

# **CSCI 6470: Algorithms**

## **Project Report By *Dhaval Bhanderi***

Recursive Solution:

The main objective of RNA structure prediction is to maximize the score.

There are four possible cases, the optimal score  $S(i, j)$  is the maximum of the four possibilities. The defined function  $S(i, j)$  recursively as a function only of optimal score of smaller subproblems. Then, the recursive function would be as below:

**Pseudocode:**

The pseudocode for Objective function StructPred and Traceback are as below:

Traceback(F, sgm, sequence, initi, initj):

    i = initi

    j = initj

    Struct = Structure()

    While(j != i-1)

        If(F[i, j] == F[i, j-1])

            Struct.unpair.append(j)

            J = j - 1

    Elif (F[i, j] == F[i+1, j])

        Struct.unpair.append(i)

        i = i + 1

    Elif (F[i, j] == F[i+1, j-1]) + alphabet(seq(i), seq(j))

        Struct.pair.append((i, j))

    Else:

        For h = i to j

            Cr = F[i, h] + F[h+1, j]

            S1 = Traceback(F, sgm, sequence, bi, bh)

            S2 = Traceback(F, sgm, sequence, bh+1, bj)

            Struct.merge(s1)

            Struct.merge(s2)

            Brace[min(tup)] = "{"

            Brace[max(tup)] = "}"

            Join(brace)

Return struct

StructPred(F, sgm, sequence)

N = len(sequence)

For k = 1 to n

For i = 0 to n-k

j = k+ i

Bi = []

For h =i to j

Bi.append(F[i, h] + F[h+1, j])

Bfr = comp(Bi)

If comp(comp(Bi) , c) ==Bfr

Bfr != c

F[i, j] = comp(comp(Bi), c)