Grundlagen: Rechnernetze und verteilte Systeme

Zweite Woche: 23. April 2018

Rahmenfehler, Kanalkodierung, Fourierreihe

Leo Glavinić

netze@eo.gl

eo.gl/netze

Organisatorisches

Entfall von Do-1200-B
Ersetzung durch Fr-1000-B (Raum 03.09.014)
...diese Woche Vertretung am Freitag

doch keine gedruckten Cheatsheets während des Semesters → grnvs.net/altklausuren

Taschenrechner erlaubt!

Heutige Themen

1. Rahmenfehlerwahrscheinlichkeiten

2. Kanalkodierung

3. Fourierreihe

1. Rahmenfehlerwahrscheinlichkeit

Wahrscheinlichkeit für einen Bitfehler über...

Funk: $p=10^{-4}$

Kupfer: $p=10^{-8}$

voneinander unabhängige, gleich verteilte Fehler, Rauschen mit konstanter Leistung, keine Interferenz

Rahmenlänge: $L = 1500$B

1. Rahmenfehlerwahrscheinlichkeit

a. Wahrscheinlichkeiten, dass ein Rahmen jew. fehlerfrei übertragen wird

Pr[„kein Bitfehler im Rahmen“]=$(1-p)^{8L}$

→ 99,99% über Kabel, 30,12% über Funk

1. Rahmenfehlerwahrscheinlichkeit

Versand von Bestätigungen bei korrekt übertragenen Rahmen wegen hoher Fehlerwahrscheinlichkeit

keine Bestätigungen bei Ethernet/Kabel!

1. Rahmenfehlerwahrscheinlichkeit

b. maximale Anzahl an Wiederholungen für garantiert korrekte Übertragung eines Rahmens

nicht existent, denn die einzelnen Übertragungsversuche sind statistisch unabhängig; verbleibendes Restrisiko

1. Rahmenfehlerwahrscheinlichkeit

c. Wahrscheinlichkeit, dass Rahmen genau *k* mal übertragen werden muss

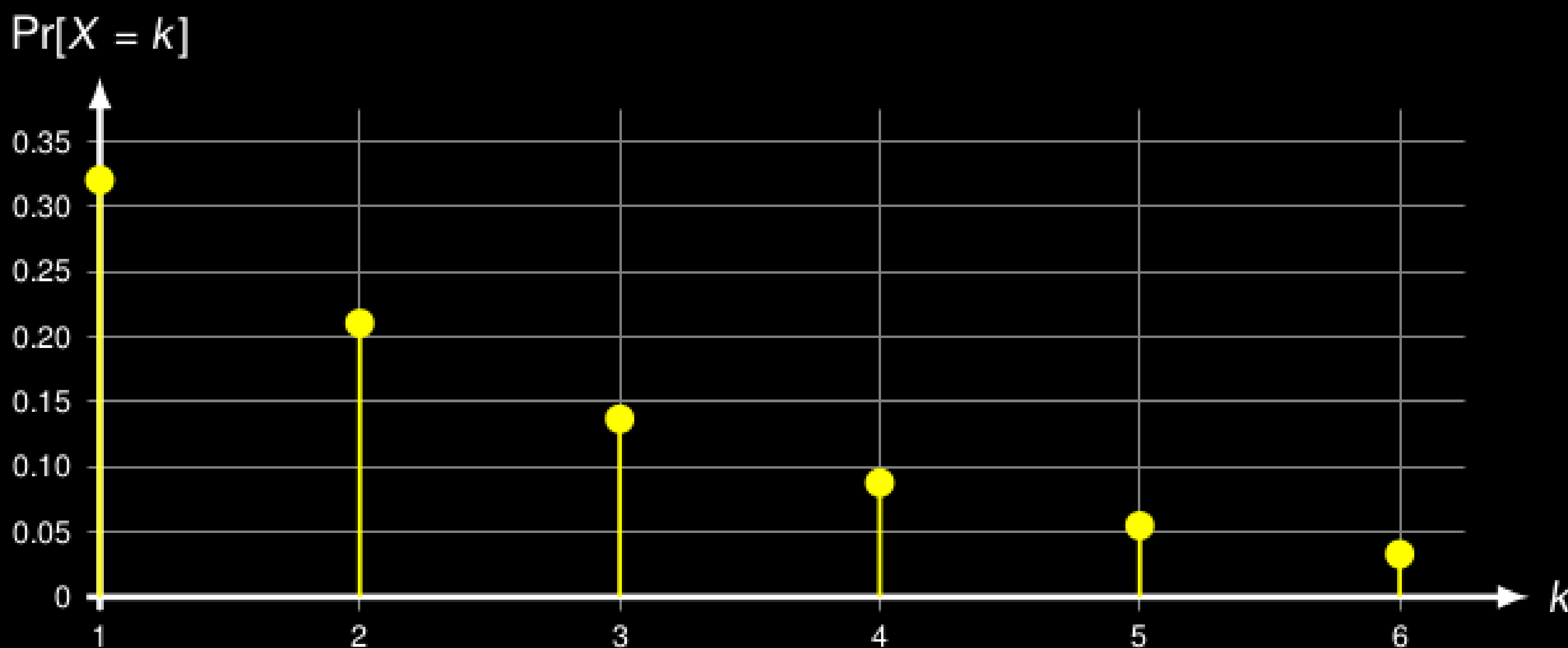
Zufallsvariable *X*: Zahl notwendiger Übertragungen
p_R=Pr[„kein Bitfehler im Rahmen"]

Pr[„Übertragung (*k*-1) mal erfolglos"]=$(1-p_R)^{k-1}$
dazu die Wahrscheinlichkeit, dass es dann klappt...

→ $Pr[X=k]=(1-p_R)^{k-1} \cdot p_R$

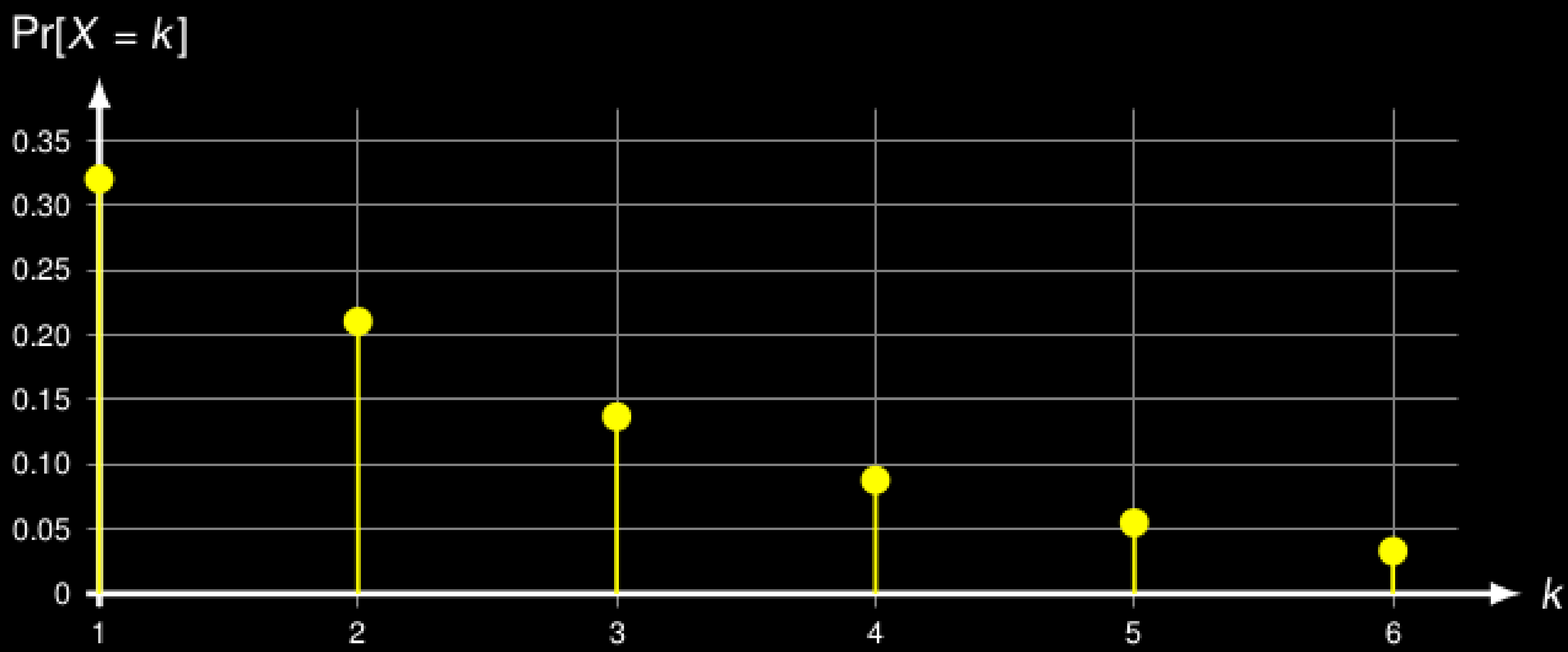
1. Rahmenfehlerwahrscheinlichkeit

d. Graph der diskreten Verteilungsfunktion (Funk)



1. Rahmenfehlerwahrscheinlichkeit

d. Graph der diskreten Verteilungsfunktion (Funk)



1. Rahmenfehlerwahrscheinlichkeit

e. Abbruch nach drei erfolglosen Versuchen: Wahrscheinlichkeit für Nichtübertragung eines Rahmens

$$\Pr[X > 3] = 1 - \Pr[X \leq 3] = 1 - \sum_{k=1}^{3} \Pr[X = k] \approx 34\,\%$$

1. Rahmenfehlerwahrscheinlichkeit

Relevanz für uns: hoher Aufwand, geringe Effizienz

Lösung: Blockcode

2. Kanalkodierung

Blockcode, Schicht 1

Beispielencoder, der jeweils Blöcke der Länge 247 bit in Kanalwörter von 255 bit umwandelt (aber nicht unbedingt nur 8 bit anhängt!)

evtl. mögliche Korrektur eines beliebigen Bitfehlers durch den Empfänger

2. Kanalkodierung

a+b. Coderate (Cheatsheet!)

$R = k/n = 247/255 \approx 0{,}97$

Coderate: Verhältnis zwischen Größe der Nutzdaten in einem Block und gesamter Blockgröße

geringere Coderate ~ höhere Redundanz

im Beispiel: jedes 255-bit-Kanalwort trägt 8 bit an Redundanz und 247 bit an Information

2. Kanalkodierung

c. Zahl der übertragenen Kanalwörter

$$N = \left\lceil \frac{1500 \cdot 8}{247} \right\rceil = 49$$

wichtig: Nutzdatengröße im Nenner, nicht die gesamte Länge inkl. Redundanz!

2. Kanalkodierung

d. prozentualer Overhead durch Padding im letzten Kanalwort

wie viel Überhang bleibt?

$49 \cdot 247$ bit $= 12103$ bit → 103 bit Padding am Ende

$\gamma = 103/(1500 \cdot 8 + 103) \approx 0{,}85\%$

2. Kanalkodierung

e. Wahrscheinlichkeit für fehlerhafte Dekodierung eines einzelnen Kanalwortes

Wahrscheinlichkeit, dass in Kanalwort (Länge 255 bit) mindestens zwei Bitfehler auftreten (Pr[*X*>1])

$$
\begin{aligned}
p_{e,Codewort} &= \Pr[X \geq 2] = 1 - \Pr[X \leq 1] = 1 - \sum_{i=0}^{1} \binom{n}{i} \cdot p_e^i \cdot (1-p_e)^{n-i} \\
&= 1 - \left(1 \cdot p_e^0 \cdot (1-p_e)^{255} + 255 \cdot p_e^1 \cdot (1-p_e)^{254}\right) \\
&\approx 1 - \left(0,9748 + 255 \cdot 10^{-4} \cdot 0,9748\right) \\
&\approx 3.18 \cdot 10^{-4}
\end{aligned}
$$

2. Kanalkodierung

f. Wahrscheinlichkeit für korrekte Übertragung des gesamten Rahmens

Pr[„kein Fehler im Rahmen“]=$(1-p_{e,\ Codewort})^N \approx 98{,}5\%$

2. Kanalkodierung

g. Beurteilung von drahtloser Übertragung mit und ohne Kanalkodierung

besser :)

deutlich erhöhte Wahrscheinlichkeit für korrekte Rahmenübertragung (ca. 30% → 98,5%)

2. Kanalkodierung

Berücksichtigung des Overheads durch Padding und Kanalkodierung → Effizienzanalyse

$$\eta = R \cdot \Pr[\text{„kein Fehler im Rahmen"}] \cdot (1-\gamma)$$
$$= 0{,}97 \cdot 0{,}985 \cdot (1 - 103/(1500 \cdot 8 + 103)) \approx 0{,}95$$

Mit Methode aus **1.e.** müsste ein Rahmen elf bis zwölf Mal wiederholt werden, um ähnliche Erfolgswahrscheinlichkeit zu erreichen!

2. Kanalkodierung

Relevanz für uns: mögliche starke Verbesserung der Datenrate durch Kanalkodierung, Unerlässlichkeit bei unzuverlässigen Verbindungen (z.B. Funk)

Unterteilung des Rahmens in Blöcke

bedeutende Reduktion der Rahmenfehlerwahrscheinlichkeit durch Korrektur eines Bitfehlers pro Codewort

3. Fourierreihe

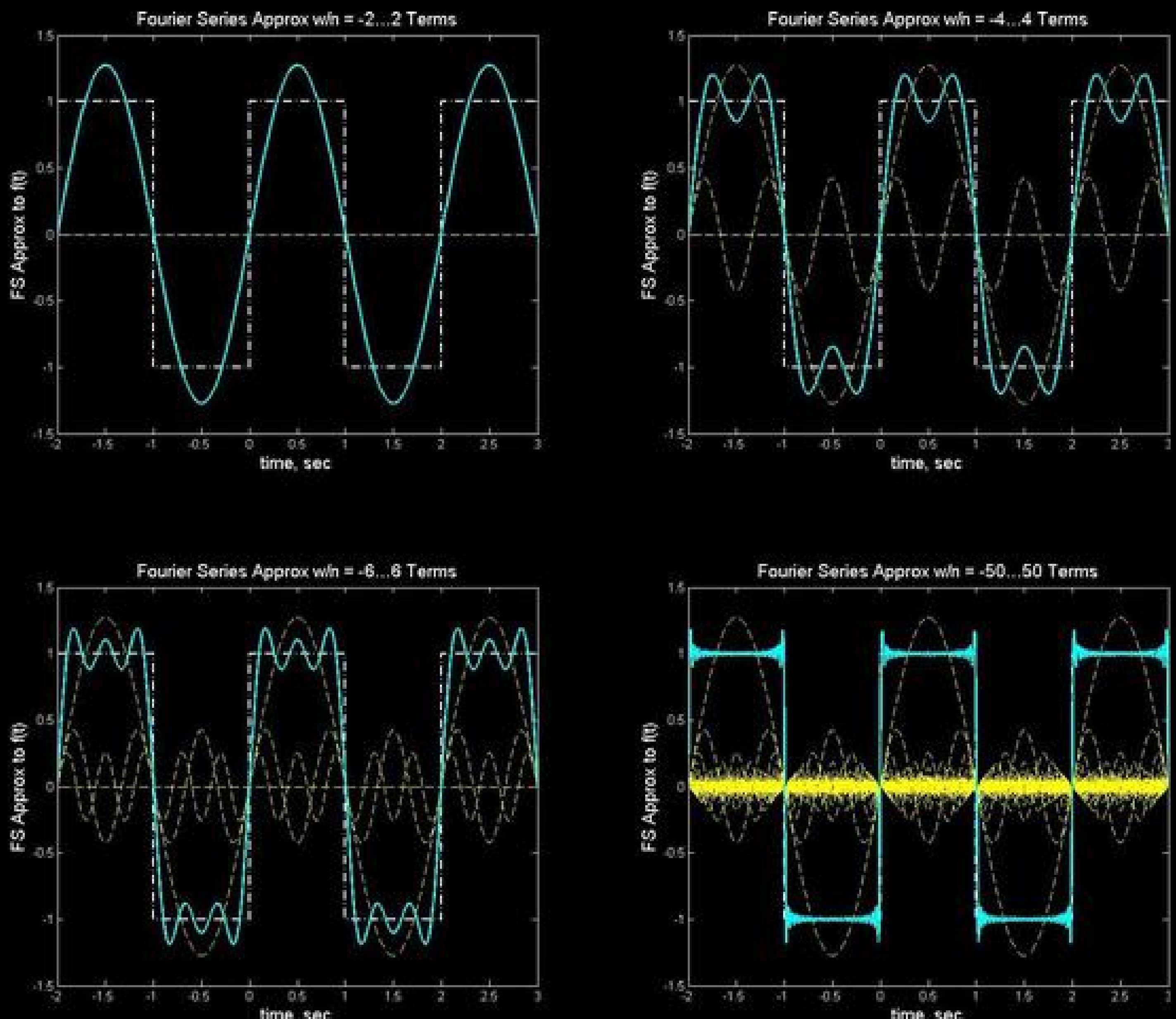
wozu das Ganze?

Übertragung periodischer (sich wiederholender) Signale

Annäherung des Signals durch Summe beliebig vieler trigonometrischer Funktionen

Fourierreihe bei periodischen Signalen;
-transformation bei nichtperiodischen Signalen (nächste Woche)

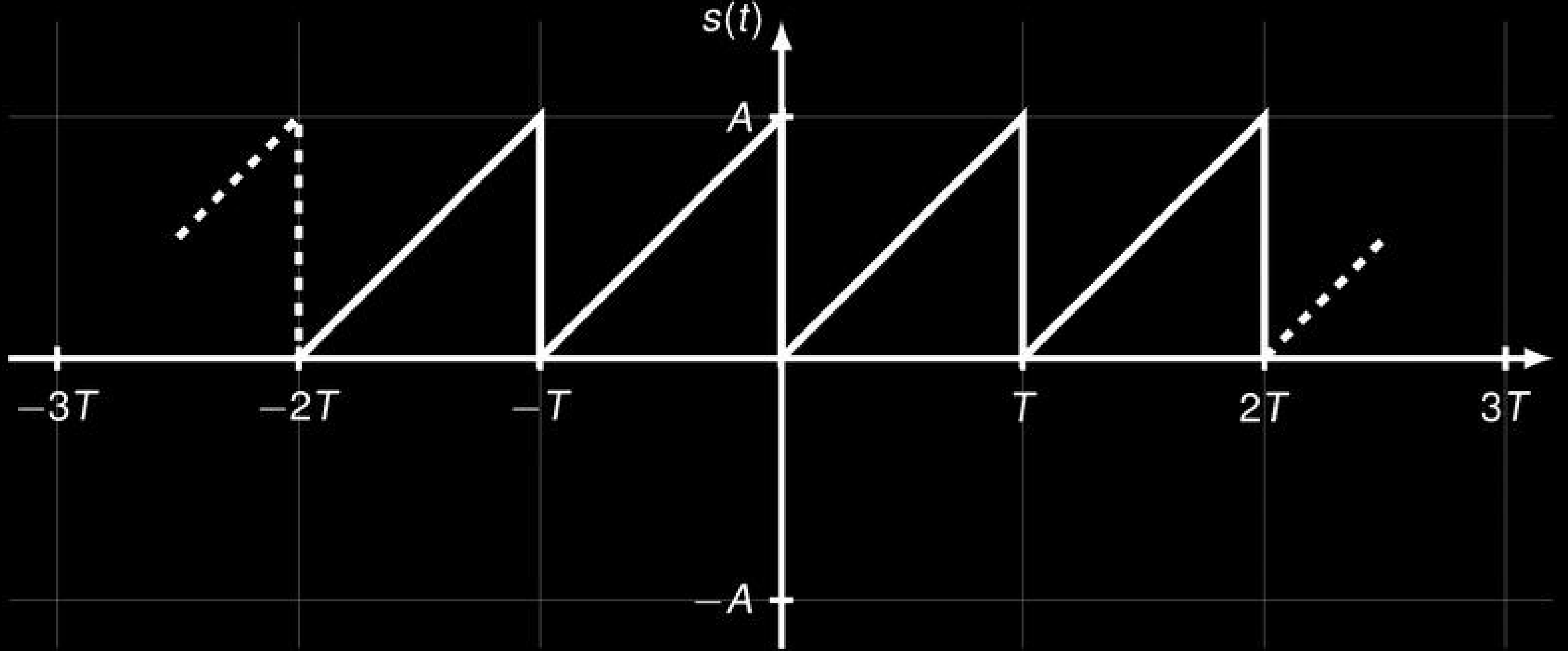
3. Fourierreihe



http://www.ee.nmt.edu/~wedeward/EE212/SP15/FourierSeriesExample6.html

3. Fourierreihe

gegebenes *T*-periodisches Signal *s(t)*



(viel komplizierter wird es in der Klausur nicht)

3. Fourierreihe

a. analytischer Ausdruck für $s(t)$ im Intervall $[0, T]$

(Tipp: Geradengleichung y=mx+t aus der Schule)

$s(t) = (t/T) \cdot A$

nur für $t \in [0, T]$!

3. Fourierreihe

Fourierreihe: (Cheatsheet!)

$$s(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos(k\omega t) + b_k \sin(k\omega t))$$

$$a_k = \frac{2}{T} \int_0^T s(t) \cdot \cos(k\omega t)\ dt$$

$$b_k = \frac{2}{T} \int_0^T s(t) \cdot \sin(k\omega t)\ dt$$

3. Fourierreihe

Fourierreihe: (Cheatsheet!)

$$s(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos(k\omega t) + b_k \sin(k\omega t))$$

$$a_k = \frac{2}{T} \int_0^T s(t) \cdot \cos(k\omega t)\, dt$$

$$b_k = \frac{2}{T} \int_0^T s(t) \cdot \sin(k\omega t)\, dt$$

3. Fourierreihe

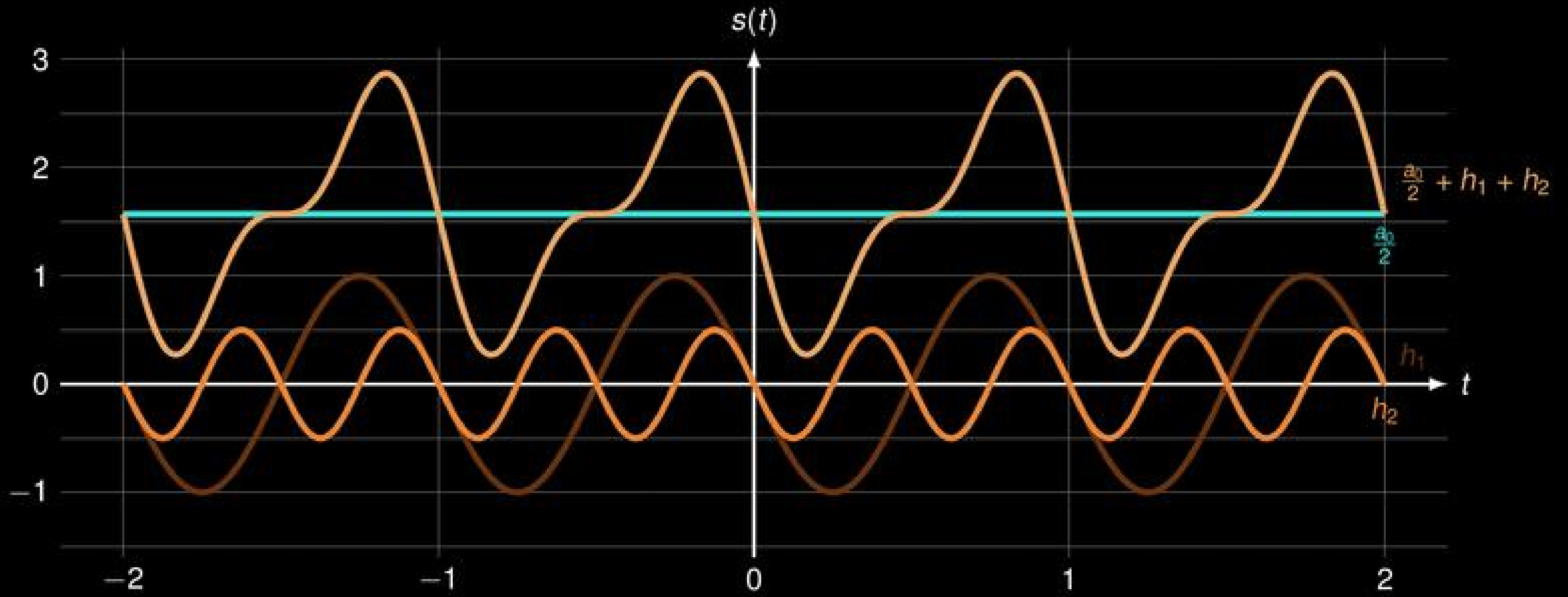
b. für den Gleichanteil verantwortlicher Koeffizient

$$s(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos(k\omega t) + b_k \sin(k\omega t))$$

3. Fourierreihe

Rest der Aufgabe: wird live vorgerechnet ;)

3. Fourierreihe



4. Quantisierung und Rauschen

„Hausaufgabe“

Musterlösung anschauen, sobald online!