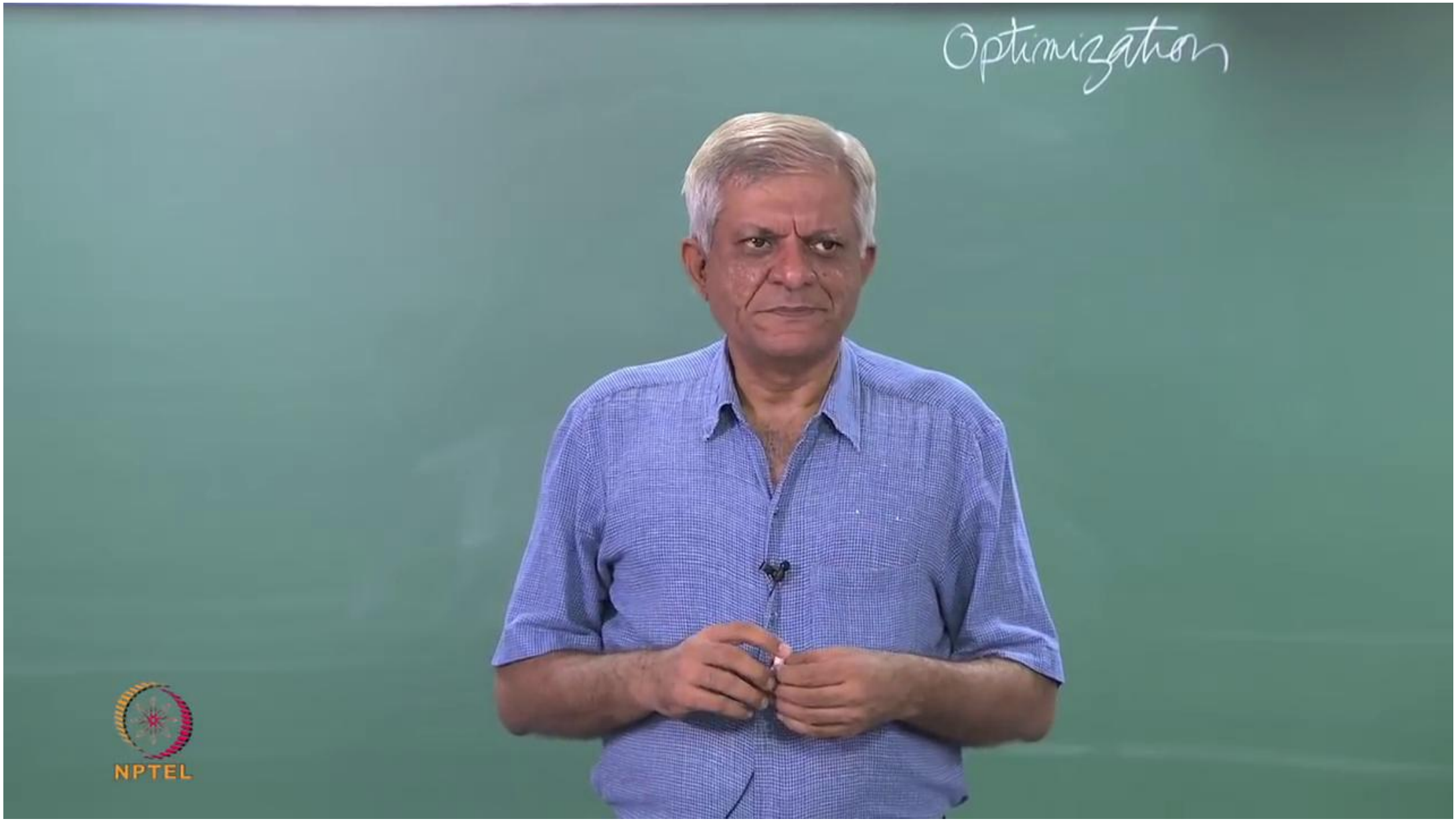
Video Lectures On

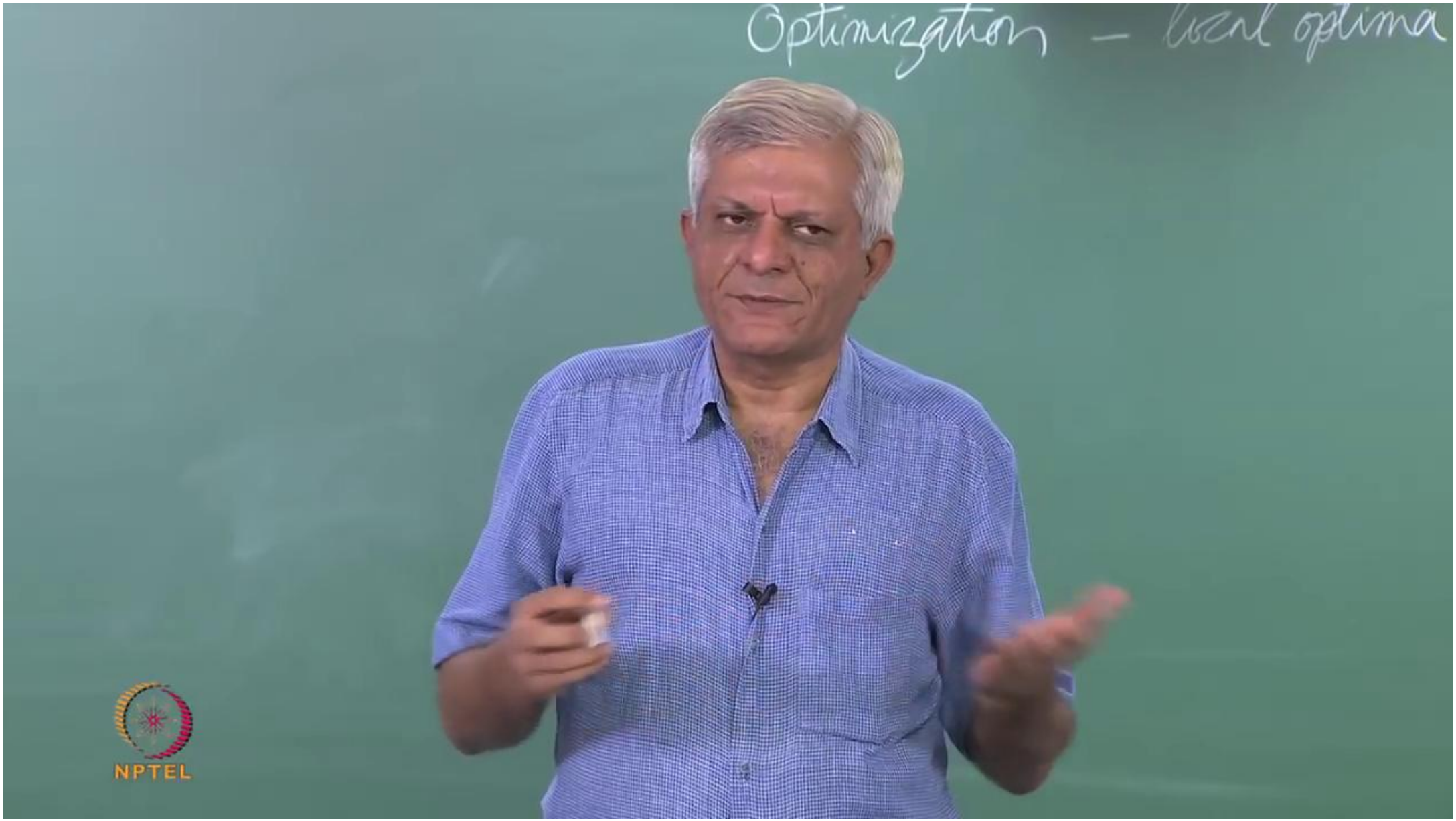
Artificial Intelligence

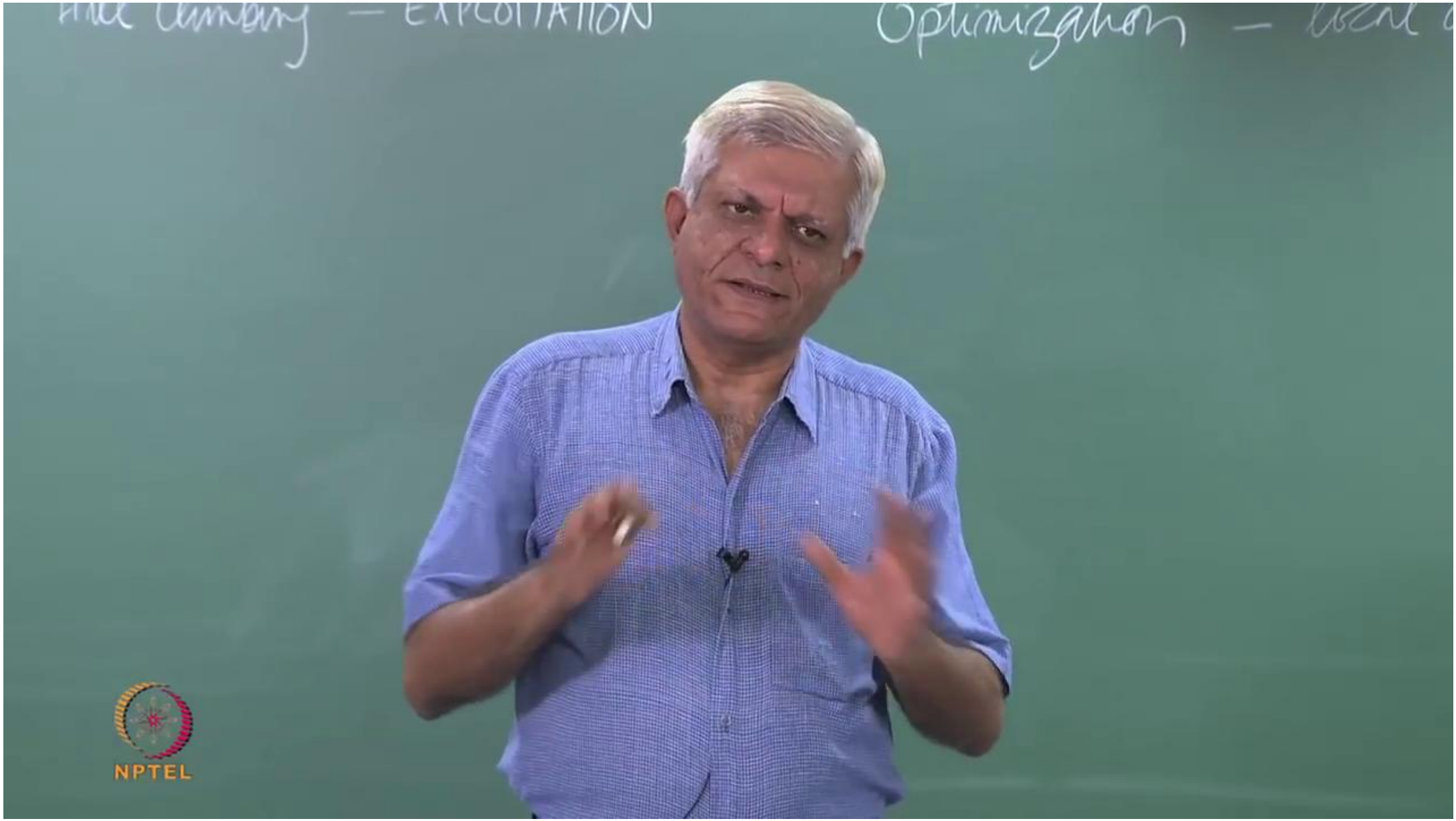
Lecture 14

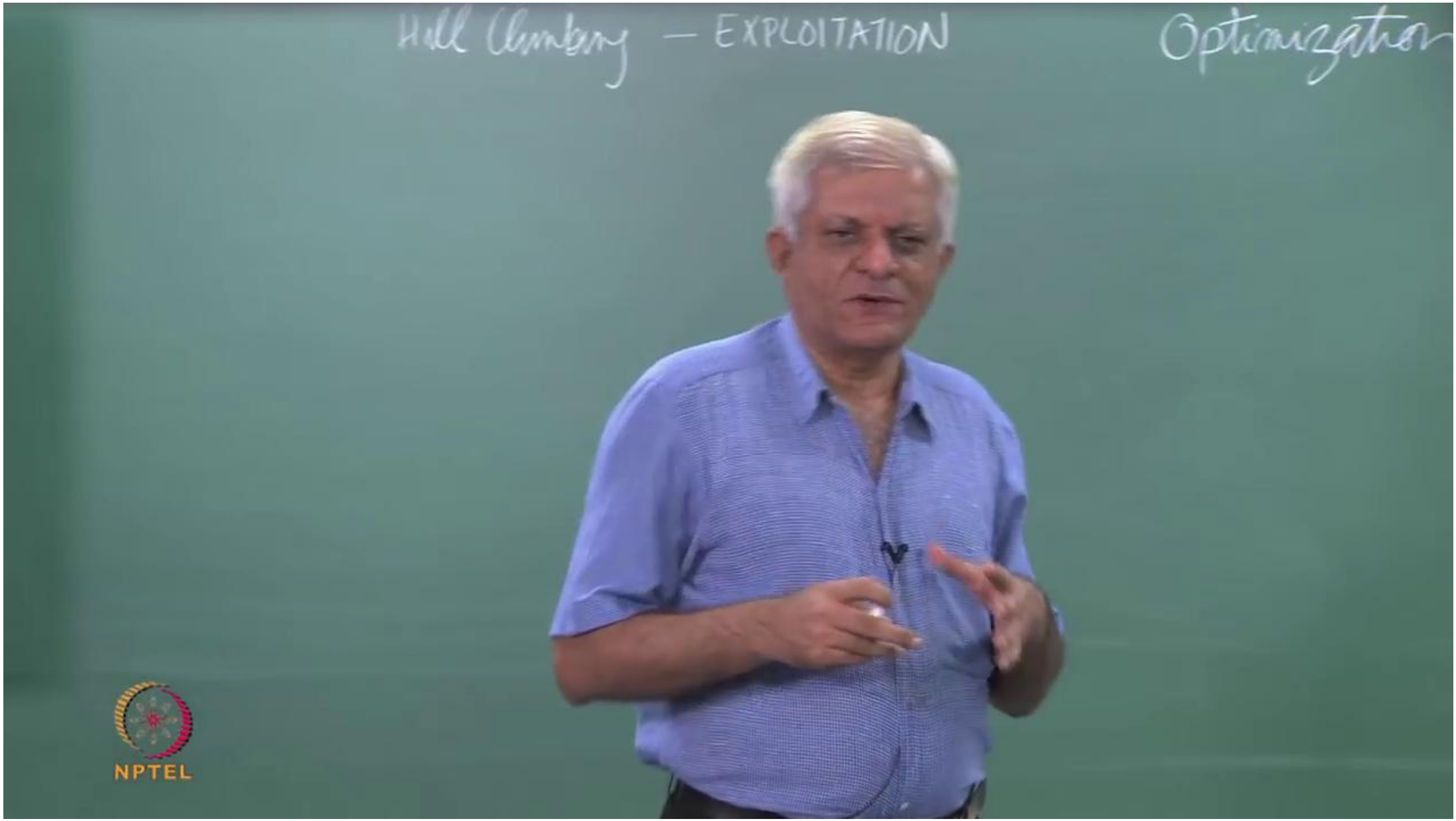
Optimization I (Simulated Annealing)

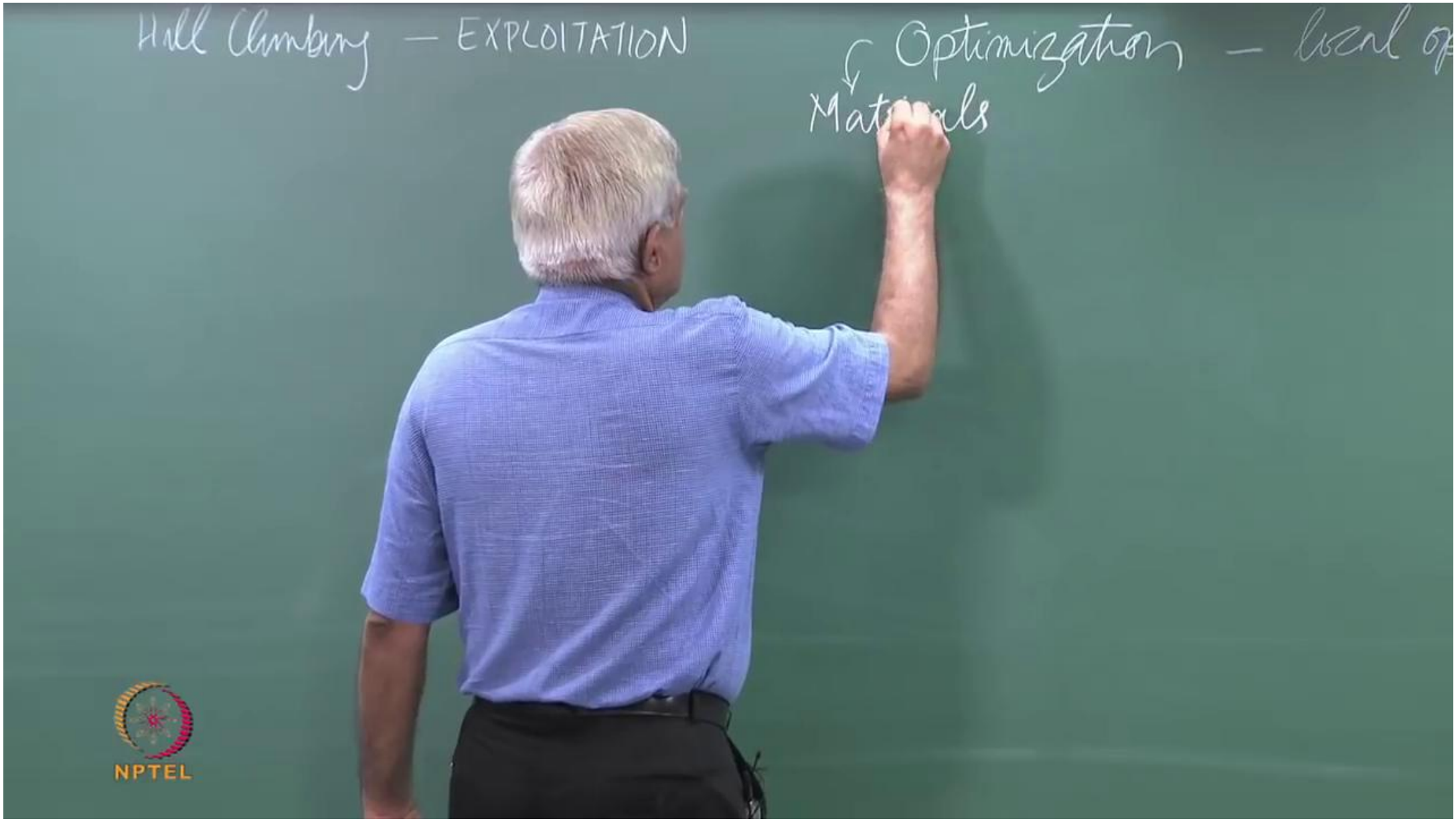
Prof. Deepak Khemani
Department of Computer Science
IIT Madras


Optimization
NPTEL


Optimization – local optima
NPTEL


Optimization
NPTEL


Hill Climbing – EXPLOITATION
Optimization
NPTEL


Hill Climbing – EXPLOITATION
Optimization
Materials
NPTEL

Optimization – local optima

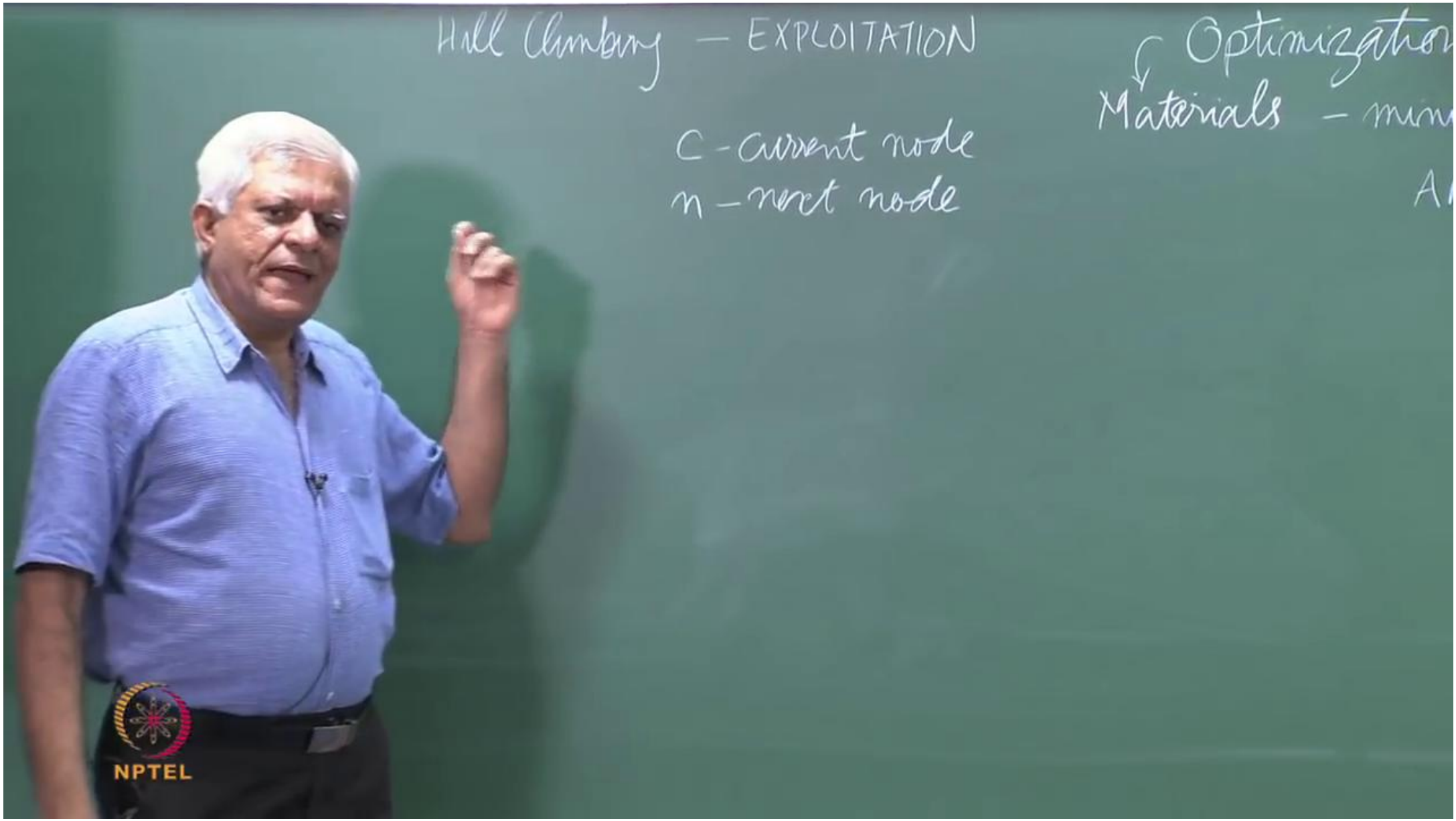
Materials – minimum energy

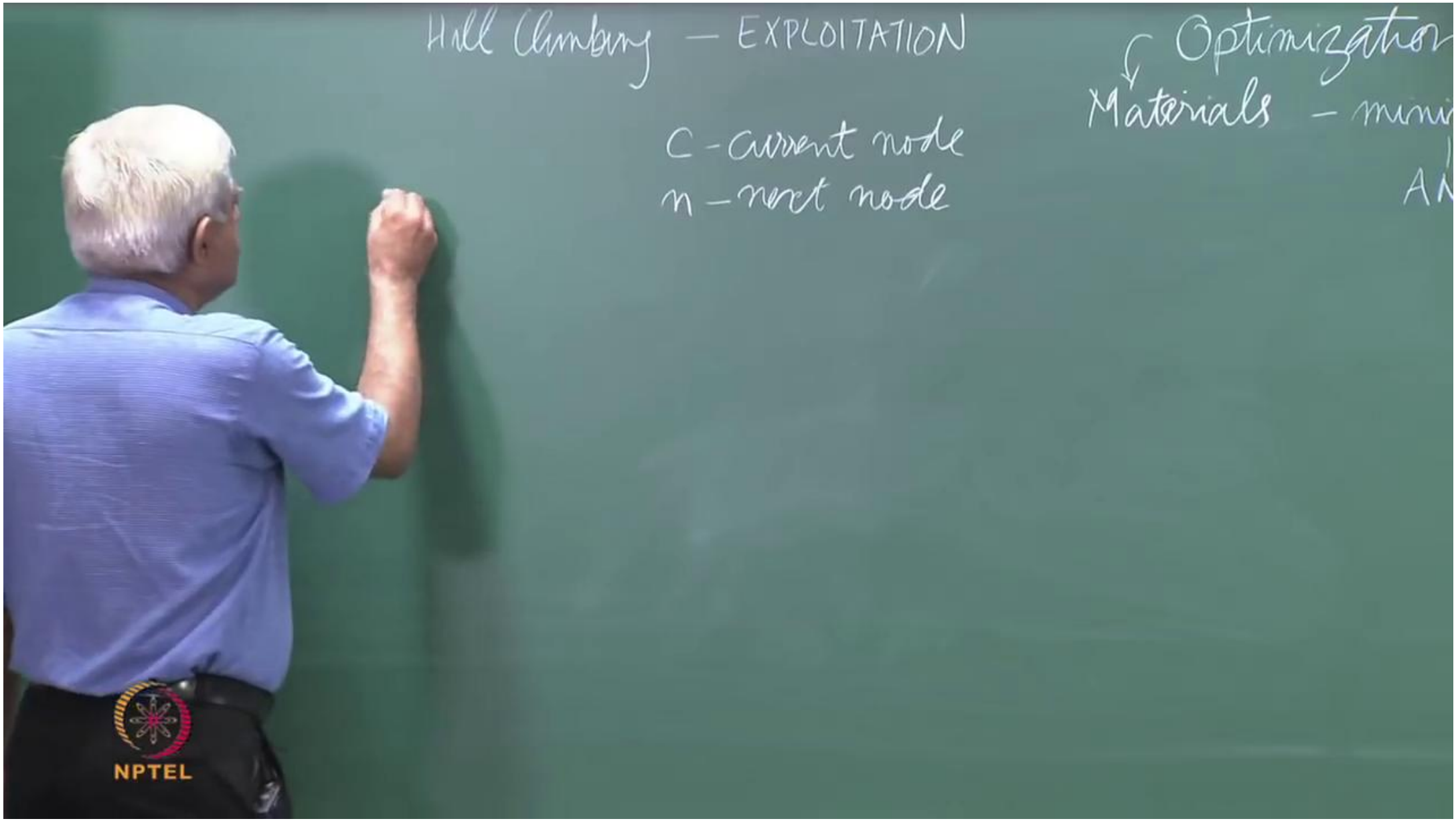
Optimization – local optima

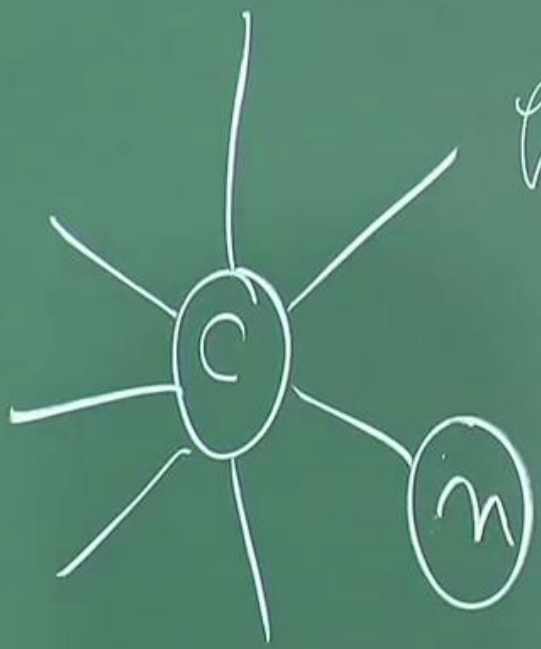
Materials – minimum energy

ANNEALING – controlled cooling

Random Walk – EXPLORATION

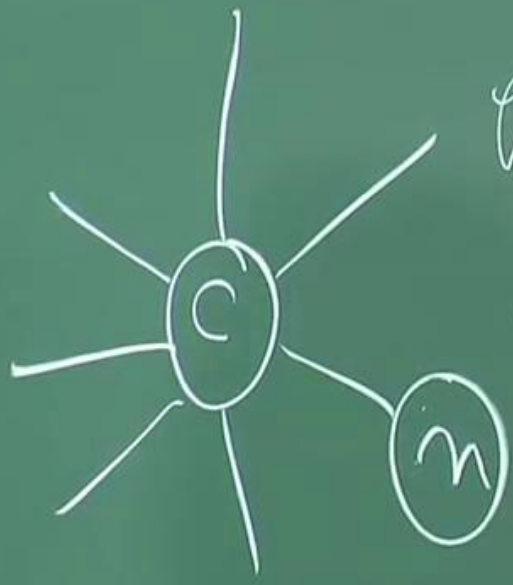

Hill Climbing – EXPLOITATION
C - current node
n - next node
Optimization
Materials – min
A
NPTEL


Hill Climbing – EXPLOITATION
C - current node
n - next node
Optimization
Materials – mini
NPTEL



val(c) → ⓒ - current node

val(n) → ⓝ - next node

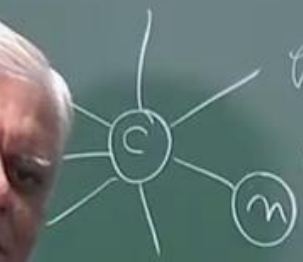


val(c) → (c) - current node

val(n) → (n) - next node

$$\Delta E = val(n) - val(c)$$

Hill Climbing – EXPLOITATION



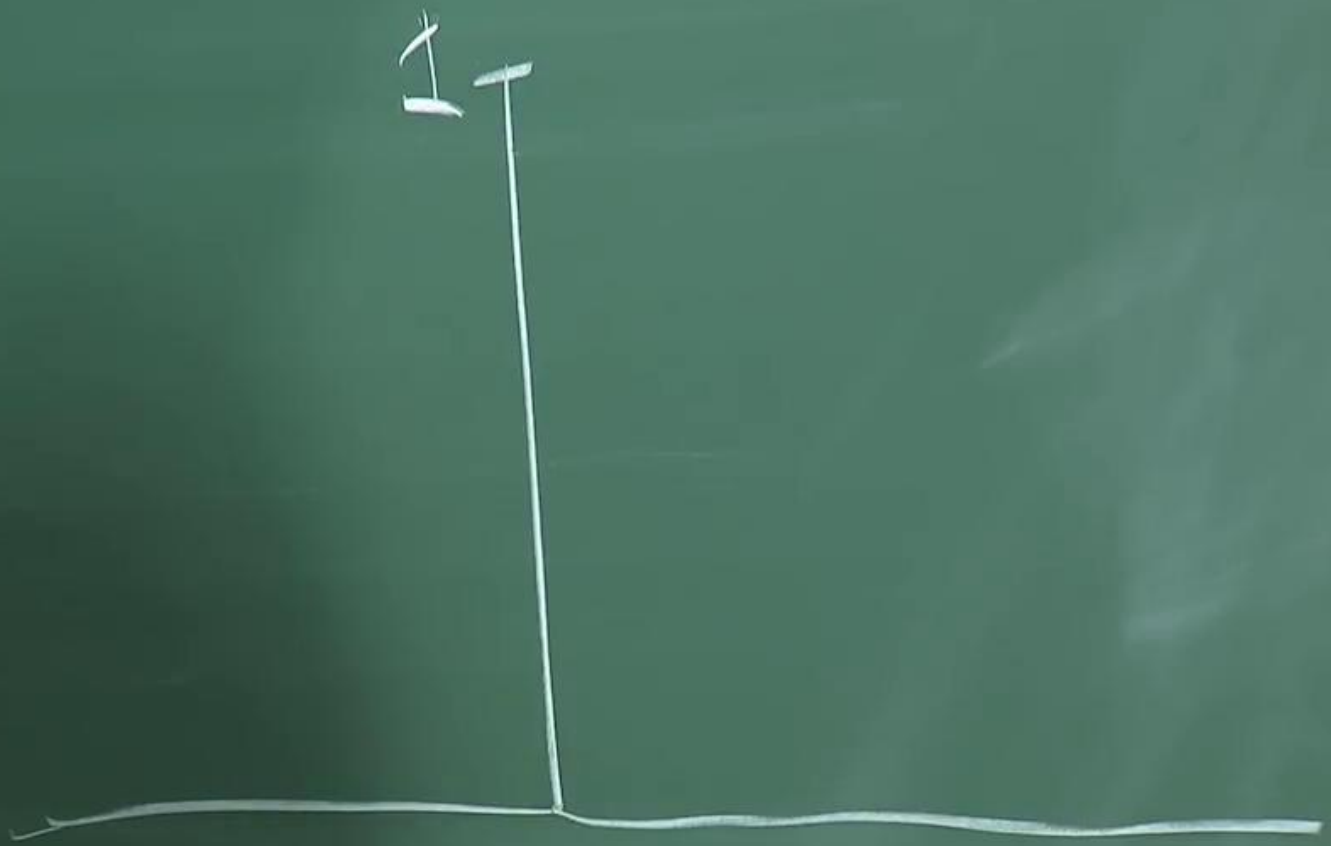
eval(c) → (c) - current node

eval(n) → (n) - next node

$\Delta E = eval(n) - eval(c)$

Optimizat

Materials – m


1
ΔE → 0
ΔE → ∞

$$P(c,m) = \frac{1}{1 + e^{-\Delta E}}$$

$$P(c,n) = \frac{1}{1 + e^{-\Delta E / T}}$$

$n \leftarrow$ random-neighbour (c)

evaluate ΔE

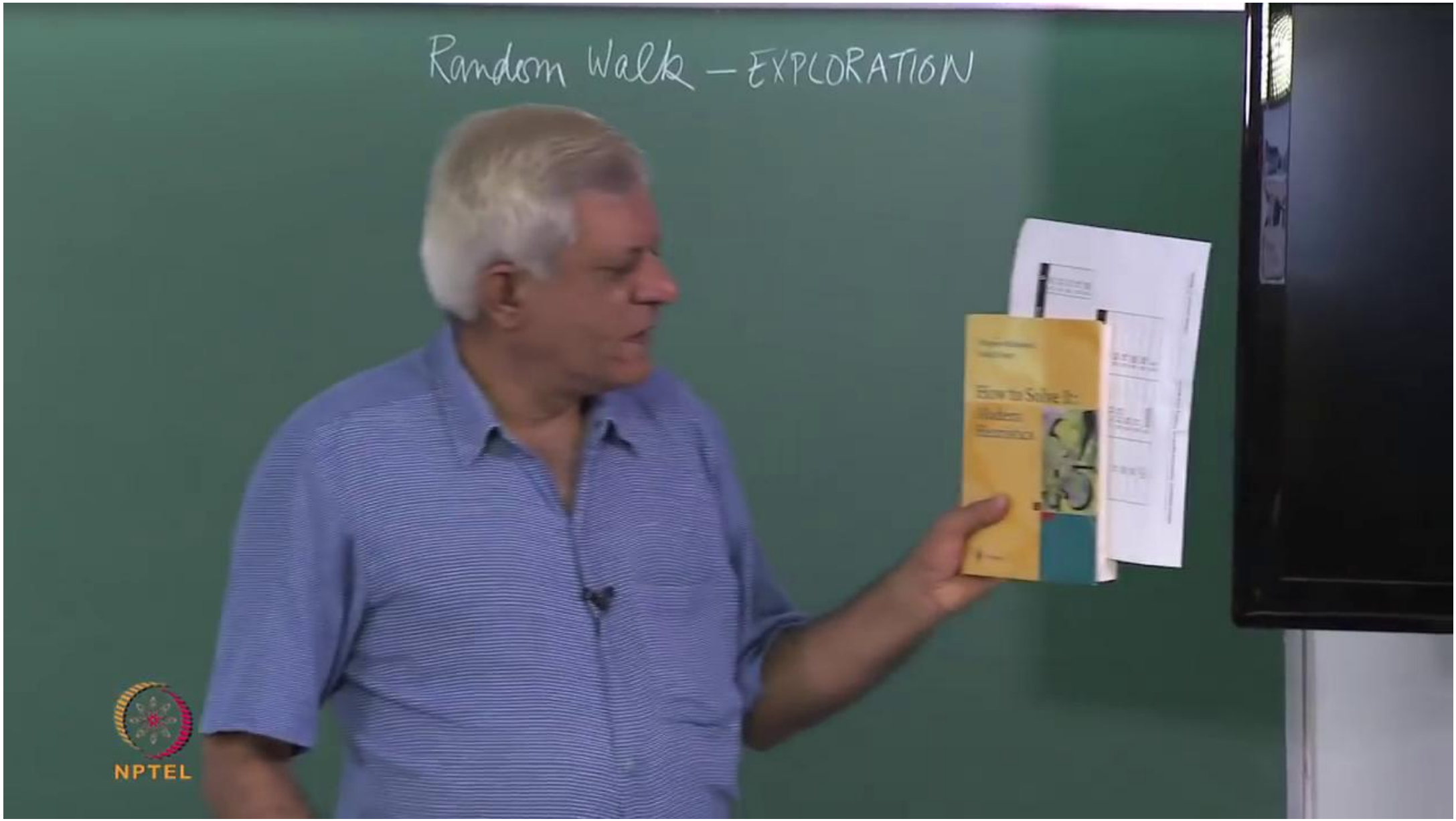
Move with probability $P(c,n) = \dfrac{1}{1 + e^{-\Delta E / T}}$

STOCHASTIC HC

$n \leftarrow$ random-neighbour (c)

evaluate ΔE

Move with probability $P(c,n) = \dfrac{1}{1 + e^{-\Delta E / T}}$


Random Walk – EXPLORATION
How to Solve It:
Modern
NPTEL

effect of ΔE, $T = 10$, eval(c) = 107

eval(n)	$-\Delta E$	$e^{-\Delta E/T}$	P
80	27	14.88	0.06

effect of ΔE, $T = 10$, eval(c) = 107

eval(n)	$-\Delta E$	$e^{-\Delta E/T}$	P
80	27	14.88	0.06
100	7	2.01	0.33

effect of ΔE, $T = 10$, eval(c) = 107

eval(n)	$-\Delta E$	$e^{-\Delta E/T}$	P
80	27	14.88	0.06
100	7	2.01	0.33
107	0	1.0	0.50

effect of ΔE, $T = 10$, $eval(c) = 107$

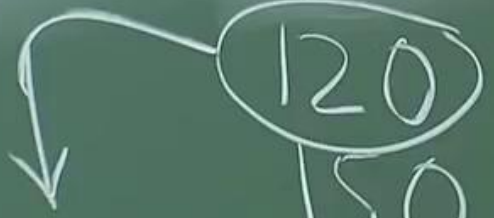
$eval(n)$	$-\Delta E$	$e^{-\Delta E/T}$	P
80	27	14.88	0.06
100	7	2.01	0.33
107	0	1.0	0.50
120	-13	0.27	0.78

effect of ΔE, $T = 10$, $eval(c) = 107$

$eval(n)$	$-\Delta E$	$e^{-\Delta E/T}$	P
80	27	14.88	0.06
100	7	2.01	0.33
107	0	1.0	0.50
120	−13	0.27	0.78
150	−43	0.01	0.99

effect of ΔE, $T = 10$, eval(c) = 107

eval(n)	$-\Delta E$	$e^{-\Delta E/T}$	P
80	27	14.88	0.06
100	7	2.01	0.33
107	0	1.0	0.50
120	-13	0.27	0.78
150	-43	0.01	0.99



effect of ΔE, $T = 10$, eval(c) = 107

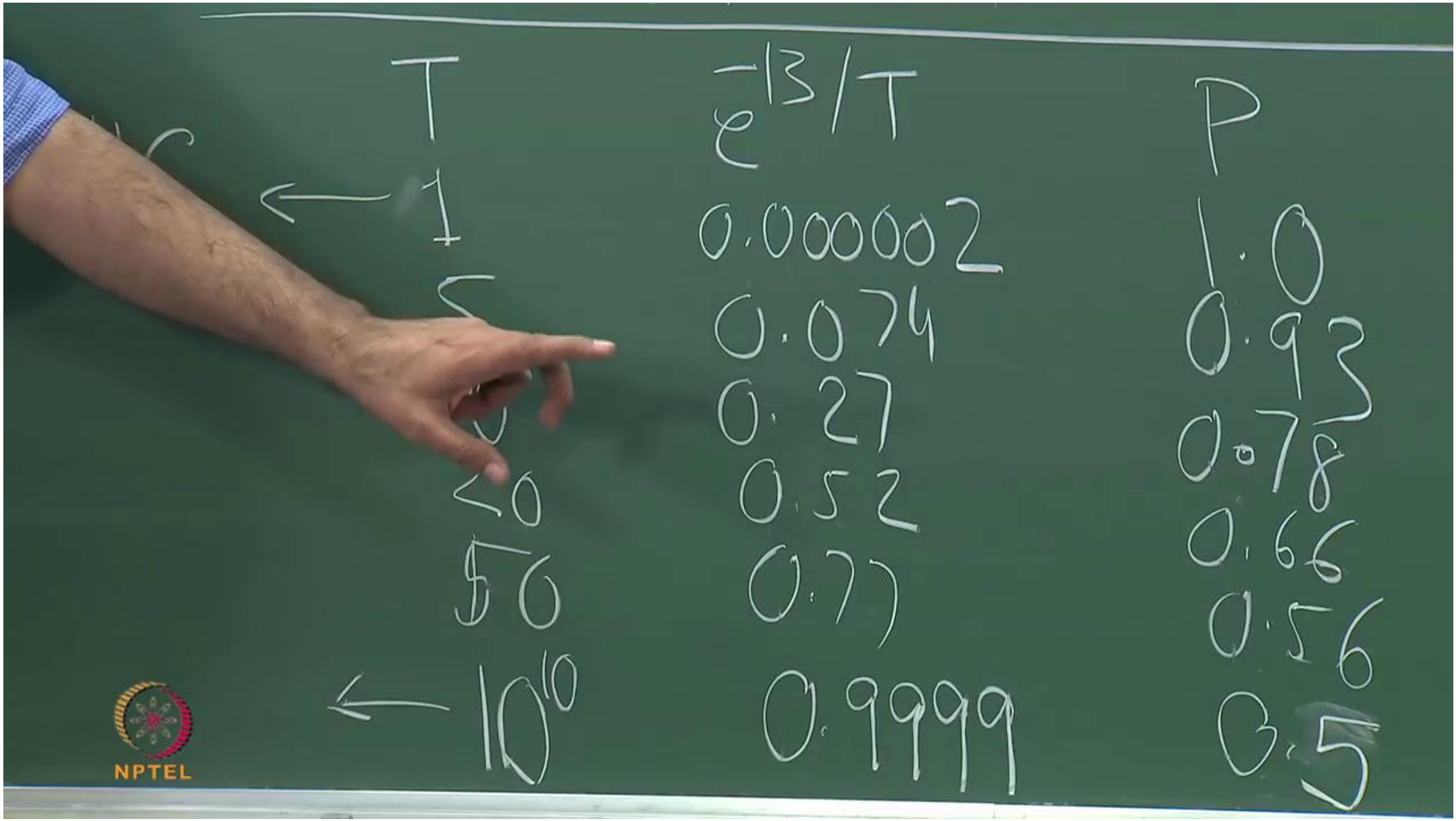
eval(n)	$-\Delta E$	$e^{-\Delta E/T}$	P
80	27	14.88	0.06
100	7	2.01	0.33
107	0	1.0	0.50
120	-13	0.27	0.78
150	-43	0.01	0.99

T	$e^{-13/T}$	P

T	$e^{-13/T}$	P
1	0.000002	1.0

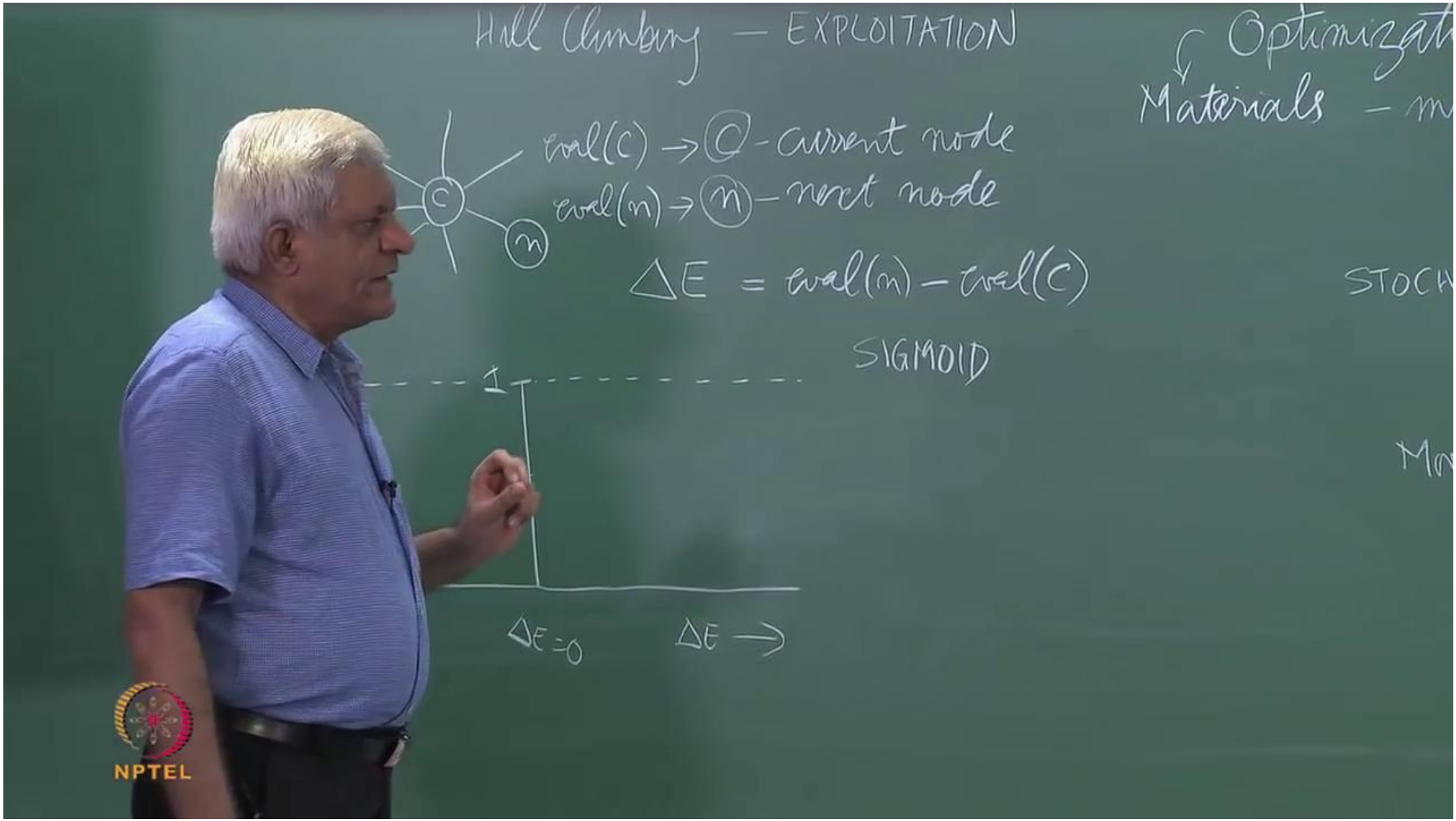
T	$e^{-\beta/T}$	P
1	0.000002	1.0
5	0.074	0.93
10	0.27	0.78
20	0.52	0.66
50	0.77	0.56
10^{10}	0.9999	0.5

HC ←	T	$e^{-B/T}$	P
	1	0.000002	1.0
	5	0.074	0.93
	10	0.27	0.78
	20	0.52	0.66
	50	0.77	0.56
	10^{10}	0.9999	0.5


T
$e^{-13/T}$
P
← 1
0.000002
1.0
5
0.074
0.93
0.27
0.78
<0
0.52
0.66
50
0.77
0.56
← 10^{10}
0.9999
0.5
NPTEL

↓	150	-43	0.01	0.9

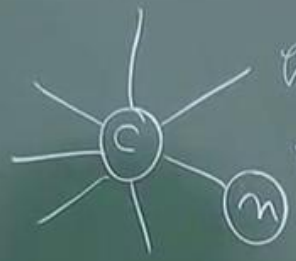
	T	$e^{-\beta/T}$	P
HC ←	1	0.000002	1.0
	5	0.074	0.93
	10	0.27	0.78
	20	0.52	0.66
	50	0.77	0.56
Random Walk ←	10^{10}	0.9999	0.5


Hill Climbing – EXPLOITATION
Optimizat
Materials – m
eval(c) → c - current node
eval(n) → n - next node
ΔE = eval(n) – eval(c)
STOCH
SIGMOID
1
ΔE = 0
ΔE →
Ma
NPTEL

Hill Climbing — EXPLOITATION

Optimizat

Materials — m



eval(c) → (c) - current node

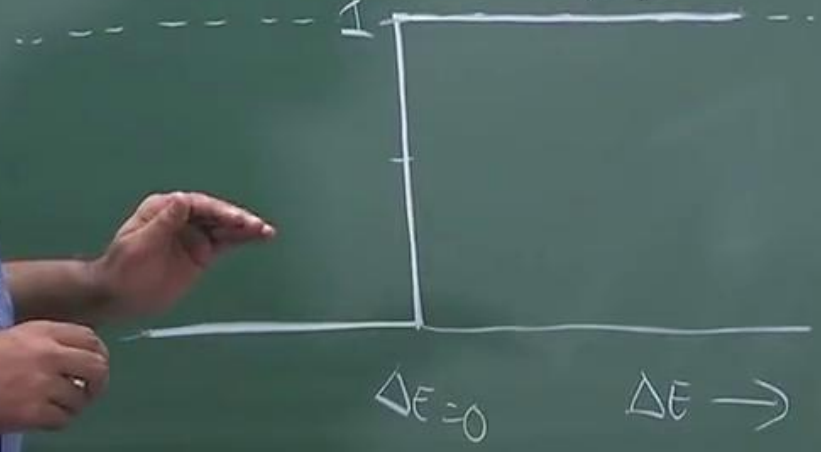
eval(n) → (n) - next node

$\Delta E = eval(n) - eval(c)$

STOCH

T=1

SIGMOID



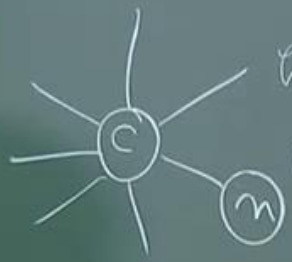
Ma

NPTEL

Hill Climbing – EXPLOITATION

Optimiz

Materials



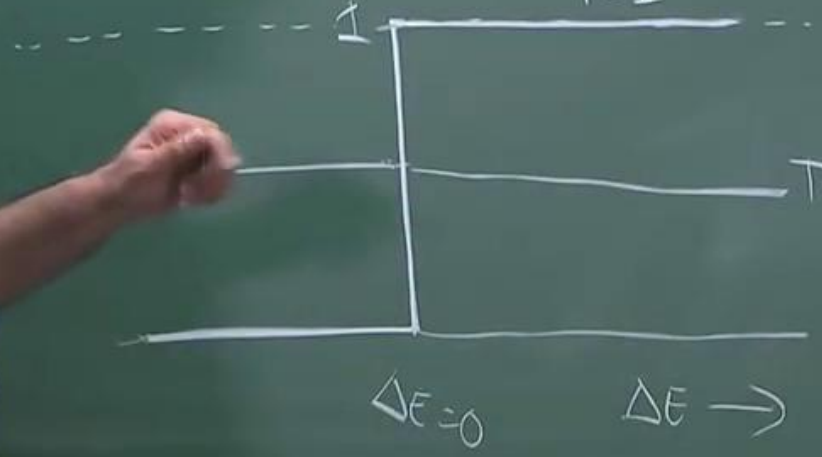
eval(C) → C - current node

eval(n) → n - next node

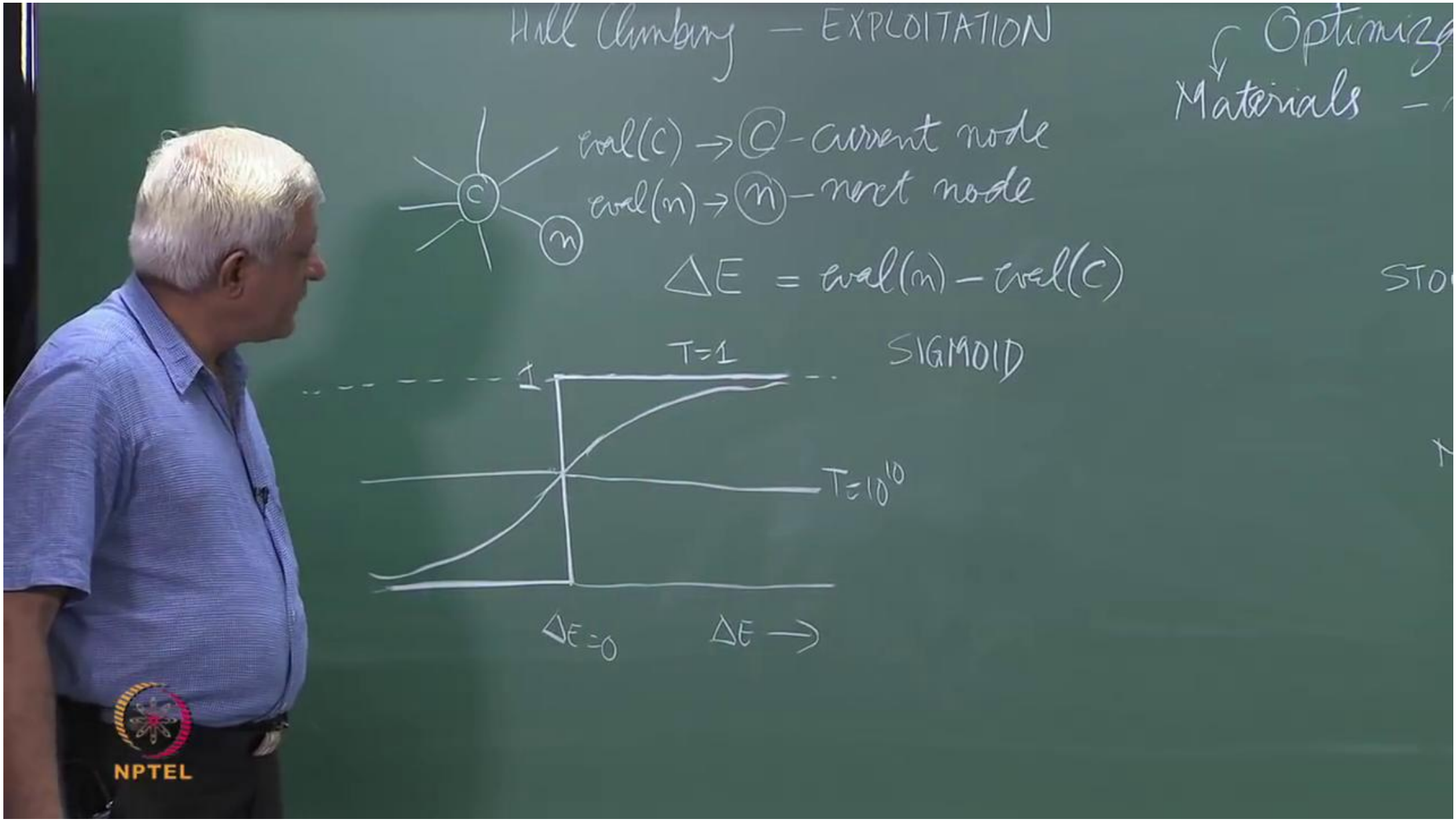
$\Delta E = eval(n) - eval(C)$

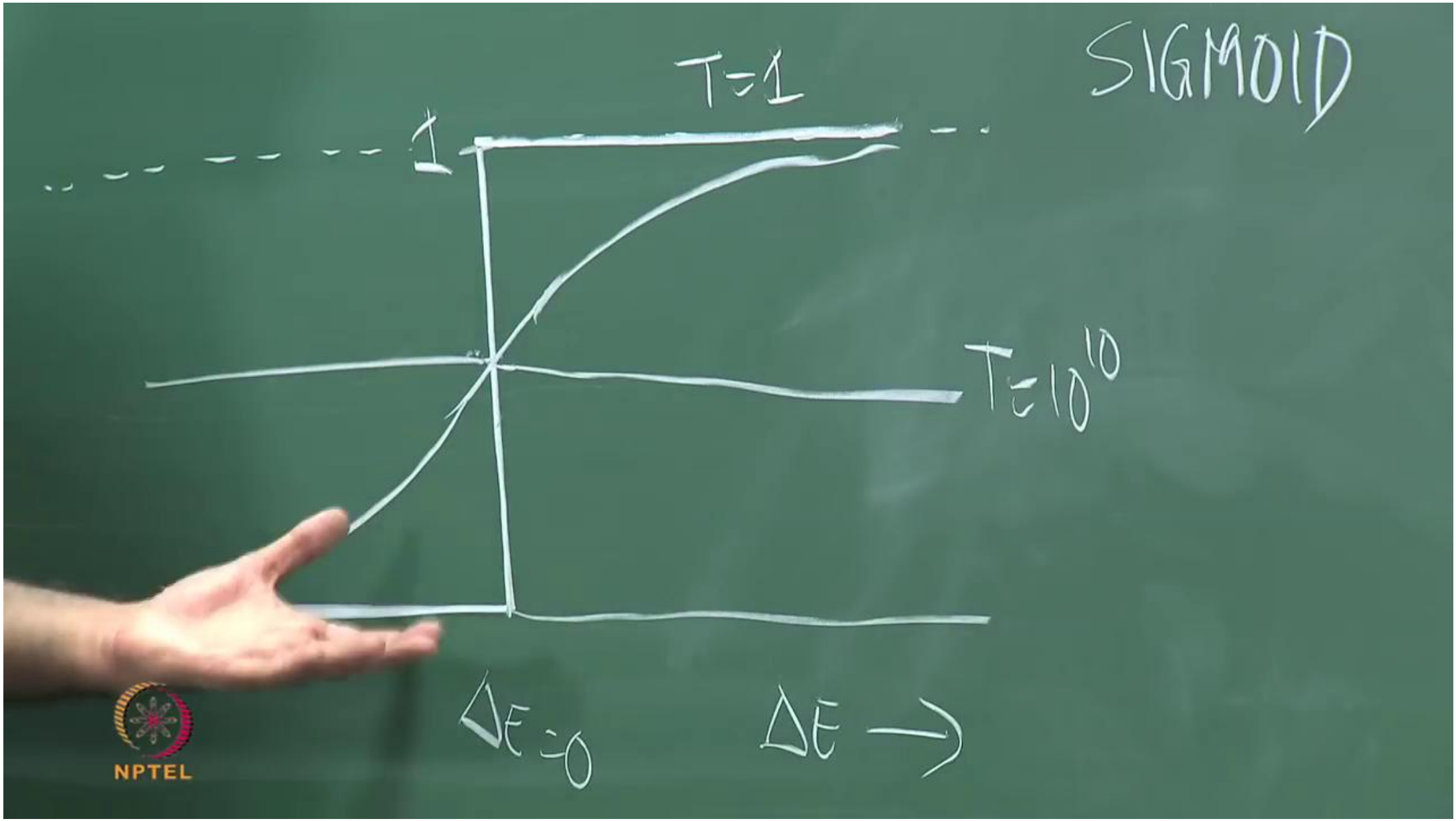
ST

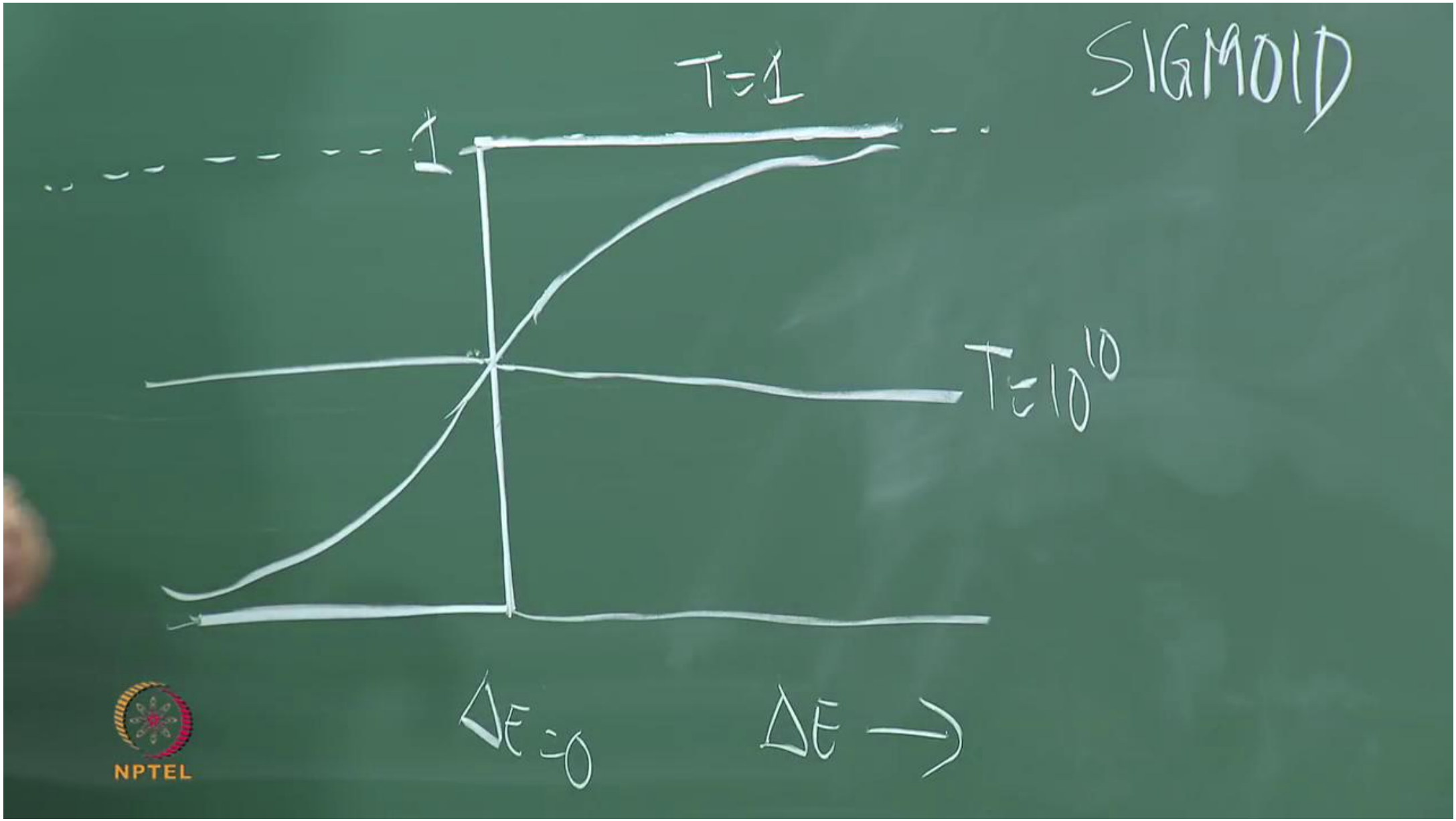
SIGMOID

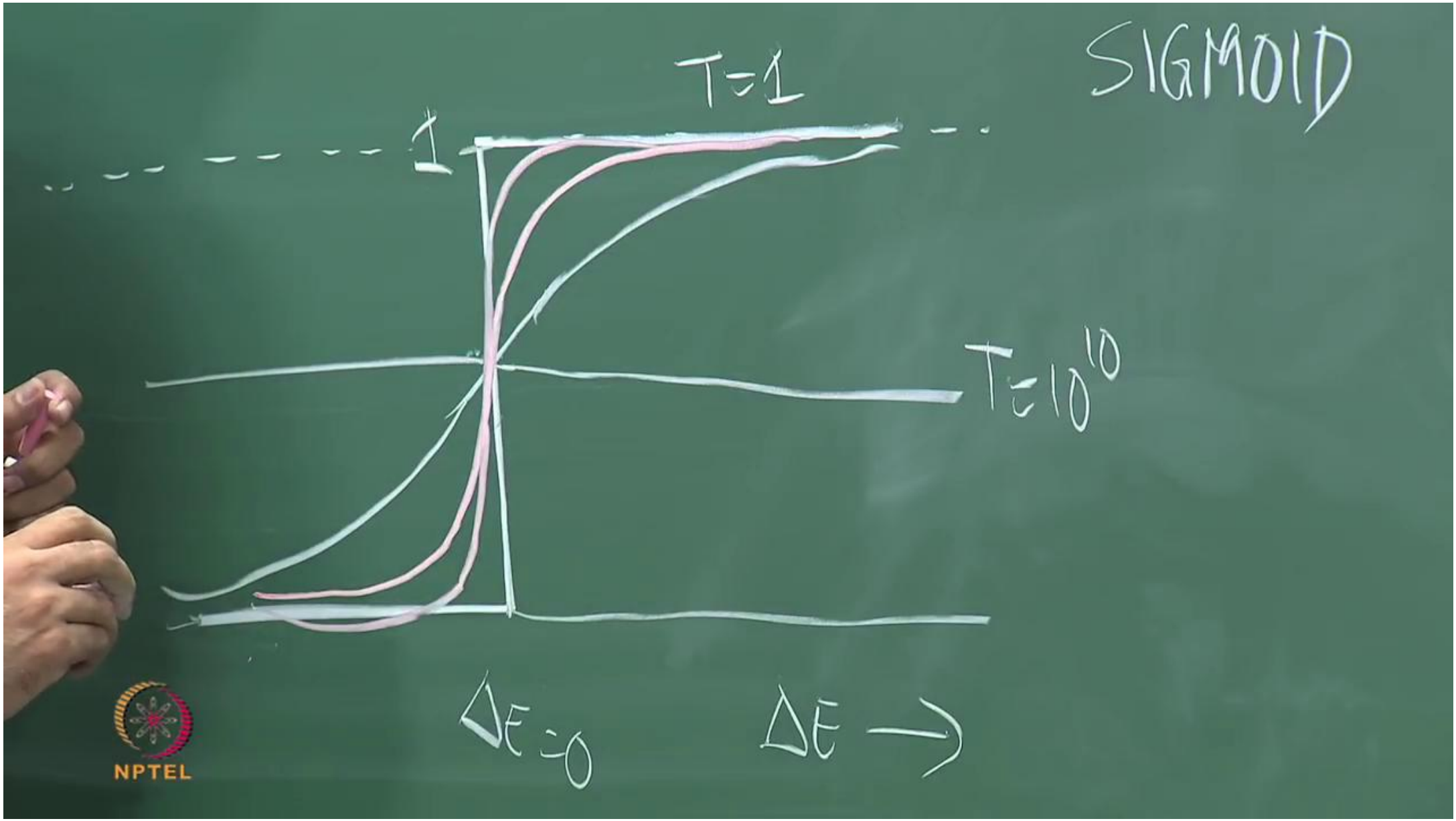


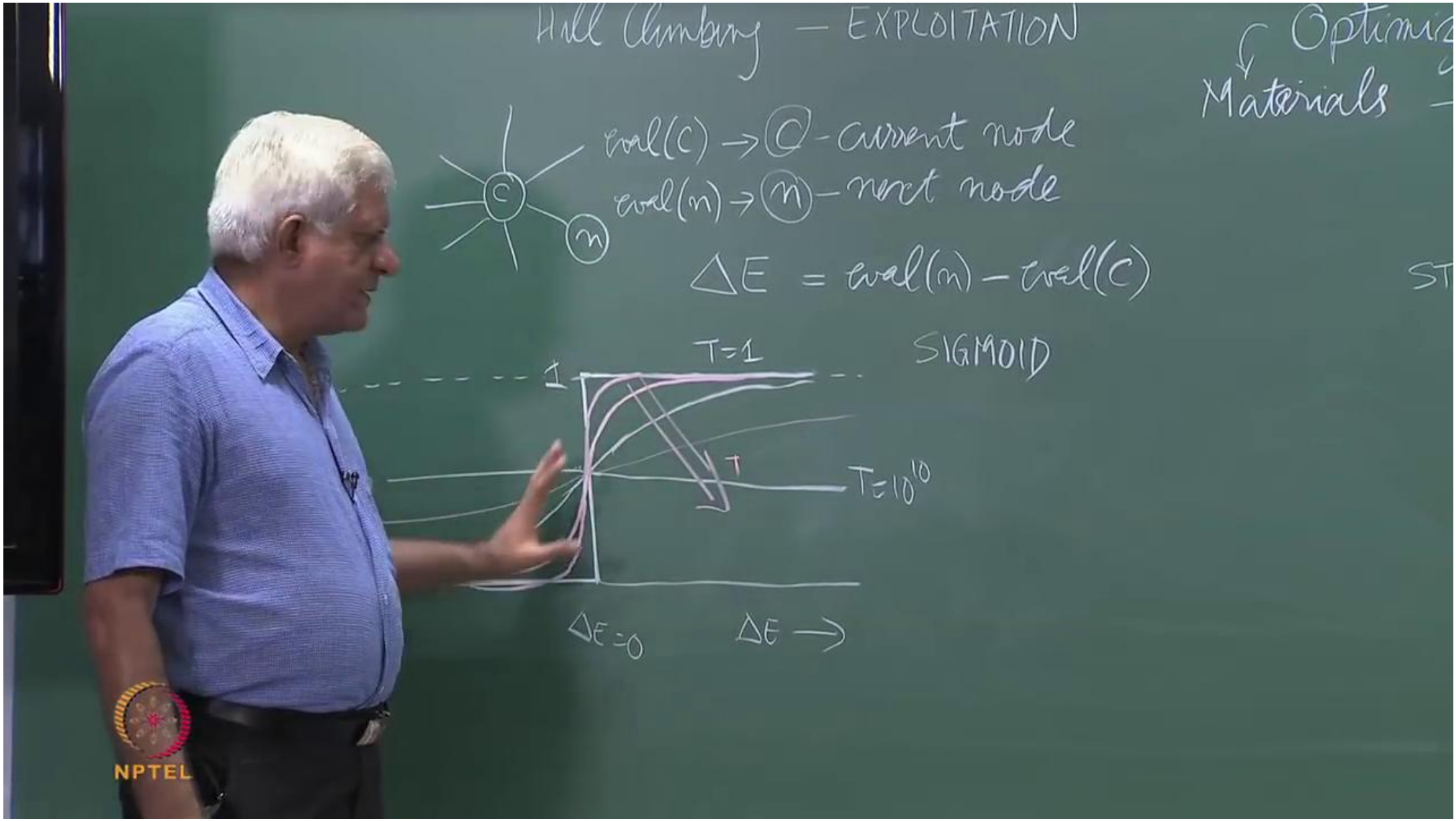
NPTEL

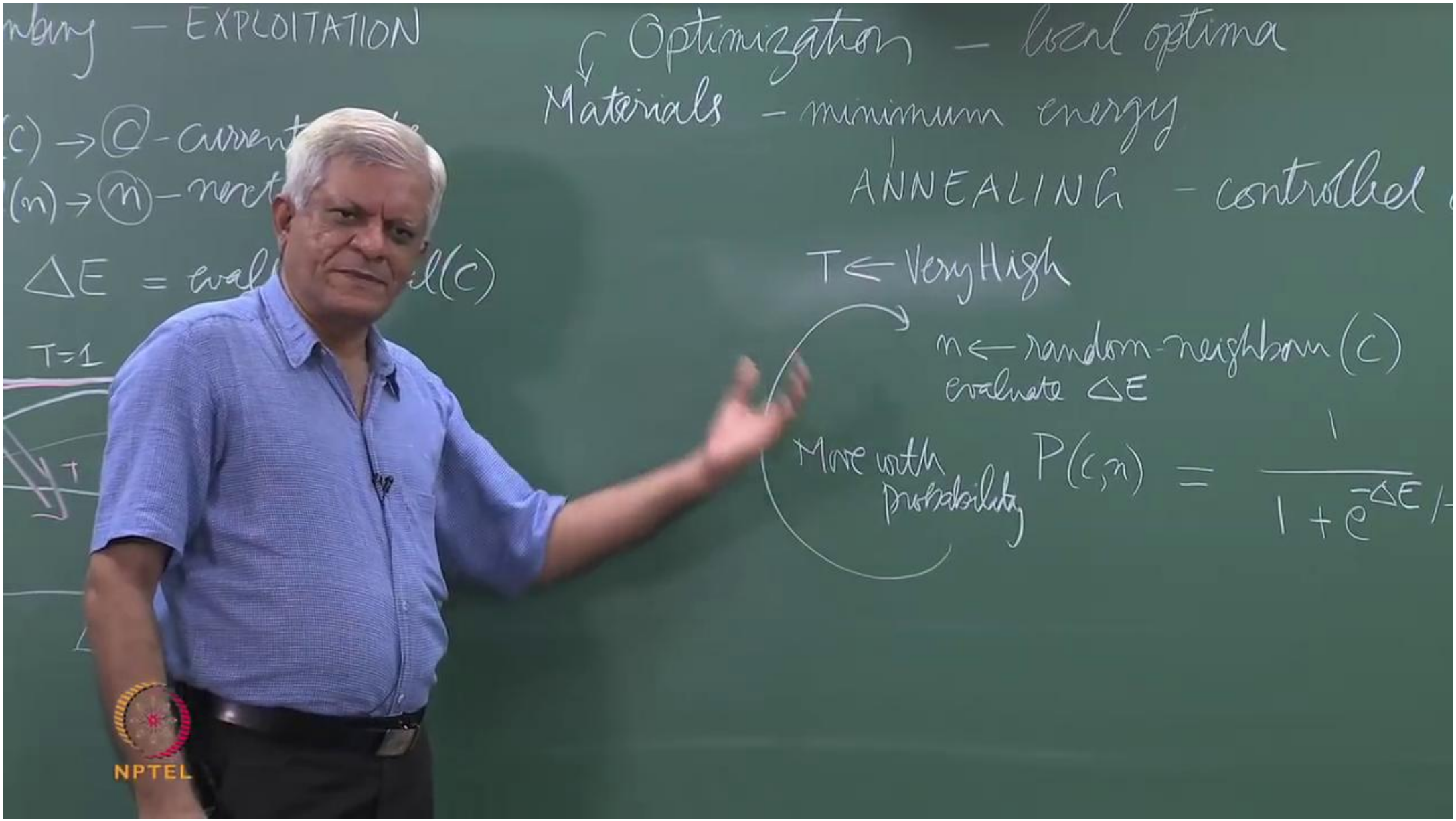

Hill Climbing – EXPLOITATION
Optimiz
Materials –
eval(c) → ⓒ - current node
eval(n) → ⓝ - next node
ΔE = eval(n) – eval(c)
STO
T=1
SIGMOID
1
T=10^10
ΔE=0
ΔE →
NPTEL

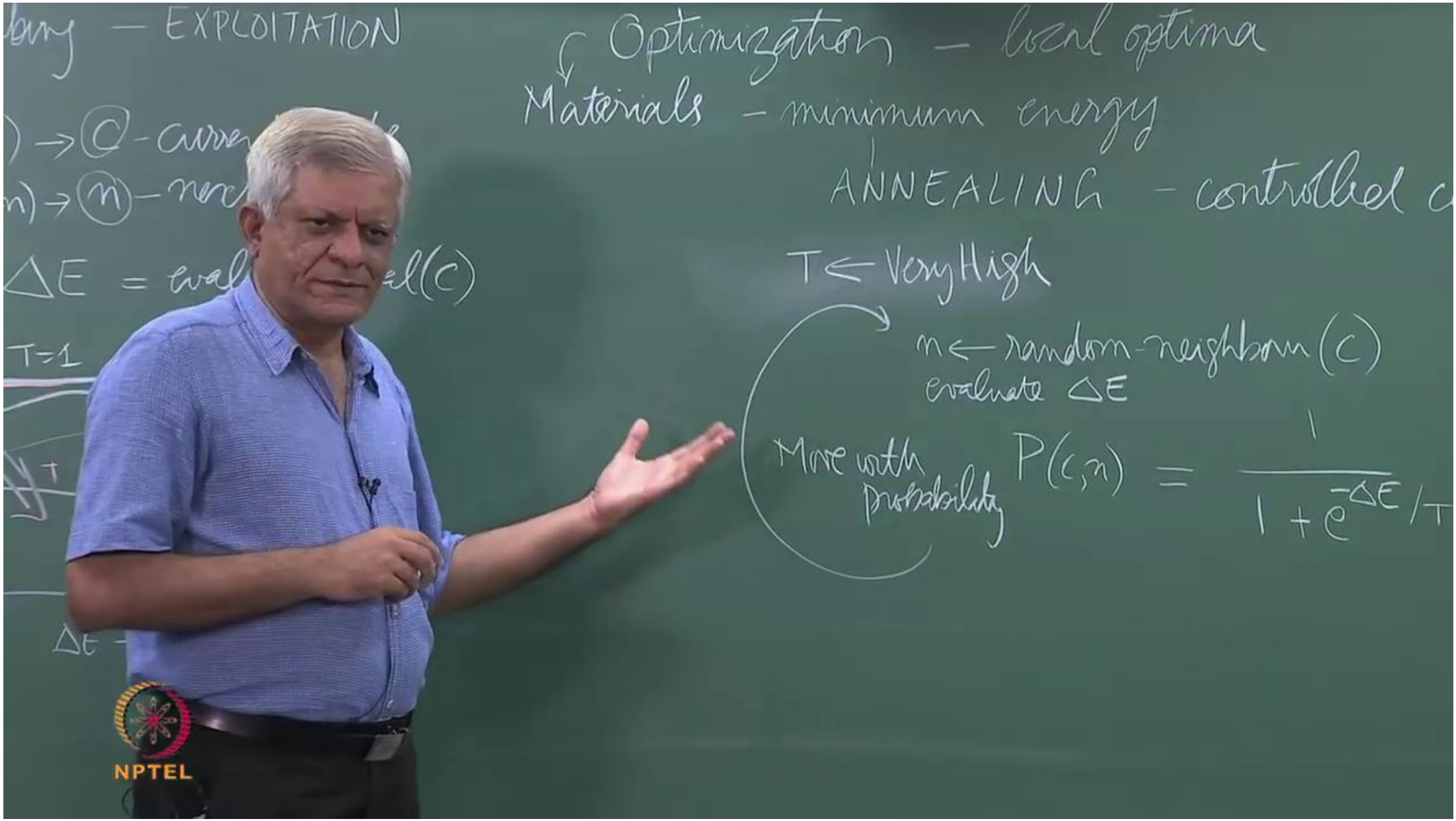

SIGMOID
T=1
1
T=10^10
ΔE=0
ΔE →

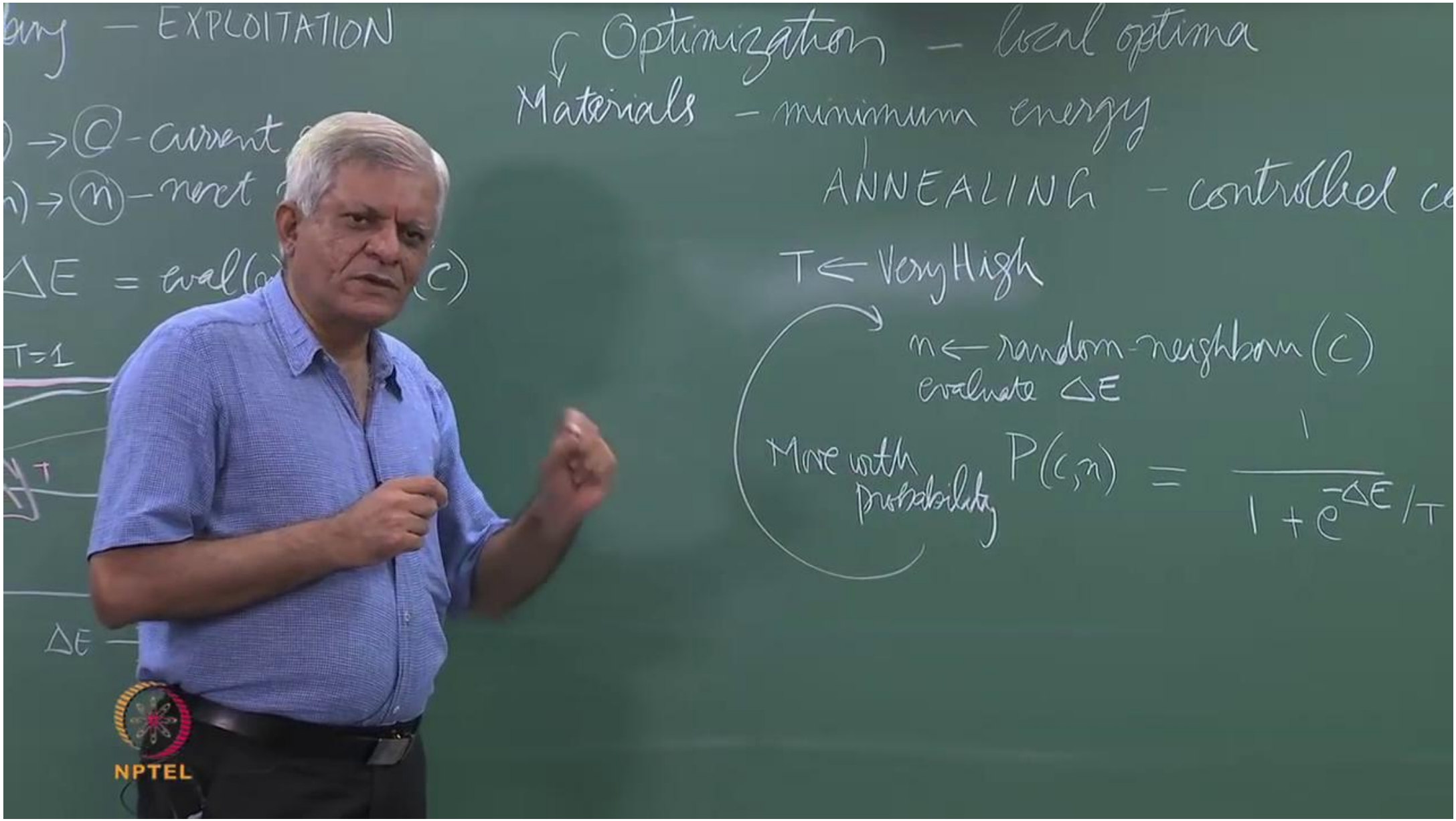

SIGMOID
T=1
1
$T=10^{10}$
$\Delta E=0$
$\Delta E \to$


SIGMOID
T=1
1
T=10^10
ΔE=0
ΔE →


Hill Climbing – EXPLOITATION
Optimiz
Materials
eval(c) → c - current node
eval(n) → n - next node
ΔE = eval(n) – eval(c)
ST
T=1
SIGMOID
1
T
T=10^10
ΔE=0
ΔE →
NPTEL


— EXPLOITATION
Optimization — local optima
Materials — minimum energy
ANNEALING — controlled
T ← Very High
n ← random-neighbour(c)
evaluate ΔE
Move with probability
P(c,n) =
1
1 + e^{-ΔE/
T=1
NPTEL


— EXPLOITATION
Optimization — local optima
Materials — minimum energy
ANNEALING — controlled
T ← Very High
n ← random-neighbour (c)
evaluate ΔE
Move with probability
P(c,n) = 1 / 1 + e^{-ΔE/T}
T=1
NPTEL


— EXPLOITATION
Optimization — local optima
Materials — minimum energy
ANNEALING — controlled
T ← Very High
n ← random-neighbour(c)
evaluate ΔE
Move with probability
P(c,n) = 1 / (1 + e^{-ΔE/T})
T=1
ΔE

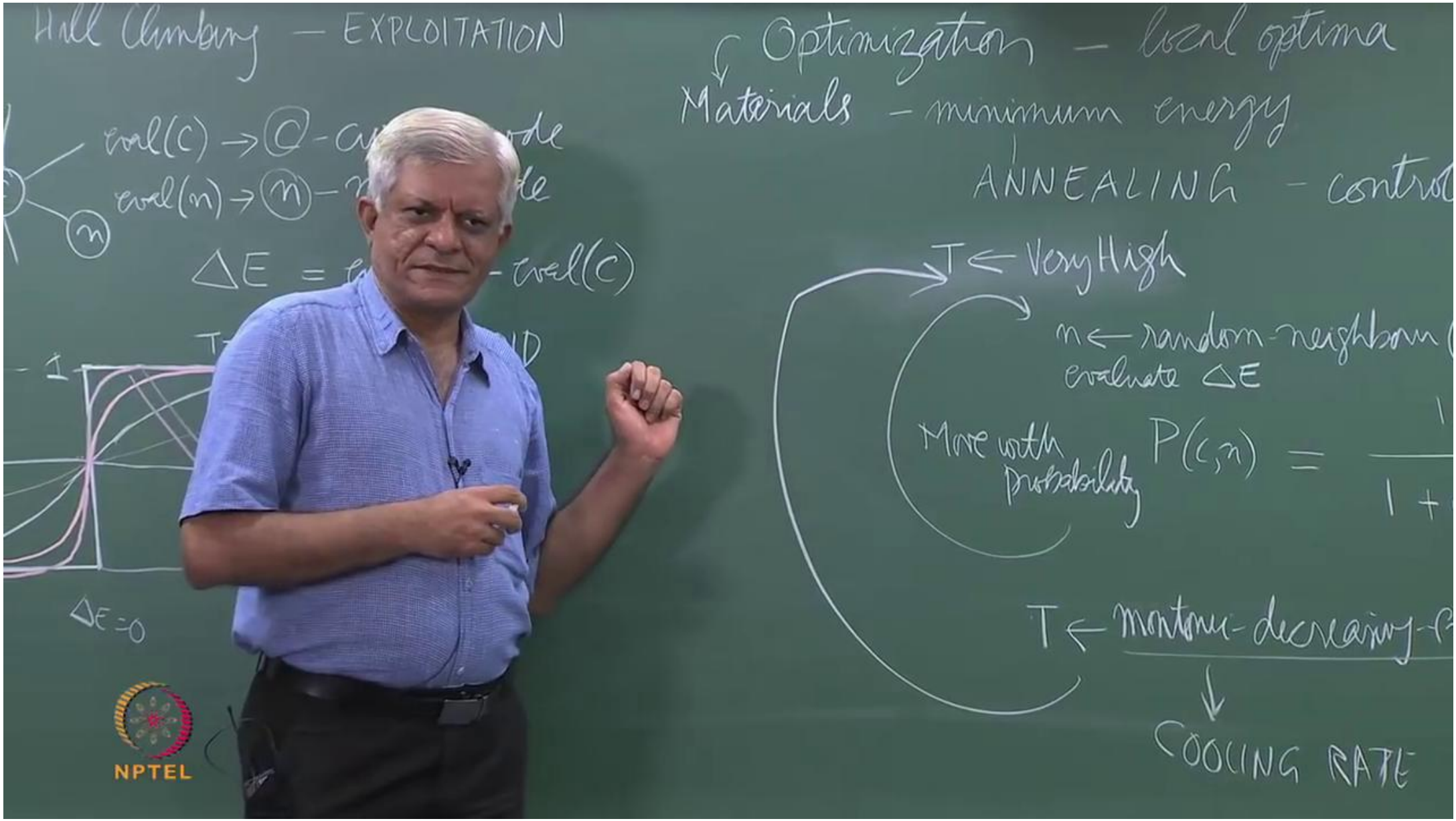
probability $\frac{1}{1+e^{-\Delta E/T}}$

$T \leftarrow$ monotonic-decreasing-fn(T)

$T \leftarrow$ monotone-decreasing-fn (T)

↓

COOLING RATE


Hill Climbing – EXPLOITATION
Optimization – local optima
Materials – minimum energy
ANNEALING – control
T ← Very High
n ← random-neighbour
evaluate ΔE
Move with probability
P(c,n) =
T ← monotone-decreasing-
COOLING RATE
ΔE = 0
NPTEL

Metropolis – Monte Carlo

SIMULATED ANNEALING – controlled cooling

T ← Very High

n ← random-neighbour(c)

evaluate ΔE

Move with probability $P(c,n) = \frac{1}{1+e^{-\Delta E/T}}$

T ← monotone-decreasing-fn(T)

Metropolis – [illegible]

SIMULATED ANNEALING – controlled cooling

T ← Very High

n ← random-neighbour(c)

evaluate ΔE

Move with probability $P(c,n) = \frac{1}{1+e^{-\Delta E/T}}$

T ← monotonic-decreasing-fn(T)

HC

Metropolis – [illegible]

SIMULATED ANNEALING – controlled cooling

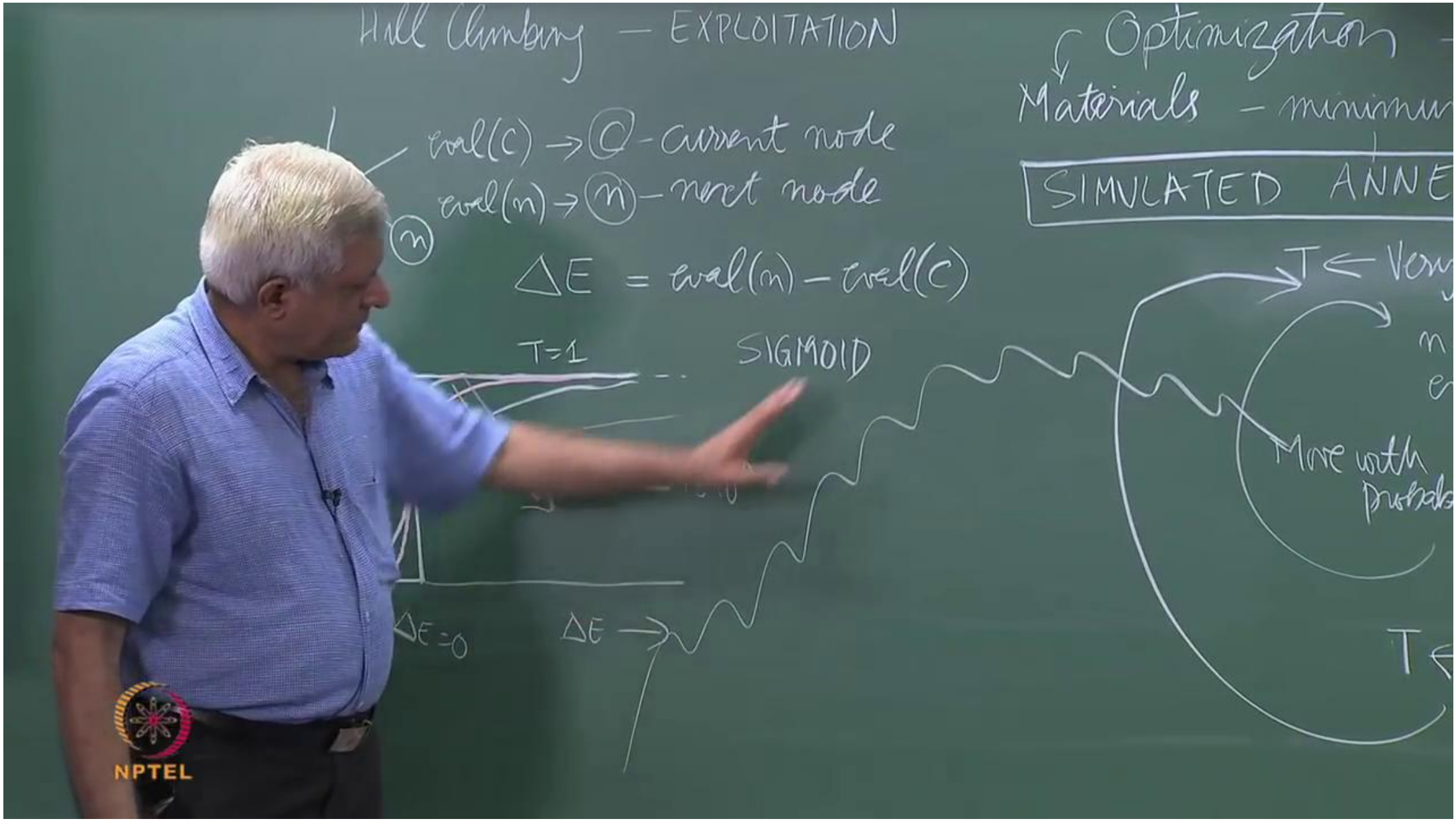
T ← Very High

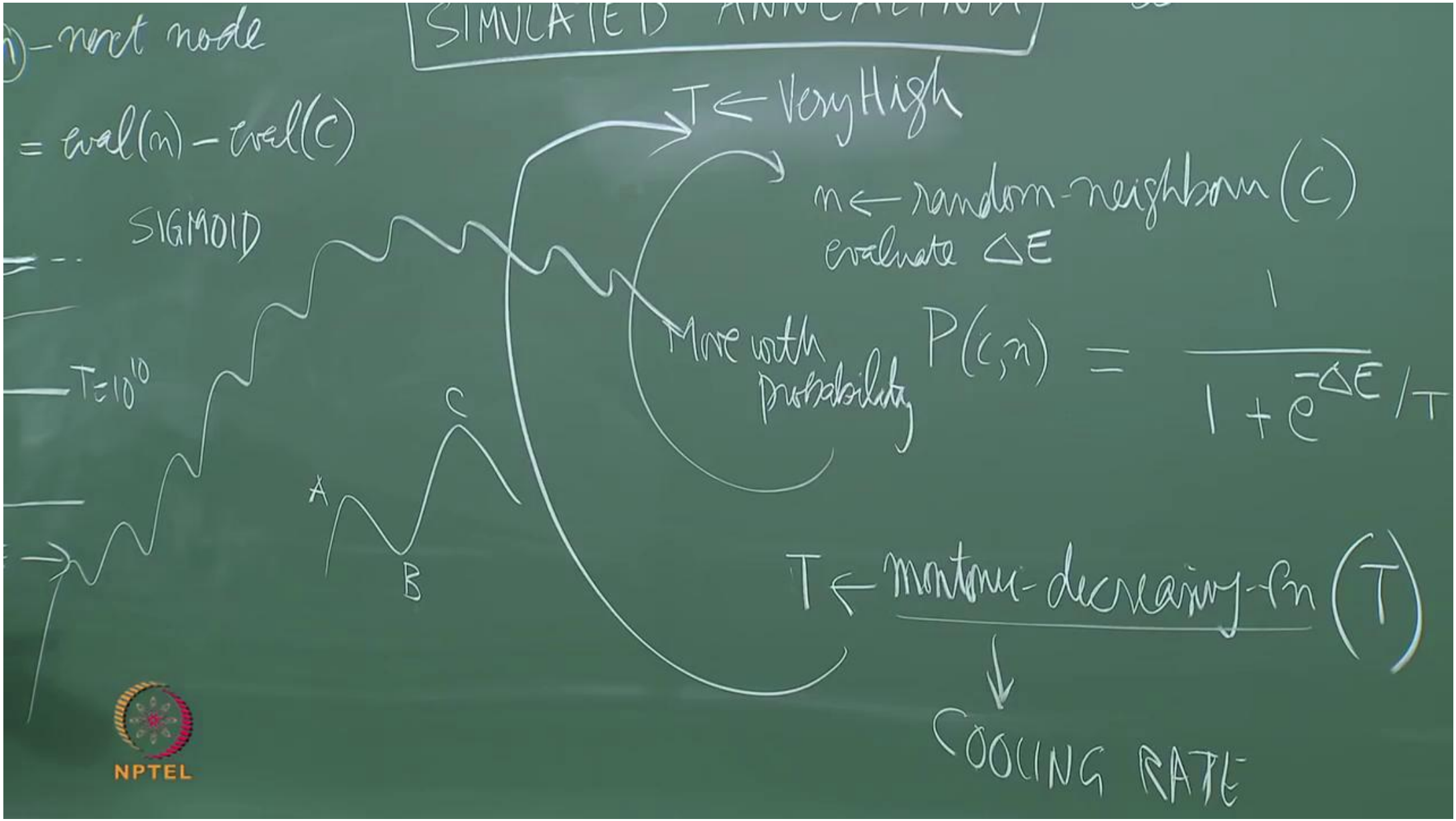
n ← random-neighbour(C)

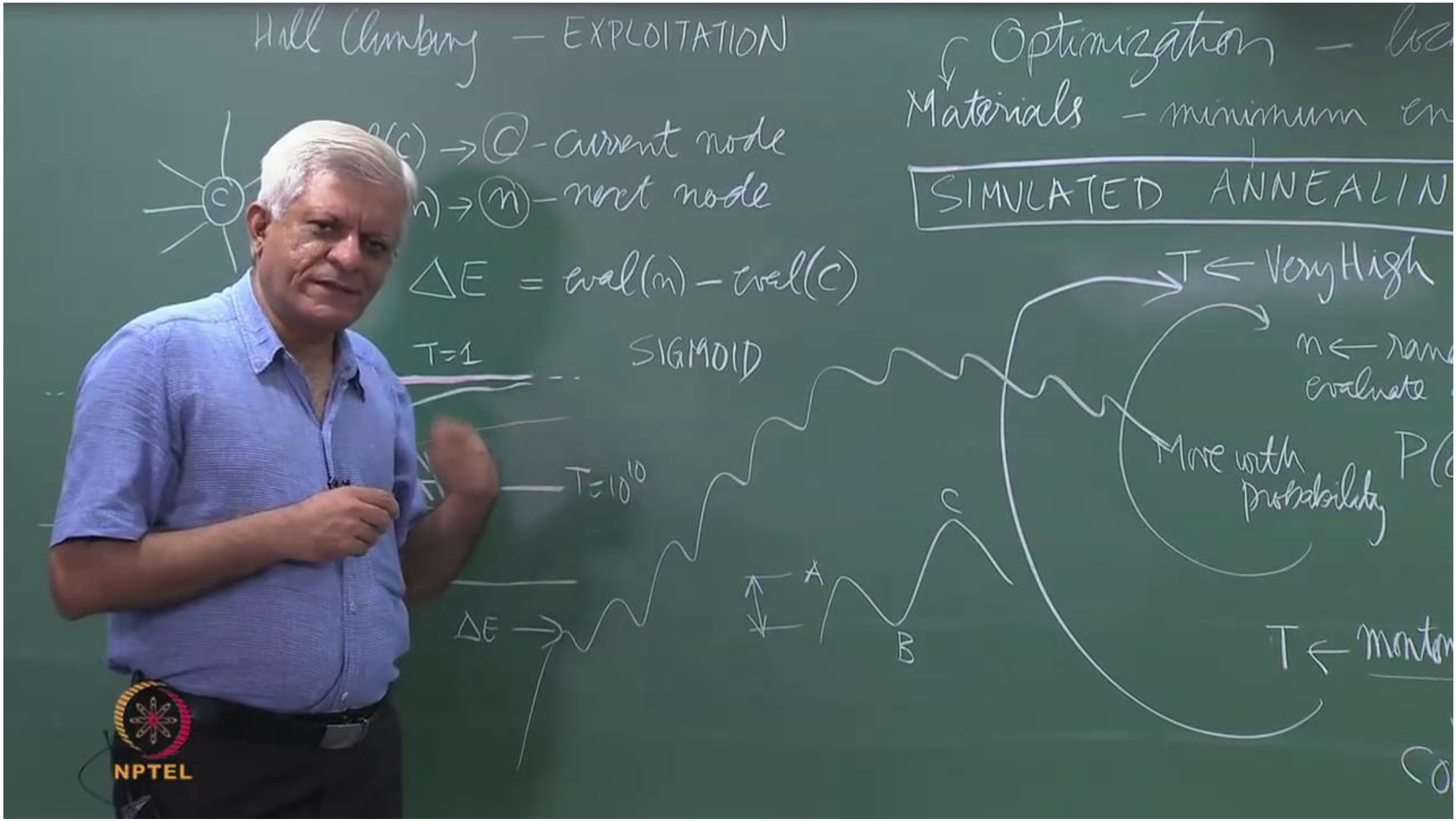
evaluate ΔE

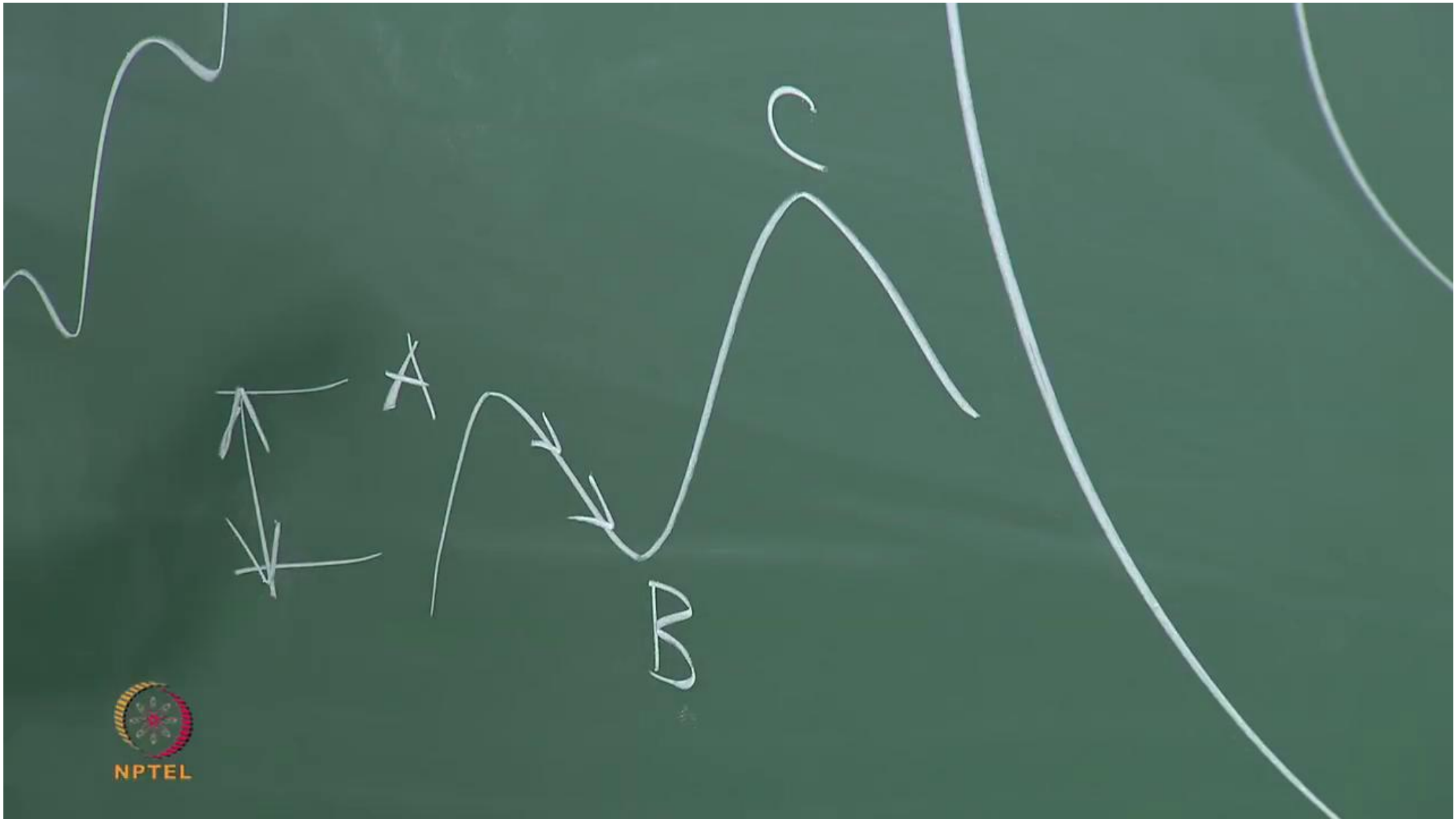
Move with probability $P(c,n) = \frac{1}{1+e^{-\Delta E/T}}$

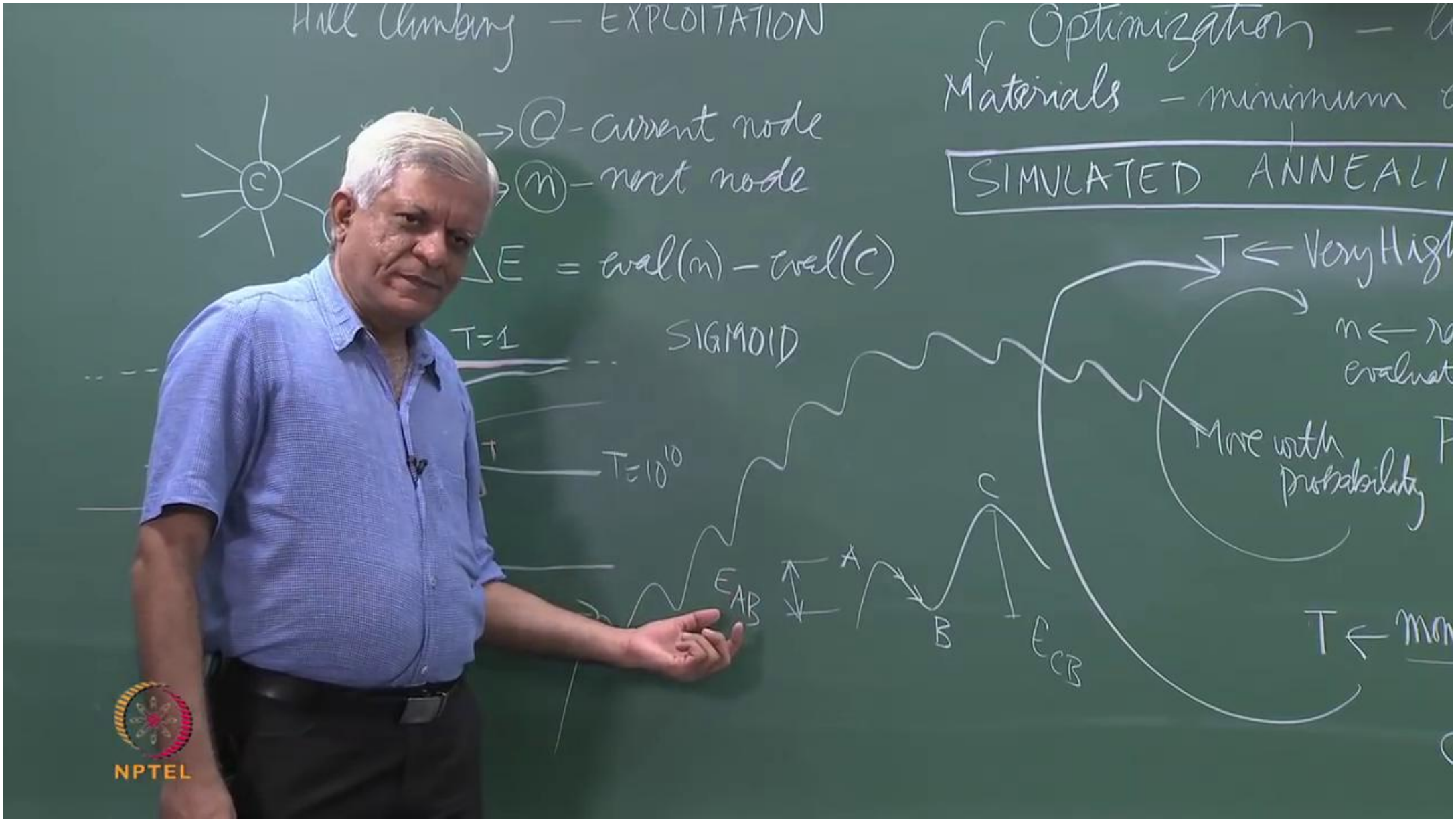
T ← monotonic-decreasing-fn(T)

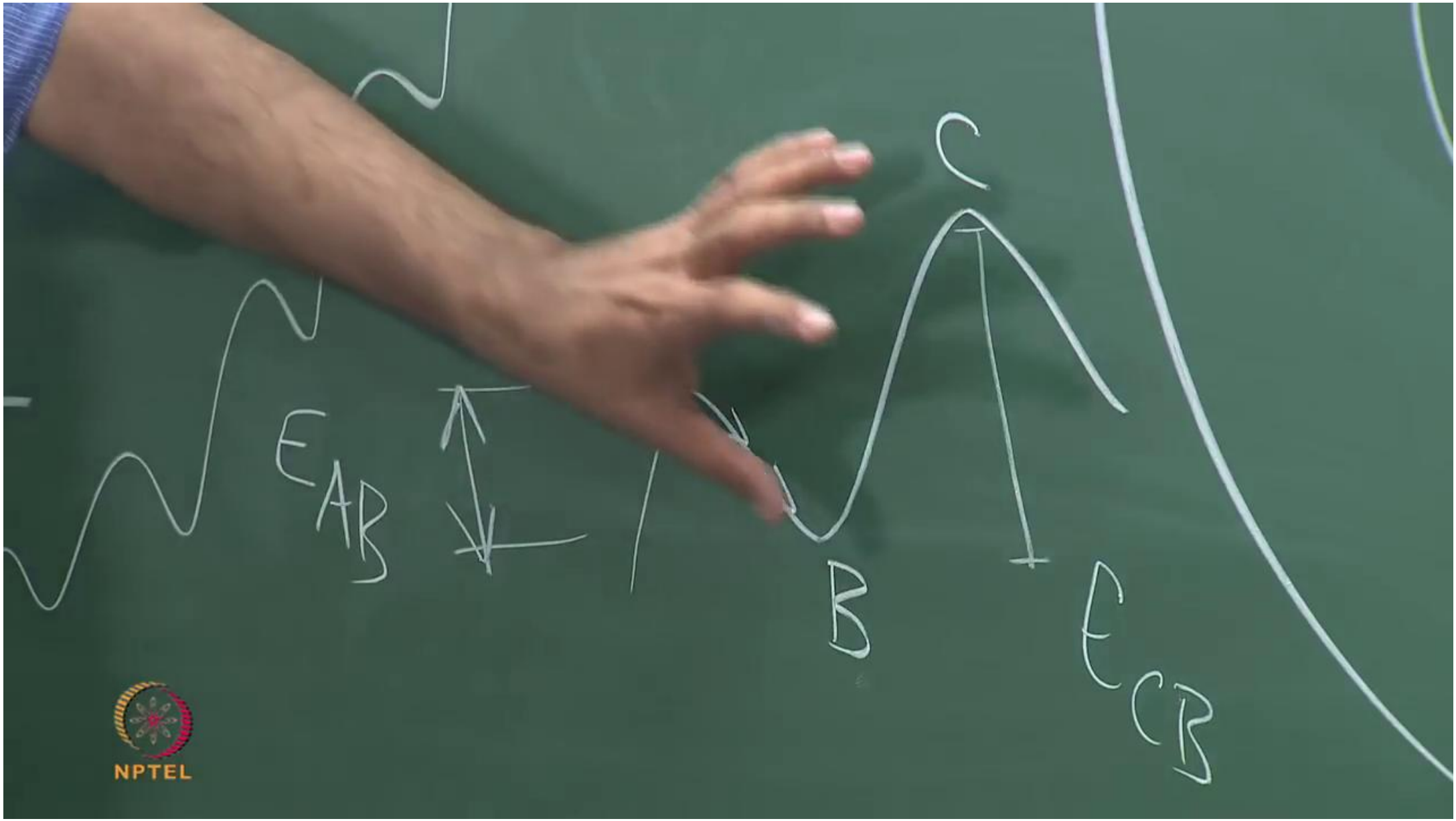

Hill Climbing – EXPLOITATION
Optimization
Materials – minimum
SIMULATED ANNE
eval(c) → c – current node
eval(n) → n – next node
n
ΔE = eval(n) – eval(c)
T=1
SIGMOID
T ← Very
More with
ΔE=0
ΔE →
T ←
NPTEL

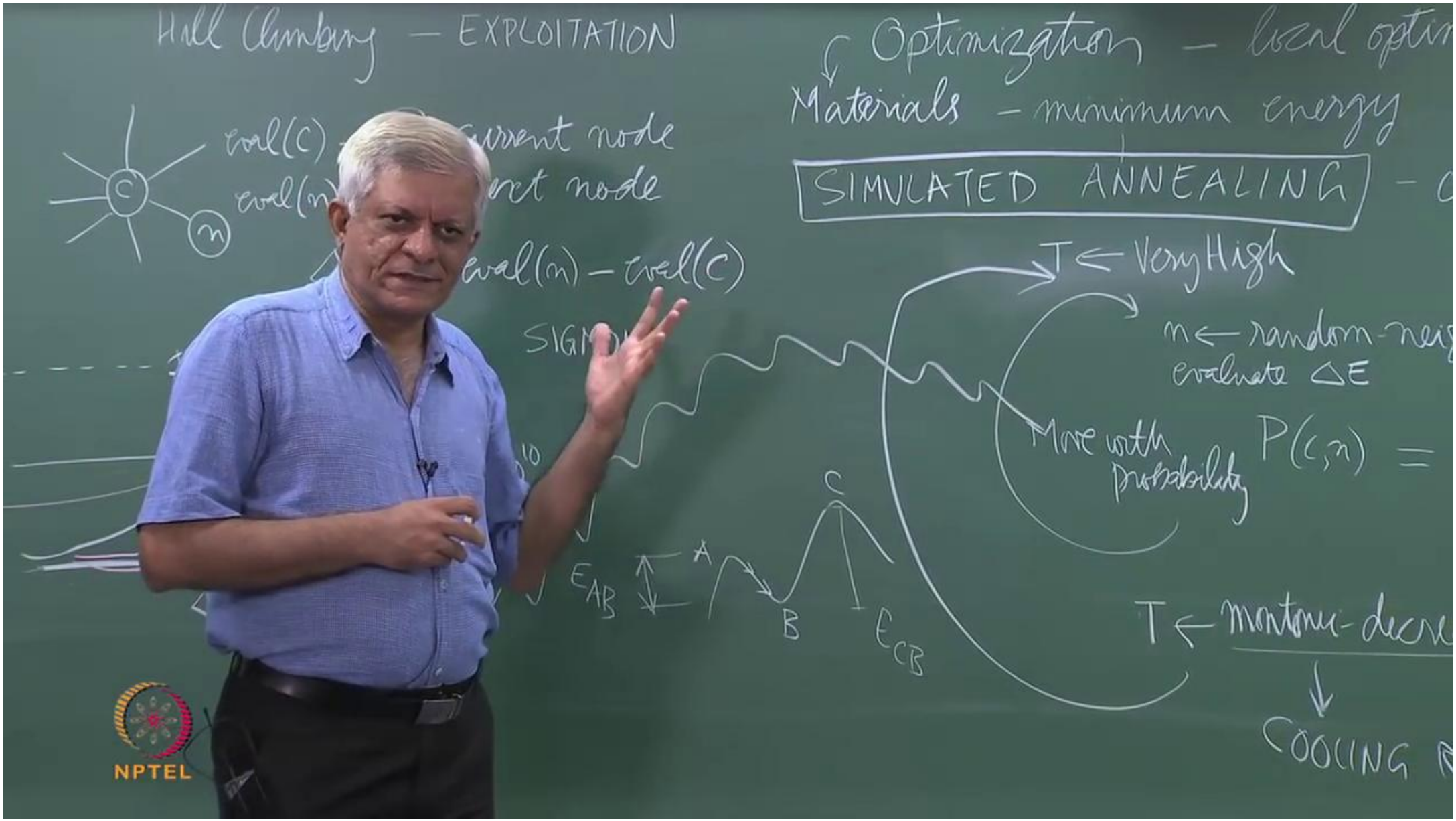

SIMULATED
n - next node
= eval(n) - eval(C)
SIGMOID
T ← Very High
n ← random-neighbour(C)
evaluate ΔE
Move with probability
P(C,n) = 1 / (1 + e^{-ΔE/T})
T = 10^10
A
B
C
T ← monotone-decreasing-fn (T)
COOLING RATE

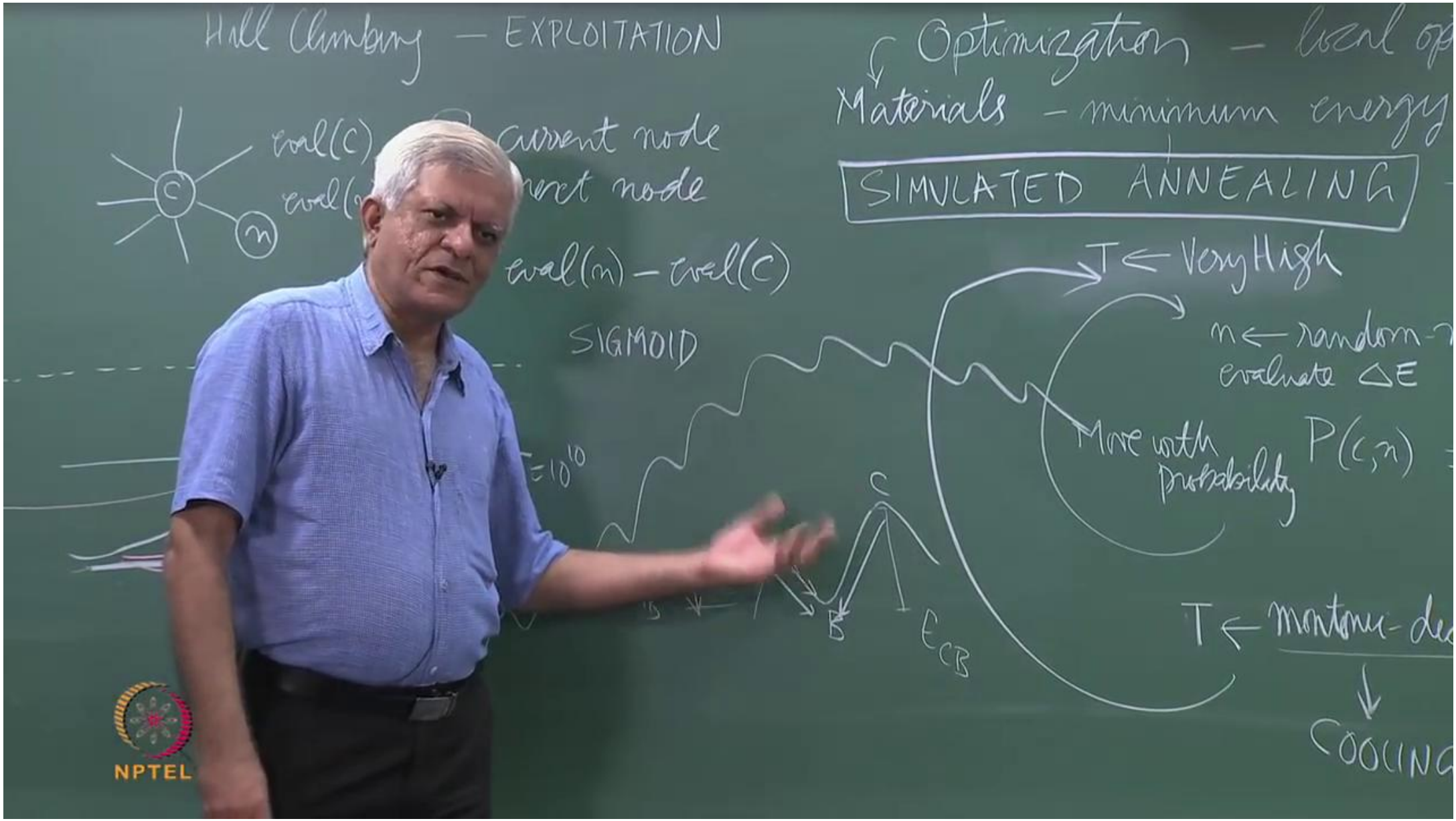

Hill Climbing – EXPLOITATION
Optimization
Materials – minimum
SIMULATED ANNEALIN
c → current node
n → next node
ΔE = eval(n) – eval(c)
T=1
SIGMOID
T=10^10
T ← Very High
n ← rand
evaluate
Move with probability
P(
ΔE
A
B
C
T ← monto
NPTEL

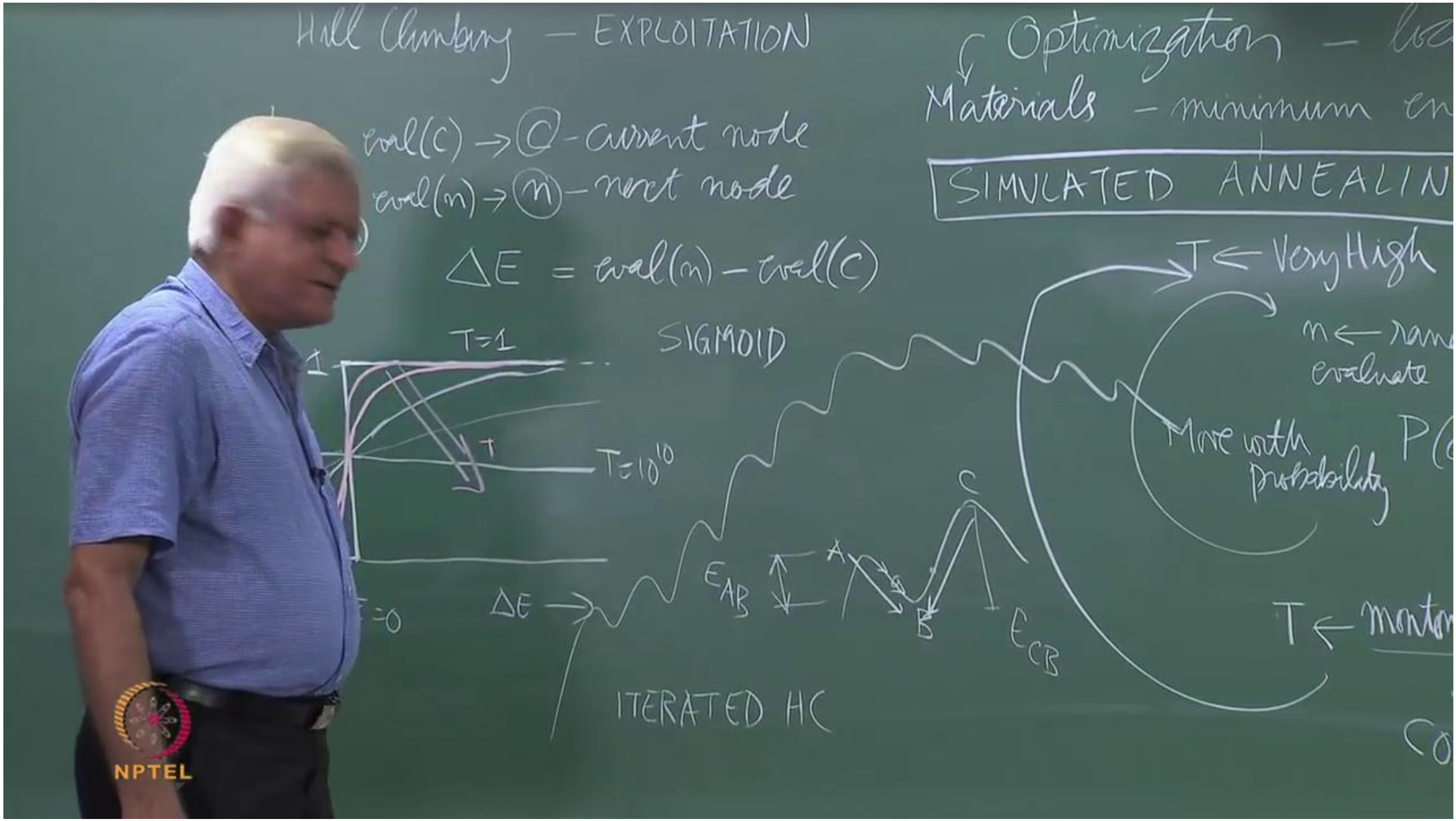

C
A
B

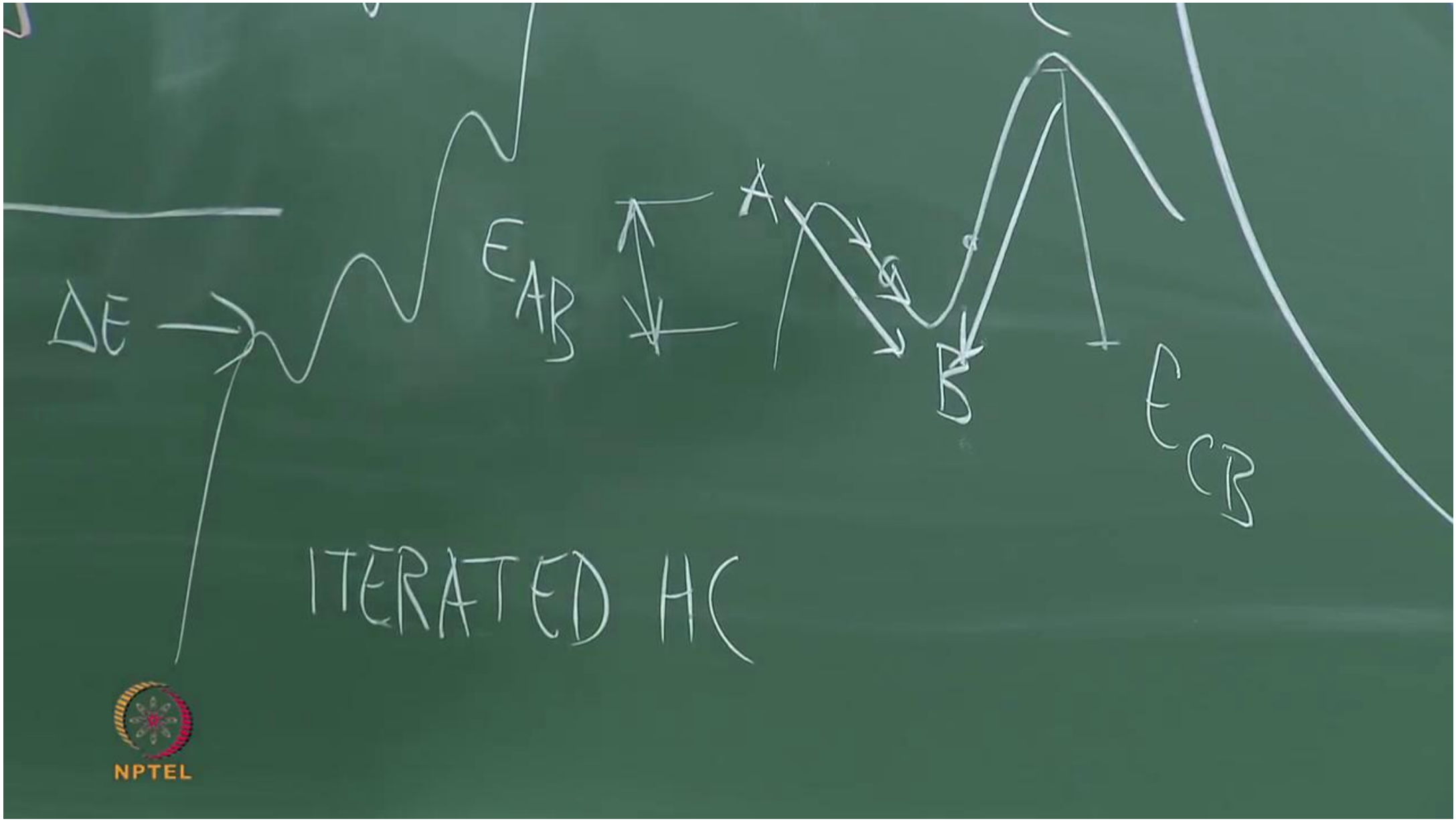

Hill Climbing – EXPLOITATION
Optimization
Materials – minimum
SIMULATED ANNEALI
c – current node
n – next node
ΔE = eval(n) – eval(c)
T=1
SIGMOID
T=10^10
T ← Very High
n ← ne
evaluat
Move with probability
E_AB
A
B
C
E_CB
T ← mo
NPTEL

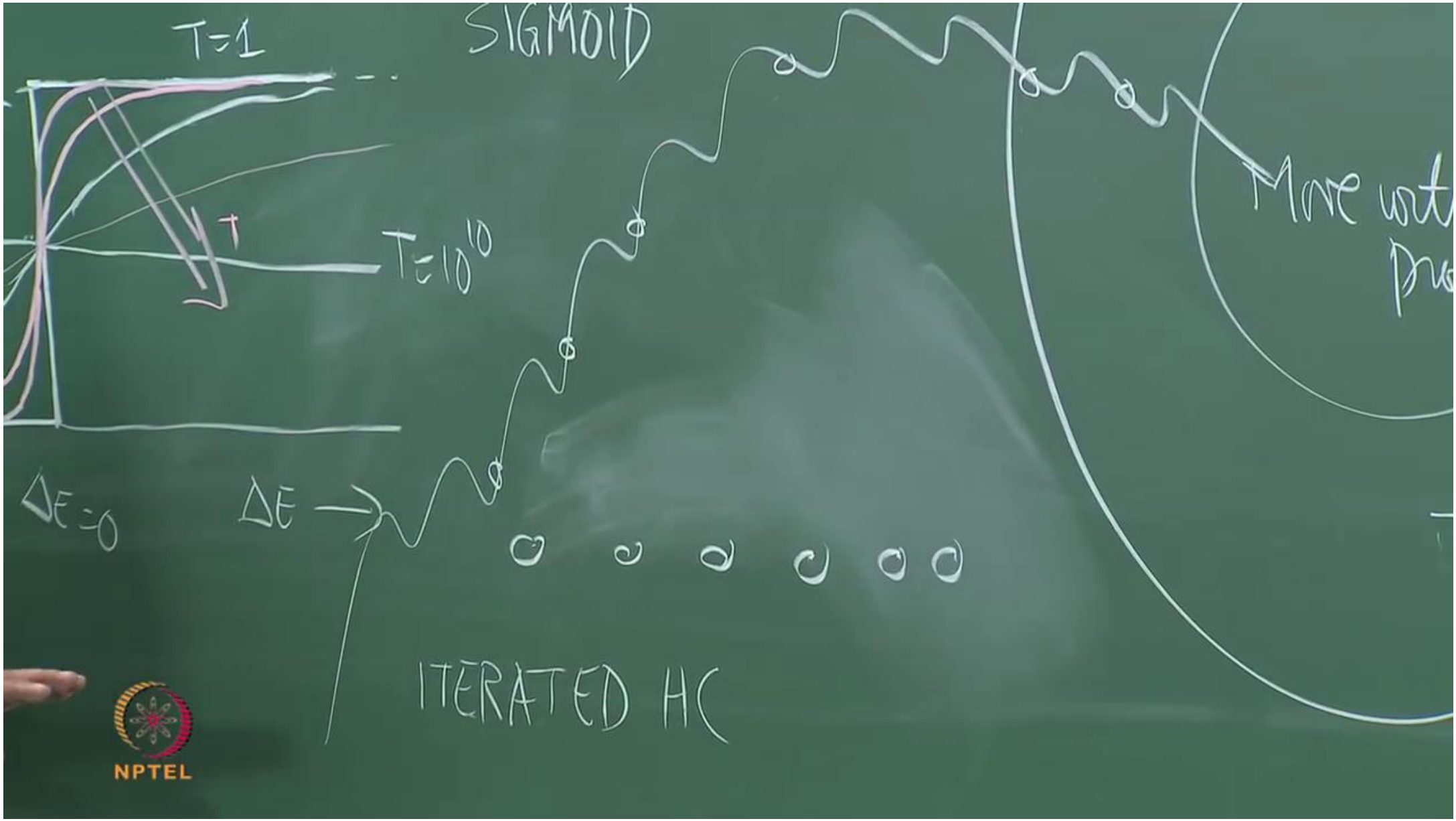

C
E
AB
B
E
CB

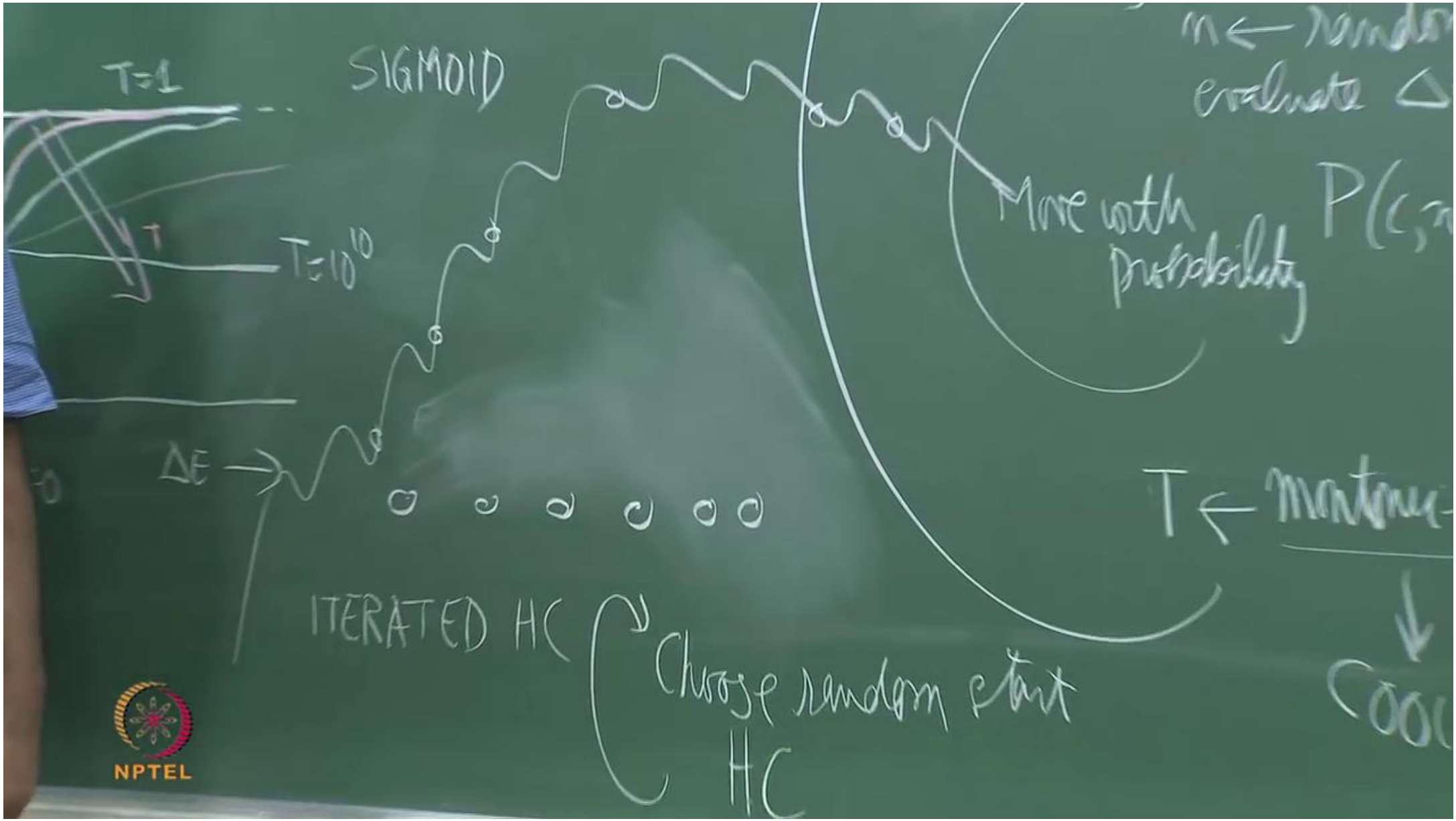

Hill Climbing – EXPLOITATION
Optimization – local optim
Materials – minimum energy
SIMULATED ANNEALING
eval(c)
eval(n)
current node
next node
eval(n) – eval(c)
T ← Very High
n ← random-neig
evaluate ΔE
Move with probability
P(c,n) =
c
A
B
E_AB
E_CB
T ← monotonic-decre
COOLING
NPTEL

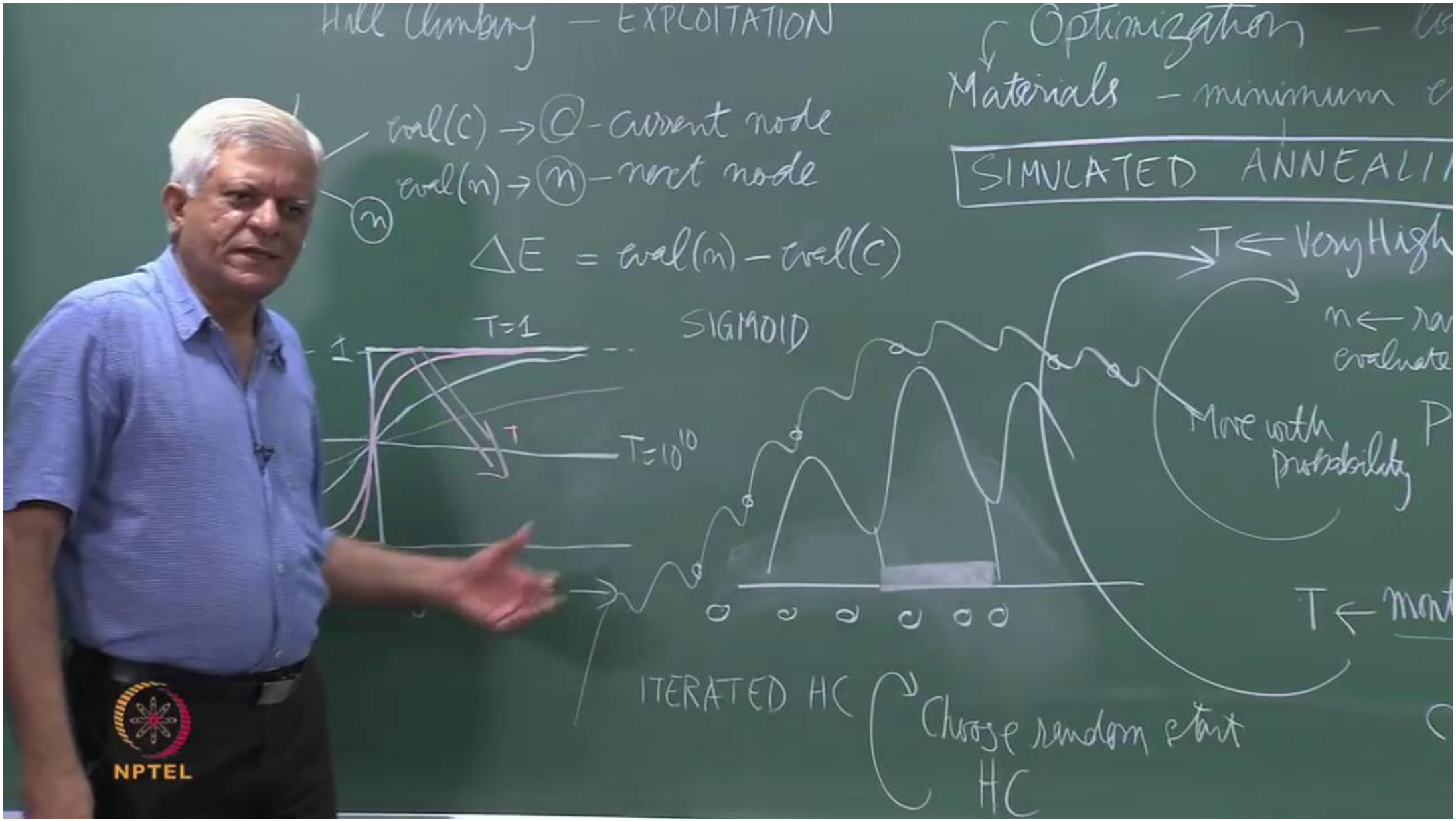

Hill Climbing – EXPLOITATION
Optimization – local op
Materials – minimum energy
SIMULATED ANNEALING
eval(c)
current node
eval(n) – eval(c)
SIGMOID
T ← Very High
n ← random
evaluate ΔE
More with probability
P(c,n)
T ← monotonic-de
COOLING
NPTEL


Hill Climbing – EXPLOITATION
Optimization –
Materials – minimum
SIMULATED ANNEALIN
eval(c) → c – current node
eval(n) → n – next node
ΔE = eval(n) – eval(c)
T ← Very High
SIGMOID
T=1
T=10^10
n ← rand
evaluate
Move with probability
P(
ΔE →
E_AB
A
B
C
E_CB
T ← monot
ITERATED HC
CO
NPTEL


ΔE
E_AB
A
B
E_CB
ITERATED HC


T=1
SIGMOID
T
T=10^0
More
ΔE=0
ΔE
ITERATED HC
NPTEL


T=1
SIGMOID
T
T=10^10
ΔE →
Move with probability
ITERATED HC
Choose random start
HC
NPTEL


Hill Climbing – EXPLOITATION
Optimization –
Materials – minimum
SIMULATED ANNEALI
eval(c) → c - current node
eval(n) → n - next node
n
ΔE = eval(n) – eval(c)
T ← Very High
SIGMOID
T=1
1
T
T=10^10
n ← ran
evaluate
Move with probability
P
T ← mon
ITERATED HC
Choose random start
HC
NPTEL