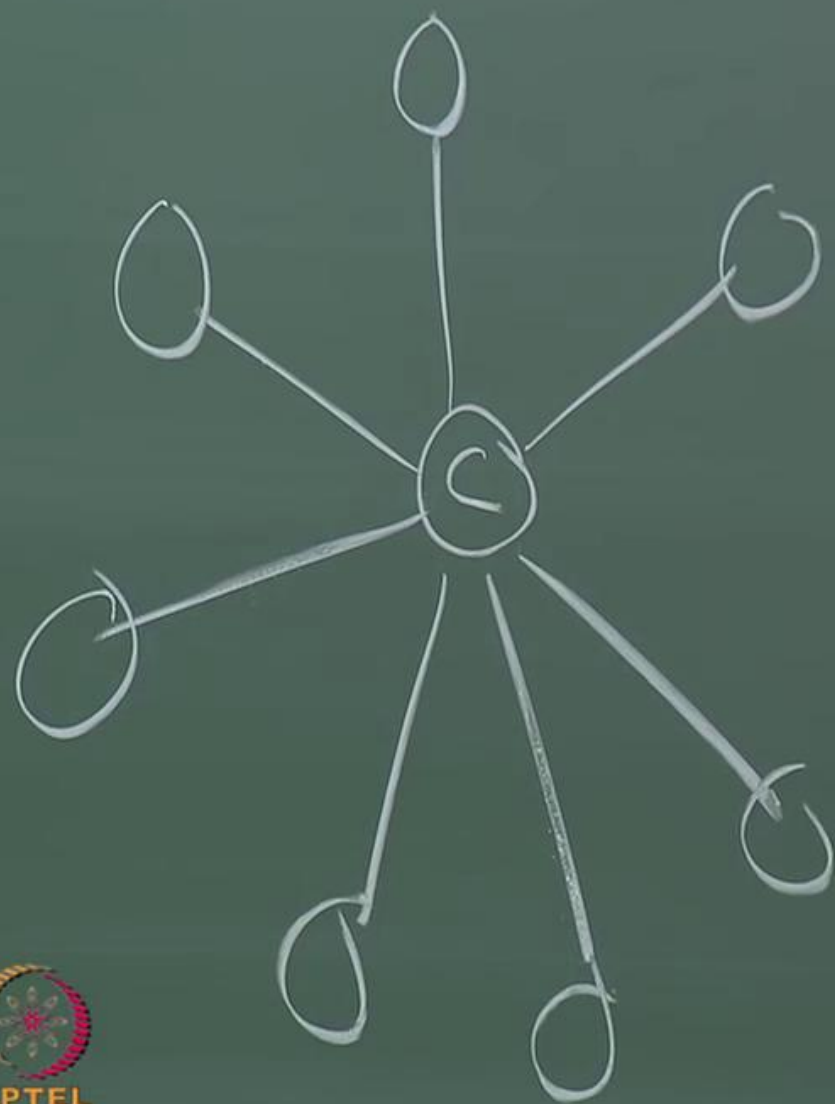
Till some termination criteria

$n \leftarrow \text{Best}(\text{Allowed}(\text{MoveGen}(c)))$

Till some termination criteria
(store best solution found)



NPTEL



Hill Climbing $c = \text{current}$
 $n = \text{next}$

$\text{Best}(\text{MoreGen}(c))$ is better than c

$n \leftarrow c$

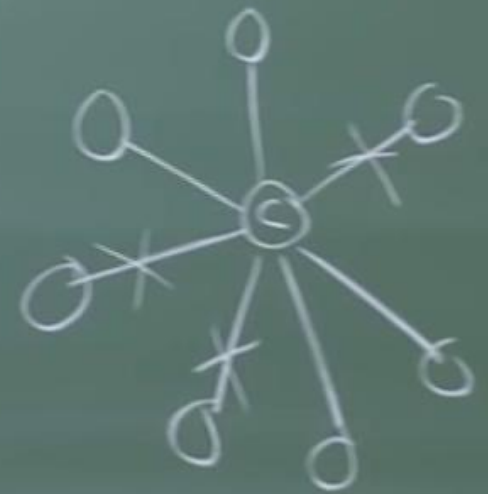
EXPLOITATION of gradient

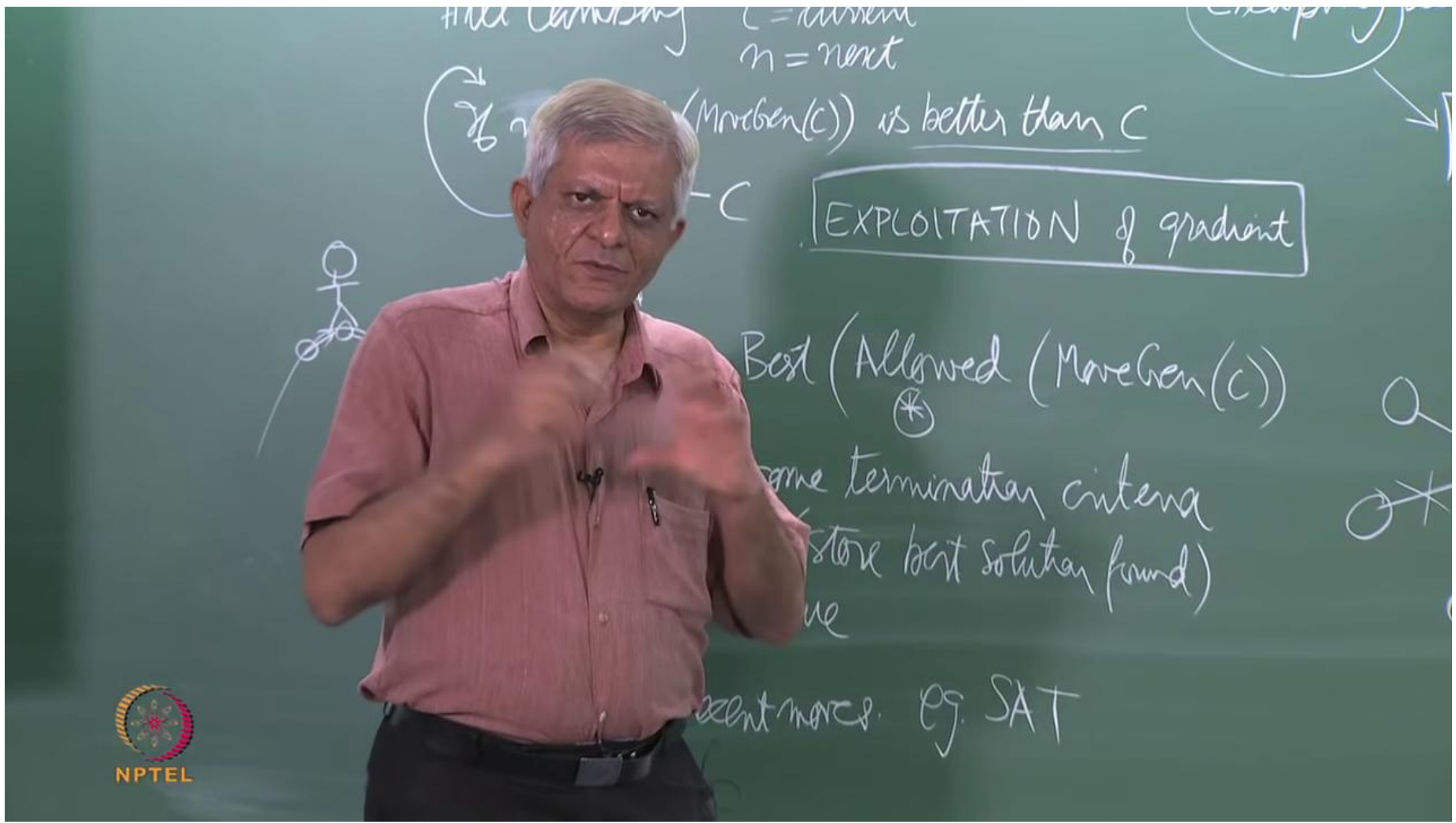
$n \leftarrow \text{Best}(\text{Allowed}(\text{MoreGen}(c)))$
(*)

Until some termination criteria
(store best solution found)
USED Queue

Escaping local Optima

EXPLORATION





Hill Climbing $c = \text{current}$
 $n = \text{next}$

\rightarrow if $n = \text{MoveGen}(c)$ is better than c

EXPLOITATION of gradient

Best (Allowed (MoveGen(c)))
*

some termination criteria
(store best solution found)
we

recent moves. eg. SAT

Until termination $c = \text{current}$
 $n = \text{next}$

$n = \text{Best}(\text{MoveGen}(c))$ is better than c

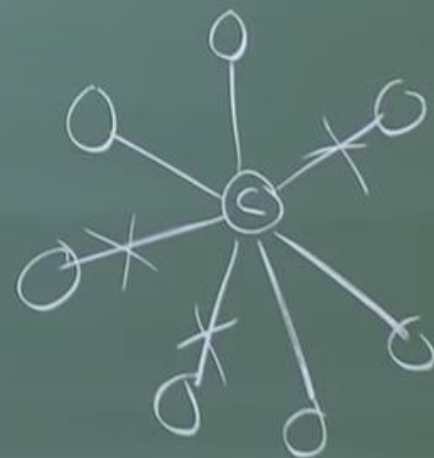
$n \leftarrow c$ **EXPLOITATION of gradient**

$n \leftarrow \text{Best}(\text{Allowed}(\text{MoveGen}(c)))$

Until some termination criteria
(store best solution found)

used Queue

Track of recent moves. eg. SAT - if a bit is changed in last t moves
then disallow it



$c = \text{current}$
 $n = \text{next}$

Best (MoveBest(c)) is better than c

$n \leftarrow c$

EXPLOITATION

$n \leftarrow \text{Best}(\text{All moves from } n(c))$

Until some condition is met
(stop condition)
CLOSED Queue

if recent move

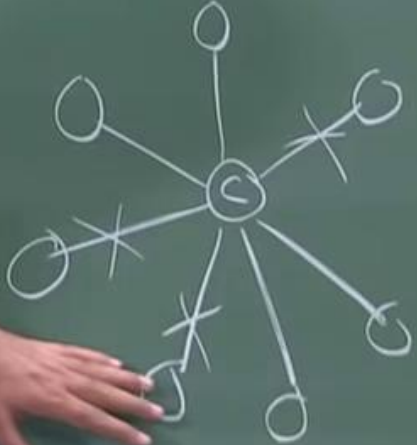
a bit is changed in last t moves
then disallow it

Exploring all options

EXPLORATION

9-bit SAT

M 000000000



eg. SA

EXPLORATION

9-but SAT

M 0 0 0 0 0 0 0 0 0



changed in last 4 moves

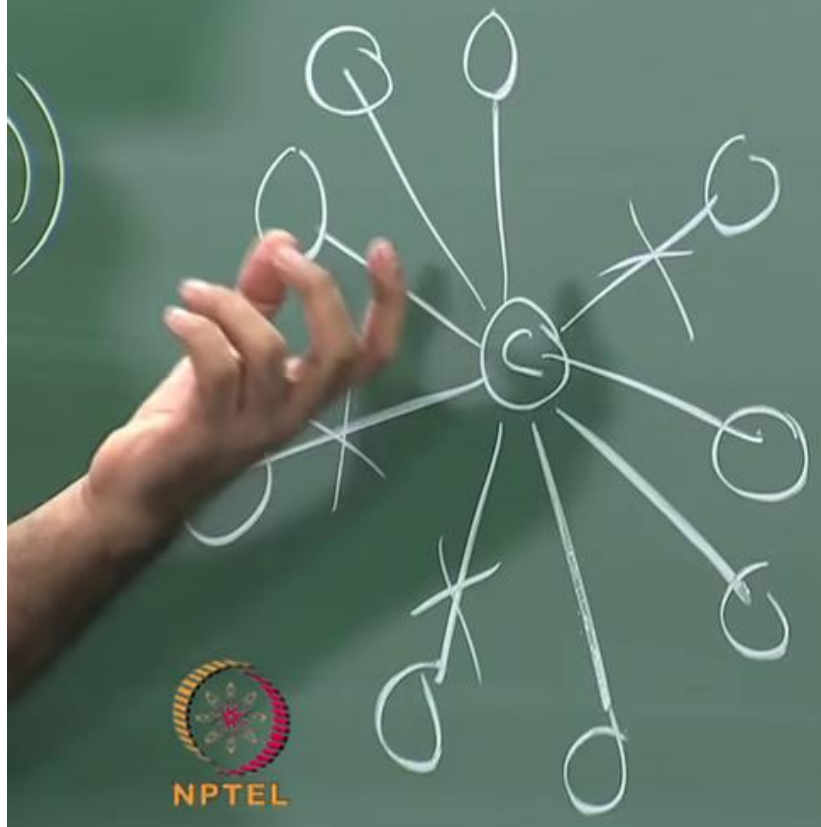
it



nt

9-bit SAT

M 0 0 0 0 0 0 0 0 0

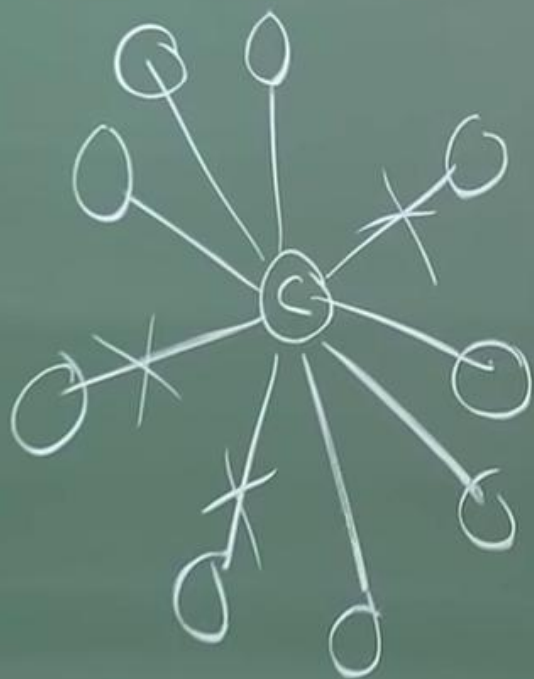


gradient

EXPLORATION

9-bit SAT

M 0 0 0 0 0 0 0 0 0
↓



tabu tenure

a bit is changed in last (+) moves



while (terminating) $c = \text{current}$
 $n = \text{next}$

Best (MoveGen(c)) is better than c
 $n \leftarrow c$

EXPLOITATION of gradient

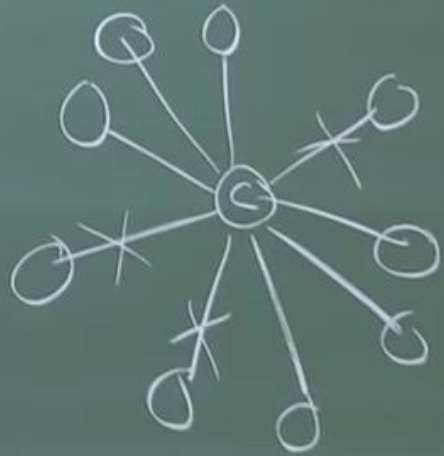
EXPLORATION

TABU SEARCH M

Best (Allowed (MoveGen(c)))
 (*)

till some termination criteria
 (store best solution found)
 return

of recent moves. eg. SAT - if a bit is changed in last (t) moves
 then disallow it



NPTEL

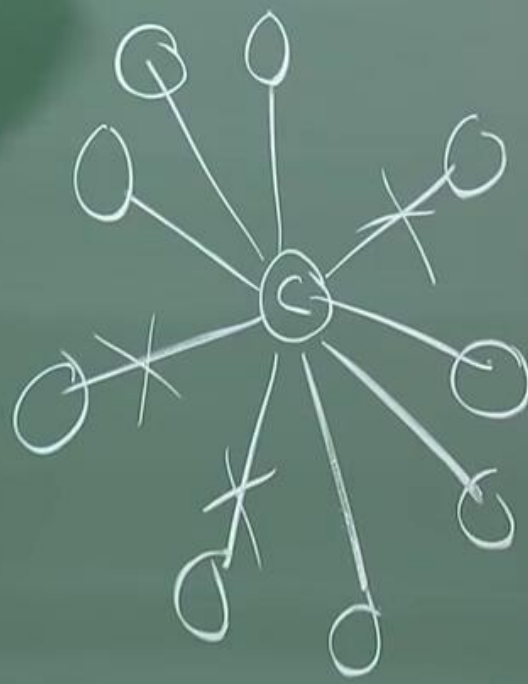
of gradient

TABU SEARCH

M

9-bit SAT

0 0 0 0 0 0 0 0 0



len(c))

end
(round)

tabu tenure

if a bit is changed in last t moves



ABU SEARCH

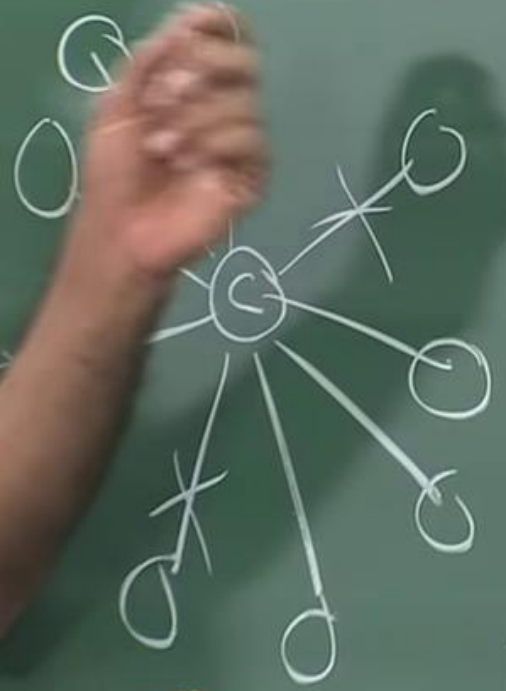
M

9-bit SAT

0 0 0 0 0 0 0 0 0



0 0 0 4 0 0 0 0 0



tabu tenure

used in hill moves



ABU SEARCH

M

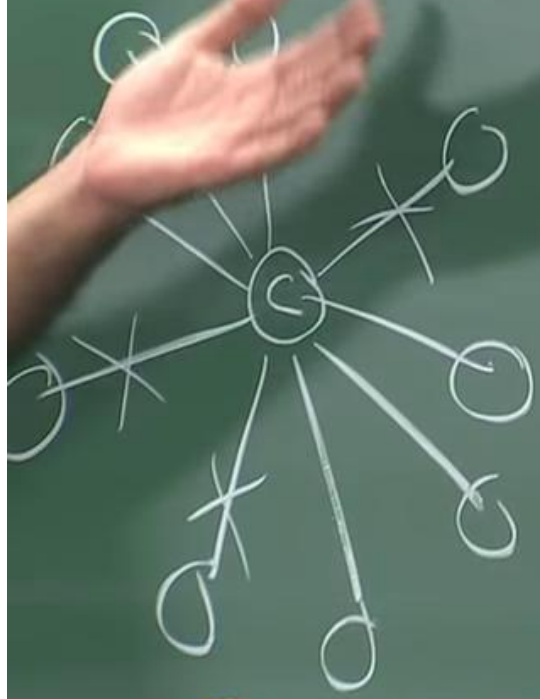
9-bit SAT

0 0 0 0 0 0 0 0 0

0 0 0 4 0 0 0 0 0

3

4



tabu tenure

used in bit moves



TABU SEARCH

M

0 0 0 0 0 0 0 0 0

t=1

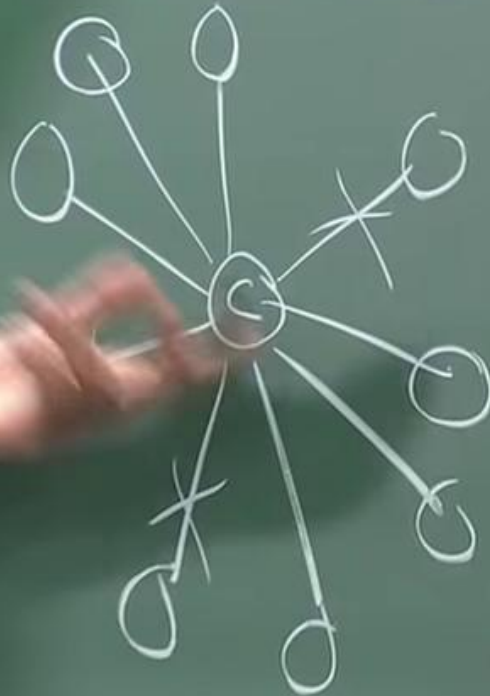
0 0 0 4 0 0 0 0 0

t=2

3

4

t=3



tabu tenure

changed in last moves

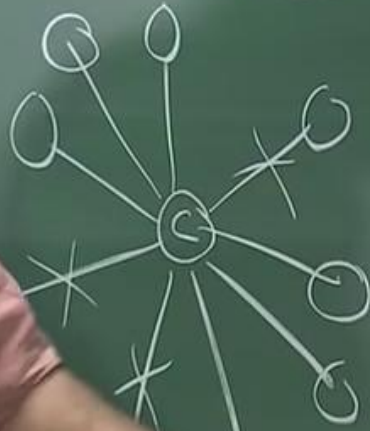
TABU SEARCH

9-bit SAT

M 000000000 t=1

000400000 t=2

3 4 t=3
2 3 4
1 0



tabu tenure
is changed in last moves
allow it

of gradient

TABU SEARCH

M

9-bit SAT

0 0 0 0 0 0 0 0 0 $t=1$

0 0 0 4 0 0 0 0 0 $t=2$



$t=3$

when(c)

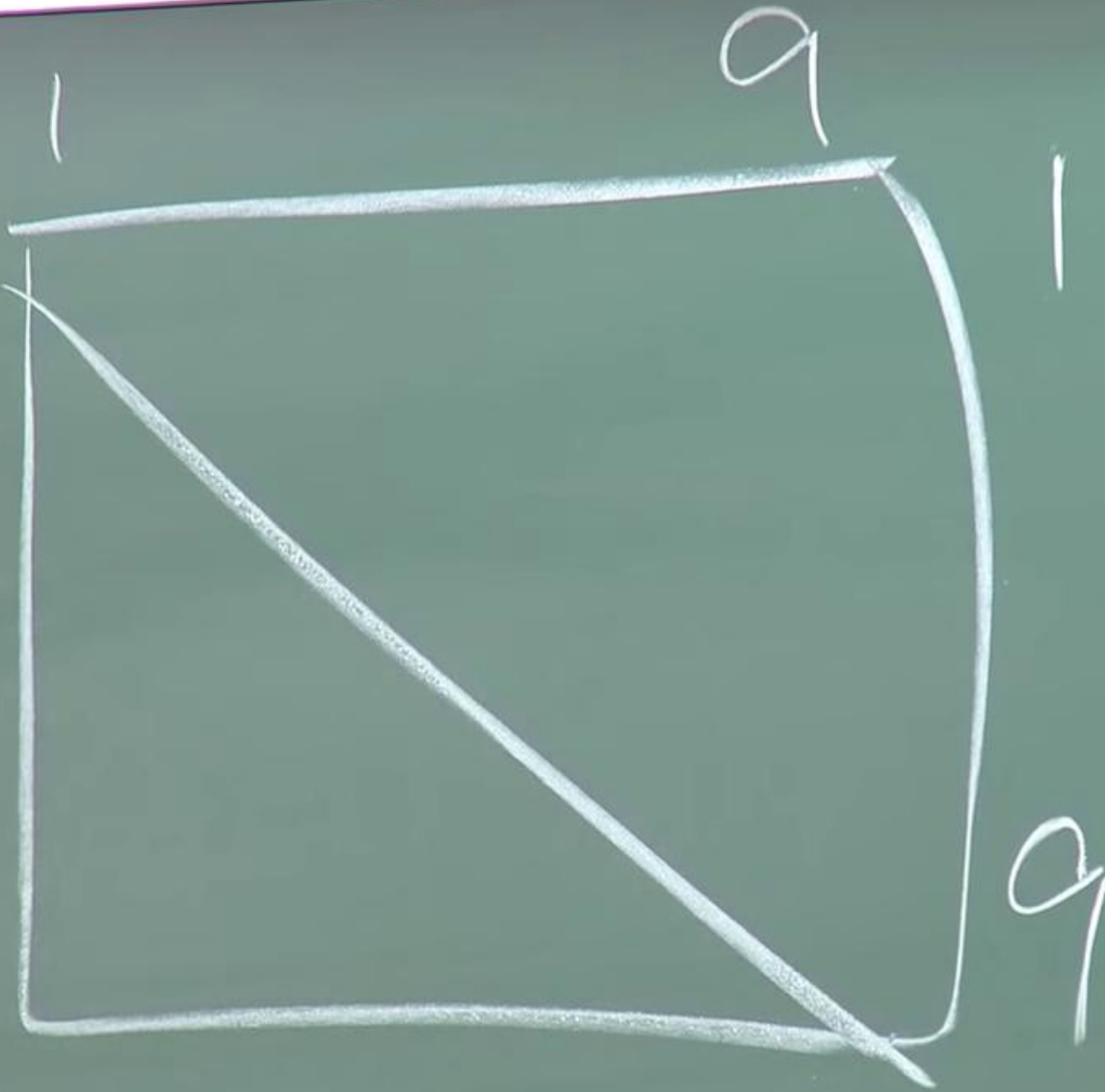
if a bit is changed in last moves then disallow it

tabu tenure

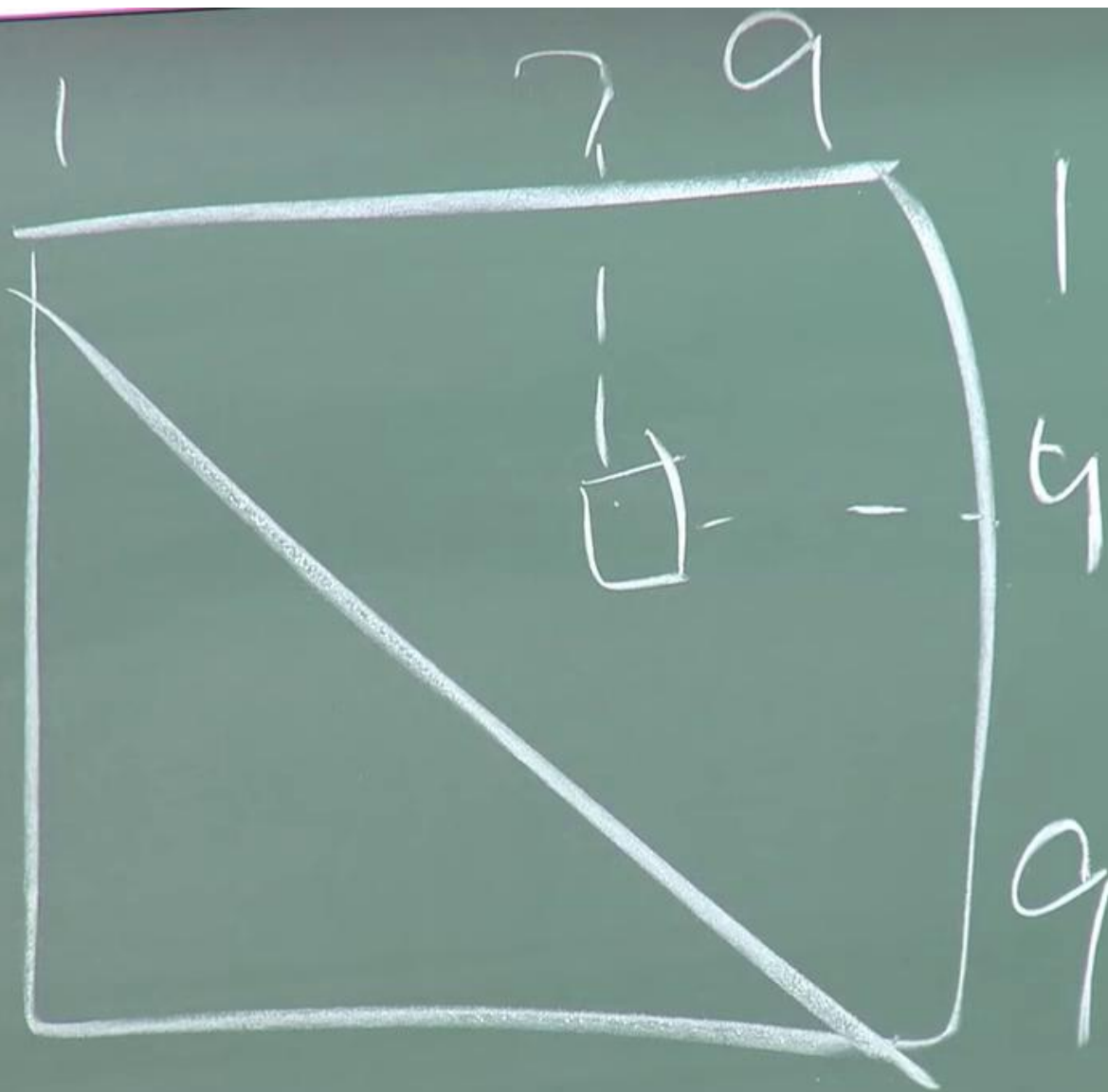


NPTEL

TSP



TSP



9-bit SAT

HM

000000000

$t=1$

000400000

$t=2$

3

4

$t=3$



Championing local optima

better than C

EXPLORATION

TABU SEARCH

9-bit SAT

$t=1$

000000000

$t=2$

000400000

$t=3$

tabu tenure

used in last moves

NPTEL

The chalkboard contains the following content:

- EXPLORATION**: A word in a box, with an arrow pointing to it from the text 'Championing local optima'.
- TABU SEARCH**: A word in a box.
- 9-bit SAT**: A title for a sequence of bit strings.
- Bit strings and transitions**:
 - $t=1$: 000000000
 - $t=2$: 000400000
 - $t=3$: A sequence of arrows pointing down from the bit strings, with numbers 3, 4, and 1 written next to them.
- Diagram**: A star graph with a central node 'C' and several peripheral nodes, some of which are marked with an 'X'.
- tabu tenure**: A phrase written below the bit strings.
- used in last moves**: A phrase written below the bit strings.
- NPTEL**: A logo in the bottom left corner.

ASPIRATION CRITERIA

↓
IF all allowed are bad
and a tabu move
leads to n
which is
better than best
THEN
allow the (tabu) move.



T
0000 t=1

0000 t=2

4 t=3

3



Hill Climbing $C = \text{current}$
 $n = \text{next}$

$n = \text{Best}(\text{MoveGen}(C))$ is better than C

$n \leftarrow C$

EXPLOITATION of gradient

$n \leftarrow \text{Best}(\text{Allowed}(\text{MoveGen}(C)))$

Until some termination criteria
(store best solution found)

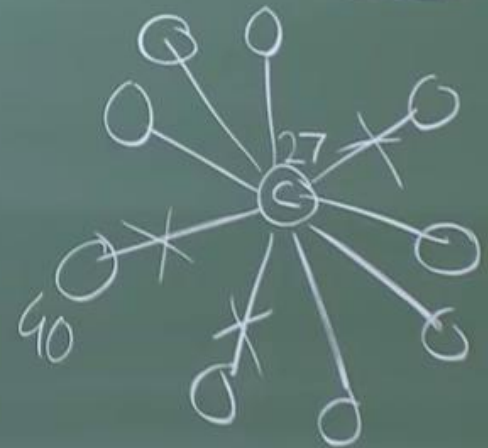
CLOSED Queue

Mark recent moves. eg. SAT - if a bit is changed in last Time
then disallow it

Escaping local optima

EXPLORATION

TABU SEARCH



ASPIRATION
CRITERIA

↓
IF all allowed and
and a time

Frequency

frequency
F = [18 7 6 7 (2) -11 -14 18]
↓

t=1

t=2

t=3

Frequency

frequency

F

18	7	6	7	2	11	14	18
----	---	---	---	---	----	----	----



Frequency

frequency

F

18	7	6	7	2	11	-14	18
----	---	---	---	---	----	-----	----

↓

d

$$eval'(n) \leftarrow eval(n) - k F(n)$$

ASPIRATION
CRITERIA

↓
IF all allowed
and a tabu
lead

t=1

t=2

t=3

THEN
al

Frequency

frequency

18	7	6	7	2	11	14	18
----	---	---	---	---	----	----	----

$$eval'(n) \leftarrow eval(n) - k F(b_n)$$



Frequency

frequency

F	18	7	6	7	2	11	14	18
---	----	---	---	---	---	----	----	----

nd

$$\text{eval}'(n) \leftarrow \text{eval}(n) - k F(b_n)$$



Penalty for
frequent moves

STOCHASTIC / RANDOMIZED methods

C = current
 n = next
 $f(\text{Move}(C))$ is better than $f(C)$
 $\leftarrow C$

Escaping local Optima

TSP

EXPLORATION

EXPLORATION of gradient

TABU SEARCH

9-bit SAT

000000000	$t=1$
000400000	$t=2$
34	$t=3$

tabu tenure

but is changed in last move
disallow it

STOCHASTIC / RANDOMIZED methods



→ Random Walk

$n \leftarrow \text{random-neighbour}(c)$

Random Walk

$n \leftarrow \text{random-neighbor}(c)$

STOCHASTIC / RANDOMIZED methods



Random Walk

$n \leftarrow \text{random-neighbour}(c)$

Make a random move with a
probability \propto improvement
in $\text{eval}(n)$

bring back Optimal TSP

EXPAN

TABU SEARCH

9-bit SAT

000000000 t=1

000400000 t=2

3 4 t=3

2 3 4

1 2 3

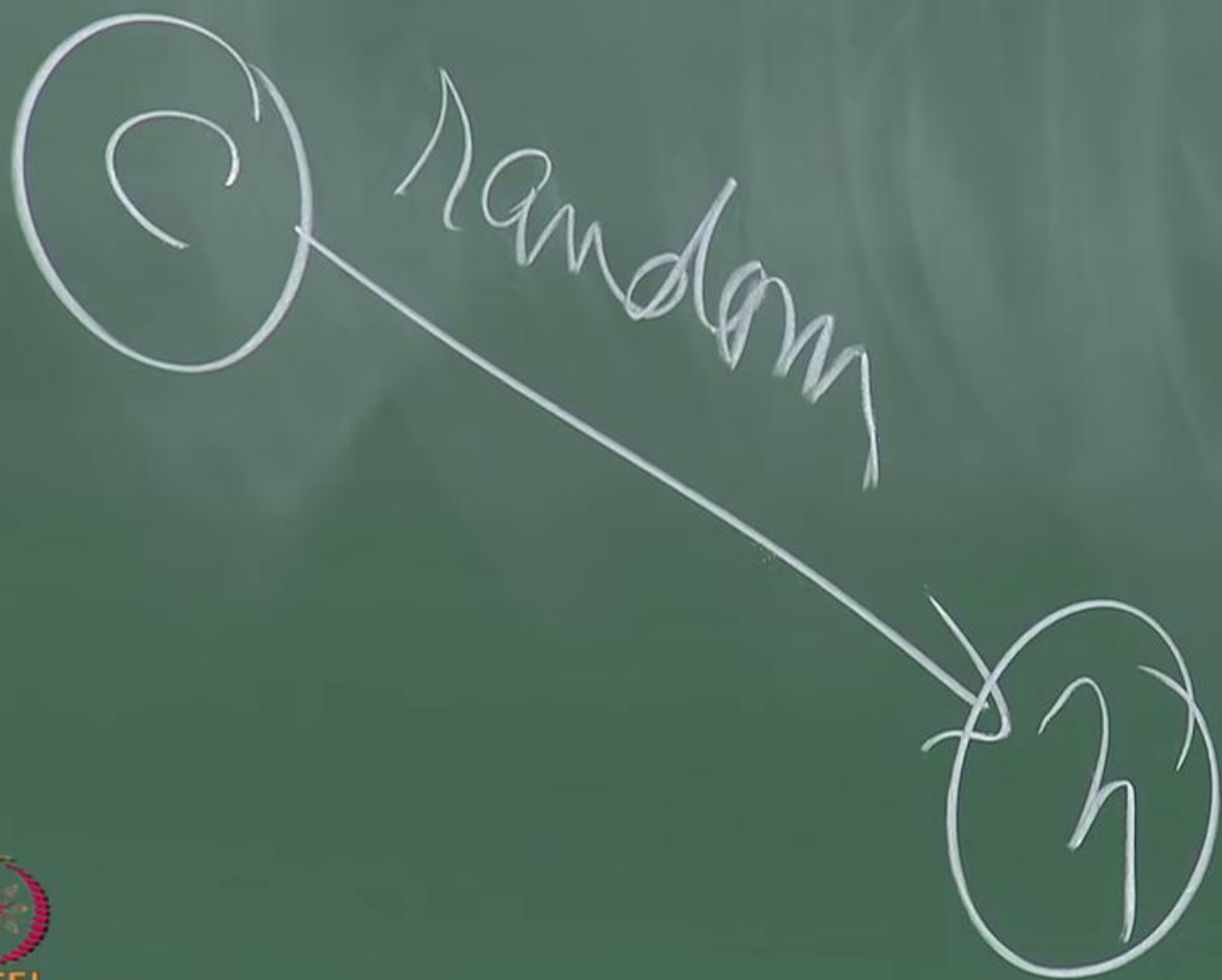
tenure

STOCHASTIC / RANDOMIZED

Random

Make a random move probability α

random



in eval(n)

$$\text{eval}(n) - \text{eval}(c)$$

STOCHASTIC / RANDOMIZED methods



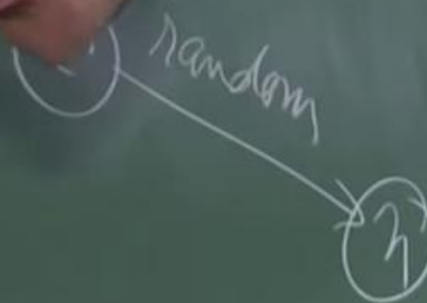
Random Walk



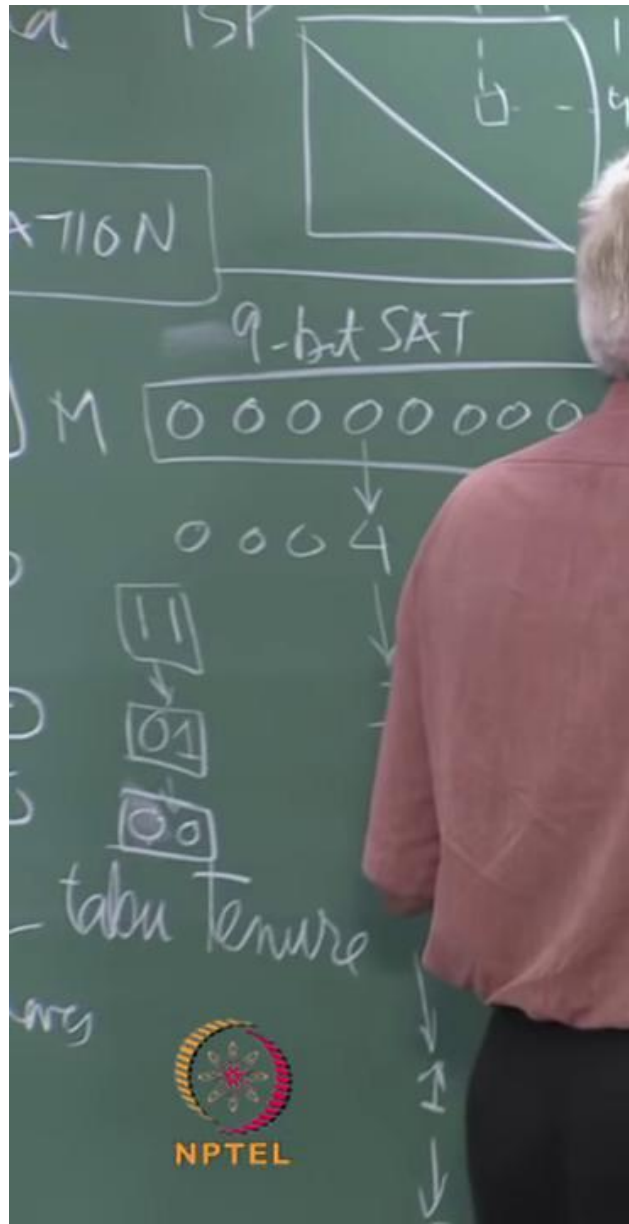
$n \leftarrow \text{random-neighbor}$



Make a random move with a
P probability \propto improvement
in $\text{eval}(n)$



$$\boxed{\text{eval}(n) - \text{eval}(c)}$$



STOCHASTIC / RANDOMIZED methods



Random Walk

$n \leftarrow \text{random-neighbour}$



Make a random move with a
 P probability \propto improvement
in $\text{eval}(n)$

① random

$$\boxed{\text{eval}(n) - \text{eval}(c)}$$

"If the number happens to be greater than p then you DON'T make the move"
"If the number happens to be less than p then you make the move"

Random number.



O

P