Lösungvariante zur Demo-Klausur (2. Fassung)

René Pönitz
rene (@) renephoenix.de

15. Februar 2005

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Aufgabe 1: Was ist Sensitivitätsanalyse

- Ausgangspunkt: Simulationsmodell steht - und Simulation wurde durchgeführt
- übliche Frage: was passiert, wenn ich dies oder jenes verändere?
- Sensitivitätsanalyse = Analyse, was passiert, wenn man die Einflußfaktoren ändern
- in der Regel werden nur kleine Unterschiede der Einflußfaktoren genommen -> d.h. die Reaktionen sollten auch nur relativ klein dazu sein (also kein Sprung gegen +/-∞)
- je größer die Reaktion ist, desto sensibler ist ein Einflußfaktor. D.h. dieser sollte bei einer Realisierung größere Bedeutung bekommen.
- ergibt damit ein gewisses Maß der Robustheit und Stabilität des Computermodells

2 Aufgabe 2: Pseoduzufallszahlen

Warum Gleichverteilung? Wie prüfbar?

- Zufallszahlen im Computersystem sind aufgrund der verwendeten Algorithmen **nicht zufällig**. Der PC kann keinen Zufall herstellen (selbst Uhrzeit ist ermittelbar).
- Damit dennoch nutzbar, muß ein gleichwahrscheinlicher Zufall generiert werden, d.h. alle Zahlen gleich groß - keine Häufung bestimmter Werte (überhöhte Dichtefunktion)
- Bei Gleichverteilung sehr geringer Aufwand in jede andere Verteilung umzuwandeln
- Nachweisbar über Chi-Quadrat-Anpassungstest.
- Man bildet (möglichst !) gleichgroße Teilbereiche (ca. Anzahl der Zahlen / 10..100)
- in jedem Intervall sollte nun gleiche Anzahl Elemente sein
- $z = \sum_{k=1}^{m} (h_k - e_k)^2 / e_k$
- man nimmt Irrtumswahrscheinlichekti $\alpha = 0,05...0,01$
- $z < \chi^2_{\alpha,m-1}$
- wenn drin enthalten, dann ist Verteilung unter Annahme der Irrtumswahrscheinlichkeit gleichverteilt
- Nachweisbar auch über Runs-Test (Die Verzeichen Differenz der benachbarten Zufallszahlen und in Bereiche eingeteilt), Autokorrelation (Entdeckung von Abhängigkeiten zwischen Zufallszahlen), Kolmogorov-Anpassungstest (max. Abweichung zwischen empirischer und theoratischer Verteilungsfunktion)

3 Aufgabe 3: Inversion der Verteilungsfunktion am Beispiel

- Verteilungsfunktion ist Integral der relative Dichtefunktion
- Verteilungsfunktion ist demnach monoton steigend.
- Verteilungsfunktion hat Wertebereich [0,1].
- Umkehrfunktion F-1(x) hat nur Definitionsbereich [0,1]
- wir "schießen"nun auf diese Funktion - und Kugel sammelt sich da, wo sie den Graphen berührt. Die eingehenden Werte müssen gleichverteilt sein. Das Ergebnis wird unsere gewünschte Verteilung sein.

- da wo die meisten Kugeln liegen, ist Anstieg der Verteilungsfunktion am größen - folglich Dichtefunktion groß.
- als Dichtefunktion kann nun jede beliebige genommen werden - damit entstehen unterschiedliche Verteilungen (z.B. Gaußche Glockenkurve)

Beispiel:

- Exponentialverteilung: $f(y) = \lambda e^{-\lambda x} fuer\ x > 0,\ sonst\ 0$
- Verteilungsfunktion: $F(x) = 1 - e^{-\lambda x}$
- Umkehrfunktion F-1(x): $F^{-1}(x) = \frac{ln(1-y)}{-\lambda}$
- Exponentialverteilung Zufallszahlen durch $F^{-1}(x) = \frac{ln(y)}{-\lambda}$

4 Aufgabe 4: Säulenproblem

```
float s1;
float s2;
while (computer != absturz)
{
s1 = rnd () * 10 - 5;
s2 = rnd () * 10 - 5;

if( abs (s1 - s2) > 6)
nacharbeit();
else
lager();
}
```

5 Aufgabe 5: Rundungsfehler

(wie schätzen, wie reduzieren)

- PC hat nur endliche Genauigkeit (Gleitkommawerte)
- der mathematische Wert hat - wenn auch nur kleine - Abweichungen zum Wert, der abgespeichert werden kann
- kontinuierliche Simulation ist iterativ -> baut auf vorherige Werte auf
- folglich: kleiner Fehler wird multipliziert
- Schätzung möglich über Vergleich zwischen Schrittweite im Modell und Schrittweite auf dem Speicher
- Reduzierung z.B. durch Erhöhung der Genauigkeit im Rechner (double statt float), Erhöhung der Schrittweise (Folge: Approximationsfehler steigt)

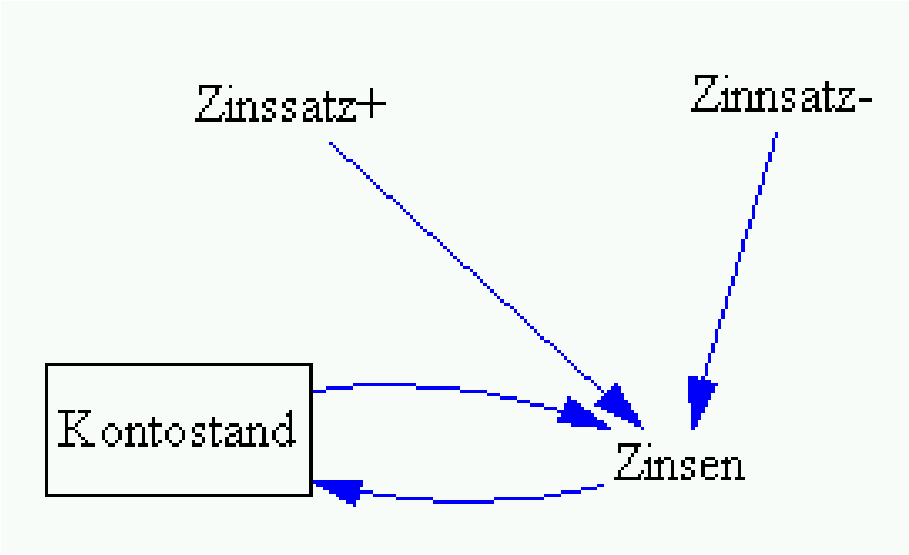

Abbildung 1: Vensim-Modell

6 Aufgabe 6: Vensim-Modell

```
Zinsen =
IF THEN ELSE( Kontostand > 0 ,
Kontostand * "Zinssatz+",
Kontostand * "Zinnsatz-" )

Kontostand = INTEG ( Zinsen, -100)
```

Zu beachten sei noch die Einstellung der richtigen Intervalle (in dem Fall Jahre). Die verwendeten Variablen:

- Normale Variable nimmt einen zugewiesenen Wert an - und überschreibt ihn
- Levelvariablen addieren den zugewiesen Wert. Sie stellen das Integral dar.

7 Aufgabe 7: SLX - Zeitverzögerung eines Objektes

- `advance 10` (Wartet die Zeit ab, bestimmte Zeitverzögerung)
- `wait until (bedingung )`(wartet solange, bis die Bedingung war wird, dazu gibt es control-Variable, die den Prüfprozeß anstößt, unbestimmte Zeitverzögerung)
- `wait` (deaktiviert Prozeß)
- `reactivate pcustomer` (Reaktivierung nur durch anderen Prozeß)

8 Aufgabe 8: Einsatz von Zufallszahlen bei Optimierungsverfahren auf Evolutionsbasis

- Evolution beruht auf den Schritten Reproduktion, Rekombination, Mutation und Selektion.
- bei Mutation
  - bei Rekombination bzw. während der Lebensdauer erfolgt zufällige Veränderung eines Parameters. Also sowohl welcher Parameter als auch der Wert des Parameters

  - Art der Veränderung

- bei der Selektion
  - zufällige Auswahl der Nachkommen
  - Auswahl meist in Zusammenhang mit Fitneß -> gewichteter Zufall (schlechte haben noch eine Chance zum Überleben, aber keine hohe)

- Optimierung und Simulation / Zukunft
  - wichtig
  - Optimierung ist auch ein Optimierungsprozeß
  - gibt noch Performanceproblem bei mehrfachen Simulationen