Dijkstra's Algorithm adapted for E-Truck charging in Smart Energy Hubs

Automating the process of decision making for optimal vehicle charging route for truck drivers

Termohlen, R. P. J., Supervisor: Cliteur, M.
Project/Research Group: Project KOP, Smart Energy
Contact information: rpj.termohlen@student.avans.nl
Date: January 16th, 2025

In light of the current industry push towards electric vehicles, trucks will also be electrified. For electric cars, drivers can use programs such as *A Better Route Planner* to find the best places to charge. However, software and simulation models like this are not yet available for truck drivers.

Currently, Smart Energy Hubs are being developed. These are energy systems designed to reduce grid congestion, by locally sharing produced energy and routing the energy flow to balance supply and demand, in order to be able to still meet the contracted power take-off. Charging stations can be a part of this as well to sell excess energy. The aforementioned model will be geared towards such energy hubs, specifically De Waterlaat, in the Eindhoven Brainport region.

Methodology

In order to automate the route selection process, the shortest route along the charging points must be found, while considering whether or not the vehicle is able to reach that point. This consideration depends on the State of Charge.

A suitable method to determine this, is graph theory, specifically, *Dijkstra's Algorithm*. This mathematical algorithm find the least costly (e.g. shortest, or cheapest, etc.) route from starting point to destination [1]. This reduces the model to the following problem: Find the fastest route from the starting point *S* to destination *T*, while determining where to charge if necessary.

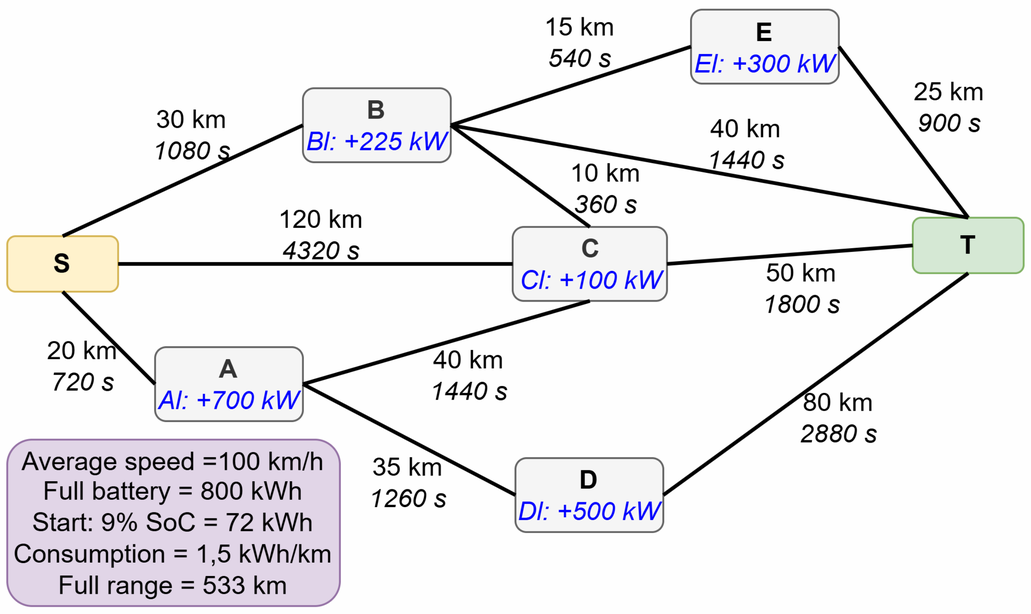

Figure 1: Example of the graph that needs to be solved in order to adapt Dijkstra's Algorithm. The weight of the graph is the calculated travelling time in seconds (in italics) with the original distance above. Points *A* to *E* have two possibilities: "charging" and "not charging". The default is "not charging". In order to distinguish between "charging" and "not charging", the charging option is treated as a separate point with a lower case *l* added to the letter (from Dutch "laden"). The available charging power is listed in blue. The purple block (bottom-left) contains the constraints the algorithm must take into consideration.

Resulting algorithm

To solve this problem systematically, Dijkstra's Algorithm has been modified to take the vehicle's State of Charge into account.

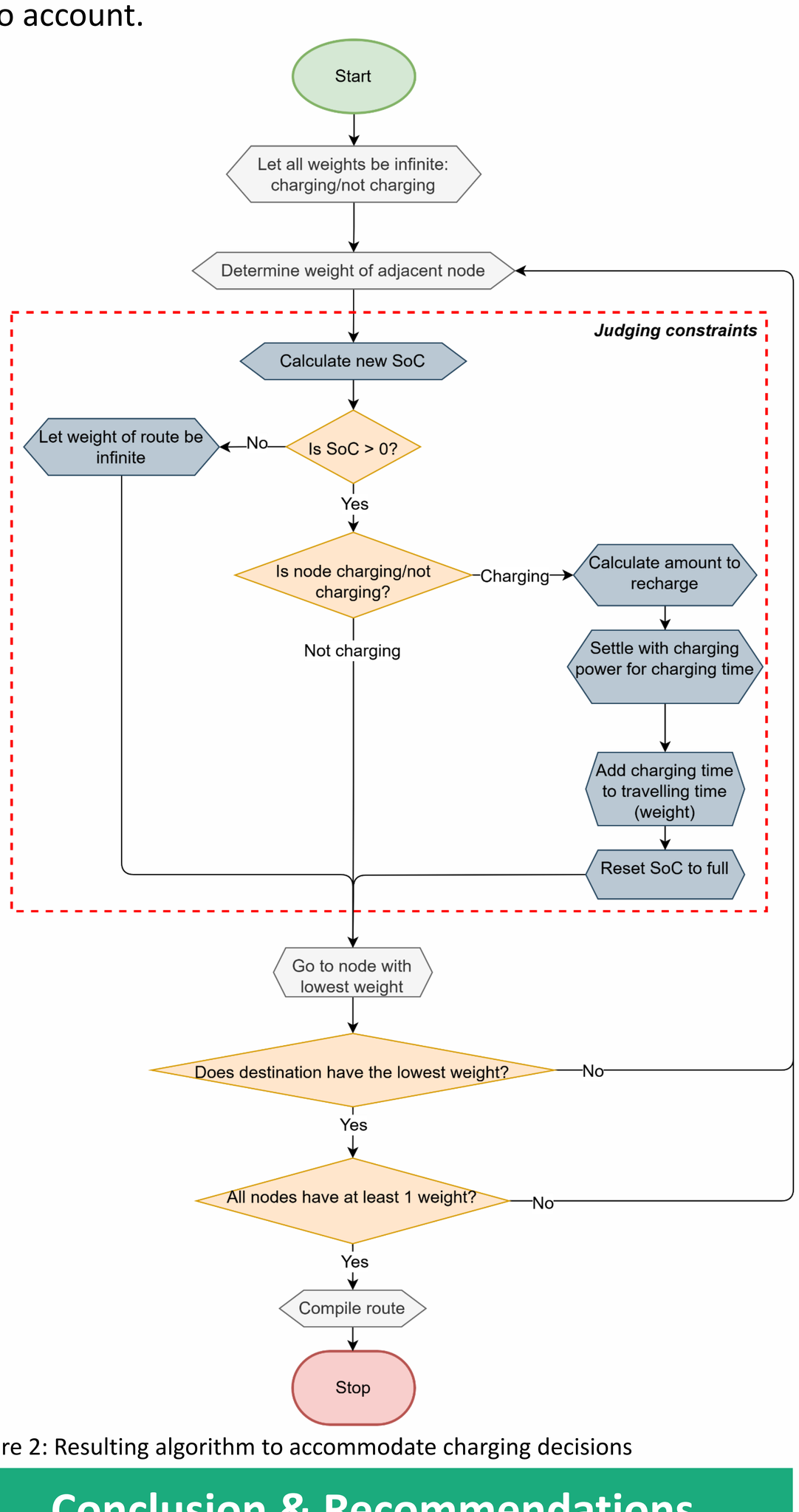

Figure 2: Resulting algorithm to accommodate charging decisions

Conclusion & Recommendations

After test runs, this algorithm turns out to be adequate to solve the problem at hand.

It is recommended for this project that the implementation be done by either IT students or programmers

References

1. Wikipedia, Kortstepad-algoritme, (n.d.), https://nl.wikipedia.org/wiki/Kortstepad-algoritme