A Heuristic Graph-Based Shortest Path Algorithm for Optimizing Routing Problems

E. E. Ogheneovo¹, E. Seetam²

¹Department of Computer Science University of Port Harcourt, Nigeria
²Department of Computer Science University of Port Harcourt, Nigeria

ABSTRACT: Route optimization is a process of considering all possible routes connecting the source and the destination and looks at the heuristic cost of each route and selecting the least cost route. Route planners depend principally on past occurrence of events associated with route optimization; hence they often use local knowledge, simple procedures, and ad hoc procedures to optimize the routes. In this paper, we proposed a graph-based shortest path algorithm for optimizing route directory. The algorithm is based on the Dijkstra algorithm. It is an improved shortest-path algorithm proposed as initially proposed by Dijkstra. In order to determine the shortest route and the most cost effective route, the algorithm is used to determine the shortest path that a traveler or someone going to a particular destination for the first time. The algorithm is tested by comparing its results with existing route algorithms and the results are presented and discussed.

Keywords: Dijkstra algorithm, route optimization, shortest path, cost effective, heuristic graphs

I. INTRODUCTION

Route optimization [1] [2] is the process of considering all possible routes that connects the source and the destination and looks at the heuristic cost of each route by selecting the least cost route [3]. Therefore, two or more arrays are used in the implementation of vertices, edges, etc., to generate a graph. A graph is a diagrammatical representation of mathematical structure in grid, chart, diagram, or table which gives visual interpretation that expresses relationship among entities in real world [4] [5]. Though, in object oriented programming language like Java programming language, graph can also be implemented using object; for instance, considering Cities as object of State, population and Capital; then it can be presented as below

State city0 = new City (“Rivers”, 20850045, “PH”)

...State city5 = new City (“Cross River”, 10850045, “Calabar”)

Graph algorithm [6] [7] [8] is the tool we adopt in solving the problem of route optimization by modeling the problem into a graph model. For instance, we model all the State Capitals in Nigeria into a heuristic graph model of the form G(V, E, W), such that the vertex correspond to the state capital while the edge correspond to the pathway between vertices with a cost. A path finding technique searches a graph from one vertex to the other most times from the root node and check the neighboring nodes in anticipation of the goal target node, more often with the intent of resolving the shortest pathway [9] [10]. Thus many constraints affect the search method. However, while some techniques would take enough time to find a route, other methods, may decide to use the graph technique the graph and reach the destination in lesser time.

Dijkstra's Algorithm [11] is the foundation algorithm for optimizing route directory. However, we graph-based technique to optimize the best optimal and least cost pathway that lead from a source to a target point. Thus combining weighted graph algorithm and adaptive routing provides the best means of achieving heuristic cost on itinerary optimization for diminutive pathway problems. The heuristic attribute may be physical road condition, space that is distance, route crime rate, and time delay which signify the cost of using the route. Actually, the benefit of using a pathway is the summation of its heuristic constraints [12] [13] [14]. Thus, the software selects the best path from source A to destination B with least heuristic cost. The current routing schemes are faced with problem and limitation that affect it usage by populace, this include accessibilities and user friendly. Traveling across the road is time consuming when the best route is not detected; tourist also missed their destination, consequently, transport cost is inflected by transport operator. The road network and routing system of Nigeria is not accessible to the populaces as well as complex, difficult to interpret and error prompt which contradict the federal government policy in tourism [15] [16].
Several techniques have been proposed for optimum route detection for cities travelers. However, varieties of recent navigation system have also embedded graph algorithm that decrease space and travel time delay. Moreover, with the increase in fuel rates, time delay in traffic congestions and pollution emissions, road user have become most concerned with route selection that give least cost [17] [18] [19]. In this paper, we proposed a graph-based algorithm for solving the problem of route optimization. The paper is concerned mainly with optimal route director in Nigeria tourism. Hence, it provides a model of synthesis of heuristic graph and adaptive routing. It targets a software application that propose possible path that lead to a destination using multi criterion approach optimizer that required two variables input (source and destination) [20] [21]. Route planners depend principally on past occurrence of event associated with the subject matter; hence they used local knowledge, simple procedure, and ad hoc procedures to optimize the route [22]. The use of computer-based tools for evaluating and estimating public transit routes optimization gives the best designed recommendation as stated by [23] [24]. This work is very important since it provide solution to real live problem, boaster transit and tourism sector of the economic which promote the national policy, aim and objective in tourism [25]. Thus, it provides the solution that aid government policy in tourism to daylight. Thus it increases the rate and benefit of road transit and reduce overhead enchanter in the road network routing.

II. STATEMENT OF THE PROBLEM

Several techniques have been proposed for optimum route detection for cities travelers. However, varieties of recent navigation system have also embedded graph algorithm that decrease space and travel time delay. Moreover, with the increase in fuel rates, time delay in traffic congestions and pollution emissions, road user have become most concerned with route selection that give least cost. In this paper, we proposed a graph-based algorithm for solving the problem of route optimization. The paper is concerned mainly with optimal route director in Nigeria tourism. Hence, it provides a model of synthesis of heuristic graph and adaptive routing. It targets a software application that propose possible path that lead to a destination using multi criterion approach optimizer that required two variables input (source and destination). Route planners depend principally on past occurrence of event associated with the subject matter; hence they used local knowledge, simple procedure, and ad hoc procedures to optimize the route. The use of computer-based tools for evaluating and estimating public transit routes optimization gives the best designed recommendation. This work is very important since it provide solution to real live problem, boaster transit and tourism sector of the economic which promote the national policy, aim and objective in tourism. Thus, it provides the solution that aid government policy in tourism to daylight. Thus it increases the rate and benefit of road transit and reduce overhead enchanter in the road network routing.

III. METHODOLOGY

Figure 1 shows the architecture of the proposed model of the route optimizer using graph-based technique. The model is grouped into three stages: stage 1 comprises of the evaluation of route network and route network collection. The route network evaluator is used to evaluate the different routes that are available from the source to the destination and the interconnection between these routes. After the route network valuator has determine the various routes, the route network collector is then used to collect information on all the routes and the information is then forwarded to the data optimization and data optimization which of course comprises stage 2. At the data optimization level, the route, there are four (4) components: the route node graph/map, the results of the computation of the various route lengths and resources used in the computation, the results of the various node clusters, and the component that extract the various node from tourist attractions.

Fig. 1: System Architecture of the proposed optimal route directory
This is followed by the route optimizer which contains two (2) components: the candidate route graph and the route graph optimization. The candidate route graph plots the individual lines and the interconnection to various notes in the graph while the route graph optimizer helps to determine the paths that are short and cost effective. It then selects that path that is shortest, consume less resources, and most cost effective. The last stage (stage 3) is then used to select the desires route from source to destination that will best suit the traveler from the plotted graph.

3.1 Algorithm for Simulating Optimal Route

This Algorithm provides logically an unambiguous, step-by-step procedure for performing a Routing Analyzing task and simulation of required route graph from a start to a destination base on the user option, which guaranteed termination after a finite numbers of steps.

**Algorithm 1:** A routine algorithm for graphically simulating optimal route

1: **Input:** Different routes from source to destination
2: **Output:** Distances between these routes from source to destination
3: // Initialize Variables and Storage space
4: Dist, Speed: int
5: State(Os): String
6: State(Dg): String
7: Vertices: String Sequence
8: Edges: Integer sequence
9: Edge_Weight_AdjacentMatrix: Integer Sequence
10: Os_NeighborState : String Sequence
11: Dg_NeighborState : String sequence
12: DirectLink: Boolean
13: ClusterLink: Boolean
14: //Load Storage Space in memory
15: Collect all Vertices → Sequence
16: Collect all Edges → Sequence
17: Collect all Edges_Weight → Sequence
18: Link the Vertices()
19: while k ≤ 36 {
20:   Get Vertices.Size
21: for each Vertex {
22:     Generate x,y_Coordinate for Vertex
23:     DrawFillCycle (x1,y1) Vertex
24:     DrawString( Vertex Name ,x1,y1)  
25:   }  // end for
26:   Get Edges.Size
27:   for each Edges {  
28:     Generate x,y_Coordinate for Start and End point of an edge
29:     DrawLine (x1,y1,x2,y2)
30:     DrawString(Weight,x1,y1)  
31:   }  //end while
32:   OptimaRouteGraph(OriginSate, GoalState)
33: }
34: v ← Initial_Sate
35: u ← Goal_State
36: Dist[space] ← 0
37: Prev[space] ← undefined
38: for each vertex v in Graph
39: if v ≠ visited
40: Dist[v] ← unknown
41: Prev[v] ← open-ended
42: end if
43: add v → Q
44: end for
45: while Q ≠ ε {
46:   u ← vertex in Q with min Dist[u]
// end OptimalRouteGraph
58: OptimalGraph( u, v ) {  
59:     While k ≤ visitNodePath.size {  
60:         For each neighbor v of visit node path {  
61:             Get x,y coordinate for Vertex  
62:             DrawFillCycle(x1,y1)  
63:             DrawString(VertexName, x1,y1)  
64:             Get x,y coordinate for Edge (start and end point)  
65:             DrawLine(x1,y1,x2,y2)  
66:             DrawString(Weight, x1,y1)  
67:         }  
68:      k+=1    }  
69:  } // end Optimal graph
71: 
72: Vertex Neighbor ()  
73: {  
74:     Get Os Vertex Index in Vertices Sequence  
75:     for i = 0; i ≤ Os_outEdges.size  
76:     {  
77:         Os[] ← Os_Neighbor[Os.index][i]  
78:         Dg[] ← Dg_Neighbor[Dg.index][i]  
79:     }  
80: } // end Vertex Neighbor
72: 
81: Fuel Quantities ()  
82: {  
83:     FQty ← (Total Dist/5km)  
84: }  
85: 
86: Timer ()  
87: {  
88:     ArriverTime ← Start Time + (Km/n-hours)  
89: } // end Timer
89: 
90: Tourist Attraction()  
91: {  
92:     Collect all Hotels by State  
93:     Collect all Tourism Pack by State  
94:     Identify the goal state then published its Hotels and Tourisms Center  
95: } // end Tourist Attraction
95: 
96: Graph Traversals ()  
97: dfs(vertex v)  
98: visit v  
99: for each neighbor w of v {  
100:     if (w has not been visit) {  
101:         dfs(w)  
102:     }  
103: } // end Graph Traversals
104: 
105: Graph Reverse Traversals
106: S ← empty sequence
106: u ← target
107: while prev[u] ← known {
108: insert u at the start of S
109: u ← prev[u]
110: } // end while
111: } // end

Fig. 2: An algorithm for graphically simulating optimal route

IV. RESULTS AND DISCUSSION

The system optimize the best route from Home State to Destination state by considering the least shortest distance in kilometer after detail evaluation of all possible route from Origin to Destination state extracted by system. Plot the graph of the best shortest route. In general view, all graph nodes are obtained easily. Each node have a precise key called index and a specified name, the neighbor of the vertex and the degree of the vertex are easily obtained as shown in figure 3. The graph interface model is an abstract as shown in figure 3 where all its internal data and frequent actions and event carry out on graphs, that moderately the implementation of graph interface.

Rivers State and Enugu State Routes

Consider the route between Rivers State and Enugu State; we have all possible routes. There are seven (7) possible routes between Rivers State and Enugu State (Route 1 to Route 7). Route 1: Rivers – Bayelsa – Delta – Enugu with a total distance of 338 kilometers; Route 2: Rivers – Imo – Abia – Enugu with a total distance of 301 kilometers; Route 3: Rivers – Abia – Enugu with a total distance of 281 kilometers; Route 4: Rivers – Abia – Ebonyi – Enugu with a total distance of 294 kilometers; Route 5: Rivers – Akwa-Ibom – Abia – Enugu with a total distance of 396 kilometers; Route 6: Rivers – Akwa-Ibom – Abia – Ebonyi – Enugu with a total distance of 409 kilometers; Route 7: Rivers – Akwa-Ibom – Abia – Cross River – Ebonyi – Enugu with a total distance of 392 kilometers. As can be seen, route 3 is the least and the shortest route with a total of three nodes (Rivers – Abia – Enugu) and two edges with total a distance of 281 kilometers from Rivers State to Enugu State. The system output the optima graph as shown in figure 3. The red lines represent the optimal path which passes through …

![Optimal route between Rivers State and Enugu State](image)

Rivers State and Lagos State Routes

Considering the route between Rivers State and Lagos State there are three possible routes label Route 1 to Route 3. Route 1 and route 2 have equal nodes and edges but difference distances. Route 1: Rivers – Bayelsa – Delta – Edo – Lagos with a total distance of 678km; Route 2: Route 2 which have 672 kilometer is choose to be the best optima route by the system as shown in figure 4.7 below. The red lines represent the optima path which passes through Rivers – Imo – Delta – Edo – Lagos) with weight cost of 672km.
Let us consider the route between Rivers State and Borno State, there are seven routes that connects these two states. We called them route 1 to route 7. For Route 1, we have: (Rivers – Bayelsa – Delta – Enugu – Benue – Taraba – Adamawa – Borno) with a total distance of 1527 kilometers; Route 2: (Rivers – Imo – Anambra – Taraba – Adamawa – Borno) with a total distance of 1490 kilometers; Route 3: (Rivers – Abia – Enugu – Benue – Taraba – Adamawa – Borno) with a total distance of 1470 kilometers; Route 4: (Rivers – Abia – Ebonyi – Benue – Taraba – Adamawa – Borno) with a total distance of 1483 kilometers; Route 5: (Rivers – Akwa-Ibom – Abia – Enugu – Benue – Taraba – Adamawa – Borno) with a total distance of 1585 kilometers; Route 6: (Rivers – Akwa-Ibom – Abia – Ebonyi – Enugu – Benue – Taraba – Adamawa – Borno) with a total distance of 1598 kilometers; and finally Route 7: (Rivers – Akwa-Ibom – Abia – Enugu – Benue – Taraba – Adamawa – Borno) with a total distance of 1581 kilometers. It must be noted that Route 3 is the least and the shortest route, it has total of seven nodes (Rivers – Bayelsa – Delta – Enugu – Benue – Taraba – Adamawa – Borno) and six edges with weight cost of 1470 kilometers from Rivers State to Borno State. The system output the optimal graph as shown in figure 4. The red lines represent the optima path which passes through …
V. CONCLUSION

Several techniques have been proposed for optimum route detection for cities travelers. However, varieties of recent navigation system have also embedded graph algorithm that decrease space and travel time delay. Moreover, with the increase in fuel rates, time delay in traffic congestions and pollution emissions, road user have become most concerned with route selection that give least cost. In this paper, we proposed a graph-based algorithm for solving the problem of route optimization. The paper is concerned mainly with optimal route director in Nigeria tourism. Hence, it provides a model of synthesis of heuristic graph and adaptive routing. It targets a software application that propose possible path that lead to a destination using multi criterion approach optimizer that required two variables input (source and destination). Route planners depend principally on past occurrence of event associated with the subject matter; hence they used local knowledge, simple procedure, and ad hoc procedures to optimize the route. The use of computer-based tools for evaluating and estimating public transit routes optimization gives the best designed recommendation as stated. This work is very important since it provide solution to real live problem, boaster transit and tourism sector of the economic which promote the national policy, aim and objective in tourism. Thus, it provides the solution that aid government policy in tourism to daylight. Thus it increases the rate and benefit of road transit and reduce overhead enhancer in the road network routing.

REFERENCES


