Sommersemester 2003 | Prof. Dr. Susanne Albers
23. Juni 2003 | Markus Büttner
Institut für Informatik

Approximations- und Online-Algorithmen

Übung 4

Abgabe am 7. Juli 2003 (in der Vorlesung)

Aufgabe 1: (5+5 Punkte)

Ein Roboter steht in der Entfernung n vor einer Wand, an welcher sich in gerader Linie vor ihm ein Zielpunkt befindet. In seinem Weg befinden sich beliebig viele quadratische Hindernisse (mit Wänden parallel/senkrecht zur Wand), die sich (und die Wand) jedoch nicht berühren. Wenn der Roboter ein Hindernis berührt, erkennt er, in welcher Richtung sich die nächste Ecke befindet.
a. Geben Sie zunächst eine einfache 2-kompetitive deterministische Strategie an, um dieses Ziel zu erreichen.
b. Entwickeln Sie eine deterministische Strategie, die besser als 2-kompetitiv ist!(Hinweis: Der Roboter muss nicht immer nur in horizontaler/vertikaler Richtung laufen)

Aufgabe 2: (5+5 Punkte)
a. Auf einem Gitter befinden sich jeweils auf der Kreuzung zweier Gitterlinien ein Roboter und ein Ziel. Der Roboter kann sich nur entlang der Gitterlinien bewegen, und weiß nicht, wo sich das Ziel befindet. Geben Sie eine möglichst gute Strategie für den Roboter an, um das Ziel zu finden, und schätzen Sie diese gegen OPT ab. Die Gitterlinien haben den Abstand 1.
b. Wir wollen nun auch allgemeinere Bewegungen zulassen (ohne Gitter). Der Roboter darf sich mit konstanter Geschwindigkeit in beliebige Richtungen bewegen, und sieht alle Objekte in einer Entfernung $d \leq 0.5$. Das Ziel befinde sich an einem beliebigen Punkt der Ebene. Geben Sie wieder eine möglichst gute Strategie an, und schätzen Sie sie die zu laufende Strecke ab.

Aufgabe 3: (10 Punkte)
Lastverteilungsproblem: Jobs unbekannter Dauer, die nacheinander eintreffen sollen auf Maschinen verteilt werden. Jeder Job kann nur von bestimmten Maschinen bearbeitet werden.
Zeigen Sie, dass der Algorithmus $GREEDY$, der jeden eintreffenden Job auf die zulässige Maschine mit der geringsten Last verteilt, nicht besser als $\Omega(N^{2/3})$-kompetitiv ist. Modifizieren Sie dazu das Verfahren aus der Vorlesung in geeigneter Weise:

Verwenden Sie folgende Invarianten:
- OPT hat stets auf jeder Maschine m_i nur einen Job j_i laufen.
- auch $GREEDY$ hat, wenn die Last von m_i größer 0 ist,diesen Job auf m_i laufen.

Zeigen Sie dabei:
Wenn für den absteigend sortierten Lastvektor von $GREEDY$ für und eine nichtleere Maschine m_i gilt:$L_{i-1}(t) < L_i(t) + i - 1$, dann lässt sich eine Anfragesequenz angeben, um einen lexikographisch größeren Lastvektor für $GREEDY$ zu erhalten.

Hinweis: Kann man die Jobs j_i von Maschinen mit größerer Last als Maschine m_k verwenden, um Last von m_k auf m_{k-1} umzuverteilen?

Aufgabe 4: (5+5 Punkte)
Bei Umtauschen einer Währung in eine andere kann zu mehreren Zeitpunkten jeweils ein Teil einer festen Geldmenge in eine andere Währung getauscht werden. Beim Suchproblem muss dieser Umtausch zu einem einzelnen Zeitpunkt erfolgen

Beweisen Sie:
a. Sei ALG_R ein randomisierter Algorithmus für das Geldumtauschproblem. Dann existiert ein deterministischer Algorithmus ALG_D, so dass gilt: $E[ALG_R(\sigma)] = ALG_D(\sigma)$ für alle σ.
b. Sei ALG_D ein deterministischer Algorithmus für das Geldumtauschproblem, dann existiert ein randomisierter *Suchalgorithmus* ALG_R, so dass gilt: $ALG_D(\sigma) = E[ALG_R(\sigma)]$ für alle σ.