Psudo code of simulated annealing algorithm

> temp = 100

> t = 1:(5e4)

> tmax = 1000

> newInit = c(init, init[1])

>

> y = temp / log(((t-1) %/% tmax)\*tmax + exp(1))

> bestInit = newInit

> bestObj = obj.fun.new(newInit)

> mem.bestInit = numeric(5e4)

> mem.newInit = numeric(5e4)

>

> ################################################################################

>

> extractAniTrt(newInit[-length(newInit)])

[,1] [,2] [,3] [,4]

[1,] "AA" "AB" "AC" "AD"

[2,] "AB" "AA" "AD" "AC"

[3,] "AE" "AF" "AG" "AH"

[4,] "AF" "AE" "AH" "AG"

[5,] "AI" "AJ" "AK" "AL"

[6,] "AJ" "AI" "AL" "AK"

[7,] "AM" "AN" "AO" "AP"

[8,] "AN" "AM" "AP" "AO"

[9,] "AQ" "AQ" "AR" "AR"

[,1] [,2] [,3] [,4]

[1,] "a" "b" "c" "d"

[2,] "b" "a" "d" "c"

[3,] "e" "f" "a" "b"

[4,] "f" "e" "b" "a"

[5,] "c" "d" "e" "f"

[6,] "d" "c" "f" "e"

[7,] "a" "b" "c" "d"

[8,] "b" "a" "d" "c"

[9,] "e" "e" "f" "f"

>

> (x1 = obj.fun.new(newInit) )

[1] 97.61799

>

> #First swap of the first stage

> newInit1 = swap.stage1.new(newInit)

> extractAniTrt(newInit1[-length(newInit1)])

[,1] [,2] [,3] [,4]

[1,] "AA" "AB" "AC" "AD"

[2,] "AB" "AA" "AD" "AC"

[3,] "AE" "AF" "AG" "AH"

[4,] "AF" "AE" "AH" "AG"

[5,] "AI" "AK" "AJ" "AL"

[6,] "AK" "AI" "AL" "AJ"

[7,] "AM" "AN" "AO" "AP"

[8,] "AN" "AM" "AP" "AO"

[9,] "AQ" "AQ" "AR" "AR"

[,1] [,2] [,3] [,4]

[1,] "a" "b" "c" "d"

[2,] "b" "a" "d" "c"

[3,] "e" "f" "a" "b"

[4,] "f" "e" "b" "a"

[5,] "c" "e" "d" "f"

[6,] "e" "c" "f" "d"

[7,] "a" "b" "c" "d"

[8,] "b" "a" "d" "c"

[9,] "e" "e" "f" "f"

>

>

> obj.fun.new(newInit1)

[1] 96.31085

>

> #Check for based on the simulated annealing Temperture of 100

>

> exp((obj.fun.new(newInit1) - x1)/y[1])

[1] 0.9870136

>

> (acceptancy.prop = runif(1,0,1))

[1] 0.4240715

> if(exp((obj.fun.new(newInit1) - x1)/y[1]) > acceptancy.prop){

+ print("improved 1!")

+ mem.newInit[1] = obj.fun.new(newInit1)

+ }

[1] "improved 1!"

>

> if((obj.fun.new(newInit1) > bestObj)){

+ print("improved 2!")

+

+ bestObj = obj.fun.new(newInit1)

+ bestInit = newInit1

+ mem.bestInit[1] = bestObj

+ print(bestObj)

+

+ }

>

> #Second swap

> newInit2 = swap.stage1.new(newInit1)

> extractAniTrt(newInit2[-length(newInit2)])

[,1] [,2] [,3] [,4]

[1,] "AA" "AB" "AC" "AD"

[2,] "AB" "AA" "AD" "AC"

[3,] "AE" "AF" "AH" "AG"

[4,] "AF" "AE" "AG" "AH"

[5,] "AI" "AK" "AJ" "AL"

[6,] "AK" "AI" "AL" "AJ"

[7,] "AM" "AN" "AO" "AP"

[8,] "AN" "AM" "AP" "AO"

[9,] "AQ" "AQ" "AR" "AR"

[,1] [,2] [,3] [,4]

[1,] "a" "b" "c" "d"

[2,] "b" "a" "d" "c"

[3,] "e" "f" "b" "a"

[4,] "f" "e" "a" "b"

[5,] "c" "e" "d" "f"

[6,] "e" "c" "f" "d"

[7,] "a" "b" "c" "d"

[8,] "b" "a" "d" "c"

[9,] "e" "e" "f" "f"

>

>

> obj.fun.new(newInit2)

[1] 96.31085

>

> #Check for based on the simulated annealing Temperture of 100

>

> exp((obj.fun.new(newInit2) - obj.fun.new(newInit1))/y[2])

[1] 1

>

> (acceptancy.prop = runif(1,0,1))

[1] 0.3830102

>

> if(exp((obj.fun.new(newInit2) - obj.fun.new(newInit1))/y[2]) > acceptancy.prop){

+ print("improved 1!")

+ mem.newInit[2] = obj.fun.new(newInit2)

+ }

[1] "improved 1!"

>

> if((obj.fun.new(newInit2) > bestObj)){

+ print("improved 2!")

+

+ bestObj = obj.fun.new(newInit2)

+ bestInit = newInit2

+ mem.bestInit[2] = bestObj

+ print(bestObj)

+

+ }

>

> #Third swap

> newInit3 = swap.stage1.new(newInit2)

> extractAniTrt(newInit3[-length(newInit3)])

[,1] [,2] [,3] [,4]

[1,] "AA" "AB" "AC" "AD"

[2,] "AB" "AA" "AD" "AC"

[3,] "AE" "AF" "AH" "AG"

[4,] "AF" "AE" "AG" "AH"

[5,] "AI" "AK" "AJ" "AL"

[6,] "AK" "AI" "AL" "AJ"

[7,] "AO" "AN" "AM" "AP"

[8,] "AN" "AO" "AP" "AM"

[9,] "AQ" "AQ" "AR" "AR"

[,1] [,2] [,3] [,4]

[1,] "a" "b" "c" "d"

[2,] "b" "a" "d" "c"

[3,] "e" "f" "b" "a"

[4,] "f" "e" "a" "b"

[5,] "c" "e" "d" "f"

[6,] "e" "c" "f" "d"

[7,] "c" "b" "a" "d"

[8,] "b" "c" "d" "a"

[9,] "e" "e" "f" "f"

>

>

> obj.fun.new(newInit3)

[1] 96.50794

>

> #Check for based on the simulated annealing Temperture of 100

>

> exp((obj.fun.new(newInit3) - obj.fun.new(newInit2))/y[3])

[1] 1.001973

>

> (acceptancy.prop = runif(1,0,1))

[1] 0.752059

>

> if(exp((obj.fun.new(newInit3) - obj.fun.new(newInit2))/y[3]) > acceptancy.prop){

+ print("improved 1!")

+ mem.newInit[3] = obj.fun.new(newInit3)

+

+ }

[1] "improved 1!"

>

> if((obj.fun.new(newInit3) > bestObj)){

+ print("improved 2!")

+

+ bestObj = obj.fun.new(newInit3)

+ bestInit = newInit3

+ mem.bestInit[3] = bestObj

+ print(bestObj)

+

+ }

> extractAniTrt(bestInit[-length(bestInit)])

[,1] [,2] [,3] [,4]

[1,] "AB" "AD" "AA" "AC"

[2,] "AD" "AB" "AC" "AA"

[3,] "AE" "AG" "AF" "AH"

[4,] "AG" "AE" "AH" "AF"

[5,] "AK" "AJ" "AL" "AI"

[6,] "AJ" "AK" "AI" "AL"

[7,] "AO" "AN" "AP" "AM"

[8,] "AN" "AO" "AM" "AP"

[9,] "AR" "AR" "AQ" "AQ"

[,1] [,2] [,3] [,4]

[1,] "b" "d" "a" "c"

[2,] "d" "b" "c" "a"

[3,] "e" "a" "f" "b"

[4,] "a" "e" "b" "f"

[5,] "e" "d" "f" "c"

[6,] "d" "e" "c" "f"

[7,] "c" "b" "d" "a"

[8,] "b" "c" "a" "d"

[9,] "f" "f" "e" "e"

>

> obj.fun.new(bestInit)

[1] 97.67238

> newInit3 = swap.stage2.new(bestInit)

>

> print(obj.fun.new(newInit3))

[1] 97.87345

> extractAniTrt(newInit3[-length(newInit3)])

[,1] [,2] [,3] [,4]

[1,] "AB" "AD" "AA" "AC"

[2,] "AD" "AB" "AC" "AA"

[3,] "AE" "AG" "AF" "AH"

[4,] "AG" "AE" "AH" "AF"

[5,] "AK" "AJ" "AL" "AI"

[6,] "AJ" "AK" "AI" "AL"

[7,] "AO" "AR" "AP" "AM"

[8,] "AR" "AO" "AM" "AP"

[9,] "AN" "AN" "AQ" "AQ"

[,1] [,2] [,3] [,4]

[1,] "b" "d" "a" "c"

[2,] "d" "b" "c" "a"

[3,] "e" "a" "f" "b"

[4,] "a" "e" "b" "f"

[5,] "e" "d" "f" "c"

[6,] "d" "e" "c" "f"

[7,] "c" "f" "d" "a"

[8,] "f" "c" "a" "d"

[9,] "b" "b" "e" "e"

>

> #Check for based on the simulated annealing Temperture of 100

> (acceptancy.prop = runif(1,0,1))

[1] 0.0220835

>

> print(exp((obj.fun.new(newInit3) - obj.fun.new(bestInit))/y[1]))

[1] 1.002013

>

>

> if(exp((obj.fun.new(newInit3) - obj.fun.new(bestInit))/y[1]) > acceptancy.prop){

+ newInit3 = newInit3

+ print("improved 1!")

+ }

[1] "improved 1!"

>

> if((obj.fun.new(newInit3) > bestObj)){

+ print("improved 2!")

+

+ bestObj = obj.fun.new(newInit3)

+ bestInit = newInit

+ print(bestObj)

+

+ }

[1] "improved 2!"

[1] 97.87345