Gruppe Di-T14

Tutorübung zu
Grundlagen: Rechnernetze und Verteilte Systeme
(SS 16)

Michael Schwarz

Institut für Informatik
Technische Universität München

24.05.2016

Cyclic Redundancy Check (CRC) [3]

Im Gegensatz zu fehlerkorrigierenden Codes (Kanalcodes, Kapitel 1), handelt sich sich bei CRC um eine Familie fehlererkennender Codes. Mit ihrem Einsatz werden folgende Ziele verfolgt:

- Eine grosse Anzahl von Fehlern (Einbit-, Mehrbit-, Burstfehler) sollen erkannt werden.
- Die zugefügte Redundanz soll gering sein.
- Fehler sollen lediglich erkannt aber nicht korrigiert werden können.

Wie funktioniert CRC?

Angenommen wir haben ein Reduktionspolynom $r(x)$ des Grads n und eine Nachricht $m(x)$ der Länge $n - 1$ bit, die mittels CRC gesichert werden soll:

1. Hänge n Nullen an $m(x)$ an:
$$m'(x) = m(x) \cdot x^n.$$
2. Bestimme den Divisionsrest
$$c(x) = m'(x) \bmod r(x),$$
welcher der Checksumme entspricht.
3. Die zu sendende Nachricht besteht aus der Summe
$$s(x) = m'(x) + c(x).$$

Der Empfänger prüft die eingehende Nachricht $s'(x) = s(x) + e(x)$, welche möglicherweise einen Übertragungsfehler $e(x) \neq 0$ enthält:

1. Er bestimmt den Divionsrest
$$c'(x) = s'(x) \bmod r(x) = (s(x) + e(x)) \bmod r(x).$$
2. Ist $c'(x) = 0$, so ist mit hoher Wahrscheinlichkeit kein Übertragungsfehler aufgetreten. Ist $c'(x) \neq 0$, so ist sicher ein Fehler aufgetreten.

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

1. Koeffizienten bestimmen: $r(x) \hat{=} 1101$ und $m(x) \hat{=} 10100101$

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

1. Koeffizienten bestimmen: $r(x) \mathrel{\hat{=}} 1101$ und $m(x) \mathrel{\hat{=}} 10100101$
2. $\text{grad}(r(x)) = 3 \Rightarrow$ Daten mit x^3 multiplizieren. Dies entspricht dem „Anhängen“ von 3 Nullen: $m'(x) = m(x) \cdot x^3 \mathrel{\hat{=}} 10100101000$

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

1. Koeffizienten bestimmen: $r(x) \hat{=} 1101$ und $m(x) \hat{=} 10100101$
2. $\text{grad}(r(x)) = 3 \Rightarrow$ Daten mit x^3 multiplizieren. Dies entspricht dem „Anhängen" von 3 Nullen: $m'(x) = m(x) \cdot x^3 \hat{=} 10100101000$
3. Polynomdivision $m'(x)/r(x)$ ausführen und den Rest (Checksumme) $c(x)$ bestimmen.

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$$\overbrace{1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0}^{m'(x)} : \overbrace{1\ 1\ 0\ 1}^{r(x)} =$$

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$$\overbrace{1\;0\;1\;0\;0\;1\;0\;1\;0\;0\;0}^{m'(x)} \;:\; \overbrace{1\;1\;0\;1}^{r(x)} \;=\; 1$$

1	0	1	0	0	1	0	1	0	0	0
1	1	0	1							
0	1	1	1							

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$$
\begin{array}{ccccccccccc}
\multicolumn{11}{c}{m'(x)} \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & & & & & & & \\
\hline
0 & 1 & 1 & 1 & 0 & & & & & & \\
 & 1 & 1 & 0 & 1 & & & & & & \\
\cline{2-5}
 & 0 & 0 & 1 & 1 & & & & & &
\end{array}
\; : \;
\begin{array}{cccc}
\multicolumn{4}{c}{r(x)} \\
1 & 1 & 0 & 1
\end{array}
\; = \; 1 \;\; 1
$$

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$m'(x)$												$r(x)$		
1	0	1	0	0	1	0	1	0	0	0	:	1 1 0 1	=	1 1 0
1	1	0	1											
0	1	1	1	0										
	1	1	0	1										
	0	0	1	1	1									

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$$\overbrace{1\,0\,1\,0\,0\,1\,0\,1\,0\,0\,0}^{m'(x)} : \overbrace{1\,1\,0\,1}^{r(x)} = 1\,1\,0\,1$$

1	0	1	0	0	1	0	1	0	0	0
1	1	0	1							
0	1	1	1	0						
	1	1	0	1						
	0	0	1	1	1	0				
		1	1	0	1					
		0	0	1	1					

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$$
\begin{array}{ccccccccccc}
\multicolumn{11}{c}{\overbrace{\hspace{15em}}^{m'(x)}} \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & & & & & & & \\
\hline
0 & 1 & 1 & 1 & 0 & & & & & & \\
 & 1 & 1 & 0 & 1 & & & & & & \\
\cline{2-5}
 & 0 & 0 & 1 & 1 & 1 & 0 & & & & \\
 & & & 1 & 1 & 0 & 1 & & & & \\
\cline{4-7}
 & & & 0 & 0 & 1 & 1 & 1 & & &
\end{array}
\quad : \quad
\overbrace{1\ 1\ 0\ 1}^{r(x)}
\quad = \quad 1\ 1\ 0\ 1\ 0
$$

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$m'(x)$: $r(x)$

$\underbrace{1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0}_{m'(x)} : \underbrace{1\ 1\ 0\ 1}_{r(x)} = 1\ 1\ 0\ 1\ 0\ 1$

1	0	1	0	0	1	0	1	0	0	0
1	1	0	1							
0	1	1	1	0						
	1	1	0	1						
	0	0	1	1	1	0				
			1	1	0	1				
			0	0	1	1	1	0		
					1	1	0	1		
					0	0	1	1		

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$m'(x)$												$r(x)$		
1	0	1	0	0	1	0	1	0	0	0	:	1 1 0 1	=	1 1 0 1 0 1 0
1	1	0	1											
0	1	1	1	0										
	1	1	0	1										
	0	0	1	1	1	0								
			1	1	0	1								
			0	0	1	1	1	0						
					1	1	0	1						
					0	0	1	1	0					

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

$$
\begin{array}{ccccccccccccccc}
\multicolumn{11}{c}{m'(x)} & & \multicolumn{1}{c}{r(x)} & & \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & : & 1\;1\;0\;1 & = & 1\;1\;0\;1\;0\;1\;0\;1 \\
1 & 1 & 0 & 1 & & & & & & & & & & & \\
\cline{1-4}
0 & 1 & 1 & 1 & 0 & & & & & & & & & & \\
 & 1 & 1 & 0 & 1 & & & & & & & & & & \\
\cline{2-5}
 & 0 & 0 & 1 & 1 & 1 & 0 & & & & & & & & \\
 & & & 1 & 1 & 0 & 1 & & & & & & & & \\
\cline{4-7}
 & & & 0 & 0 & 1 & 1 & 1 & 0 & & & & & & \\
 & & & & & 1 & 1 & 0 & 1 & & & & & & \\
\cline{6-9}
 & & & & & 0 & 0 & 1 & 1 & 0 & 0 & & & & \\
 & & & & & & & 1 & 1 & 0 & 1 & & & & \\
\cline{8-11}
 & & & & & & & 0 & 0 & 0 & 1 & = c(x) & & &
\end{array}
$$

Adressierung und Fehlererkennung

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

1. Koeffizienten bestimmen: $r(x) \mathrel{\hat{=}} 1101$ und $m(x) \mathrel{\hat{=}} 10100101$
2. $\mathrm{grad}(r(x)) = 3 \Rightarrow$ Daten mit x^3 multiplizieren. Dies entspricht dem „Anhängen" von 3 Nullen: $m'(x) = m(x) \cdot x^3 \mathrel{\hat{=}} 10100101000$
3. Polynomdivision $m'(x)/r(x)$ ausführen und den Rest (Checksumme) $c(x)$ bestimmen.

Beispiel: Reduktionspolynom $r(x) = x^3 + x^2 + 1$, Daten $m(x) = x^7 + x^5 + x^2 + 1$

1. Koeffizienten bestimmen: $r(x) \mathrel{\hat{=}} 1101$ und $m(x) \mathrel{\hat{=}} 10100101$
2. $\mathrm{grad}(r(x)) = 3 \Rightarrow$ Daten mit x^3 multiplizieren. Dies entspricht dem „Anhängen" von 3 Nullen: $m'(x) = m(x) \cdot x^3 \mathrel{\hat{=}} 10100101000$
3. Polynomdivision $m'(x)/r(x)$ ausführen und den Rest (Checksumme) $c(x)$ bestimmen.
4. Die zu sendende Nachricht ist $s(x) = m'(x) + c(x)$. Die Addition reduziert sich auf ein XOR, da wir auf GF(2) arbeiten.

$$
\begin{array}{c|ccccccccccc|l}
 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & = m'(x) \\
\oplus & & & & & & & & & 0 & 0 & 1 & = c(x) \\
\hline
 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & = s(x)
\end{array}
$$

Verfahren

- ALOHA: Letzte Übung
- CSMA
  - “Nur” CSMA
  - CSMA/CA
  - CSMA/CD

CSMA, CSMA/CD, CSMA/CA

Carrier Sense Multiple Access (CSMA)

Eine einfache Verbesserung von Slotted ALOHA: „Listen Before Talk“

- Höre das Medium ab
- Beginne erst dann zu senden, wenn das Medium frei ist

Non-persistent CSMA:

1. Wenn Medium frei, übertrage im nächstmöglichen Intervall
2. Wenn belegt, warte eine feste Zeitspanne (länger als ein Zeitintervall, typischerweise mit zufälliger Dauer), dann 1.

1-persistent CSMA:

1. Wenn Medium frei, übertrage im nächstmöglichen Intervall
2. Wenn Medium belegt, warte bis frei und übertrage im nächstmöglichen Intervall

p-persistent CSMA:

1. Wenn Medium frei, übertrage mit Wahrscheinlichkeit p oder verzögere mit Wahrscheinlichkeit $1 - p$ um eine Slotzeit
2. Wenn Medium belegt, warte bis frei, dann 1.
3. Warte eine Slotzeit, dann 1.

CSMA/CD (Collision Detection)

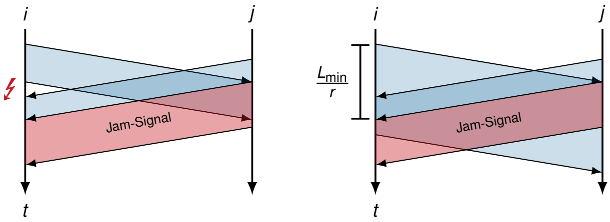
- Erkenne Kollisionen und wiederhole die Übertragung, wenn eine Kollision erkannt wird
- Verzichte auf das Senden von Bestätigungen
- Wird keine Kollision erkannt, gilt die Übertragung als erfolgreich

Problem: Der Sender muss die Kollision erkennen, während er noch überträgt

Voraussetzung für CSMA/CD [2]

Angenommen zwei Stationen *i* und *j* kommunizieren über eine Distanz *d* mittels CMSA/CD. Damit Kollisionen erkannt werden können, müssen Nachrichten folgende Mindestlänge $L_{\min}$ aufweisen:

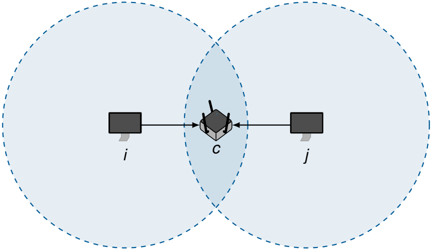
$$L_{\min} = \frac{2d}{\nu c} r$$



CSMA/CA (Collision Avoidance)

In Funknetzwerken funktioniert CSMA/CD nicht, da der Sender einer Nachricht eine Kollision auch bei ausreichender Nachrichtenlänge nicht immer detektieren kann.

„Hidden Station“:



- Knoten *i* und *j* senden gleichzeitig
- Knoten *c* erkennt die Kollision
- Weder *i* noch *j* bemerken die Kollision

CSMA/CA basiert auf *p*-persistentem CSMA, d. h.

1. Wenn Medium frei, übertrage mit Wahrscheinlichkeit p oder verzögere mit Wahrscheinlichkeit $1 - p$ um eine Slotzeit
2. Wenn Medium belegt, warte bis frei, dann (1)
3. Warte eine Slotzeit, dann (1)

Fallbeispiel: IEEE 802.11 DCF (Distributed Coordination Function)

- Festes Zeitintervall zwischen Rahmen: DIFS (DCF Interframe Spacing).
- Wenn Medium mind. für DIFS idle ist, dann wähle unabhängig und gleichverteilt eine Anzahl von Backoff-Slots aus dem Intervall $\{0, 1, 2, \ldots, \min\{2^{c+n} - 1, 255\}\}$.
- c ist abhängig vom PHY (z.B. $c = 4$), n ist der Retry Counter des Binary Exponential Backoffs.

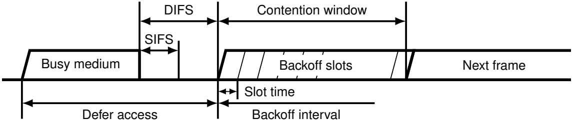


Abbildung: IEEE 802.11 DCF

- Medienzugriff hat durch festes c stets ein Contention Window.
- Ein Rahmen gilt in IEEE 802.11 als erfolgreich übertragen, wenn
  - im Fall von Unicasts der Empfänger eine Bestätigung schickt (Link-Layer Acknowledgements) oder
  - im Fall von Broadcasts die Übertragung eines Frames störungsfrei abgeschlossen wird.
- Da i.d.R. nicht gleichzeitig gesendet und das Medium geprüft werden kann (anders bei Ethernet), ist die zweite Bedingung praktisch bereits erfüllt, wenn ein Knoten zu senden beginnt.