"Vergleich von linearen Prädiktionsstrategien (NLMSvariants)"

Das Voraussagen von Signalwerten ist ein Schlüsselelement in modernen Kompressionssysteme. Aber auch in anderen Anwendungsgebieten ist die Prädiktion von Ereignissen erforderlich.

Aufgabe ist es, ein Matlab/SciLab/Octave/C-Programm zu schreiben, das Varianten eines bestimmten Prädiktionsverfahrens miteinander vergleicht.

Adaptive Prädiktionsfilter passen sich der Charakteristik der Signale an und gehören somit in den Bereich des **maschinellen Lernen**s. So genannte Least-Mean-Square-Filter (LMS) sind adaptive Filter mit einer vergleichsweisen geringen Komplexität. Im Wesentlichen werden M vorangegangene Signalwerte gewichtet überlagert, um einen Schätzwert für die aktuelle Position zu generieren

$$\hat{x}[n] = \sum_{j=1}^{M} a_j[n] \cdot x[n-j]$$

Der Prädiktionsfehler lautet: $\mathrm{e}[n] = x[n] - \hat{x}[n]$. Um die Prädiktionsfehlerenergie zu minimieren müssen die Filterkoeffizienten nachgeführt (aktualisiert) werden

$$a_j[n+1] = a_j[n] + \mu \cdot e[n] \cdot \frac{x[n-j]}{||\mathbf{x}[n]||^2} \qquad \text{mit} \qquad ||\mathbf{x}[n]||^2 = \sum_{j=1}^{M} (x[n-j])^2$$

während $0 < \mu \leq 1$ die Lernrate ist.

Leider funktioniert das nur gut, wenn der Mittelwert von $x[n]$ gleich Null ist. Für Bilder und auch teilweise für Sprachsignale ist das nicht gegeben. Als Lösung kommen drei Varianten in Frage, welche die obigen Formeln leicht abwandeln:

1. lokalen Mittelwert abziehen

$$\hat{x}[n] = \bar{x}[n] + \sum_{j=1}^{M} a_j[n] \cdot (x[n-j] - \bar{x}[n]) \quad \text{mit} \quad \bar{x}[n] = \frac{1}{M}\sum_{j=1}^{M} x[n-j]$$

und

$$a_j[n+1] = a_j[n] + \mu \cdot e[n] \cdot \frac{x[n-j] - \bar{x}[n]}{||\mathbf{x}[n]||^2} \qquad \text{mit} \qquad ||\mathbf{x}[n]||^2 = \sum_{j=1}^{M} (x[n-j] - \bar{x}[n])^2$$

2. Bezug auf direkten Vorgänger nehmen

$$\hat{x}[n] = x[n-1] + \sum_{j=1}^{M} a_j[n] \cdot (x[n-1] - x[n-j-1])$$

und

$$a_j[n+1] = a_j[n] + \mu \cdot e[n] \cdot \frac{x[n-1] - x[n-j-1]}{||\mathbf{x}[n]||^2} \quad \text{mit } ||\mathbf{x}[n]||^2 = \sum_{j=1}^{M} (x[n-1] - x[n-j-1])^2$$

3. differentiellen Bezug auf Vorgänger nehmen

$$\hat{x}[n] = x[n-1] + \sum_{j=1}^{M} a_j[n] \cdot (x[n-j] - x[n-j-1])$$

und

$$a_j[n+1] = a_j[n] + \mu \cdot e[n] \cdot \frac{x[n-j] - x[n-j-1]}{||\mathbf{x}[n]||^2} \quad \text{mit } ||\mathbf{x}[n]||^2 = \sum_{j=1}^{M} (x[n-j] - x[n-j-1])^2$$

Das originale Verfahren und die drei Varianten sind zu implementieren und mit verschiedenen Testsignalen (synthetisierte und reale, N>= 500) und verschiedene M zu prüfen. Als Gütekriterium ist die mittlere Energie des Schätzfehlers $E = {}^{1}/_{N} \sum_{n=1}^{N} (e[n])^2$ heranzuziehen. Bei selbstgenerierten Signalen könnte auch die Konvergenz der Filterkoeffizienten zu den richtigen Werten untersucht werden.

Weitere Unterstützung wird bei Bedarf gegeben. Alle Untersuchungen sind schriftlich zu dokumentieren. Neben der schriftlichen Arbeit sind alle Quellen (Programmcode, Texte, Testsignale) und Tools abzugeben, damit eine Reproduktion der Ergebnisse möglich ist.

Teilaufgaben:

- Koordination
- Recherche
- Programmierung
- Dokumentation (Grundlagen, Methode, Änderungen am Quellcode, Kompressionsergebnisse)

Max. 5 Personen,
Max. 2.5 Zusatzpunkte für Klausur