

Bundeswettbewerb Informatik: Aufgabe 4

Lösungsidee

Zum Finden der Senderstandorte benutze ich folgende Methode:

Trifft man beim Durchgehen der Karte auf einen Pixel der noch nicht in einem Sendegebiet eines Senders liegt, dann setzt man einen Sender so, dass dieser den Pixel und noch möglichst viele andere unbedeckte Land-Pixel (also „Landmasse“) bedeckt. So wird gewährleistet, dass jeder Sender voll ausgelastet ist und man somit eine sehr geringe Anzahl an Sendern erhält, die ganz Europa mit dem Internetzugang versorgen.

Diese Lösungsmethode beachtet jedoch nicht, dass es eventuell mehrere Senderstandorte gibt, die die selbe Anzahl an Pixeln neu bedecken würden. In meiner Umsetzung merke ich mir immer nur den ersten optimalen Senderstandort.

Auf diese Weise wird errechnet, dass 80 Sender ausreichen, um ganz Europa mit Sendegebietern zu bedecken.

Da es für jeden zubedeckenden Landpixel mehrere optimale Senderstandorte gibt, habe ich testweise statt des Ersten einen Zufälligen ausgewählt. Nach mehreren Durchläufen konnte die Anzahl der Sender, die nötig sind ganz Europa abzudecken, auf 75 reduziert werden. Da allerdings auch Lösungen mit 90 Sendern auf diese Weise gefunden wurden, entschied ich mich für die Methode den ersten optimalen Standort zu wählen.

Mein erster Ansatz dieses Problem zu lösen war ein Brute-Force-Algorithmus. Dieser Algorithmus müsste alle Möglichkeiten durchgehen, eine bestimmte Anzahl an Sendern auf der Karte zu platzieren.

Findet der Algorithmus eine Möglichkeit, in der alle Landpixel von Sendegebietern bedeckt sind, so müsste die Anzahl der zu platzierenden Sender reduziert und die Suche neu gestartet werden. Ergibt eine Suche keine Möglichkeit die Sender zu platzieren, so dass die ganze Landfläche bedeckt ist, so hat man die optimale Lösung im Durchgang zuvor bereits gefunden.

Bei diesem Brute-Force-Algorithmus erhält man die Anzahl der möglichen Senderstandorte durch folgende Formel:

$$M = \binom{P}{S} = \binom{57160}{74} = \frac{57160!}{74! (57160 - 74)!} \approx 3 \cdot 10^{244}$$

Formel zur Bestimmung der Anzahl der Möglichkeiten beim Brute-Force-Algorithmus ausgerechnet mit 74 Sendern auf allen Landpixeln.

M: Anzahl der Möglichkeiten

P: Anzahl der Pixel (57.160 Landpixel)

S: Anzahl der Sender (74)

Schon bei der Platzierung von 74 Sendern auf 57160 Landpixel (von mir ermittelt) ergeben sich zu viele Möglichkeiten. Wenn die Wasserfläche noch dazukommen würde ergebe das einen Binomialkoeffizienten von 153182 über 74. Da diese Berechnung sehr lange dauern würde, entschied ich mich für eine schnellere Methode, die allerdings nicht das optimale Ergebnis liefert.

Programm-Dokumentation

Meine Lösungsidee ist von mir in der Programmiersprache C umgesetzt. Zum Lösen dieser Aufgabe werden die Strukturen **Kreis**, **Europa** und **Flag** benötigt. In der Struktur **Kreis** werden die „Rohdaten“ von der Bitmap-Datei **circle.pbm** in Form eines 2-dimensionalen **char**-Arrays gespeichert. Dabei enthält das erste Array-Element auch den ersten Pixelwert aus der Bitmap-Datei. Die Struktur **Europa** besteht aus einem Array der Struktur **Flag**. Diese Struktur **Flag** beinhaltet wiederum die Information, ob ein Pixel zur Landmasse gehört, der Pixel von einem Sendegebiet bedeckt wird und/oder auf dem Pixel selbst ein Sender steht. Außerdem besitzt die Struktur **Europa** noch eine Variable, wo die Anzahl der gesetzten Sender gespeichert wird. In den Strukturen **Europa** und **Kreis** wird zusätzlich noch die Höhe und Breite des jeweiligen Arrays gespeichert.

Das Programm kann mit bestimmten Optionen als Parameter aufgerufen werden. Eine Beschreibung aller möglichen Optionen bekommt man, wenn man das Programm mit der Option **-h** aufruft. Werden dem Programm keine Parameter übergeben, dann werden die Dateien **map.pbm** und **circle.pbm** als „Eingabedateien“ verwendet und nur nach Standorten auf der Landfläche gesucht. Das Bild mit den eingezeichneten Sendern wird standardmäßig in **standorte.ppm** abgespeichert.

In der main-Funktion wird die Funktion **finde_standorte** aufgerufen, die die Senderstandorte herausfindet, diese in eine Struktur **Europa** „einzeichnet“ und schließlich ein Zeiger auf die Struktur zurückgibt. Je nach Option platziert die Funktion **finde_standorte** die Sender entweder nur auf dem Festland oder auch auf der Wasserfläche.

Die Funktion **finde_standorte** lädt zunächst die Pixeldaten der einzelnen **pbm**-Dateien in die Strukturen **Kreis** (Variable **kr**) und **Europa** (Variable **eu**) und initialisiert diese und eine weitere temporäre Struktur Europa (Variable **temp**).

Die Variable **temp** wird zum Suchen der besten Senderstandorte benötigt. Auf der **temp**-Europakarte wird ein Sender auf einen „Land-Pixel“ gesetzt, der noch nicht von einem Sendegebiet bedeckt ist. Mit Hilfe der Funktion **setze_flags_fuer_ein_sender** wird in **temp** das Sendegebiet als bedeckt markiert: Die bedeckt-Flags der Struktur **Flag** innerhalb der Struktur **Europa** werden fuer alle gesetzten Pixel der Variable **kr** gesetzt.

Es werden nun nacheinander Standorte für Sender gesucht und bewertet. Alle in **temp** als bedeckt markierten Standorte werden dabei in Betracht gezogen. Je nach Operationsmodus kommen als Senderstandorte jedoch nur Landmasse oder der gesamte bedeckte Bereich in Frage. Als Bewertungskriterium dient dabei die Anzahl der neu bedeckten Landpixel. Wurden alle möglichen Standorte bewertet wird der erste Senderstandort, der die meisten Landpixel neu bedeckt in die **eu**-Karte gesetzt und sein Sendegebiet als bedeckt markiert. Die Variable **temp** wird wieder initialisiert und der Vorgang wiederholt sich bis ganz Europa von Sendegebieten bedeckt wird.

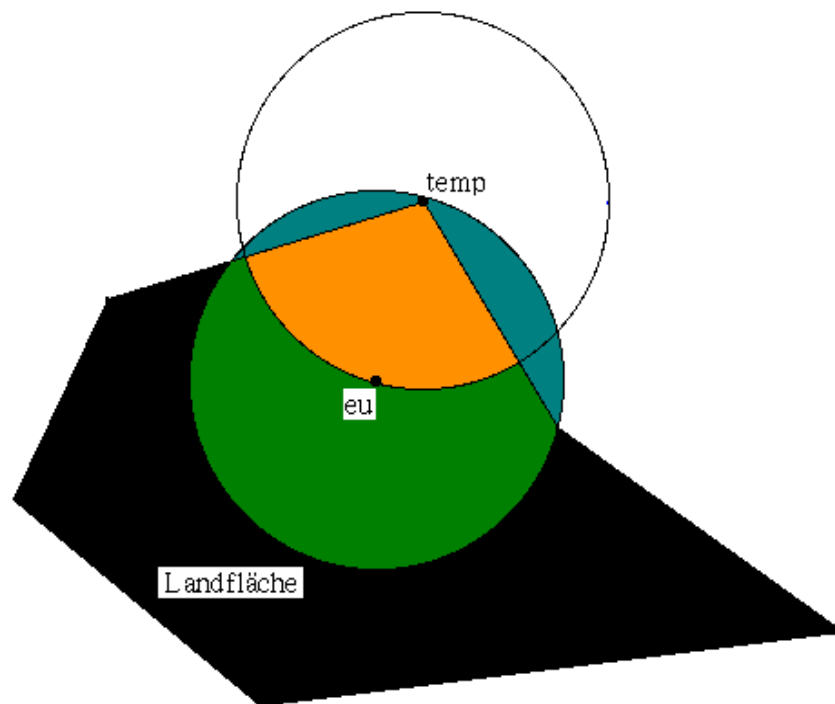


Abbildung 1: ein optimaler Senderstandort auf der Landfläche

Bedeutung der Farben:

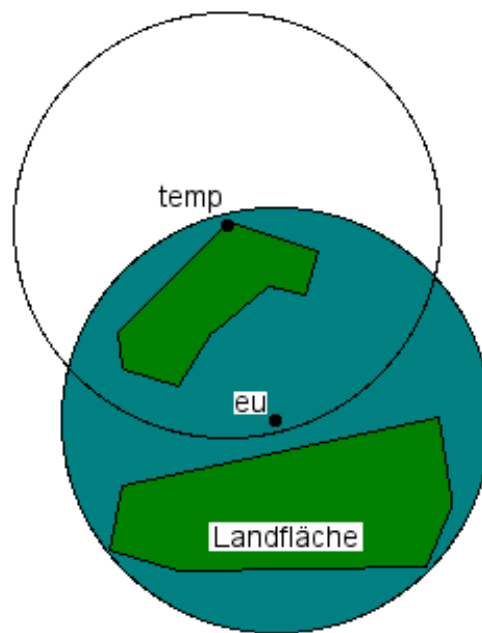
schwarz: Landpixel/Festland

grün: Festland wird vom Sendegebiet bedeckt

türkis: Wasserfläche wird vom Sendegebiet bedeckt

orange: mögliche Senderstandorte

Der Punkt temp ist hier der temporär auf dem ersten unbedeckten Land-Pixel in der Europakarte gesetzte Sender. Der optimale Sender eu wurde in diesem Beispiel auf dem Sendekreis von temp platziert, denn dort ist die bedeckte Landfläche maximal und der anfangs unbedeckte Landpixel bei temp ist noch im Sendegebiet von eu enthalten.



*Abbildung 2: ein optimaler
Senderstandort im Wasser*

Wieder wurde hier zunächst ein Sender auf dem zuerst gefundenen unbedeckten Landpixel gesetzt (in die Europastruktur **temp**) und danach nach einem optimalen Standort für ein Sender gesucht (in diesem Beispiel durfte der Sender auch auf der Wasserfläche platziert werden).

Sucht man also für jeden einzelnen Sender einen optimalen Standort, so wird auch ganz Europa mit einer sehr geringen Anzahl an Sendern bedeckt.

Programm-Ablaufprotokoll

Das Programm wird ohne Parameter gestartet und die Dateien **map.pbm** und **circle.pbm** liegen im selben Verzeichnis wie das Programm selbst:

```
$ ./aufgabe4
Starte Suche fuer Senderstandorte (nur Landflaeche)
Senderstandorte wurden in standorte.ppm eingezeichnet
Anzahl Sender: 80
```

Um die zweite Teilaufgabe (Sender dürfen auch auf Wasserfläche platziert werden) zu lösen, gibt man dem Programm die Option **-w** als Parameter an:

```
$ ./aufgabe4 -o standorte_wasser.ppm -w
Starte Suche fuer Senderstandorte (auch Wasserflaeche)
Senderstandorte wurden in standorte_wasser.ppm eingezeichnet
Anzahl Sender: 80
```

In diesem Fall wurde auch noch die Option **-o standorte_wasser.ppm** angegeben, was bedeutet, dass das Bild mit den eingezeichneten Sendern in der Datei **standorte_wasser.ppm** abgespeichert wird.

Mein Programm errechnet, dass man mit 80 Sendern ganz Europa mit dem drahtlosen Internet versorgen kann. Dabei ist es egal, ob man Sender nur auf dem Festland oder auch im Meer (bzw. außerhalb der EU) platzieren darf.

Zur visuellen Kontrolle, dass alle Landpixel der Europakarte von Sendegebietten bedeckt sind und um zu sehen wie die Sender platziert wurden, habe ich die Karten jeweils in den Dateien **standorte.ppm** und **standorte_wasser.ppm** als Bild abgespeichert.

In den folgenden Bildern bedeutet ein grüner Pixel ein Land-Pixel, der von einem Sendegebiet bedeckt ist. Wäre ein Land-Pixel nicht von einem Sendegebiet bedeckt, würden dieser rot gekennzeichnet sein. Blaue Pixel stehen für Wasserfläche bzw. Länder außerhalb der EU. Ist ein blauer Pixel von einem Sendegebietten bedeckt, so wurde dieser hellblau markiert. Die schwarzen Kreuze sind die Senderstandorte.

Übergibt man dem Programm falsche Parameter oder die Option **-h** erscheint folgende Hilfe:

```
$ ./aufgabe4 -h
Benutzung: ./aufgabe4 [Optionen]
Optionen:
-k <Dateiname>   Dateiname der Karte

-s <Dateiname>   Dateiname des Sendegebietes

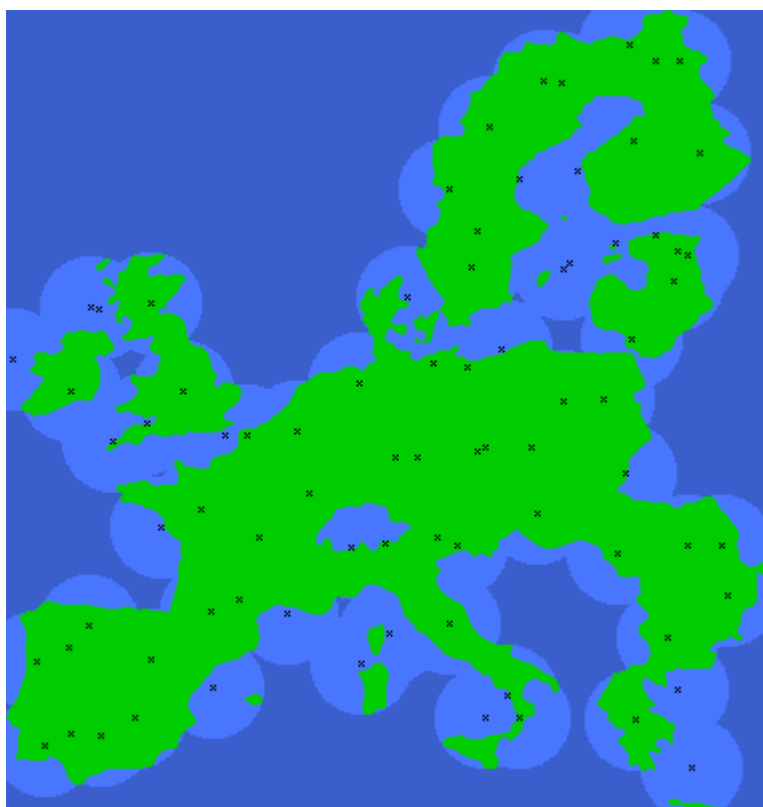
-o <Dateiname>   Dateiname der Ausgabedatei

-l oder -w       l: Sender nur auf Landflaeche platzieren
                  w: Sender duerfen auch auf Wasserflaeche
                    platziert werden

-h              Ausgeben dieser Hilfe
```



standorte.ppm



standorte_wasser.ppm