Let be a solution to a bus transit problem, where is a graph representing bus stops and their connections and is a set of paths in representing bus lines. Let be a fitness function that evaluates solution .

We transform into a simplified version of the graph, , by “collapsing” the paths without branching into a single vertex. First, let us introduce the equivalence relation which “merges” all vertices that form a “simple path”. We define:

1. is a path in
2. for all (order is important? < ⬄ i<j?)

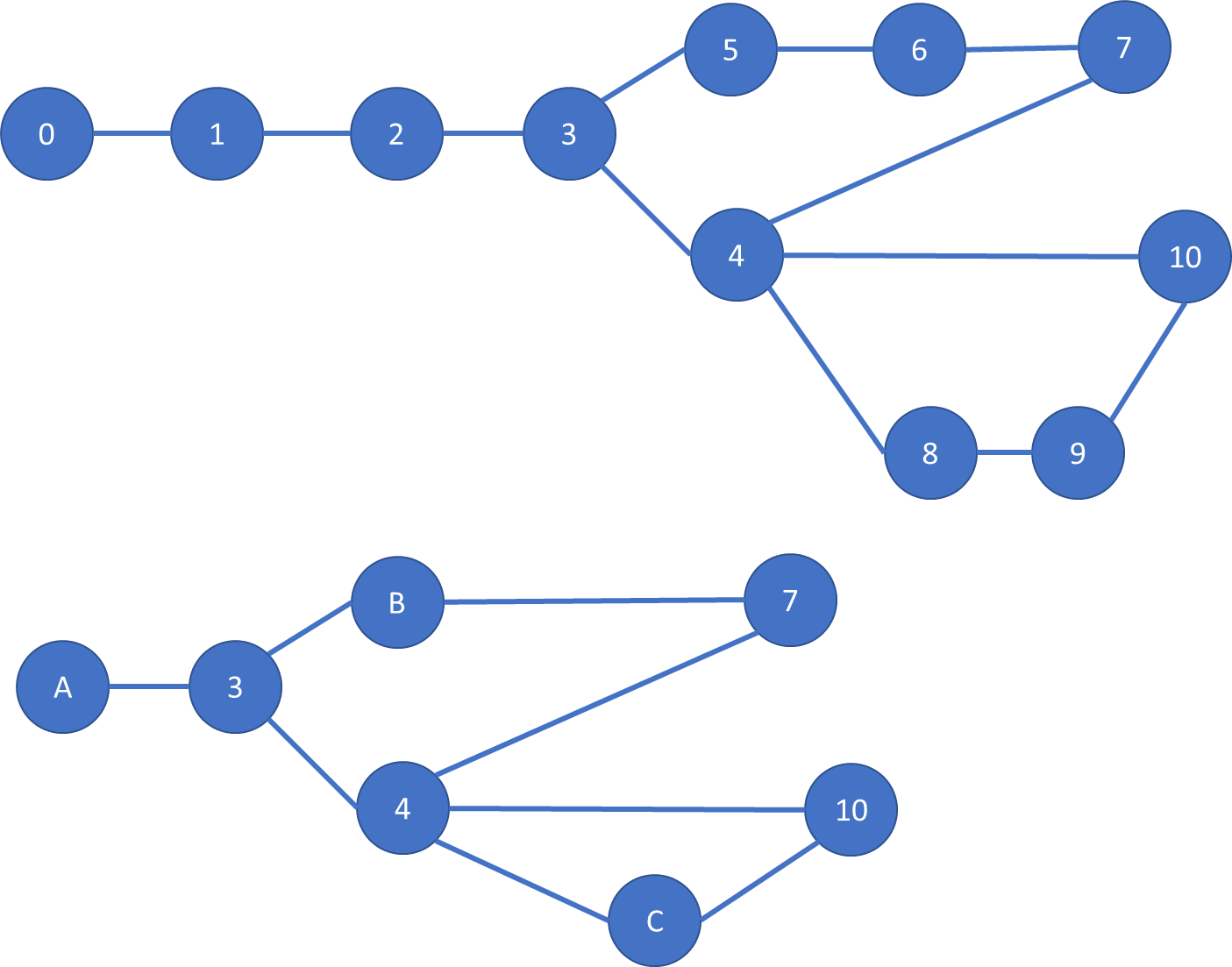
This means that all “internal” vertices have no other connections than to other (or or ) and that are either the “ends” of the graph (with deg=1) or have deg=2.

Let be a function defined as follows:

where is an equivalence class of the relation .

For a given graph we define the simplified graph , where is defined as follows: .

Example: consider the following graph (upper part). Its simplified version is shown below.



We have , and , so , , . All other equivalence classes are singletons. So:

,

,

,

,

,

.

->?

The order of vertices are considered in this example.

For example, h(8)=h(9)=[8]

We can think as h(10)=h(9)=[10]. Is it impossible? If then why?

-?

We can also reduced h(8)=h(9)=h(10)=[8]

Since it satisfies the definition of ~. Is it possible? Reason?

The simplification procedure includes the following activities:

1. Simplification of a given to
2. Transforming problem parameters in into problem parameters of (for example, the entries in the OD-matrix, information about travel time between vertices etc.)
3. Calculating fitness function for , where is a solution given in terms of the simplified graph. For example, if there is a bus line (0, 1, 2, 3, 5, 6, 7) in the corresponding solution in will be ([0], [3], [5], [7]) and we calculate fitness function for this solution in a simplified graph
4. The method of transforming simplified solution to a “full” solution (that is, a process of restoration of the OD-matrices elements, for example)

Point 4) may be not necessary, if we are able to show the following theorem:

Theorem 1. Let and be the fitness function values for two solutions , in the original graph such that . Let and be the corresponding solutions which use the simplified graph . Then .

This property of preserving the monotonicity of the fitness function allows us to transform a problem into a simplified version, solve it and then go back to the original problem – and the theorem guarantees us that if we find the best solution for a simplified version, then this is also the best solution for the original problem. Or, in general, if we have several solutions for the simplified version, ordered by the fitness function, then the original versions will also be ordered in the same way.

Of course, the property from the theorem will depend on the way we define fitness function for the simplified graph. So, the theorem may hold only when some assumptions holds regarding the fitness function . The research should investigate which properties of give us that the theorem holds.

For example, suppose the fitness function for counts only the average number of transits, suppose that and suppose that vertices form the equivalence class for and that all other vertices form singleton equivalence classes. Suppose we have some solution . Fitness function can be defined as , where is a set of passengers and is the number of transits for a passenger in a given network .

For a simplified graph we define the corresponding in the same way as , only some origins and destinations for some passengers may be changed, since some vertices may be represented by equivalence classes in (for example, in original graph we have three vertices: 1, 2, 3, but in each of them is transformed to the same vertex .

We can define as a sum of four different parts: , that count: number of transits for passengers going from some vertex from to another vertex from ; for passengers going from to some vertex outside (that is, to 4, 5, 6, 7, 8, 9 or 10); for passengers going from outside to a vertex from ; and for passengers going from outside to some other vertex outside . Let us analyze these four parts one by one, looking at the behavior of the fitness function:

Part : by definition, vertices from form a “simple” path, so these passengers will have no transfers needed, since each connection that goes from some to some such that must be a single line [note: we assume that there is no end-point in the middle! If this is the case, this won’t be true; to make our life easier we can add an additional property for the relation :

1. are not the end-stops

Hence, for the original solution and also in this case.

Part : similarly, if there’s some route for a passenger going from some v in Q to some v’ in , then in a simplified graph there will be the same number of transfers, because within Q there are no transfers needed (even if a passenger has to make a transfer, it can be easily shown that the transfer bus stop may be outside Q). So for this part as well.

Part : similar argument

Part : similar argument – if a given passenger does not travel through any vertex from , then . If a passenger travels through some vertex from , then we use a similar argument as above – the number of transfers will not change in .

Putting all four parts together, we conclude that in this case The fitness functions are equal for both original and simplified solutions.

Of course, it may happen that the equality does not hold, but, for example, is some function of . Other factors for fitness function may be more complicated for the analysis. For example, if a fitness function counts the average travel time, the question is: how should we transform into ? We don’t have a simple way as in the previous case (number of transfers), because we have somehow “average” the mean travel time for all passengers from all vertices from – the question is: can we do this in a way that will assure that the fitness monotonicity holds for ? This should be upon a research!

Other useful theorems that can be shown:

1. If a path in consists of different vertices, then a corresponding path in will also consist of different vertices (because otherwise some vertices in one equivalence class would have degree greater than 2, which contradicts the definition of )
2. [I THINK this is true – to be double-checked!] If Theorem 1 holds for some fitness function and for some other , then it holds for their linear combination . This theorem allows us to prove theorem 1 for some different “parts” of the evaluation and then “combine” them into a more complicated fitness function that takes into account some more complicated factors. For example, if we know that Theorem 1 holds for that evaluates, let’s say, the average number of transits and for other that evaluates, let’s say, average travel time, then we know that it will hold also for some fitness function that is defined as a linear combination of these two factors, so we may safely use our simplification method for such a fitness function.

Can we use the same fitness function of the paper in the simplified graph?

Can the solutions have limitation for lengths and numbers of routes?

Is it impossible that the routes can have end-stops on the nodes which don’t have branch?