Projektgruppe 369
Pflichtenheft

December 1, 2000

Contents

Part I

Zielbestimmung

Busliniennetze leisten einen erheblichen Beitrag bei der Bewältigung von Mobilitätswünschen. Ein zunehmendes Umweltbewußtsein hat bei vielen Betrieben die Fahrgastzahlen steigen lassen. In Großstädten existieren ausgeprägte Netze, die zumeist aus historischen Netzen über viele Jahre hinweg entstanden sind und nur geringfügige Änderungen erfahren haben. Diese Netze sind jedoch häufig nicht mehr in der Lage, den heutigen Mobilitätsanforderungen in ausreichendem Maße gerecht zu werden. Gleichzeitig sehen sich die Verkehrsbetriebe aufgrund geänderter finanzieller Rahmenbedingungen einem höheren Kostendruck ausgesetzt. Aus diesen Gründen werden viele Busnetze in den nächsten Jahre reformiert werden müssen. Neben großstädtischen Netzen werden ferner in zunehmendem Maße auch in Mittelstädten eigene Stadtbusnetze gefordert und umgesetzt.

Die Festlegung der Linienwege stützt sich im großen Umfang auf die Intuition und Erfahrung von Angebotsplanern. Eine Rechnerunterstützung der eigentlichen Planung wird nur in geringem Maße geboten und dient hauptsächlich zur Veranschaulichung der Planerideen und deren Wirkungsgefüge.

Die Komplexität des Planungsproblems hängt stark davon ab, welche Zielgrößen bei der Planung berücksichtigt werden. Ein Auswahl möglicher, teilweise konkurrierender Ziele sind: Kostenminimierung, Erschließungsmaximierung, Fahrzeugparkminimierung, Reisezeitminimierung und die Minimierung der Umstiegshäufigkeit.

Eine manuelle Planung kann diesen Zielen nur sehr schwer gerecht werden. Aus der Literatur sind einige Verfahren zur Linienplanung bekannt, die beispielsweise eine manuelle Planung durch eine Bewertung unterstützen (Krug87) oder automatisch Linienpläne erstellen (Simonis81, Sonntag77).

In diesem Umfeld stellt sich für die Projektgruppe die Aufgabe:

- Das Planungsproblem bzgl. Steuergrößen, Zielgrößen und Randbedingungen genau einzugrenzen, ggfs. in verschiedene Planungsprobleme zu differenzieren.

- Die aus der Literatur bekannten Verfahren zu sichten und bzgl. ihrer Eignung zu bewerten und auszuwählen. Dies beinhaltet selbstverständlich auch die Freiheit, eigene Ideen zu entwickeln und weiterzuverfolgen.

- Ausgewählte Verfahren zu implementieren und anhand von ausgewählten Beispiel-städten zu evaluieren.

- Die Bewertung eines Liniennetzes anhand statisch ermittelbaren Größen, wie z. B. der Anzahl einzusetzender Fahrzeuge, und ggfs. auch von Simulationsstudien zu ermitteln.

Die letzten beiden Punkte bestimmen den praktischen Anteil der Projektgruppe: Er erfordert einen Software-Entwurf (bei dem moderne Entwurfstechniken wie UML (Oestereich99) eingesetzt werden sollen), der in eine Umgebung aus einem geographischen Informationssystem und einem Bewertungswerkzeug eingebettet ist. Das geographische Informationssystem dient zur Übermittlung der Daten und der Visualisierung der Planungsergebnisse. Das Bewertungswerkzeug dient dazu, die Güte des geplanten Liniennetzes ggfs. auch per Simulation zu ermitteln. Hierbei bietet es sich an, auf Werkzeuge zurückzugreifen, die allgemein zur Modellierung und Bewertung von Logistiksystemen (Kuhn95) – und hierzu gehören auch Buslinien-netze – entwickelt wurden.

Diese Projektgruppe ist thematisch verwandt mit Fragestellungen, die auch im aktuellen, interdisziplinären Sonderforschungsbereich 559 "Modellierung großer Netze der Logistik" an der Universität Dortmund unter Beteiligung der Informatik verfolgt werden. Das Thema der PG bietet somit die Möglichkeit, Kenntnisse auf einem interessanten Anwendungs- und Forschungsgebiet der Informatik zu erwerben und dabei auch klassische Themen der Informatik wie dem Softwareentwurf und Informationssysteme kennenzulernen.

Die Erstellung eines kommerziellen Produkts wird mit dieser PG nicht angestrebt.

1 Musskriterien

Minimalziel ist die Implementierung eines zumindest halb-automatischen Planungsverfahrens, das für zumindest eine Beispielstadt ein nach zumindest einer Zielgröße zufriedenstellendes Liniennetz entwirft. Diese prototypische, aber lauffähige Implementierung setzt einen entsprechenden Softwareentwurf ebenso voraus wie eine Dokumentation.

2 Modul Programm (GUI, Datenbank, Schnittstelle)

2.1 Musskriterien

2.1.1 GUI

/AP10 ANZEIGE_GEOGR

Beschreibung Dem Benutzer muss es auf einfachem Weg ermöglicht werden, die geografischen Daten der zu betrachtenden Stadt in einem Fenster anzeigen zu lassen.

Priorität 10

Rationale Im bisher verwendeten System sind zu viele Schritte notwendig, um eine grafische Ausgabe zu erhalten. Dies verringert nicht nur die Benutzerakzeptanz, sondern erhöht neben dem Zeitaufwand bei der Entwicklung der anderen Module auch die Zeit der Einarbeitung der anderen, am Projekt beteiligten Personen

/AP20 ANZEIGE_DATEN

Beschreibung In einem entsprechenden Fenster müssen die, vorwiegend in der Datenbank anfallenden, zusätzlichen Daten übersichtlich und für den Benutzer verständlich einblenden lassen.

Priorität 9

Rationale Im bisherigen System ist eine Anzeige von nicht-geografischen Daten nicht möglich. Eine Kontrolle der gespeicherten Daten ist somit nicht möglich.

/AP30 ANZEIGE_ERGEBNISSE

Beschreibung In einem entsprechenden Fenster müssen sich die, von den anderen Modulen errechneten Ergebnisse, visuell anzeigen lassen.

Priorität 10

Rationale Für die Ergebnisse steht bislang keine Ausgabemöglichkeit zur Verfügung. Durch eine entsprechende, visuelle Anzeigemöglichkeit kann der Benutzer eine wesentlich bessere Bewertung der Ergebnisse vornhemen.

/AP40 MANIPULATION_MENÜ

Beschreibung Dem Benutzer muss zur Bedienung des Systems ein Menü zur Seite gestellt werden, in dem er a) die Berechnungsprozesse der einzelnen Module starten, b) die verschiedenen, anzuzeigenden Daten auswählen und c) weitere notwendige Einstellungen vornehmen kann.

Priorität 8

Rationale Im bisherigen System fehlt ein einheitliches Menü, von dem aus der gesamte Programmablauf steuerbar ist. Ein solches Menü führt zu einer einfacheren Bedienung.

/AP50 MANIPULATION_GEOGR_LAGE

Beschreibung Dem Benutzer muss es in einem entsprechenden Fenster ermöglicht werden, alle verwendeten Objekte (hierzu zählen: Knotenpunkte, Haltestellen, Linien und Straßen) zu markieren und in ihrer Position zu verändern. Die relative Lage der betroffenen, bzw. angrenzenden oder verbundenen Objekte zum veränderten Objekt muss dabei jederzeit berücksichtigt bzw. gewährleistet bleiben.

Priorität 9

Rationale Eine Veränderungsmöglichkeit von geografischen Daten ist im bisherigen System nicht möglich. Insbesondere die Anforderungen der anderen Module machen derartige Manipulationsmöglichekiten durch den Benutzer notwendig und tragen gleichzeitig zu einer Steigerung der Benutzerakzeptanz bei.

/AP60 MANIPULATION_GEOGR_HINZUFÜGEN

Beschreibung Dem Benutzer muss es in einem entsprechenden Fenster emöglicht werden, neue Objekte (hierzu zählen: Knotenpunkte, Haltestellen, Linien und Straßen) zu erstellen und somit auch in das bestehende Umfeld einzugliedern.

Priorität 10

Rationale Eine Eingabemöglichkeit ist im bisherigen System nur über den Dateiimport möglich. Insbesondere die Anforderungen der anderen Module machen derartige Manipulationsmöglichkeiten notwendig und tragen gleichzeitig zu einer Steigerung der Benutzerakzeptanz bei.

\AP70 MANIPULATION_GEOGR_RESTRIKTIONEN

Beschreibung Bei der Veränderung oder Erzeugung eines Objektes (hierzu zählen: Haltestellen, Knotenpunkte, Linien und Straßen) durch den Benutzer oder eines Moduls, muss sichergestellt werden, dass die gewünschten Änderungen erlaubt sind und nicht gegen geografische Restriktionen verstoßen. Beispielsweise darf eine Haltestelle nicht außerhalb einer Straße liegen.

Priorität 10

Rationale Würde eine Überprüfung der Eingaben auf ihre Korrektheit nicht stattfinden, so würden diese Fehleingaben die Ergebnisse der einzelnen Module negativ beeinflussen. Insbesondere bei der Haltestellen- und Linienplanung könnten die generierten Daten durch Häuser oder andere Hindernisse hindurch verlegt werden.

/AP80 MANIPULATION_DATEN

Beschreibung Der Benutzer muss möglichst einfach und komfortabel, zu einem Objekt zusätzliche Daten eingeben, verändern und dauerhaft speichern können. Dies umfasst insbesondere die Daten, die von den einzelnen, verbundenen Modulen angefordert werden.

Priorität 8

Rationale Die angeschlossenen Module benötigen neben den geometrischen Daten auch erweiterte Daten, wie z.B. den Verkehrsfluss oder die Reichweite einer Haltestelle. Hätte der Benutzer eine solche Eingabemöglichkeit nicht, könnten vorhandene Daten nicht ergänzt und korrigiert werden. Dies hätte Auswirkungen auf die Ergebnisse der einzelnen Module.

/AP90 MANIPULATION_DATEN_IMPORT

Beschreibung Dem Benutzer muss es ermöglicht werden, bereits als Datei vorliegende, zusätzliche Daten zu einem oder mehreren Objekten, vom System einlesen zu lassen ("Importfunktion").

Priorität 10

Rationale Da der Benutzer nicht jedes notwendige Datum zu einem Objekt selbst eingeben muss, stellt eine solche Importfunktion eine enorme Erleichterung in der Bedienung dar.

/AP100 ANSICHT_FAHRPLAN

Beschreibung Der vom System generierte Fahrplan muss für den Benutzer visuell und leicht verständlich einsehbar sein.

Priorität 10

Rationale Würde diese Ausgabe nicht geschehen, könnte der Benutzer ein Hauptziel des Gesamtsystems, sowie weitere Ausgaben des Fahrplanmoduls nicht kontrollieren.

/AP110 MANIPULATION_MODULANFRAGE

Beschreibung Sofern ein Modul eine Eingabe durch den Benutzer wünscht, muss der benötigte Wert durch ein entsprechendes Fenster beim Benutzer abgefragt und an das entsprechende Modul zurückgeliefert werden.

Priorität 9

Rationale Da die einzelnen Module keine Möglichkeit zur direkten Kommunikation mit dem Benutzer haben, würden Ausgabemeldungen den Benutzer nicht erreichen. Dies gilt insbesondere für Fehlermeldungen.

/AP120 MANIPULATION_GEOGR_AUSWAHL

Beschreibung Es muss dem Benutzer ermöglicht werden, geografische Objekte (dazu zählen: Punkte, Haltestellen, Linien, Straßen) als "ausgewählt" zu markieren.

Priorität 10

Rationale Die einzelnen Module verlangen nach einer Möglichkeit, Objekte als "ausgewählt" zu markieren. Würde diese Möglichkeit nicht zur Verfügung stehen, könnten die Module nicht optimal ausgeführt werden.

2.1.2 Datenbank

/AP130 DATENBANK_DATEN_SPEICHERUNG

Beschreibung Zu den einzelnen, geografischen Objekten (dazu zählen: Punkte, Haltestellen, Linien, Straßen) müssen zusätzliche Daten dauerhaft speicherbar und jederzeit abrufbar sein.

Priorität 10

Rationale Die an das Programmmodul angebundenen Module benötigen über die geografischen Daten hinaus, zwingend zusätzliche Daten, ohne die sie ihre Arbeit nicht korrekt ausführen könnten.

/AP140 DATENBANK_FAHRPLAN_SPEICHERUNG

Beschreibung Es müssen die vom Fahrplanmodul erzeugten Fahrpläne dauerhaft speicherbar und jederzeit abrufbar sein.

Priorität 10

Rationale Das Modul zur Fahrplanerzeugung würde ohne eine Speicherung seine Ergebnisse verlieren. Darüber hinaus können die Daten auf diese Weise von anderen Modulen genutzt bzw. vom Benutzer abgefragt werden.

/AP150 DATENBANK_VERSIONIERUNG

Beschreibung Es muss eine Versionierung der erzeugten Daten möglich sein.

Priorität 8

Rationale Können bereits in einem früheren Lauf erzeugte Daten, durch eine Versionierung nicht verwaltet werden, so kann eine Rückkopplung nicht einwandfrei durchgeführt werden.

2.1.3 Interface

/AP160 INTERFACE_DATEN_LIEFERUNG

Beschreibung Die von einem Modul benötigten Daten müssen in der verlangten Form geliefert werden.

Priorität 10

Rationale Ein Modul kann ohne die Daten von Außen seine Arbeit nicht aufnehmen.

/AP170 INTERFACE_DATEN_WEITERLEITUNG

Beschreibung Daten, die von einem Modul geliefert werden, müssen an die entsprechenden Stellen (dies sind die Datenbank, die Benutzeroberfläche oder ein anderes Modul) weitergeleitet werden.

Priorität 10

Rationale Würden die Daten nicht weitergeleitet, wären die Daten verloren und somit die Arbeitsweise des gesamten Systems beeinträchtigt bzw. gefährdet.

/AP180 INTERFACE_MODUL_STEUERUNG

Beschreibung Die im System enthaltenen Module müssen verbunden und verwaltet werden. Dies beinhaltet auch die Steuerung der einzelnen Module.

Priorität 10

Rationale Würden die Module nicht verwaltet, käme ein kontrolliertes Starten der Berechnungen in den angebundenen Modulen nicht zustande, wodurch das System seine Hauptziele nicht erreichen würden.

/AP190 INTERFACE_MODUL_ANFRAGE

Beschreibung Benötigt ein Modul eine Eingabe vom Benutzer, so muss dies an die Benutzeroberfläche weitergeleitet und die von dort eingelesenen Daten wieder an das Modul zurückgeliefert werden.

Priorität 9

Rationale Ein manuelles Eingreifen des Benutzers wäre nicht möglich, wodurch wiederum Optimierungen in den Modulen nicht durchgeführt werden könnten (insbesondere im Hinblick auf die Rückkopplungen zwischen den Modulen).

/AP200 INTERFACE_MODUL_MELDUNG

Beschreibung Möchte ein Modul eine Meldung an den Benutzer senden, so muss dieser Wunsch an die Benutzeroberfläche weitergeleitet werden.

Priorität 10

Rationale Würden Meldungen an den Benutzer nicht weitergeleitet, würden diese als Folge vor dem Benutzer verborgen bleiben. Dies gilt insbesondere für Fehlermeldungen.

/AP210 INTERFACE_NETZGENERATOR

Beschreibung Es muss eine Möglichkeit geschaffen werden, die Ausgangsdaten der einzelnen Module manuell oder automatisch mit manueller Unterstützung zu generieren. Der Benutzer soll in diesem Vorgang zu Rate gezogen werden.

Priorität 10

Rationale Wäre es nicht möglich die Ausgangsadaten manuell oder automatisch erstellen zu lassen, würde dies den Test erheblich erschweren.

2.2 Wunschkriterien

/AP300 WUNSCH_NAVIGATION

Beschreibung Die Navigation innerhalb der Fenster und Menüs muss dem Benutzer so einfach, intuitiv und einheitlich wie möglich gestaltet werden.

Priorität 5

Rationale Im bisher verwendeten System ist die Benutzeroberfläche und die Bedienung durch die Mischung von textuellen und grafischen Bedienelementen nicht konsistent und daher nur schwer erlenbar.

/AP310 WUNSCH_ABSCHOTTUNG

Beschreibung Der Benutzer soll so gut wie möglich, von dem verwendeten GIS, der verwendeten Datenbank und den darunter liegenden Modulen abgeschottet werden.

Priorität 2

Rationale Da das GIS primär der Datenspeicherung dient, braucht der Benutzer nicht zusätzlich die Bedienung des GIS zu erlernen.

/AP320 WUNSCH_LÖSCHEN

Beschreibung Dem Benutzer muss es in einem entsprechenden Fenster ermöglicht werden, bereits existierende Objekte (hierzu zählen: Knotenpunkte, Haltestellen, Linien und Straßen) zu entfernen. Die relative Lage der betroffenen, bzw. angrenzenden oder verbundenen Objekte zum entfernten Objekt muss dabei jederzeit berücksichtigt bleiben.

Priorität 5

Rationale Der Benutzer konnte im bisherigen GIS einmal gemachte Änderungen nicht wieder Rückgängig machen. Gerade im Hinblick auf die Schaffung einer Einfügemöglichkeit von Objekten durch den Benutzer, trägt diese Forderung zu einer Erhöhung des Bedienkomforts bei. Darüber hinaus kann das Entfernen von Objekten (insbesondere von Haltestellen und Linien) im Hinblick auf die Rückkopplung der einzelnen Module notwendig werden.

/AP330 WUNSCH_FAHRPLAN_ÄNDERUNG

Beschreibung Der vom System generierte Fahrplan muss durch den Benutzer änderbar sein.

Priorität 5

Rationale Aufgrund der Rückkopplung der Module könnte eine Änderung des Fahrplans durch den Benutzer das Gesamtergebnis optimieren.

/AP340 WUNSCH_ZOOM

Beschreibung Das Fenster, das für die Anzeige der geografischen Daten zuständig ist, muss seine Daten in mehreren Stufen sowohl vergrößert, als auch verkleinert anzeigen können.

Priorität 2

Rationale Könnte der Benutzer den angezeigten Stadtplan nicht vergrößern, wären bestimmte Teilgebiete der Karte, aufgrund ihrer Größe nicht mehr einsehbar.

2.3 Abgrenzungskriterien

Das Modul "Programm" ist nicht zuständig für

- die Berechnung und die automatische Erzeugung der Haltestellen
- die Berechnung und die automatische Erzeugung der Buslinien
- die Berechnung und die automatische Erzeugung der Fahrpläne

3 Modul Haltestellen

3.1 Musskriterien

/AH10

Beschreibung Die Haltestellen werden gemäss den Fahrtwünschen gesetzt.

Prioritaet 10

/AH20

Beschreibung Die Pflichthaltestellen werden gemäss der von Modul Programm übergebenen Vorgaben gesetzt.

Prioritaet 10

/AH30

Beschreibung Es werden Prioritäten für die einzelnen Haltestellen ermittelt und zugewiesen.

Prioritaet 10

/AH40

Beschreibung Alle Haltestellen werden an das Strassennetz angepasst.

Prioritaet 9

/AH50

Beschreibung Abhängig von der Besiedlung wird die Haltestellenvergabehäufigkeit parametrisiert.

Prioritaet 7

/AH60

Beschreibung Ein Strassenplan, in dem die kürzesten Wege zwischen jeweils zwei Haltestellen in einem durch den User einzugebendem Radius angegeben sind, wird berechnet.

Prioritaet 8

/AH70

Beschreibung Aus dem Strassenplan wird ein Wegenetzgraph erzeugt.

Prioritaet 7

3.2 Wunschkriterien

/AH80

Beschreibung Der Fussweg überschreitet nicht eine durch den User vorgegebene Länge

Prioritaet 4

/AH90

Beschreibung Die Haltestellen werden moeglichst kreuzungsnah gesetzt.

Prioritaet 5

/AH100

Beschreibung Implementierung eines Algorithmus, der die Pflichhaltestellen automatisch setzt

Prioritaet 2

/AH110

Beschreibung Genuegend Flaechen- /Einwohnererschliessung.

Prioritaet 5

3.3 Abgrenzungskriterien

/AH120

Beschreibung Es wird weder ein Liniennetz, noch ein Fahrplan erstellt.

/AH130

Beschreibung Keine Fahrtzeitenermittlung fuer den Wegenetzgraphen.

/AH140

Beschreibung Keine Beschriftung der Knoten des Wegenetzgraphen mit Bedarfswerten.

/AH150

Beschreibung Der Wegenetzgraph enthaelt bei Abschluss des Moduls Haltestelle nur die Werte:

- ID (Knoten)
- x,y-Position (Knoten)
- Prioritaet (Knoten)
- Reisezeit in km (Kanten)
- Liste der benutzten Strassenzuege (Kanten)

4 Modul Linienplanung

4.1 Musskriterien

/AL10

Beschreibung

Linienplan gemäß den Fahrtwünschen generieren unter den Kriterien

- Direktfahrermaximierung
- Max. Linienlänge beachten

Priorität

10

/AL20

Beschreibung

Max. Linienanzahl beachten bei der Erstellung eines Linienplans

Priorität

9

/AL30

Beschreibung

Pflichtlinien beachten bei der Erstellung des Linienplans

Priorität

8

/AL40

Beschreibung

Der Linienplan soll auf Güte hin bewertet werden können

Priorität

6

/AL50

Beschreibung

Max. Linienfahrtzeit beachten bei der Erstellung eines Linienplans

Priorität

5

4.2 Wunschkriterien

/AL60

Beschreibung Es wird zur Generierung des Linienplans mindestens einer der folgenden Algos zusätzlich verwendet:

- Simonis
- Marcus' Verfahren
- eigene Verbesserungen an Sonntag

Priorität 5

/AL70

Beschreibung Statt Dijkstra wird der k-kürzeste Wege-Algorithmus von Weigand verwendet

Priorität 3

/AL80

Beschreibung verbesserter Rueckkopplungsalgorithmus

Priorität 4

4.3 Abgrenzungskriterien

- Keine gleichzeitige Beruecksichtigung von Fahrplan oder Haltestellen

5 Modul Fahrplan

5.1 Musskriterien

/AF10 Fahrplanerzeugung

Beschreibung Erzeugung eines Fahrplans durch die Angabe der räumlichen und zeitlichen Festlegung von Fahrten.

Priorität 10

/AF20 Zeitabhängige Taktänderung

Beschreibung Der Fahrplan muß für unterschiedliche Zeitgruppen erstellt werden können. Dies bedeutet, daß der Fahrplan in den Morgenstunden anders sein kann, als der Fahrplan in den Abendstunden.

Priorität 7

/AF30 Maximale Fahrzeuganzahl

Beschreibung Einhaltung einer maximalen Fahrzeuganzahl. Das ist die Anzahl von gleichzeitig fahrenden Fahrzeugen im Fahrplan.

Priorität 7

/AF40 Allgemeiner Mindesttakt

Beschreibung Ein für alle Linien gleicher Takt (allgemeiner Mindesttakt) muß berücksichtigt werden.

Priorität 10

/AF50 Auswahl von Linienpaaren

Beschreibung Es müssen Linienpaare berücksichtigt werden, die direkten Anschluß aneinander haben müssen.

Priorität 10

/AF60 Umsteigezeiten

Beschreibung Es müssen Umsteigezeiten in den Haltestellen berücksichtigt werden.

Priorität 10

/AF70 Verschiedene Takte auf verschiedenen Linien

Beschreibung Manche Linien dürfen einen größeren Takt haben als den allgemeinen Mindesttakt. Dies muß berücksichtigt werden.

Priorität 6

/AF80 Interaktive Fahrplanoptimierung

Beschreibung Der Benutzer muß immer die Möglichkeit haben in den Planungsprozeß manuell eingreifen zu können. Dies muß besonders an wichtigen Stellen geschehen.

Priorität 10

/AF90 Ausgabe eines beliebigen Haltestellenplans

Beschreibung Es muß möglich sein den Fahrplan für eine beliebige Haltestelle auszugeben, wobei die Abfahrtszeiten der einzelnen Linien an der Haltestelle angezeigt werden müssen.

Priorität 10

/AF100 Ausgabe eines beliebigen Linienfahrplans

Beschreibung Es muß möglich sein den Fahrplan für eine beliebige Linie auszugeben, wobei die Abfahrtszeiten an den einzelnen Haltestellen ausgegeben werden müssen.

Priorität 10

/AF110 Bewertung des Fahrplans

Beschreibung Der Fahrplan muß bewertet werden können. Die Bewertung muß anhand der Reisezeit, Wartezeit und Umsteigeaktivitäten erfolgen.

Priorität 10

/AF120 Bewertung von Routen

Beschreibung Es muß möglich sein eine einzelne Route eines Fahrplans bewerten zu können. Dabei gelten die gleichen Kriterien wie bei /AF110.

Priorität 7

5.2 Wunschkriterien

/AF130 Automatische Berechnung des Mindesttaktes

Beschreibung Es kann eine automatische Berechnung des allgemeinen Mindesttaktes anhand des Verkehrsstromes erfolgen.

Priorität 5

/AF140 Feste An- und Abfahrtszeiten

Beschreibung Es können feste Abfahrts- bzw. Ankunftszeiten an bestimmten Haltestellen berücksichtigt werden.

Priorität 3

/AF150 Ausgabe Reiseroute

Beschreibung Ausgabe einer Reiseroute von einer Haltestelle a zu einer Haltestelle b kann erfolgen.

Priorität 4

5.3 Abgrenzungskriterien

Part II
Produkt-Funktionen

Im folgenden wird für die Funktionen das Kürzel /F<X><Nr> (**F**unktion Modul **X**) und für die Schnittstellen das Kürzel /S<X><Nr> (**S**chnittstellen Modul **X**) gewählt. Die Nummerierung erfolgt dabei in 10er Schritten, wodurch eine spätere Änderung gewährleistet werden kann.

6 Modul Programm

6.1 Funktionalität

Das Ziel bzw. die Aufgabe dieses Moduls ist die Weiterleitung und Speicherung von Daten, die interaktion mit dem Benutzer und die Steuerung der beteiligten Komponenten.

Dem “Programmmodul” liegt eine komponentenbasierte Softwarearchitektur zugrunde. Dabei findet eine Aufteilung in drei Komponenten statt: der Benutzeroberfläche, der Datenbank und der Schnittstelle.

Die Benutzeroberfläche ist für die Darstellung der Daten (sowohl geografisch, als auch Daten aus der Datenbank) und den Interaktionen mit dem Benutzer zuständig. Die Datenbankkomponente kapselt das GIS und die, für Zusatzdaten notwendige Datenbank, und versieht sie mit einer einheitlichen Schnittstelle nach Außen.

Die Schnittstellenkomponente dient als Verbindungselement: diese Komponente übernimmt die Steuerung der anderen beteiligten Komponenten, leitet Anfragen an das entsprechende Modul weiter, konvertiert Daten in das angeforderte Format und dient gleichzeitg als Verknüpfungspunkt zu den anderen, berechnenden Modulen. Diese weiteren Module kommunizieren nur über diese Schnittstelle mit dem Programmmodul.

Alle von den Modulen gelieferten Daten werden je nach Datentyp in der entsprechenden Datenbank gespeichert und können von einem Modul von dort auch wieder abgefragt werden

Jedem Layer des GIS wird ein Typ zugordnet, damit die enthaltenen Daten im späteren Verlauf die Layer unterschieden werden können. So enthält z.B. ein Layer alle Haltestellen, ein anderer alle Straßen etc.

6.2 Schnittstellen

Zu den anderen Modulen, nicht intern zwischen GUI, Schnittstelle und Datenbank

/SP10 Haltestellen_Wertanforderung

Eingaben -

Ausgaben Übergibt die folgenden Daten ab das aufrufende Modul:

- Fahrtwünsche (=Liste von Haltestellen, die jeweils wieder eine Liste mit ihren Zielhaltestellen enthalten, entspricht "Fahrtwunschmatrix")
- Pendlermatrix (=Liste von Stadtteilen, die jeweils die Bevölkerungsdichte als Datum aufweisen)
- Straßennetz (=Liste von Lienien, die wiederum aus einer Liste von Punkten bestehen)
- Pflichthaltestellen (=Liste von Punkten)
- max. Radius des Einzugsgebietes einer Haltestelle (ein Wert)
- min. Radius des Einzugsgebietes einer Haltestelle (ein Wert)
- Radius, indem Haltestellen miteinander verbunden werden sollen (=zwei Werte, der Erste gibt den Radius im Stadtgebiet an, der Zweite den Radius in ländlichen Gebieten)

/SP20 Haltestellen_Ergebnisrückgabe

Eingaben aufrufendes Modul übergibt folgende Daten:

- Wegenetzplan
- Straßenplan (= Graph mit allen Straßen, auf denen Busse fahren können (=alle möglichen Buslinien als Straßen); dient primär zur Anzeige)

Ausgaben -

/SP30 Linienplanung_Wertanforderung

Eingaben -

Ausgaben Übergibt die folgenden Daten ab das aufrufende Modul:

- Wegenetzplan
- Liniengesamtlänge (= ein Wert, entspricht der max. zu fahrenden Kilometer einer Linie)
- maximale Linienanzahl (=ein Wert)
- minimale Linienanzahl (=ein Wert)
- maximale Linienfahrtzeit (=ein Wert)
- Verkehrsbedarfsmatrix für alle Haltestellen (=Liste von Haltestellen, wobei jede Haltestelle eine Liste/Tabelle mit den Verkehrsbedarfen zu allen anderen Haltestellen angibt
- vorgebebene Linien (= eine Liste von Linien, wobei jede Linie aus einer Liste von Haltestellen besteht)

/SP40 Linienplanung_Ergebnisrückgabe

Eingaben aufrufendes Modul übergibt folgende Daten:

- Liste von Buslinien, die selbst jeweils durch eine Liste von Haltestellen dargestellt werden
- modifizierter Wegenetzplan
- Versionierungsdatum

Ausgaben -

/SP50 Fahrplanerstellung_direkter_Anschluss

Eingaben -

Ausgaben aufrufendes Modul erhält zurück: Liste von Tripeln ("direkter Anschluss", Inhalt: zwei Buslinien und eine Haltestelle, an der sich die beiden Linien kreuzen)

/SP60 Fahrplanerstellung_max_Fahrzeuganzahl

Eingaben -

Ausgaben aufrufendes Modul erhält zurück: maximale Fahrzeuganzahl (ein Wert)

/SP70 Fahrplanerstellung_allgemeiner_Takt

Eingaben -

Ausgaben aufrufendes Modul erhält zurück: allgemeiner Takt über den gesamten Fahrplan (ein Wert)

/SP80 Fahrplanerstellung_Wegenetzplan

Eingaben -

Ausgaben aufrufendes Modul erhält zurück: Wegenetzplan

/SP90 Fahrplanerstellung_Ergebnisrückgabe

Eingaben aufrufendes Modul übergibt folgende Daten:

- eine Liste von Haltestellen, wobei zu jeder Haltestelle die Linien und die Abfahrtszeit dieser jeweiligen Linie gespeichert ist
- eine Liste von Buslinien, wobei zu jeder Buslinie die Haltestellen und die Abfahrtszeit von jeder dieser Haltestellen gespeichert ist.

Ausgaben -

/SP100 Fahrplanerstellung_Ausgangslösung_Eingaben

Eingaben aufrufendes Modul übergibt folgende Daten:

- netzbedingte Wartezeit in den Knoten
- Wartezeitaufwand im Netz
- Mindestwarteaufwand im Netz
- gesamte Wartezeitaufwand im Netz

Ausgaben -

/SP101 Fahrplanerstellung_Ausgangslösung_Ausgaben

Eingaben -

Ausgaben an das aufrufende Modul werden folgende Daten übergeben:

- netzbedingte Wartezeit in den Knoten
- Wartezeitaufwand im Netz
- Mindestwarteaufwand im Netz
- gesamte Wartezeitaufwand im Netz

/SP110 Fahrplanerstellung_Starthaltestelle

Eingaben -

Ausgaben dem aufrufenden Modul werden folgende Daten übergeben:

- Eine Haltestelle, die als Starthaltestelle dienen soll
- Abfahrtszeit
- Ankunftszeit
- Linie, an der die Zeit gilt

/SP120 Fahrplanerstellung_Menüaktivierung

Eingaben dem aufrufenden Modul werden folgende Daten übergeben:

- Menuepunkt, der geändert werden soll
- neuer Zustand (aktiviert/grau)

Ausgaben -

/SP130 Fahrplanerstellung_Wartezeiten

Eingaben -

Ausgaben Liste mit Haltestellen, wobei zu jeder Haltestelle gespeichert wird:

- Mindestwartezeit (ein Wert)
- Mindestumsteigezeit (ein Wert)

/SP140 Fahrplanerstellung_neuer_Linienplan

Eingaben -

Ausgaben -

/SP150 Meldung

Eingaben die auszugebende Meldung

Ausgaben Solbald der Benutzer due Meldung bestätigt hat, wird dies zurückgemeldet

/SP160 Laufzeit_Ausgabe

Eingaben die auszugebende Meldung (z.B. Fortschritt des Berechnnungsprozesses in Prozent)

Ausgaben -

/SP170 Versionierung_Status

Eingaben -

Ausgaben Anzahl der Pakete, entspricht somit der Identifikationsnummer des zuletzt angelegten Paketes

6.3 Funktionen

6.3.1 GUI

/FP10 Anzeige eines Layers
Diese Funktion dient dazu, einen oder mehrere, im GIS gespeicherten Layer in einem Fenster auf dem Bildschirm anzuzeigen.

Preconditions Die anzuzeigenden Layer müssen im GIS vorliegen.

Postconditions Der Layer wurde korrekt angezeigt

/FP20 Anzeige der Daten
Diese Funktion dient dazu, Daten, die in der Datenbank gespeichert sind, in einem Fenster auf dem Bildschirm anzuzeigen.

Preconditions Die Daten müssen in der Datenbank vorliegen.

Postconditions Die Daten werden in einem Fenster korrekt angezeigt.

/FP30 Anwendungsstart
Diese Funktion startet die Anwednungen, nimmt ggf. Initialisierungen der Module vor und startet die Benutzeroberfläche.

Preconditions -

Postconditions Die Anwendung mit der Benutzeroberfläche wurde gestartet.

/FP40 Modulstart
Diese Funktion stößt die Berechnungen des entsprechenden Moduls an.

Preconditions Alle zur Berechnung notwendigen Daten liegen vor.

Postconditions -

/FP50 Eigenschaftseinstellungen
Diese Funktion ruft ein Dialogfenster auf, indem der Benutzer globale Einstellungen vornehmen kann.

Preconditions -

Postconditions Die geänderten Einstellungen werden gespeichert.

/FP60 Positionsverschiebung von geografischen Daten mit der Maus
Diese Funktion erlaubt es dem Benutzer geografische Objekte in einem Fenster mit der Maus zu verschieben. Dazu wählt der Benutzer den entsprechenden Layer, klickt dann das betreffende Objekt an und verschiebt es mit seiner Maus an die neue Stelle.

Preconditions Ein Layer mit Objekten wird angezeigt.

Postconditions Der veränderte Layer wird im GIS abgelegt, sofern die Änderung gültig war.

/FP70 Positionsverschiebung durch Werteingabe
Diese Funktion erlaubt es dem Benutzer die Lage von geografischen Objekten durch die Eingabe von neuen Positionswerten in einem Fenster vorzunehmen. Dazu wählt der Benutzer den entsprechenden Layer, klickt dann das betreffende Objekt an und gibt dann in einem Fenster mit den entsprechenden Eingabefeldern die neue Position ein.

Preconditions Ein Layer mit Objekten wird angezeigt

Postconditions Der veränderte Layer wird im GIS abgelegt, sofern die Änderung gültig war.

/FP80 Gültigkeitsprüfung eines Layers
Diese Funktion prüft einen veränderten Layer, ob die veränderte, geografische Lage der Objekte gültig ist. Z.B. darf eine Haltestelle nicht innerhalb von Gebäuden platziert werden.

Preconditions Ein veränderter Layer liegt vor

Postconditions Im Falle einer nicht korrekten Eingabe wird ein Fehler zurückgegeben.

/FP90 Layerwahl
Diese Funktion erlaubt es dem Benutzer, aus einer Liste einen Layer auszuwählen. Hierzu wird ein entsprechendes Fenster geöffnet. Diese Funktion wird von anderen Funktionen benötigt.

Preconditions Es existiert mindestens ein Layer

Postconditions Es wird der ausgewählte Layer zurückgegeben

/FP100 Objekterzeugung mit der Maus
Diese Funktion erlaubt es dem Benutzer, ein neues Objekt auf einem Layer zu erzeugen. Dazu wählt der Benutzer zunächst den Layer, den er verwenden möchte. Dadurch ist die Objektart, die er hinzufügen kann festgelegt. Danach legt er durch einfaches klicken mit der Maus fest, wo ein neues Objekt erstellt werden soll.

Preconditions Es existiert mindestens ein Layer

Postconditions Der veränderte Layer wird im GIS abgelegt, sofern die Änderung gültig war.

/FP110 Objekterzeugung durch Werteingabe
Diese Funktion erlaubt es dem Benutzer, ein neues Objekt auf einem Layer zu erzeugen. Dazu wählt der Benutzer zunächst den Layer, den er verwenden möchte. Dadurch ist die Objektart, die er hinzufügen kann festgelegt. Danach wird

ihm ein Fenster mit entsprechenden Eingabefeldern angezeigt, in das er seine Eingaben vornehmen kann.

Preconditions Es existiert mindestens ein Layer

Postconditions Der veränderte Layer wird im GIS abgelegt, sofern die Änderung gültig war.

/FP120 Datenmanipulation

Diese Funktion zeigt zu einem gewählten Objekt die in der Datenbank gespeicherten Werte in einem Fenster an, in dem der Benutzer selbst Änderungen durchführen kann.

Preconditions Es exisitiert mindestens ein Objekt.

Postconditions Die Werte liegen innerhalb der erlaubten Wertebereiche und werden in der Datenbank abgelegt.

/FP130 Import

Diese Funktion importiert bestehende, Daten aus einer externen Textdatei in die Datenbank.

Preconditions Die Objekte, zu denen die Daten gehören exisitieren. Zudem muss der Benutzer angeben, zu welchen Objekten diese Daten gehören

Postconditions Die Daten wurden eingelesen und, sofern keine Fehler die Datenbestand auftraten, in der Datenbank gespeichert.

/FP140 Fahrplananzeige

Diese Funktion zeigt den Fahrplan in einem Fenster an.

Preconditions Der Fahrplan liegt vor.

Postconditions -

/FP150 Objektauswahl mit der Maus

Diese Funktion erlaubt es dem Benutzer nach der Auswahl eines Layers, mehrere Objekte durch das anklicken mit der Maus als “markiert” zu kennzeichnen

Preconditions Mindesten ein Objekt auf einem Layer liegt vor

Postconditions Die entsprechenden Objekte sind als “markiert” gekennzeichnet und werden im entsprechenden Fenster entsprechend farblich markiert.

/FP160 Objektauswahl über eine Liste

Diese Funktion erlaubt es dem Benutzer nach der Auswahl eines Layers, mehrere Objekte durch das Auswählen von Einträgen aus einer Liste als “markiert” zu kennzeichnen

Preconditions Mindesten ein Objekt auf einem Layer liegt vor

Postconditions Die entsprechenden Objekte sind als “markiert” gekennzeichnet und werden im entsprechenden Fenster entsprechend farblich markiert.

/FP170 Abfragefunktion

Diese Funktion erlaubt es anderen Modulen, eine Eingabe vom Benutzer anzufordern. Dazu wird ein entsprechendes Fenster generiert und angezeigt, in das der Benutzer die benötigten Werte eingibt.

Preconditions -

Postconditions Der vom Modul geforderte Wert, wird zurückgegeben.

/FP180 Meldungsausgabe

Diese Funktion erlaubt es anderen Modulen, dem Benutzer eine Meldung anzuzeigen. Dazu wird ein entsprechendes Fenster generiert und angezeigt. Der Benutzer muss diese Meldung über einen entsprechenden Schalter bestätigen.

Preconditions -

Postconditions Die Bestätigung des Benutzers wird an das entsprechende Modul zur Weiterverarbeitung weitergeleitet.

6.3.2 Datenbank

/FP500 Datenänderung
Die von der Schnittstelle geliefert Daten werden je nach Typ entweder im GIS oder in der Datenbank, zunächst in das Format der jeweils passenden Tabelle umgewandelt und danach die entsprechende Tabelle mit den gelieferten Daten aktuallisisert.

Preconditions Die Daten sind korrekt; es werden keinerlei Überprüfungen vorgenommen, die eine Korrektur bzgl. des verwendeten Kontextes durchgeführt.

Postconditions Daten sind in der Datenbank abgelegt

/FP510 Datenerstellung
Die von der Schnittstelle geliefert Daten werden je nach Typ entweder im GIS oder in der Datenbank, zunächst in das Format der jeweils passenden Tabelle umgewandelt, danach wird die benötigte Tabelle erzeugt und die Daten dort abgespeichert.

Preconditions Die Daten sind korrekt; es werden keinerlei Überprüfungen vorgenommen, die eine Korrektur bzgl. des verwendeten Kontextes durchgeführt.

Postconditions Daten sind in der Datenbank abgelegt

/FP520 Datenlieferung
Diese Funktion liefert die, in der Datenbank oder im GIS gespeicherten Daten in der von der Schnittstelle verlangten Form. Dazu wird, sofern nötig, eine entsprechende Umwandlung in die benötigte Datenstruktur vorgenommen.

Preconditions Die angeforderten Daten liegen bereits in der Datenbank vor

Postconditions Die Daten werden in Form der benötigten Datenstruktur zurückgeliefert.

/FP530 Initialisierung
Diese Funktion legt in der Datenbank und im GIS alle grundlegenden Tabellen bzw. Layer an und füllt diese mit evtl. vorhandenen Ausgangswerten oder mit Standardwerten.

Preconditions -

Postconditions Die grundlegenden Tabellen, auf denen die anderen Module im Folgenden aufbauen, sind in der Datenbank verfügbar.

/FP540 Versionierung
Diese Funktion sogt für die Versionsverwaltung bereits erstellter Daten. Dazu werden alle Daten eines Durchlaufes zu einem Paket zusammengefasst und mit einer Identifikationsnummer versehen, die die Reihenfolge der Erstellung des Paketes widerspiegelt.

Preconditions -

Postconditions -

6.3.3 Schnittstelle

/FP1000 Datenweiterleitung
Die von einem Modul angeforderten Daten werden an das entsprechende Modul, dass diese Daten liefern kann, weitergeleitet. Die von dem aufgerufenen Modul, aufgrund der Anfrage gelieferten Daten, werden an das ursprüngliche Modul anschließend zurückgeliefert.

Preconditions -

Postconditions Die angeforderten Daten wurden an das aufrufende Modul zurückgeliefert.

/FP1010 Modulstart
Diese Funktion stöst die Ausführung eines Moduls an.

Preconditions Das Modul, dass diesem Modul in der Rangfolge vorgelagert ist, wurde bereits erfolgreich ausgeführt.

Postconditions Das Modul startet mit seinen Berechnungen

/FP1020 Modulstopp
Diese Funktion stoppt die Ausführung eines Moduls.

Preconditions Das Moduls befindet sich in der Ausführungsphase

Postconditions Das Modul stoppt.

/FP1030 Netzgenerator
Diese Funktion erzeugt die benötigten Ausgangsdaten der einzelnen Module, wie z.B. den Wegenetzplan automatisch. Ggf. wird der Benutzer um entsprechende Eingaben gebeten. Bereits vorliegende oder berechnete Daten werden dabei als Grundlage verwendet.

Preconditions -

Postconditions Alle, von den Modulen berechneten Daten liegen in der Datenbank vor.

6.4 Vorgehensweise

Nach dem Starten der Benutzeroberfläche wird das Menü geöffnet, in dem der Benutzer das System steuern kann. Wird von dort aus das System angestoßen, werden nacheinander die einzelnen Module gestartet. Sobald ein Modul mit seinen Berechnungen fertig ist, werden diese Ergebnisse in der Datenbank gespeichert und dem Benutzer zur Korrektur vorgelegt. Danach wird das nächste Modul mit diesen Daten aufgerufen.

7 Modul Haltestellen

7.1 Funktionalität

Das Modul Haltestellen soll dazu dienen, aus einem vorgegebenen Strassennetz eine (moeglichst) optimale Besetzung von Haltestellen zu ermitteln. Dabei wird jedoch keine minimale Liste erstellt, sondern eher, im Hinblick auf das Modul Linienplan, eine erweiterte Auswahl der fuer die Betrachtung relevanten Haltestellen.

Zuerst berechnet das Modul aus dem uebergebenen Strassennetz die Kreuzungen und daraus resultierend die Entfernungen zwischen Ihnen.

Nach dem Setzen der Pflichthaltestellen, die die hoeschste Prioritaet erhalten, werden weitere Haltestellen gesetzt.

Abschliessend werden der Wegenetzgraph und der Strassenplan berechnet.

Zur Datenvor- und Nachbereitung benoetigt das Modul die Schnittstelle Anlieferung(2.1.6) und Ablieferung(2.1.6). Insbesondere die zweite ist fuer das Modul Linienplan von enormer Bedeutung und stellt somit einen Grenzfall zwischen beiden Modulen dar.

Die Funktionalitaet kann im zweiten Prototypen gemaess den Wunschkriterien erweitert und verfeinert werden.

7.2 Funktionen

/FH10 Kreuzungen bestimmen
Da das uebergebene Strassennetz keine Kreuzungen enthaelt, muessen sie von uns selbst errechnet werden, damit man sieht, wo sich Strassen kreuzen.

Preconditions Strassennetz als Liste von Linien

Postconditions Strassennetz mit Kreuzungen

/FH20 Entfernungen berechnen
Es werden die Strassenkilometer der Strecken zwischen den ermittelten Kreuzungen berechnet.

Preconditions Strassennetz mit Kreuzungen

Postconditions Strassennetz mit Kreuzungen und Entfernungen.

/FH30 Anpassen einer Haltestelle an das Strassennetz
Da nicht gewaehrleistet werden kann, dass die Haltestellenkoordinaten exakt den Koordinaten(-liste) einer Strasse entsprechen, wird die Haltestellenkoordinate bei Bedarf angepasst.

Preconditions Koordinaten einer Haltestelle, Strassennetz

Postconditions angepasste Haltestellenkoordinaten

/FH40 Setzen der Pflichhaltestellen
Die vom User bestimmten Pflichthaltestellen werden gesetzt, bei Bedarf an das Strassennetz angepasst und erhalten die Prioritaet 10. Beim Anpassen der Haltestellen wird /FH30 benutzt.

Preconditions Koordinaten aller Pflichhaltestellen, Strassennetz

Postconditions (angepasste) Koordinaten

/FH50 Setzen der restlichen Haltestellen
Anhand von Fahrtwunschmatrix und Bevoelkerungsdichte werden einzelne Koordinaten von Haltestellen ermittelt. Anschliessend werden sie mit Hilfe von /FH30 ans Strassennetz angepasst. Die Hoehe der Fahrtwuensche bestimmt die zu vergebende Prioritaet. Ein von der Schnittstelle gegebener maximaler und minimaler Radius der Erschließung kann die Haltestellensetzung beeinflussen. Jede Haltestelle erhaelt eine eindeutige Nummerierung (ID)

Preconditions Fahrtwunschmatrix, Bevoelkerungsdichte, Strassennetz

Postconditions (angepasste) Koordinaten von Haltestellen mit Prioritaet von 1-10 (1 niedrigste Prioritaet, 10 hoechste Prioritaet) und ID

/FH60 Berechnen des Strassenplans
Der Strassenplan enthaelt alle moeglichen Strassen, auf denen Busse fahren koennten. Er wird nur zur Anzeige von der GUI benoetigt.

Preconditions abgeschlossene Haltestellenvergabe, Verbindungsradius, Strassennetz

Postconditions Graph wie oben beschrieben.

/FH70 Berechnen des Wegenetzgraphen
Der fuer das Modul Linienplan erforderliche Wegenetzgraph wird erstellt. Ein Knoten repraesentiert eine Haltestelle in dem zusaetzlichen Daten abgespeichert sind: eindeutige ID, Koordinaten, Prioritaet und Bedarf. Zwei Haltestellen erhalten eine verbindende Kante, wenn sie sich innerhalb einer manuell bestimmten Entfernung zueinander befinden. An der Kante wird gespeichert: Verkehrsfluss, Liste der realen Strassen, Luftlinienentfernung und Strassenentfernung.

Preconditions alle HS gesetzt und angepasst, Strassennetz mit Entfernungen, Pendlermatrix, Strassenplan

Postconditions Wegenetzgraph (s.o.)

7.3 Preconditions des Moduls

Eingabe

- Fahrtwuensche
- Strassennetz
- Pendlermatrix
- min. Radius des Einzugsgebietes einer HS
- Radius, in dem Haltestellen miteinander verbunden werden sollen

Optionale Eingabe

- Pflichthaltestellen (Schulen, Denkmaeler,...)
- Bevoelkerungsverteilung
- Toleranzgrenze des Fussweges (= max. Radius des Einzugsgebietes einer HS)

Bedingungen

- Pflichthaltestellen koennen, muessen aber nicht an das Strassennetz angepasst sein.

7.4 Postconditions des Moduls

Ausgabe

- Strassenplan
- Wegenetzgraph

Bedingungen

- Alle Haltestellen sind an das Strassennetz angepasst
- Wenigstens eine Haltestelle vorhanden
- Graphen beinhalten korrekte Werte und sind zusammenhaengend

Randbedingungen Es wird keine minimale Anzahl von Haltestellen ausgegeben, sondern eher eine erweiterte Liste mit Auswahlmoeglichkeiten fuer das Modul Linienplan.

Noetige Benutzerinteraktionen:

- Pflichthaltestellen (Mauseingabe)
- max. Radius des Einzugsgebietes einer Haltestelle (Werteeingabe)
- min. Radius des Einzugsgebietes einer Haltestelle (Werteeingabe)
- Radius, in dem Haltestellen miteinander verbunden werden sollen (Werteeingabe)

Alle Benutzeraktionen erfolgen am Anfang, vor dem Start des Moduls.

7.5 Schnittstellen

/SH10 Init
Initialisieren des Moduls, Vorbereitungen

/SH20 Start
Starten der Berechnung

/SH30 Stopp
Anhalten des Moduls

/SH40 Werteuebergabe
Uebergabe der erforderlichen Werte (s. Preconditions des Moduls)

/SH50 Ergebnisrueckgabe
Rueckgabe des Strassenplans und des Wegenetzgraphen an die Schnittstelle

7.6 Team und Rollen

- Joerg Draeger
- Marc Pompl
- Jin-Ha Tchoe

8 Modul Linienplan

8.1 Funktionalität

Das Modul erzeugt aus einem vorgegebenen Wegenetz mit vorgegebenen Verkehrswünschen und einschränkenden Parametern (Liniengesamtlänge, Linienanzahl) nach dem Verfahren von Sonntag einen Linienplan. Dabei kann es auch auf einen vorgegebenen Linienplan zurückgreifen und ihn bearbeiten. Zum Zweck der Rückkopplung kann der Algorithmus mit neuen Parameter aufgerufen werden.

8.2 Schnittstellen

/SF10 Initialisierung
Initialisiert das Modul Linienplan.

Eingabe Wegenetzplan mit Linien.

Ausgabe Boolean: *True* wenn Initialisierung erfolgreich abgeschloßen wurde. Sonst *False*

/SF20 Start
Startet das Modul Linienplan.

Eingabe NULL

Ausgabe Linienplan.

/SF30 Stopp
Stoppt das Modul Linienplan.

Eingabe NULL

Ausgabe Boolean: *True* wenn das Modul erfolgreich gestoppt wurde.

Diese Schnittstellen werden später spezifiziert:

Anlieferung Aus der Schnittstelle werden folgende Daten übernommen:

- Wegenetz
- Linien (Optional)
- Parameter (Linienlänge, Linienanzahl)

Ablieferung An die Schnittstelle werden folgende Daten übergeben:

- Liste von Linien, welche wiederum Haltestellenfolgen darstellen mit Fahrgastanzahlen zwischen den Haltestellen
- Der Wegenetzplan ist entsprechend modifiziert

8.3 Funktionen

/FL10 Initialisierung

Preconditions das Modul ist noch nicht gestartet worden

Postconditions die initialen Variablen und Umgebungen sind gesetzt

/FL20 Benutzer-Eingaben
Diese Funktion koordiniert die Benutzereingaben, d.h. sie weiss wann eine Anfrage an den Benutzer abzusenden ist, bzw. ▷ Klassen innerhalb des Moduls Linienplan. (unter Umständen wird so was gebraucht)

Preconditions Keine

Postconditions Die Anfrage an den Benutzer wurde an die GUI weitergegeben.

/FL30 Linienplanausgabe
In dieser Funktion wird der vorhandene Linienplan so aufbereitet, daß die GUI ihn entsprechend darstellen kann.

Preconditions Ein gültiger Linienplan liegt vor.

Postconditions Die Daten wurden entsprechend aufbereitet an die GUI weitergegeben,d.h. der Wegenetzplan ist um die entsprechenden Daten erweitert worden und eine Liste von Linien erzeugt.

/FL40 Kontrolle
Diese Funktion koordiniert die komplette Berechnung. (Hauptfunktion des Sonntags)

Preconditions Graph liegt vor, allgemeine Preconditions wurden beachtet (z.B. liegen Daten über die Liniengesamtlänge vor)

Postconditions Die Berechnung wurde durchgeführt

/FL50 Kontrolle-überarbeitung
Diese Funktion koordiniert die Berechnung, wenn ein Linienplan schon vorliegt bzw. erweitert werden soll. (es muß noch geklärt werden was genau bei dieser Funktion aufgerufen und gemacht werden muß (Aufbereitung))

Preconditions Graph liegt vor, vorgegebene Linien sind bestimmt

Postconditions Die Berechnung wurde durchgeführt

/FL60 Datenbeschaffung
Diese Funktion führt Anfragen an die im Modul Programm gelagerte Datenbank via Schnittstelle durch. Dabei handelt es sich hauptsächlich um den Wegenetzplan, die F_{ij} und die Umgebungsvariablen des Algorithmus.

Preconditions Daten werden benötigt.

Postconditions Daten wurden beschafft und, entsprechend aufbereitet, richtig weitergeleitet.

/FL70 Netzplan
ist von allen internen Klassen aus erreichbar. Verwaltet den internen Graphen und übernimmt das Mappen auf die globale Datenstruktur Wegenetzplan (noch sehr viel darüber nachzudenken und zu entscheiden)

Preconditions Wegenetzplan liegt vor

Postconditions Wegenetzplan ist im Bezug auf Linienplan angepaßt

/FL80 Knotenreduktion
Überflüssige Knoten werden entfernt und deren Bedarf auf die Nachbarknoten umgelegt.

Preconditions Wegenetzplan liegt vor.

Postconditions Systemknotennetzplan liegt nun vor und kann bearbeitet werden.

/FL90 Umlage
Legt die Nachfrage auf dem Systemknotennetzplan um, in dem sie auf dem kürzesten Weg zwischen allen Haltestellen die zu den bereits erzeugten Strom auf den betroffenen Kanten den jeweiligen Fahrtwunsch hinzu.

Preconditions Systemknotennetzplan liegt vor.

Postconditions Die entsprechenden Bewertungen liegen vor.

/FL100 Knotenaufspaltung
Die Systemknoten werden entsprechend der Vollvermaschungin Umsteigeknoten aufgespalten.

Preconditions Systemknotennetzplan liegt vor.

Postconditions Es liegt ein Graph vor, der vom Algorithmus nach Sonntag bearbeitet werden kann. Es handelt sich hierbei um den angepassten Wegenetzplan.

/FL110 Basislinienbestimmung
Diese Funktion berechnet alle möglichen Basislinien und sortiert sie in einer Liste der Größe des Basisstroms nach.

Preconditions angepasster Wegenetzplan liegt vor.

Postconditions Liste der Basislinien

/FL120 Basislinenwahl
Entlang der Liste werden die Basislinien ausgewählt und anschließend entsprechend der Restriktionen (Gesamtlinienlänge, Haltestellenanschluß) angepaßt.

Preconditions Liste der Basislinien und angepasster Wegenetzplan liegen vor.

Postconditions Eine Auswahl von Basislinien wird als Liste zurückgegeben.

/FL130 EindeutigeLinienverknüpfung Endkanten von Basislinien werden verknüpft, wenn sie eindeutig und erlaubt sind.

Preconditions Auswahl von Basislinien sind gegeben.

Postconditions Eine Menge von Linien (bereits verknüpfte Basislinien) und Basislinien liegen vor.

/FL140 MehrdeutigeLinienverknüpfung
Die optimalen Verknüpfungen von mehrdeutigen Basislinienüberdeckungen (an den Endkanten) werden gewählt.

Preconditions Eine Menge von Linien und Basislinien liegen vor.

Postconditions Es liegen nur noch Linien vor.

/FL150 Attraktivitätsberechnung
Der Attraktivitätsgewinn bei der Verknüpfung zweier Linien auf einer Kante wird berechnet.

Preconditions Zwei sich überlagernde Linien werden übergeben.

Postconditions Der Wert wurde berechnet.

/FL160 Aufbruchverknüpfung
Die optimalen Aufbruchverknüpfungen (Definition siehe Sonntag) werden gefunden und durchgeführt.

Preconditions Es liegen Linien vor.

Postconditions Es liegen immer noch Linien vor. ;-)

/FL170 Bewertung
Legt die Nachfrage auf den Wegenetzplan um, wobei aber nur die Kanten des Linienplans benützt werden dürfen Berechnet die durchschnittliche Fahrzeit, die durchschnittliche Umsteigehäufigkeit und andere Kriterien.

Preconditions Wegenetz und Linienplan liegen vor.

Postconditions Die entsprechenden Bewertungen sind berechnet und eingetragen.

8.4 Vorgehensweise

1. Datenbeschaffung
2. Datenaufbereitung
3. Verknüpfungsphase
4. Bewertungsphase
5. Datenrückübertragung
6. InteraktionVorgegebenerLinienplan
7. InteraktionVeränderungderRandbedingungen(Rückkopplung)

Datenbeschaffung

1. Einlesen des Wegenetzplans in die interne Datenstruktur “Netzplan”.

Datenaufbereitung

1. Reduktion
   (a) Umlegung der Verkehrsbedarfswerte auf die Systemknoten
   (b) Aufspaltung der Systemknoten in Umsteigeknoten

Verknüpfungsphase:

1. Basislinienbestimmung
   (a) Basislinienwahl
   (b) Eindeutige Linienverknüpfung
   (c) Mehrdeutige Linienverknüpfung
   (d) Aufbruchsverknüpfung

Bewertungsphase

1. Umlegung der Verkehrsbedarfmatrix auf den Linienplan
   (a) Bewertung und Berechnung von Gütekriterien

InteraktionVorgegebenerLinienplan

1. Anwender gibt einen Linienplan vor
   (a) Aufbereitungsphase
   (b) eventuell Basislinienwahl, wenn es die Gesamtlänge zuläßt
   (c) Mehrdeutige Verknüpfungsphase
   (d) Aufbruchsverknüpfung
   (e) Bewertungsphase

InteraktionVeränderungderRandbedingungen(Rückkopplu

1. veränderte Liniengesamtlänge
   (a) veränderte Linienanzahl
   (b) veränderte maximale Linenfahrzeit
   (c) Verknüpfungsphase
   (d) Bewertungsphase

8.5 Team und Rollen

- Reza Scheikh-ol-Eslami
- Michael Schlüter
- Stefan Magerstedt
- Alexander Schluck

9 Modul Fahrplan

9.1 Funktionalität

Das Ziel bzw. die Aufgabe unseres Moduls ist die Minimierung des Reisezeitaufwandes. Der Reisezeitaufwand ergibt sich hierbei aus der Summe der Produkte aus Anzahl Reisender und Reisezeit.

Da die Fahrzeiten zwischen den Haltestellen, sowie die Mindesthaltezeiten und Mindestumsteigezeiten vorgegeben sind, kann mittels des Fahrplans nur die netzbedingte Wartezeit

(Wartezeit die über die Mindestwartezeit und Mindestumsteigezeit hinausgeht) beeinträchtigt werden.

Ziel des Fahrplans ist die Minimierung der netzbedingten Wartezeit (natürlich unter Beachtung des Verkehrsstroms).

Die Erreichung dieses Ziels soll durch Darstellung der zeitlichen Abhängigkeiten des vorgegebenen Liniennetzes bzw. Netz der Transportverbindungen mittels des sogenannten Transportkettengraphen (TKG) erfolgen.

Der TKG läßt sich aus dem Netz der Transportverbindungen dadurch ableiten, daß für jede Kante ein Knoten erzeugt wird, der somit die Fahrtzeit zwischen zwei Haltestellen symbolisiert. Knoten werden dann durch eine gerichtete Kante verbunden, wenn eine Verbindung hinsichtlich des gegebenen Liniennetzes möglich ist. Die Bewertung der sich ergebenen Kanten spiegelt die netzbedingte Wartezeit (zusätzlich multipliziert mit dem entsprechenden Verkehrsstrom) wieder. Dieses bedeutet, daß die Bindungen den Warteaufwand angeben.

9.2 Schnittstellen

/SF10 Initialisierung
Initialisiert das Modul Fahrplan.

Eingabe Wegenetzplan mit Linien.

Ausgabe Boolean: *True* wenn Initialisierung erfolgreich abgeschloßen wurde. Sonst *False*

/SF20 Start
Startet das Modul Fahrplan.

Eingabe NULL

Ausgabe Fahrplan.

/SF30 Stopp
Stoppt das Modul Fahrplan.

Eingabe NULL

Ausgabe Boolean: *True* wenn das Modul erfolgreich gestoppt wurde.

/SF40 Heuristische Fahrplanerstellung
Erstellt eine Ausgangslösung mit heuristischen Mitteln. Es wird dabei kein Optimum garantiert.

Eingabe NULL

Ausgabe Boolean: *True*, wenn die Rechnung beginnt.

/SF50 Sukzessive Lösungsverbesserung
Vebessert die Lösung der Heuristischen Fahrplanerstellung

Eingabe NULL

Ausgabe Boolean: *True*, wenn die Rechnung beginnt.

/SF60 Entscheidungsbaum Größenbeschränkt
Es wird ein Optimaler Fahrplan erstellt.

Eingabe Anzahl der Knoten die eine Zusammenhangskomponente höchstens haben darf damit auf dieser Zusammenhangskomponente das Entscheidungsbaumverfahren angewandt werden soll.

Ausgabe Boolean: *True*, wenn die Rechnung beginnt.

/SF70 Entscheidungsbaum Stopkriterium
Wie Schnittstelle /SF70 ausser das nach endlich vielen Schritten gestoppt wird.

Eingabe Iterationsschritte nach denen das Entscheidungsbaumverfahren stoppen soll.

Ausgabe Boolean: *True*, wenn die Rechnung beginnt.

/SF80 Fahrplan Ausgeben
Gibt den Fahrplan aus. Nur möglich nach /SF40 oder /SF60 oder /SF70

Eingabe NULL

Ausgabe Gibt den erzeugten Fahrplan an die Schnittstelle.

9.3 Funktionen

/FF10 Einlesen eines Graphen
Diese Funktion dient dazu, den von der Schnittstelle des Programmes gelieferten Graphen einzulesen und entsprechend der, im Modul Fahrplan vereinbarten Datenstruktur aufzubereiten.

Anforderung /AF10

Preconditions Graph gemäß Schnittstellenvereinbarung

Postconditions Bereitstellung eines Graphen gemäß interner Schnittstellenvereinbarung (Netz der Transportverbindungen).

/FF20 Transportkettengraph
Diese Funktion dient dazu, den Aufbau eines Transportkettengraphen (TKG) aus dem, von der Schnittstelle erhaltenen Graphen zu steuern. Dabei werden auch Verkehrswünsche usw. entsprechend interner Vereinbarung umgelegt.

Anforderung /AF10; /AF60

Preconditions Der eingelesene Graph (Linienplan) wurde von /FF10 entsprechend aufbereitet.

Postconditions Ein TKG gemäß Vereinbarung wurde erstellt.

/FF30 Anforderung des Mindestverkehrsstroms
Erfragt bei dem Modul Programm den zu berücksichtigenden Mindestverkehrsstrom auf einer Linie.

Anforderung /AF10

Preconditions Keine.

Postconditions Der Mindestverkehrsstrom ist bekannt.

/FF40 Reduktion des TKG
Diese Funktion reduziert den, in /FF20 erstellten TKG, indem sie beispielsweise alle zweiwertigen Knoten entfernt.

Anforderung /AF10

Preconditions Ein TKG ist vorhanden.

Postconditions Die TKG wurde korrekt reduziert und ist weiterbearbeitbar.

/FF50 Zusammenhangskomponenten
Diese Funktion stößt die Berechung der Zusammenhangskomponenten eines vorhandenen TKG an.

Anforderung /AF10

Preconditions TKG ist vorhanden.

Postconditions Die Zusammenhangskomponenten des betrachteten TKG wurden berechnet (evtl. Ausgabe)

/FF60 Zyklomatische Zahl
Diese Funktion stößt die Berechnung der zyklomatischen Zahl eines vorhandenen TKG an.

Anforderung /AF10

Preconditions TKG ist vorhanden.

Postconditions Die zyklomatische Zahl des betrachteten Transportkettengraphen wurde berechnet.

/FF70 Startlösung
Diese Funktion stößt die Berechnung einer Ausgangslösung, sowie des zugehörigen Simplextableaus mittels Maximalgerüsten an, dabei können einzelne Zusammenhangskomponenten einzeln betrachtet werden.

Anforderung /AF10

Preconditions TKG vorhanden, zyklomatische Zahl bekannt, Zusammenhangskomponenten bekannt.

Postconditions Für betrachteten Graphen wurde zulässige Ausgangslösung, sowie Simplextableau berechnet.

/FF80 Ausgabe Kennwerte der Lösung
Diese Funktion berechnet die folgende Kennwerte einer vorhandenen Lösung und gibt sie an die Schnittstelle weiter, diese läßt sie auf dem Monitor ausgeben:

- Die netzbedingte Wartezeit in den Knoten.
- Der Wartezeitaufwand, d.h. das Produkt aus netzbedingten Wartezeiten und der Anzahl der Reisenden für alle Bindungen.
- Der netzbedingte Wartezeitaufwand im Netz.
- Der Mindestwarteaufwand im Netz.
- Der gesamte Wartezeitaufwand im Netz.

Anforderung /AF80; /A110

Preconditions Eine Lösung wurde berechnet.

Postconditions Die aufgezählten Werte wurden berechnet und an die Schnittstelle weitergegeben.

/FF90 Interaktive Lösungverbesserung
Diese Funktion, am Besten aufrufbar mittels eines Menüs auf GUI-Ebene, ermöglicht dem Benutzer im Anschluß an /FF70 und /FF80 (vorher entsprechendes Menü, oder einzelne Menüpunkte auf GUI-Ebene disabled) die erhaltene Ausgangslösung wahlweise und abwechselnd wie folgt zu verbessern:

1. Die Lösung interaktiv und schrittweise verbessern (Auswahl von Bindungen ohne netzbedingte Wartezeiten). Aufruf /FF120.
2. Heuristisches Verfahren der sukzessiven Lösungsverbesserung (Tausch von Knotenbindungen, siehe Weigand Seite 65).
3. Das exakte Entscheidungsbaumproblem bei kleinen Problemen (Test, ob dies möglich übernimmt nach Auswahl des entsprechenden Menüpunktes übernimmt im zweiten Prototyp /FF170).
4. Entscheidungsbaumverfahren mit heuristischen Abbruchkriterien. Aufruf /FF170

Im Anschluss an die Verbesserung wird /FF80 mit der neu erhaltenen Lösung aufgerufen. Der Benutzer kann die interaktiv berechnete Lösung behalten, oder verwerfen, bzw. als alternative Lösung zwischenspeichern. Zu diesem Zweck wird Funktion /FF210 verwendet.

Anforderung /AF80

Preconditions Eine Ausgangslösung ist vorhanden, sowie deren Kennwerte sind dem Benutzer bekannt.

Postconditions Eine Lössungsverbesserung wurde gemäß Benutzerwunsch berechnet, die Kennwerte wurden ausgegeben. Mit erhaltener Lösung wurde entsprechend Benutzerwunsch verfahren.

/FF100 Einlesen Mindesthalte- und Umsteigezeit
Diese Funktion gibt die Anforderung der Mindesthalte- und Umsteigezeit, einzugeben durch den Benutzer, an die Schnittstelle und nimmt von ihr schließlich den angeforderten Wert entgegen.

Anforderung /AF60

Preconditions Werte für Mindesthalte- und Umsteigezeit werden benötigt.

Postconditions Die Werte für Mindesthalte- und Umsteigezeit sind im Modul verfügbar.

/FF110 Einlesen Taktzeit
Diese Funktion gibt die Anforderung der Taktzeit, einzugeben durch den Benutzer, an die Schnittstelle weiter, und nimmt schließlich den angeforderten Wert von ihr entgegen.

Anforderung /AF40; /AF70; /AF130

Preconditions Die Taktzeit wird benötigt.

Postconditions Die Taktzeit ist im Modul verfügbar.

/FF120 Einlesen von Bindungen ohne netzbedingte Wartezeit
Diese Funktion gibt die Anforderung an die Schnittstelle, den Benutzer zu fragen, ob er oder sie Bindungen ohne netzbedingte Wartezeiten auswählen möchte und falls ja, welche dies seien sollen. Für den Fall, daß der Benutzer Bindungen gewählt hat, stößt diese Funktion die Funktion /FF190 an. Sollten gewählten Bindungen nicht plausibel sein, wird eine erklärende Fehlermeldung ausgegeben. Neuwahl dann möglich.

Anforderung /AF50

Preconditions ???

Postconditions Wurden Bindungen ohne netzbedingte Wartezeit gewählt, so stehen diese bereit. Durch die Kontrollfunktion ist die Auswahl korrekt.

/FF130 Fahrplanaufbereitung
Diese Funktion stößt die Aufarbeitung des bisher berechneten Fahrplanes an.

Anforderung /AF90; /AF100; /AF150

Preconditions Ein gültiger Fahrplan wurde berechnet.

Postconditions Die Daten wurden, entsprechend aufbereitet, an die GUI weitergegeben.

/FF140 Qualitätssicherung
Diese Funktion übernimmt die Qualitätskontrole eines berechneten Fahrplans. Hierunter fällt unter anderem die Überprüfung, ob die maximale Fahrzeuganzahl eingehalten werden kann und die Bewertung des Fahrplans. Weitere Kriterien sind denkbar.

Anforderung /AF30; /AF110; /AF120

Preconditions Ein gültiger Fahrplan ist vorhanden.

Postconditions Die Funktion liefert ,,Qualitätsverletzungen" zurück.

/FF150 Kontrolle
Diese Funktion koordiniert die komplette Berechnung.

Anforderung

Preconditions Keine

Postconditions ???

/FF160 Simplexberechnungen
Diese Funktion führt die im Verfahren von Weigang notwendigen Simplex Berechnungen durch. Dabei können bestimmte Parameter gesetzt werden.

Anforderung /AF10

Preconditions Zumindest ein TKG ist vorhanden.

Postconditions Die Simplexberechnungen wurden korrekt durchgeführt und die Ergebnisse zurückgegeben.

/FF170 Entscheidungsbaum
Diese Funktion führt die im Verfahren von Weigang beschriebenen Entscheidungsbaum Berechnungen durch. Dabei hat diese Funktion die Fähigkeit unter bestimmten Kriterien abzubrechen, bzw. sie kann vom Benutzer angehalten werden. Eine Entscheidung, ob eine Heuristik verwendet werden soll oder nicht, vorgenommen durch den Benutzer, ist angedacht.

Anforderung /AF10

Preconditions Ein TKG existiert.

Postconditions Die Berechnungen wurden korrekt durchgeführt und die Ergebnisse zurückgegeben.

/FF180 Kruskal
Diese Funktion dient der Berechnung eines Maxiamlgerüstes, dabei können vom Benutzer bestimmte Verbindungen vorgegeben werden.

Anforderung /AF10; /AF50

Preconditions TKG, sowie evetl. Benutzereingaben sind vorhanden.

Postconditions Das Maximalgerüst wurde entsprechend der Preconditions berechnet.

/FF190 Plausibilität
Da das Verfahren von Weigand interaktiv arbeitet, an mehreren Stellen also Benutzereingaben vorgenommen werden können, welche direkten Einfluß auf die Berechnung haben, werden diese mittels dieser Funktion daraufhin untersucht, ob sie zulässig sind. Es ist denkbar, diese Funktion zu zerlegen und entsprechenden anderen Funktionen quasi einen Vor-Ort-Test zu spendieren.

Anforderung

Preconditions Eine Benutzereingabe wurde gemacht.

Postconditions Antwort, ob Eingabe erlaubt.

/FF200 Datenbeschaffung
Diese Funktion ist in der Lage, indirekte Anfragen, daß heißt Anfragen über eine Schnittstelle an die, im Modul Programm gelagerte Datenbank zu entrichten. Die Anfragen können dabei aus beliebigen anderen Funktionen innerhalb des Moduls Fahrplan stammen.

Anforderung /AF30; /AF40; /AF50; /AF60; /AF70; /AF140

Preconditions Daten werden benötigt.

Postconditions Daten wurden beschafft und, entsprechend aufbereitet, richtig weitergeleitet.

/FF210 Datenspeicherung
Diese Funktion ist in der Lage, zu speichernde Daten an die Schnittstelle weiterzugeben, diese legt sie in der Datenbank unter einem entsprechenden Schlüssel ab.

Anforderung /AF90; /AF100; /AF150

Preconditions Es sind zu speichernde Daten vorhanden.

Postconditions Die Daten wurden gespeichert.

/FF220 Statusmeldung
Diese Funktion teilt der Schnittstelle zu bestimmten Zeiten (müssen noch definiert werden) den Status im Berechnungsprozeß mit.

Anforderung

Preconditions Fahrplanerstellung ist an einer bestimmten Stelle angekommen

Postconditions Keine

9.4 Vorgehensweise

Modulpunkte:

1. Initialisierung und Start
2. Datenaufbereitung
3. Reduktion des Transportkettengraphen
4. Festlegen der Berechnungsart
5. Eingabe der benötigten Daten
6. Weiter je nach Berechnungsart:
   - Heuristisches Verfahren
     (a) Erzeugung einer Ausgangslösung
     (b) Ausgabe von Grundwerten der Ausgangslösung
     (c) Entscheidung über weiteren Ablauf:
         i. Heuristischen Verfahren fortsetzen
         ii. Sukzessive Lösungsverbesserung

iii. Entscheidungsbaumverfahren beginnen
iv. Fahrplanausgabe

- Entscheidungsbaumverfahren

7. Fahrplanerstellung

Beschreibung:

Initialisierung und Start

1. Das Modul muß erst vom Programm initialisiert werden. Dies geschieht über Schnittstelle /SF10, wobei der Wegenetzplan in Form eines Multigraphen übermittelt wird.
2. Start des Moduls über die Schnittstelle /SF20.

Datenaufbereitung

1. Erzeugung des vollständigen TKG

Reduktion des Transportkettengraphen

1. Alle Knoten die einen In- und einen Outgrad von eins haben werden wie in Weigand beschreiben zusammengefasst.

Festlegen der Berechnungsart

1. Das Modul wartet, bis vom Programm die Entscheidung des Planungsverfahrens mitgeteilt wird. Dies erfolgt mittels Schnittstellen /SF40, /SF60 oder /SF70.

Eingabe der benötigten Daten

1. Alle für die Berechnung benötigten Daten werden von der Schnittstelle abgefragt.
   (a) maximale Fahrzeuganzahl
   (b) Allgemeiner Takt
   (c) Mindesthalte- und Umsteigezeiten in den Haltestellen
   (d) Feste Bindungen zwischen den Linien

Weiter je nach Berechnungsart:

Heuristisches Verfahren

1. Erzeugung einer Ausgangslösung
   (a) Bestimmung der Zusammenhangskomponenten des Transportkettengraphen
   (b) Berechnung der zyklomatischen Zahl
   (c) Überprüfung auf Kreisfreiheit (ggf. Fehlermeldung)
   (d) Die vom Anwender gewählten Bindungen werden in die Gerüste aufgenommen
   (e) Berechnung der Maximalgerüste der Zusammenhangskomponenten (unter Beachtung der gewählten Bedindungen)
   (f) Erzeugung der Simplextableaus (pro Zusammenhangskomponente) der fundamentalen Kreismatrizen und Berechnung der sich ergebenen netzbedingten Wartezeiten
2. Ausgabe der Grundwerte der Ausgangslösung
   (a) netzbedingten Wartezeiten in den Knoten
   (b) Wartezeitaufwand (Produkt aus netzbedingten Wartezeiten und der Reisendenanzahl)
   (c) netzbedingter Wartezeitaufwand im Netz
   (d) Mindestwartezeitaufwand im Netz
   (e) gesamte Warteaufwand im Netz
   (f) Benötigte Fahrzeuganzahl
3. Entscheidung über weiteren Ablauf
   (a) Heuristischen Verfahren fortsetzen. Die Berechnung des Fahrplans läuft weiter. Schnittstelle /SF40
   (b) Sukzessive Lösungsverbesserung. Änderungen der Grunddaten sowie Interaktiver Eingriff in der Ausgangslösung muß erfolgen. Schnittstelle /SF50
   (c) Entscheidungsbaumverfahren. Schnittstelle /SF60 und /SF70
   (d) Fahrplanausgabe: Erstellt den Fahrplan mit den bissher Berechneten Werten und übergibt ihn

an das Programm. Schnittstelle /SF80

(e) Speichern der aktuellen Lösung. Muß vom Programm übernommen werden.

Entscheidungsbaumverfahren

1. Berechnung eines optimalen Fahrplans für eine Zusammenhangskomponente desen Knotenzahl geringer ist als die obere Schranke die in der Schnittstelle übergeben wurde. Schnittstelle /SF60.
2. Berechnung eines Fahrplans mit n-Iterationen. Schnittstelle /SF70. Optimalität des Fahrplans nicht gewährleistet.

Fahrplanerstellung

1. Abfrage einer festen Abfahrts- bzw. Ankunftszeit einer Linie an einer Haltestelle.
2. Ausgabe des Fahrplans als Linienpläne
3. Ausgabe des Fahrplans als Haltestellenpläne.

Sukzessiven Lösungsverbesserung:

Interaktion Fahrplanoptimierung
Auswahl der Optionen wahlweise und abwechselnd:

1. schrittweise interaktive Lösungsverbesserung

(a) Anwender entscheidet, welche Bindung keine Wartezeit mehr und welche Bindung stattdessen Wartezeiten erhalten soll

(b) Berechnung der neuen Lösung. Wird durch Schnittstelle /SF50 eingeleitet.

(c) Ausgabe der berechneten Daten

(d) Gegenüberstellung mit bisher gespeicherte Lösung

(e) Anwender trifft Entscheidung, ob neue Lösung übernommen werden soll

(f) Weitere Lösungsverbesserung Anwender überlassen

2. heuristische Verfahren der sukzessiven Lösungsverbesserung (Lokale Suche)

(a) Austausch von Nicht- und Basisvariablen in den jeweiligen Simplextableaus (Berücksichtigung der jeweiligen Wartezeitaufwände)

(b) Verbesserte Lösung gefunden, erneute Suche

Interaktion Veränderung der Randbedingungen
Nach den Punkten 1-3 muß eine neue Ausgangslösung berechnet werden.

1. Anwender verändert Fahrzeuganzahl
2. Anwender verändert Taktzeit
3. Anwender verändert Mindestwartezeiten und Mindestumsteigezeiten
4. Anwender fordert neuen Linienplan an mit Angabe von Eigenschaften (Linienanzahl, Linienlänge). Modul Fahrplan beendet sich und muß neu initialisiert und gestartet werden.

Part III
Modul-Design

10 UML-Design

Part IV
Benutzungsoberfläche

- ausführlich
- möglichst Screenshots, wobei jeder Button, jede Eingabezeile etc. von der Funktionalität her dokumentiert ist.

Part V

Globale Testszenarien

Hier führt jede Gruppe auf, wie sie Tests vornimmt, welche Szenarien sinnvollerweise getestet werden.

11 Modul Fahrplan

Hier führt jede Gruppe auf, wie sie Tests vornimmt, welche Szenarien sinnvollerweise getestet werden

Part VI

Ergänzungen

12 Glossar

Arc So wird unter GRASS jede Vektorlinie genannt, die selbst aus einer Folge von Punkten definiert sind.

Ausgangslösung Gültiger nicht optimaler Fahrplan.

Entscheidungsbaumverfahren Normales Branch-And-Bound-Verfahren

Haltestellenprioritaet Jede Haltestelle bekommt eine Priotritaet zugewiesen, die die Bedeutung dieser Haltestelle ausdrueckt.

Layer Ein Layer wird vom GIS bereitgestellt. Es handelt sich dabei um eine Ebene, die eine Menge von geografischen Objekten aufnehmen kann. Die verschiedenen geografischen Elemente, wie Straßen und Flüsse, lassen sich so bei der grafischen Anzeige in einem Fenster übereinander legen.

Maximalgerüst Maximaler Spannbaum eines bewerteten Graphen

Mindestumsteigezeit Mindestdauer, die ein Fahrgast benötigt, um an einer bestimmten Haltestelle die Linie zu wechseln

Mindesthaltezeit Mindestdauer, die ein Bus an einer Haltestellen warten muß (zum Ein- und Auszeigen), bis er seine Fahrt wieder aufnehmen kann

Netz der Transportverbindungen Multigraph der zwischen zwei Haltestellen soviele Kanten enthält wie dort Linien entlang fahren

Netzbedingte Wartezeit Wartezeit die über die Mindestwartezeit und Mindestumsteigezeit hinausgeht

Pflichthaltestellen Haltestellen, welche unbedingt zu beruecksichtigen sind, wie z.B. Schulen, Denkmaeler, Gewerbegebiete, etc.

Reisezeitaufwand Summe (bezüglich aller Haltestellenbeziehungen) der Produkte aus Anzahl Reisender und Reisezeit.

Site Ein einzelner Punkt unter GRASS, sofern er nicht Teil einer Arc ist.

Strassennetz Das Netz aller Strassen, die GRASS zur Verfuegung stellt.

Strassenplan Das Strassennetz, reduziert auf die kuerzesten Wege zwischen allen Haltestellen in einem vorgegebene Radius.

Taktzeit Die Taktzeit gibt an, wie oft pro Zeitintervall (normalerweise 1 Stunde) ein Bus auf einer Linie fahren soll.

Transportkettengraph Der TKG läßt sich aus dem Netz der Transportverbindungen dadurch ableiten, daß für jede Kante ein Knoten erzeugt wird, der somit die Fahrtzeit zwischen zwei Haltestellen symbolisiert. Knoten werden dann durch eine gerichtete Kante verbunden, wenn eine Verbindung hinsichtlich des gegebenen Liniennetzes möglich ist. Die Bewertung der sich ergebenen Kanten spiegelt die netzbedingte Wartezeit (zusätzlich multipliziert mit dem entsprechenden Verkehrsstrom) wieder. Dieses bedeutet, daß die Bindungen den Warteaufwand angeben.

Verkehrsstrom Anzahl Reisender auf einer bestimmten Strecke bezüglich einer bestimmten Linie

Verkehrsbedarfsmatrix vollständige Matrix zwischen allen Haltestellen (Linienbezogen) mit Angabe der jeweilige Reisendenzahl.

Wegenetz Graph mit bewerteten Kanten und Knoten. Die Knoten repräsentieren genau die Haltestellen und die Kanten eine Strassenfolge zwischen Haltestellen, markiert mit der Länge des Weges zwischen den Haltestellen und dem kürzesten Weg als Strassenfolge zwischen den Haltestellen.

Wegenetzgraph Ein Graph, dessen Konten die Haltestellen sind und die Kanten die Luftlinienverbindung aller Haltestellen in einem gewissen Umkreis. In den Konten sind die Koordinaten, die Haltestellenprioritaet und zur eindeutigen Identifizierung eine ID gespeichert. An den Kanten steht die Strassenentfernung und die Strassenabfolge.

Wegenetzplan Graph, der folgende Zusatzdaten spreichert:

an den Knoten

- ID (eindeutige Zahl)
- x,y Position (Koordinaten innerhalb des Stadtplanes)
- Priorität (Zahl)
- Bedarf (=Liste von Haltestellen, wobei zu jeder dieser Haltestellen gespeichert, wie viele Personen von dem Knoten zu dieser Haltestelle wollen, entspricht also einer Zeile der Quelle-Ziel-Matrix)
- Umsteigezeit

an den Kanten = Verbindung zwischen zwei Haltestellen

- Verkehrsfluss (Zahl)
- Reise- oder Fahrtzeit (Zahl)
- Liste der realten Straßen, über die diese Kante führt
- Mindestwartezeit in einer Haltestelle
- Liste mit Buslinien, die über diese Kante fahren, wobei zu jeder Buslinie folgendes gespeichert wird:
  1. Ankunftszeit (ein Zeitdatum)
  2. Abfahrtszeit (ein Zeitdatum)
  3. Linienfluss (Zahl; entspricht Fahrgaststrom)
  4. Takt an der Linie

Zyklomatische Zahl Überbestimmtheit bezüglich eines Gerüstes einer Zusammenhangskomponente eines Graphen, d.h. Differenz zwischen Kantenanzahl und Knotenanzahl-1

13 Literatur

Law, A. M., Kelton, W. D. Simulation modeling und analysis, McGraw-Hill 1982 bzw. 1991

Krug, S. Ein interaktives Programmsystem zur Angebotsplanung für den liniengebundenen öffentlichen Personennahverkehrs, Braunschweig 1987

Simonis, C., Wall, H. Omnibusnetze zur Erschließung von Verdichtungsräumen und deren Randzonen, Bonn 1981

Sonntag, H. Linienplanung im öffentlichen Nahverkehr, Berlin 1977

Kuhn, A. Prozeßketten in der Logistik. Verlag Praxiswissen, Dortmund 1995.

Oestereich, B. Objekt-orientierte Softwareentwicklung mit der Unified Modeling Langauge, Addison-Wesley, 1999

Stroustrup, B. The C++ programming language, Addison-Wesley 1986

Flanagan, D. Java in a nutshell, O'Reilly 1996

Harrison, M., MacLennan, M. Effektiv Tcl, Tk programmieren, Addison-Wesley, 1998

Dickmann, F., Zehner, K. Computerkartographie und GIS, Westermann, 1999

Neumann, K., Morlock, M. Operations Research, Hanser 1993

Werner Weigand Graphentheoretisches Verfahren zur Fahrplangestaltung in Transportnetzen unter Berücksichtigung von Pufferzeiten mittels Interaktiver Rechentechnik; Diss. Braunschweig; 1981