



Discrete Event Systems

Sample Exam

1 Petrinetze (26 Punkte)

Hinweis: Wir bitten Sie, die Bedeutungen der Stellen und Transitionen der von Ihnen in dieser Aufgabe entworfenen Petrinetze klar zu beschreiben.

Alice lädt Bob zum Abendessen ein. Leider hat's in Alice' WG fast kein Besteck.

- a) (10) Zur Vorspeise serviert Alice Suppe. Leider gibt es aber nur einen Löffel, es kann also zu jedem Zeitpunkt nur immer einer von beiden essen. Allerdings sind Alice und Bob sowieso eher an sich interessiert sind als am Essen. Deshalb wird, wann immer einer der beiden einen Löffel voll Suppe gegessen hat, das Besteck wieder hingelegt und ein bisschen geplaudert. Irgendwann nimmt dann wieder jemand den Löffel und isst ein wenig weiter.
- (i) Modellieren Sie die Situation mit einem Petrinetz.
 - (ii) Beweisen Sie *formal*, dass in Ihrem Petrinetz wirklich immer nur höchstens einer von beiden den Löffel besitzt.
 - (iii) Wie sähe das Petrinetz aus, wenn sich die beiden strikt abwechseln würden?
- b) (10) Alice findet in der Küche zusätzlich zum Löffel noch eine Gabel. Zur Hauptspeise gibt es Spaghetti mit Gemüse. Spaghetti isst man "natürlich" mit einem Löffel und einer Gabel. Das Gemüse kann man entweder mit nur einem Löffel oder mit nur einer Gabel essen (nicht beides). Wieder nehmen sich die beiden viel Zeit und plaudern nach jedem Bissen. Irgendwann nimmt dann jemand den Löffel oder die Gabel und isst ein wenig Gemüse. Oder er nimmt beides und isst Spaghetti. Modellieren Sie auch diese Situation als Petrinetz.
- c) (3) Angenommen es gäbe eine zweite Gabel. Überlegen Sie, ob Sie Ihr Petrinetz von der vorangehenden Teilaufgabe mit minimaler Veränderung auch für diese Situation verwenden können. Begründen Sie in 2-3 Sätzen!
- d) (3) Zurück zur Situation mit einem Löffel und einer Gabel. Überraschend kommt nun auch Trudy – Alice' WG-Kollegin – vom Ausgang nach Hause. Sie nimmt sich ein Joghurt aus dem Kühlschrank und setzt sich zu den beiden anderen. Wie Sie richtig vermuten, braucht Trudy einen Löffel, um Joghurt zu essen. Ergänzen Sie Ihr Petrinetz entsprechend!

2 Endliche Automaten und reguläre Sprachen (34 Punkte)

- a) (10) Betrachten Sie den NFA A in Abbildung 1 und nehmen Sie an, dass $\Sigma = \{0, 1\}$.

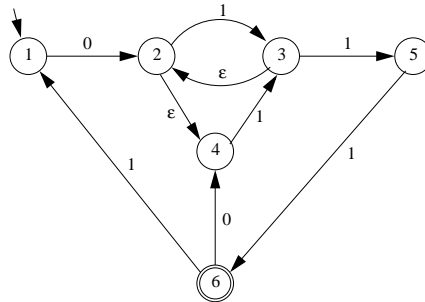


Abbildung 1: NFA A .

- (i) (7) Transformieren Sie den NFA in einen äquivalenten DFA.
 (ii) (3) Welche reguläre Sprache akzeptiert der Automat A ?
- b) (6) Gegeben sei die Sprache

$$L = \{0^a 1^b 0^c 1^d \mid a, b, c, d \geq 0 \text{ und wenn } a = 1 \text{ und } b = 2, \text{ dann } c = d\}.$$

Ist die Sprache L regulär? Beweisen Sie Ihre Antwort. (Achtung: Seien Sie vorsichtig beim Beweis!)

- c) (8) Betrachten Sie den DFA in Abbildung 3, der die Sprache L akzeptiert. Das Alphabet sei $\Sigma = \{0, 1\}$. Sei $\Phi(L)$ definiert als $\Phi(L) = \{w \in \Sigma^* \mid \exists x \in \Sigma^*, |x| = |w| \text{ und } wx \in L\}$. Mit anderen Worten, $\Phi(L)$ ist die Menge der ersten Hälften aller Strings in L .

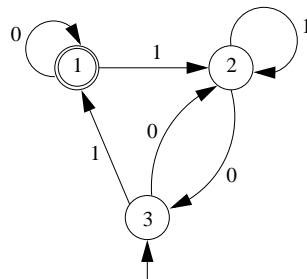


Abbildung 2: DFA B .

Konstruieren Sie einen DFA, der einen String w nur genau dann erkennt, wenn $w \in \Phi(L)$.

- d) (10) Die regulären Sprachen sind *abgeschlossen* (*closed*) bezüglich einer Operation g , wenn für jede reguläre Sprache L gilt, dass $g(L)$ wieder regulär ist. Die regulären Sprachen sind beispielsweise abgeschlossen bezüglich des Kleene star oder der Vereinigung zweier Sprachen.

Sind die regulären Sprachen abgeschlossen bezüglich der folgenden Operatoren? Beweisen Sie jeweils Ihre Antwort.

- (i) (5) $NOPREFIX(L) = \{w \in L \mid \text{es gibt keinen String } x \in L, \text{ sodass } w \text{ ein Präfix von } x \text{ ist}\}$.
 (ii) (5) In dieser Teilaufgabe betrachten wir einen binären Operator $SHUFFLE(L_1, L_2)$. Der Operator $SHUFFLE$ nimmt als Argument zwei Sprachen $L_1 \subseteq \Sigma_1^*$ und $L_2 \subseteq \Sigma_2^*$, wobei Σ_1 und Σ_2 disjunkte Alphabete sind. Der Operator $SHUFFLE(L_1, L_2)$ ist definiert als die Menge der Strings $w \in (\Sigma_1 \cup \Sigma_2)^*$ mit den folgenden Eigenschaften:
- Ersetzt man alle Symbole $x \in \Sigma_1$ in w durch ϵ ist der resultierende String in L_2 .
 - Ersetzt man alle Symbole $x \in \Sigma_2$ in w durch ϵ , ist der resultierende String in L_1 .

3 Zeitkontinuierliche stochastische Prozesse (22 Punkte)

Bei der Hotline des Internet Service Providers RedWindow arbeiten zwei Supporter: Alice und Bob. Jeder Anruf wird zufällig entweder in die Warteschlange (engl. *queue*) von Alice oder in die Warteschlange von Bob eingereiht. Die Ankunft der Kundenanrufe ist Poisson verteilt mit Rate λ . Die Servicezeit eines Anrufs ist sowohl bei Alice als auch bei Bob exponentiell verteilt mit Rate μ .

Der Chef von RedWindow bekommt nun eine Bewerbung von Trudy, welche dank ihrer guten Ausbildung Kunden doppelt so schnell abfertigen kann, d.h. mit Rate 2μ . Trudy verlangt aber auch einen Lohn in der Höhe von Alice und Bob zusammen. Der Chef fragt sich nun, ob es sich lohnt, Alice und Bob zu entlassen und dafür alleine Trudy einzustellen. Vom Lohn her würde das keinen Unterschied machen. Aber wie sieht es aus mit der Effizienz?

- a) (5) Ist die durchschnittliche Wartedauer eines Kunden grösser, kleiner, oder gleich gross falls Trudy Alice und Bob ersetzt? Begründen Sie Ihre Antwort ausführlich!

Der Chef entscheidet sich (trotz oder wegen Ihrer Empfehlung), nur noch Trudy zu beschäftigen. Ausserdem findet der Chef heraus, dass es zwei Typen von anrufenden Kunden gibt: Kunden des Typs C kennen sich bereits sehr gut aus mit Computern, wohingegen Kunden des Typs N jeder Schritt einzeln erklärt werden muss. Die entsprechenden Servicezeiten (von Trudy) seien $\mu_C = 1/3$ pro Minute und $\mu_N = 1/9$ pro Minute. Nehmen Sie an, dass die Anrufe weiterhin Poisson verteilt mit Rate λ ankommen, und dass ein Anruf mit 50% Wahrscheinlichkeit vom Typ C ist und mit 50% Wahrscheinlichkeit vom Typ N.

- b) (2) Nehmen Sie an, das System befinde sich im stationären Zustand. Wie gross ist die Wahrscheinlichkeit, dass ein zu einem gegebenen Zeitpunkt *gerade bedienter* Kunde vom Typ C ist?
- c) (2) Angenommen in der Warteschlange befindet sich mindestens ein Kunde. Mit welcher Wahrscheinlichkeit handelt es sich im stationären Zustand beim Kunden *an der Spitze* der Warteschlange um jemanden, der sich bereits sehr gut mit Computern auskennt?
- d) (7) Modellieren Sie die Hotline von RedWindow mit den beiden Kundentypen als Markov Prozess! Verwenden Sie dabei möglichst wenige Zustände, und beschriften Sie die Skizze ausführlich!
- e) (6) Die Ankunftsrate der Anrufe sei nun $\lambda := 1/8$ pro Minute. Was ist – im stationären Zustand – die Wahrscheinlichkeit, dass sich kein Kunde in der Hotline befindet, d.h. sich niemand in der Warteschlange befindet und auch niemand gerade bedient wird? (*Tipp*: Statt den Markov-Prozess aus der vorangehenden Teilaufgabe zu analysieren, verwenden Sie besser den Grundsatz “Was rein kommt, muss auch raus”!).

4 Online Algorithmen und kompetitive Analyse (35 Punkte)

Mario und Luigi sind Brüder und arbeiten als Eiscremeverkäufer an einem ein Kilometer langen Strandausschnitt an der Adria. Immer wenn ein Badetourist ein Eis kaufen möchte (Request), muss mindestens einer der beiden zu dem Touristen eilen, und ihm ein Eis verkaufen. Da der Strand sehr flach und übersichtlich ist, können Mario und Luigi immer den ganzen Strand überblicken. Das heisst, bei einem neuen Request kennen Mario und Luigi immer sofort den Ort des Requests. Requests kommen nacheinander und die Zeit zwischen zwei aufeinanderfolgenden Requests ist immer genug lange, damit der Weg zum Kunden zurückgelegt kann, bevor der nächste Request gestellt wird. Wir nehmen also an, dass es nie zwei unerfüllte Requests gibt.

In Bild 4 ist eine Beispielsituation dargestellt. Der i te Request befindet sich rechts von Luigi. Im Beispiel wird dieser Request von Luigi bedient. Dieser legt dafür einen Weg von 0.2 Kilometern zurück. Der $i + 1$ te Request liegt zwischen Luigi und Mario. Wer soll ihn bedienen?

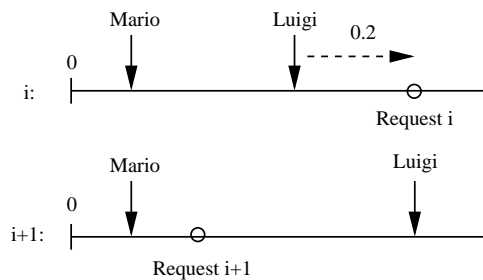


Abbildung 3: Der Strand mit Mario und Luigi's Position. Der i te Request wird von Luigi bedient.

Nehmen Sie an, dass Mario am Anfang auf Position 0 und Luigi auf Position 1 stationiert sind. Ein Request i ist definiert durch seine Position r_i im Intervall $[0, 1]$. Mario und Luigi werden konfrontiert mit einer Sequenz von Requests $\sigma = r_1, r_2, \dots$

Sei d_M resp. d_L der gesamte Weg, den Mario resp. Luigi an einem Tag (es gibt potentiell unendlich viele Requests) zurücklegen. Um am Abend nicht müde zu sein, möchten Mario und Luigi den Weg, den sie zusammen zurücklegen so kurz halten wie möglich. Das heisst, sie möchten den Wert $d_M + d_L$ minimieren. Mit welcher Strategie sollten Mario und Luigi die Requests bedienen?

- a) (10) Mario und Luigi lassen sich von der renommierten Consulting-Firma Cons-ULT beraten. Diese schlagen vor, dass immer derjenige den Request i bedienen soll, der momentan näher bei r_i ist. Der andere soll einfach an seinem gegenwärtigen Ort bleiben und warten. Gemäss Cons-ULT ist dies die bestmögliche Strategie für Mario und Luigi.

Verifizieren Sie die Aussage von Cons-ULT anhand folgender Request Sequenz σ_1 :

$$\sigma_1 = 0.4, 0.1, 0.5, 0.7, 0.0$$

Beschreiben Sie anhand von Skizzen den Weg den Mario und Luigi zurücklegen. Was wäre die optimale Lösung für Mario und Luigi gewesen? Wie kompetitiv ist der Cons-ULT-Algorithmus bezüglich der Sequenz σ_1 ?

- b) (10) Ist die Strategie von Cons-ULT c -kompetitiv für irgendeine Konstante c ? Wenn ja, für welche Konstante? Beweisen Sie ihre Antwort.
- c) (15) Es gibt einen 2-kompetitiven Algorithmus ALG für das Problem von Mario und Luigi. Wie der Cons-ULT Algorithmus ist auch ALG deterministisch, also nicht randomisiert. Wie könnte dieser 2-kompetitive Algorithmus aussehen? Schlagen Sie einen deterministischen Algorithmus vor und begründen Sie, warum dieser 2-kompetitiv ist.