In this talk I will present an alternative way of analyzing sequential algorithms: critical path analysis. Critical path analysis has already been applied to parallel algorithms since decades. The critical path of an algorithm is its longest dependency chain, i.e., the longest sequence of operations depending on each other. It is well-known that if the length of the critical path is L, then with sufficiently many processing units, the parallel runtime of an algorithm can be reduced to O(L), and this is also best possible. Recently, critical path analysis has been applied to various well-known sequential algorithms for geometric problems. However, the length of the critical path of a well-known algorithm for trapezoidal decomposition has remained an open problem. I present some ideas on how to approach this.