

Modellgestützte Analyse und Optimierung Übungsblatt 4

Ausgabe: 27.04.2015, Abgabe: 04.05.2015 (12 Uhr)

Aufgabe 4.1: Verteilungsfunktionen

(3 Punkte)

Gegeben ist die Funktion

$$f(x) = x^2 + \frac{2}{3}x + \frac{1}{3} \quad \text{für } 0 \leq x \leq c$$

1. Wie muss c gewählt werden, damit $f(x)$ die Dichtefunktion einer kontinuierlichen Zufallsvariable X ist? Setzen sie dann die Aufgabe mit dem ermittelten Wert für c fort.
2. Berechnen sie die Verteilungsfunktion $F(x)$.
3. Zeichnen sie den Graphen der Dichtefunktion und der Verteilungsfunktion.
4. Berechnen sie $P[1/3 \leq X \leq 2/3]$, sowie den Erwartungswert $E[X]$ und die Varianz $Var[X]$.

Aufgabe 4.2: Gedächtnislosigkeit der geometrischen Verteilung

(3 Punkte)

Zeigen Sie, dass die geometrische Verteilung gedächtnislos ist. Es soll für eine Zufallsvariable X mit geometrischer Verteilung also gelten:

$$P[X = t + s | X \geq t] = P[X = s] \quad \text{für alle } t, s > 0$$

(Hinweis: Nutzen sie die Definition der bedingten Wahrscheinlichkeit: $P[B|A] = \frac{P[B \cap A]}{P[A]}$)

Aufgabe 4.3: Eigenschaften von Zufallszahlengeneratoren

(2 Punkte)

Stochastische Simulation erfordert Zufallszahlengeneratoren, die folgende Eigenschaften besitzen sollten:

- Sie erzeugen unabhängige, gleichverteilte Zahlen.
- Sie erzeugen Folgen mit großer Periode.
- Sie erzeugen reproduzierbare Zahlen.
- Die Zahlen sind einfach und schnell berechenbar.

Untersuchen Sie, ob die folgenden Generatoren diese Eigenschaften besitzen:

1. Erzeugung einer Folge von "0"en und "1"en durch das Werfen einer idealen Münze. Herausschneiden einer beliebigen Teilfolge der Länge n und Interpretation dieser Teilfolge als Binärzahl.

2. Messen der natürlichen Radioaktivität. Jedes mal wenn diese einen Schwellwert S überschreitet, wird ein Impuls ausgelöst und gezählt. Nach Ablauf eines festen Zeitintervalls wird der Zählerstand abgelesen und so ein neues Element der Folge erzeugt.
3. Verfahren 1) wird einmal durchgeführt und die entstandene Zahlenfolge als Tafel gespeichert.
4. Midsquare-Verfahren nach John von Neumann
5. Kongruenzgenerator nach Lehmer, wobei der Startwert (seed) als $f(\text{Systemzeit, Datum, Process-ID})$ gewählt wird.

Aufgabe 4.4: Erste Schritte mit AnyLogic

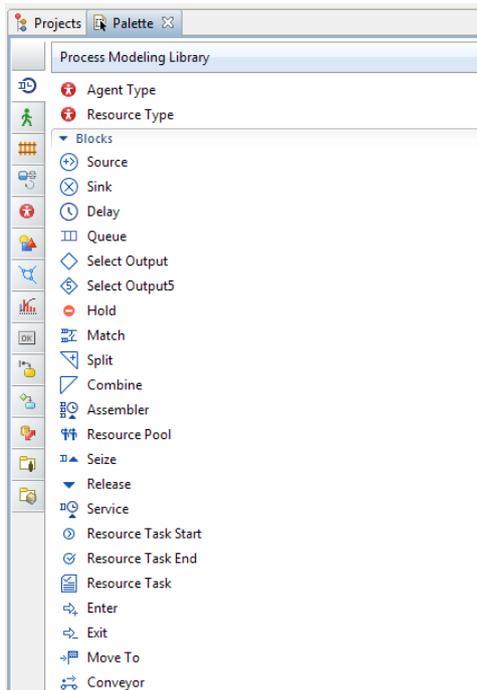
(4 Punkte)

AnyLogic ist eine Mehr-Methoden Simulationssoftware, die unter anderem ereignisorientierte und agentenbasierte Modellierung unterstützt. Für die Übungen nutzen wir die Free Personal Learning Edition (Free PLE), die kostenlos von der Webseite <http://www.anylogic.de/> heruntergeladen werden kann.

Nützliche Hinweise bieten die enthaltene Hilfe und das Buch *AnyLogic 7 in Three Days: A Quick Course in Simulation Modeling*, das ebenfalls kostenlos von <http://www.anylogic.de/free-simulation-book-and-modeling-tutorials> heruntergeladen werden kann.

Im Rahmen dieser Aufgabe soll Schritt für Schritt ein erstes einfaches Modell einer kleinen Fabrik erstellt und analysiert werden. Die Fabrik stellt aus ankommenden Bauteilen zwei unterschiedliche Produkte her, die jeweils in zwei Schritten produziert werden. Der erste Produktionsschritt ist dabei für beide Produkte identisch. Die Fabrik verfügt also über drei Pools von Maschinen, wobei Pool A den ersten gemeinsamen Produktionsschritt für beide Produkte durchführt und die Pools B und C jeweils den zweiten Produktionsschritt für eines der beiden Produkte ausführen. Im Detail ergeben sich folgende Parameter für das Modell:

- Bauteile kommen exponentialverteilt mit einer Rate von 0.5 pro Minute in der Fabrik an.
- Maschinenpool A besteht aus 7 Maschinen. Es können also bis zu 7 Bauteile gleichzeitig bearbeitet werden. Ein Bauteil wird immer nur durch maximal eine Maschine aus dem Pool bearbeitet.
- Die Bearbeitungszeit eines Bauteils durch eine Maschine aus Pool A ist gleichverteilt im Intervall $[8, 9]$ Minuten.
- 30% der von Pool A bearbeiteten Bauteile werden zu Pool B weitergeleitet, die restlichen zu Pool C.
- Maschinenpool B besteht aus 4 Maschinen.
- Die Bearbeitungszeit eines Bauteils durch eine Maschine aus Pool B ist gleichverteilt im Intervall $[15, 20]$ Minuten.
- Maschinenpool C besteht aus 2 Maschinen.
- Die Bearbeitungszeit eines Bauteils durch eine Maschine aus Pool C ist gleichverteilt im Intervall $[5, 10]$ Minuten.

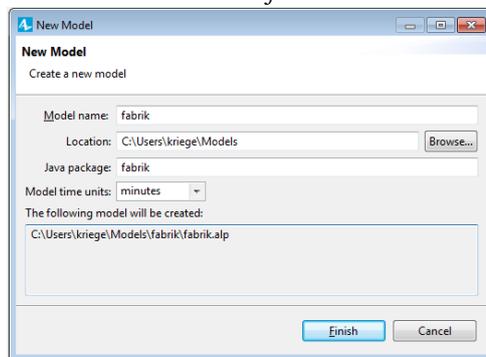


AnyLogic verfügt über Bibliotheken, die vordefinierte Modellbausteine (*Blocks*) für bestimmte Anwendungsgebiete enthalten. Die Abbildung rechts zeigt die Bausteine der *Process Modeling Library*, mit der sich Warte/Bedien-Systeme realisieren lassen.

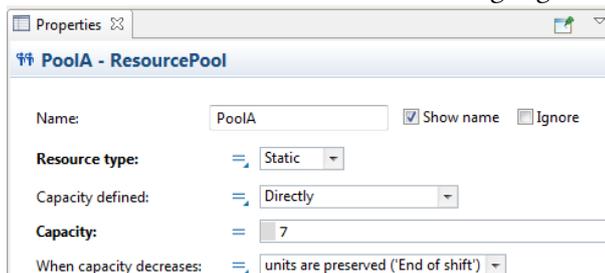
Diese Modelle bestehen aus mehreren miteinander verbundenen Blöcken. Ausgehend von einer Quelle (*Source*), die Ankünfte von Entitäten (Kunden, Bauteile, ...) erzeugt, durchlaufen diese Entitäten mehrere Bearbeitungsschritte, die durch weitere Blöcke realisiert werden, und verlassen dann an einer Senke (*Sink*) das Modell. Die *Process Modeling Library* verfügt beispielsweise über Blöcke, mit denen sich Wartezeiten (*Delay*) oder der Zugriff auf Ressourcen (*Service*) modellieren lassen. Ressourcen selbst lassen sich mit einem *Ressource Pool* abbilden. Eine Aufteilung in parallele oder alternative Pfade ist beispielsweise mit den Blöcken *Split* und *Select Output* möglich.

a) Erstellen Sie in AnyLogic ein Modell der oben beschriebenen Fabrik. Gehen Sie dafür wie folgt vor:

- Starten Sie AnyLogic und erstellen Sie ein neues Modell.
- Wählen Sie als Name *fabrik* und als Einheit für die Modellzeit Minuten.

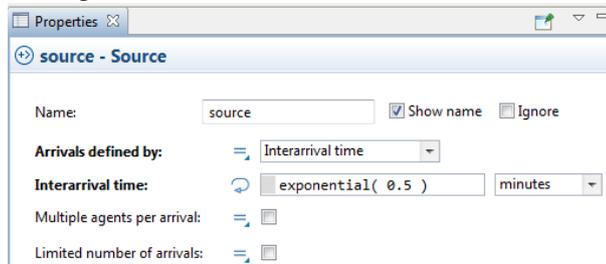


- Wählen Sie in dem Reiter *Palette* die *Process Modeling Library* aus.
- Wir modellieren zunächst die 3 Pools mit den Maschinen. Ziehen Sie dazu per Drag and Drop einen *Ressource Pool* in das Modellfenster. Unter *Properties* können die Parameter für den Ressourcen-Pool eingestellt werden. Benennen Sie den Block *PoolA*, wählen Sie als *Resource type* den Punkt *static* (die Maschinen sind fest in der Fabrik) und als *Capacity* 7, da sieben Maschinen in dem Pool zur Verfügung stehen.

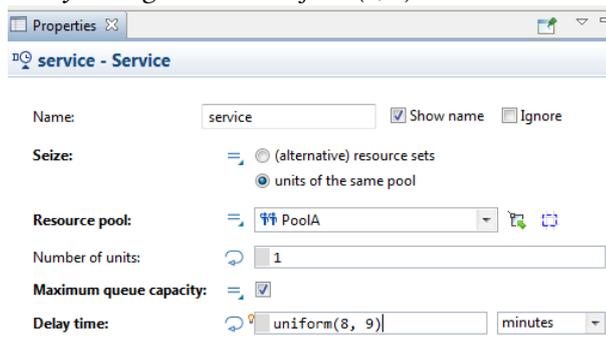


Verfahren Sie analog für die Maschinen aus den Pools B und C.

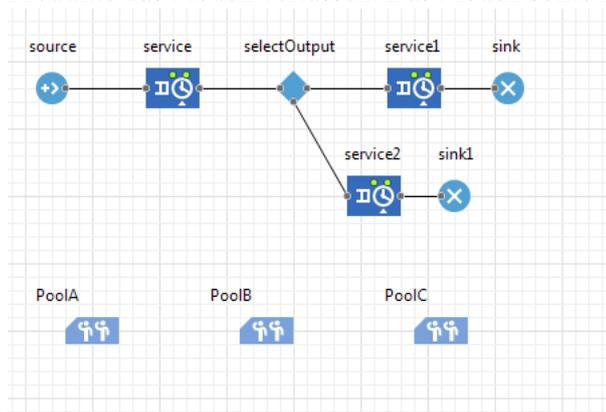
- Ziehen Sie nun per Drag and Drop eine Quelle *Source* in das Modellfenster.
- Unter *Properties* können die Parameter für die Quelle eingestellt werden. Wählen Sie für *Arrivals defined by* den Punkt *Interarrival time* und als *Interarrival time* eine Exponentialverteilung wie oben beschrieben.



- Ziehen Sie nun einen *Service* Block in die Nähe der Quelle. Die beiden Blöcke werden automatisch miteinander verbunden. Der Service Block soll die Bearbeitung der Bauteile durch eine Maschine aus Pool A modellieren. Wählen Sie dazu bei *Seize* den Punkt *units of the same pool* und als *Resource pool* den *PoolA*. Setzen Sie das Häkchen bei *Maximum queue capacity*. Als *Delay time* geben Sie *uniform(8, 9)* Minuten ein.



- Wir müssen nun modellieren, dass die Bauteile nach Bearbeitung durch Pool A aufgeteilt werden. Verbinden Sie hierzu einen *Select Output* Block mit dem Service Block. Wählen Sie als *Probability* 0.3. Hierdurch werden 30% der Bauteile zu dem rechten Ausgang (für Pool B) und 70% zu dem unteren Ausgang (für Pool C) geleitet.
- Schließen Sie an die beiden Ausgänge des *Select Output* Blocks jeweils einen weiteren *Service* Block an und modellieren Sie den Zugriff auf die Pools B und C.
- Verbinden Sie jeweils einen *Sink* Block mit den beiden *Service* Blöcken, damit die fertigen Produkte das Modell verlassen. Das Modell sollte nun folgendermaßen aussehen:



Die *Analysis* Bibliothek bietet verschiedene Blöcke zur Darstellung von Leistungsgrößen an. Hier können beispielsweise Histogramme, Plots von Zeitreihen usw. erzeugt werden.

- b) Fügen Sie einen *Time Plot* Block in das Modell ein. Fügen Sie unter *Properties* ein *Data item* hinzu. Geben Sie als *Value* den Wert *PoolA.utilization()* ein. Der Plot wird nun während der Simulation die mittlere Auslastung von Maschinenpool A visualisieren. Erzeugen Sie analog zwei weitere *Data items* für die beiden verbleibenden Maschinenpools. Starten Sie die Simulation und lassen Sie sie für 500 Zeiteinheiten laufen. Was beobachten Sie?
- c) Versuchen Sie den Engpass in der Fabrik zu beheben und simulieren Sie das Modell dann erneut.