

**Dr. Jürg M. Stettbacher**

Margrit Rainer Strasse 12a  
CH-8050 Zürich

Telefon: +41 43 299 57 23  
Fax: +41 43 299 57 25  
E-Mail: dsp@stettbacher.ch

# Arithmetische Codierung

## Praktikum

Version 2.00  
2013-06-12

**Zusammenfassung:** In diesem Praktikum geht es darum, je einen arithmetischen Encoder und einen arithmetischen Decoder zu schreiben. Es soll also möglich sein, einen Datenblock aus einer Datei in eine reelle Zahl umzuwandeln und anschliessend daraus den ursprünglichen Block wieder herzustellen.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
<b>2</b>	<b>Dateien</b>	<b>2</b>
<b>3</b>	<b>Encoder Aufgabe</b>	<b>3</b>
<b>4</b>	<b>Decoder Aufgabe</b>	<b>3</b>

## 1 Einleitung

Das arithmetische Codieren ist ein Quellencodierungsverfahren. Es wird eine Symbolsequenz definierter Länge auf eine reelle Zahl im Intervall<sup>1</sup>  $[0, 1)$  abgebildet. Ein Encoder zerlegt den Symbolstrom aus der Quelle demnach zuerst in entsprechende Symbolsequenzen fixer Länge (wir nennen diese Sequenzen auch Blöcke) und codiert danach jeden Block einzeln. Auf der Leitung wird dann ein Strom von reellen Zahlen übertragen, je eine reelle Zahl für einen Block. Der Empfänger rekonstruiert aus jeder Zahl den Block und fügt die Blöcke wieder an einander, so dass der ursprüngliche Symbolstrom wieder entsteht.

In diesem Praktikum haben wir verschiedene Symbolblöcke je in Form einer Datei zur Verfügung. Es handelt sich um ASCII-Dateien (\*.txt) und jedes ASCII-Zeichen daraus stellt ein Symbol dar. Gesucht ist ein arithmetischer Encoder und Decoder, so dass der Inhalt je einer Datei in eine reelle Zahl codiert und anschliessend wieder decodiert werden kann.

## 2 Dateien

Die folgenden Dateien stehen Ihnen für das Bearbeiten des Praktikums zur Verfügung:

<sup>1</sup> Die Intervall-Schreibweise  $[0, 1)$  bedeutet, dass die Null im Intervall eingeschlossen ist, die Eins aber gerade nicht mehr dazu zählt.

- *arithmetic\_template.php* (Encoder und Decoder)
- *data\_10.txt* bis *data\_13.txt* (Testdaten)

### 3 Encoder Aufgabe

Schreiben Sie zuerst ein PHP-Skript *arithmetic.php*, das eine bestimmte Datei von Testdaten öffnet und deren Inhalt arithmetisch codiert<sup>2</sup>. Die resultierende reelle Zahl ist auf dem Bildschirm anzuzeigen. Der Name der ASCII-Datei soll auf der Kommandozeile übergeben werden, so dass der Aufruf folgendermassen aussieht.

```
> php arithmetic.php data.txt
```

Sie erhalten eine Vorlage *arithmetic\_template.php*, in welcher einige Funktionsblöcke bereits vorhanden sind. Sie müssen nur an den bezeichneten Stellen ergänzen. Für die Erzeugung des arithmetischen Codes gehen Sie so vor:

1. Interval auf  $[0, 1)$  festlegen.
2. Aufteilen des Intervalls auf die möglichen Symbole proportional zu ihren Wahrscheinlichkeiten.
3. Einschränken des Intervalls auf jenes Teilintervall das dem aktuellen Symbol entspricht.
4. Solange der Block weitere Symbole enthält, mit dem eingeschränkten Interval bei Punkt 2 weiter fahren. Wenn im Block keine Symbole mehr übrig, weiter bei Punkt 5.
5. Der Mittelwert aus oberer und unterer Intervallgrenze entspricht dem Codewort.

Diese Aufgabe ist zu erfüllen, und wenn nötig im Selbststudium fertig zu stellen.

### 4 Decoder Aufgabe

Ergänzen Sie Ihre PHP Skript *arithmetic.php*, so dass es die reelle Zahl, welche der Encoder erzeugt hat, wieder decodiert. Der decodierte Datenblock ist mit dem ursprünglichen zu vergleichen. Machen Sie diesen Test mit allen verfügbaren Testdateien.

<sup>2</sup> Beachten Sie, dass der gesamte Inhalt der Testdatei als ein einziger Datenblock aufzufassen ist. Er ist in nur eine reelle Zahl umzuwandeln.

In der Vorlage *arithmetic\_template.php* sind einige Funktionsblöcke bereits vorhanden. Sie müssen nur an den bezeichneten Stellen ergänzen.

Diese Aufgabe ist zu erfüllen, und wenn nötig im Selbststudium fertig zu stellen.