

2022

Praktikum Betriebssicherheit

Aufgabenblatt 3

Simulation eines Systems mit Hilfe eines Markov-Modells

Anforderungen:

- Die Aufgabe wird in Python programmiert.
- Die Aufgabe wird von jedem Teilnehmer einzeln erstellt!
- Der Teilnehmer kommt rechtzeitig zur Abnahme auf den Dozenten zu. Die Abnahme erfolgt für jeden Teilnehmer einzeln. Die Kenntnis des Quellcodes wird erwartet.
- Der Teilnehmer erzeugt bei der Abnahme mit dem Programm ein Bild des Markov-Modells und erklärt dem Dozenten die Herleitung.
- Programmcode wird auf Ilias hochgeladen. Die Lokation wird im Praktikum bekanntgegeben. Das File hat folgendes Format:
 - <Name>_<Vorname>_<Matrikelnummer>_Aufgabe_3_Programmcode.py
- **Bedingung zur Abnahme ist die Fähigkeit zur Matrixmultiplikation, da dies für die Aufgabe notwendig ist! Es gibt zwei Versuche bei der Abnahme.**
- **Es gibt eine Frist für die Abnahme und Hochladen der Files. Dies wird im Praktikum bekanntgegeben.**
- **Die hochgeladenen Files werden nach der Frist nochmals kontrolliert. Erst nach dieser Kontrolle gilt die Aufgabe als vollständig bestanden.**

Einleitung

Auch sicherheitsgerichtete Systeme fallen aus und müssen repariert werden. Viele Zuverlässigkeitsmodelle berücksichtigen aber nicht den Austausch von fehlerhaften Komponenten. Das Markov Modell arbeitet mit Zuständen und kann somit den fehlerhaften Zustand und deren Reparatur berücksichtigen.

Aufgabe

Ein Fließbandsystem für den Transport von Koffern am Flughafen besitzt sechs Elektromotoren für seinen Antrieb. Ein Diagnosesystem misst über die Motorströme die Zustände der Motoren. In dem Fall, dass das Diagnosesystem ein auffälliges Verhalten bei einem oder bei zwei Motoren misst, geht das System in einen Warnzustand. Misst das Diagnosesystem ein auffälliges Verhalten bei mehr als zwei Motoren, dann geht das System in einen Fehlerzustand. Befindet sich das System in einem Warnzustand, dann wird ein Servicetechniker angerufen, der innerhalb von 24h erscheint.

Befindet sich das System in einem Fehlerzustand, dann erscheint der Servicetechniker innerhalb von 12h. Die Reparatur des Systems nach dem Eintreffen des Servicetechnikers dauert im Schnitt vier Stunden. Die Rate für die Diagnose eines, bzw. mehrerer auffälliger Motoren beträgt 2/Jahr. Bei der Reparatur kann der Servicetechniker entscheiden, ob eine Generalüberholung des Gesamtsystems durchgeführt werden muss. Dies passiert im Schnitt einmal in zehn Jahren. Eine Generalüberholung dauert drei Wochen.

- a.) Programmieren Sie das Markov Modell in Python. Verwenden Sie dafür drei Klassen: STATE, TRANSITION und MARKOV, siehe Bild 2, 3 u. 4. Die Klassen können wie folgt aufgebaut sein (Klassen sind nicht vollständig):

```
class STATE:
    def __init__(self, name, num):
        self.name = name
        self.num=num
    return
```

Bild 2: Zustand

```
class TRANSITION:
    def __init__(self, source, destination, name, rate):
        self.source = source
        self.destination = destination
        self.name = name
        self.rate = rate
    return
```

Bild 3: Transition

```
class MARKOV:
    def __init__(self, name, dt=1.0):
        self.nodes=[]
        self.transitions=[]
        self.name=name
```

Bild 4: Markov

Das Markov Modell soll wie unten dargestellt (Programmbeispiel 1) aufgebaut werden. Es werden für die Klasse Markov die Methoden state() und transition() benötigt:

```
M = MARKOV("Beispiel")

S1 = STATE('S1',0)
S2 = STATE('S2',1)
M.state(S1)
M.state(S2)
T12 = TRANSITION(S1, S2, '12', 1000)
M.transition(T12)
```

Programmbeispiel 1

- b.) Programmieren Sie eine Simulation, um die Wahrscheinlichkeiten zu ermitteln, dass das System generalüberholt wird. Dazu muss die Simulation oft wiederholt werden. Es soll dafür die Methode probability() der Klasse Markov verwendet werden. Sie können hierfür die library numpy

verwenden. Setzen Sie für die Simulation die folgende Formel ein:

$$\underline{p}(t + dt) = \underline{p}(t) \cdot P$$

mit

$$\underline{p}(0) = (1 \quad 0 \quad 0 \quad 0 \quad 0 \quad \dots)$$

- c.) Programmieren Sie den Graphen des Markov Modells mit Graphviz/Matplot/OpenCV und präsentieren diesen auf dem Bildschirm.
- d.) Erweitern Sie das Markov-Modell zu einem MDP-Modell. Führen Sie dazu neue Zustände ein, und zwar Q-State's. Das MDP soll zwei Aktionen besitzen, und zwar „betreiben“ und „außer Betrieb setzen“. Es soll dabei nur eine Transition mit Aktion „außer Betrieb setzen“ von dem Zustand „Generalüberholung“ zu einem neuen Zustand „Außerbetrieb“ geben. Alle anderen Transitionen werden der Aktion „betreiben“ zugeordnet. Stellen Sie das MDP-Modell auf dem Bildschirm dar. Die Häufigkeit für die Ausmusterung ist einmal in 20 Jahren. Rewards werden hier nicht benötigt, da keine Gesamtbelohnung berechnet werden soll.
- e.) Führen Sie die Simulation bei der Abnahme mit einem vom Dozenten vorgegebenen Markov-Modell und MDP-Modell aus. Das MDP-Modell besteht tatsächlich aus zwei Markov-Modellen; ein Modell mit der Aktion „betreiben“ (das ursprüngliche Markov-Modell) und ein Modell mit der Aktion „außer Betrieb setzen“. Führen Sie die Simulationen (siehe b)) mit beiden Modellen aus.