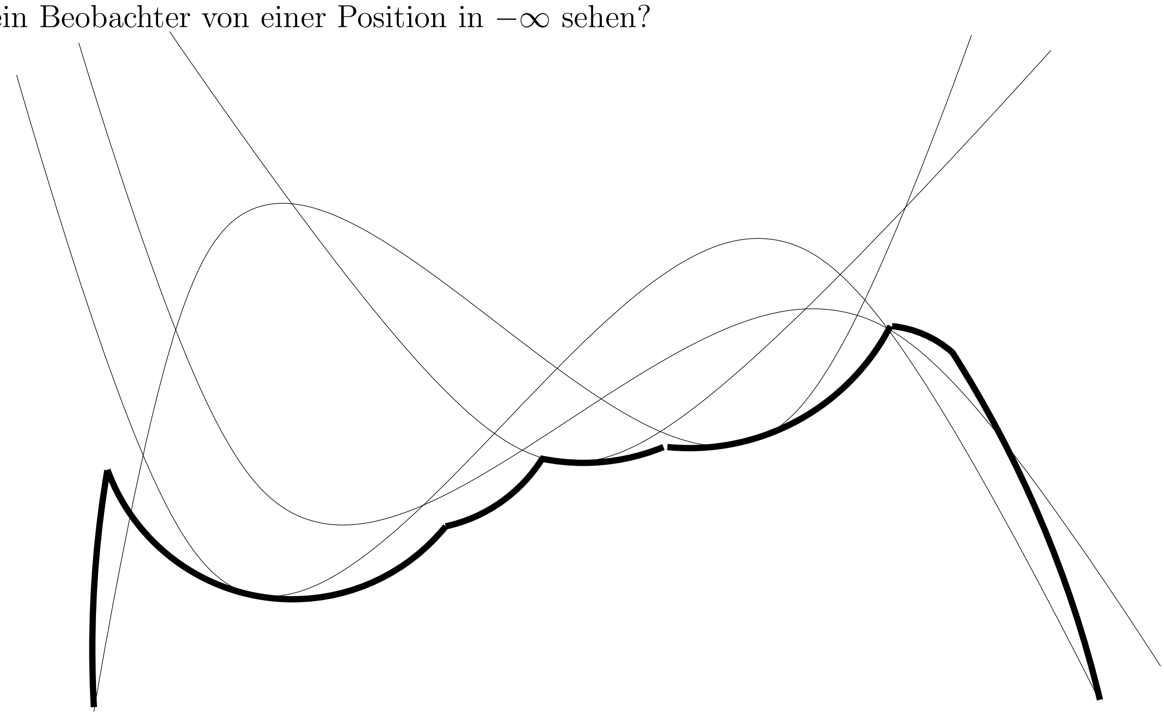
Rheinische Friedrich-Wilhelms-Universität Bonn
Institut für Informatik I
Dr. Elmar Langetepe   Annette Ebbers-Baumann

Übung 10 Bewegungsplanung für Roboter SS 2001

Mittwoch, 13.00 Uhr s.t., HS 1001

Aufgabe 1: Gegeben seinen n Polynome dritten Grades der Form $a_iX^3 + b_iX^2 + c_iX + d_i$ für $i = 1, \ldots n$. Wieviele Teilstücke der Polynome, wie in der Abbildung illustriert, kann ein Beobachter von einer Position in $-\infty$ sehen?



Aufgabe 2: Auf einer Geraden g befinden sich n Punkte p_i, an jedem Punkt soll eine bestimmte Dienstleistung mit einer bestimmten Güte $w_i \in R$ bereitgestellt werden. Je kleiner w_i ist desto schlechter ist die Dienstleistung. (Es könnte sich zum Beispiel um Kneipen handeln und die Güte einer Kneipe ergibt sich durch den Kehrwehrt des Preises des Standardbieres). Wenn wir uns an einem Punkt $x \in g$ befinden und die Dienstleistung an einem Ort p_j in Anspruch nehmen, müssen wir den Weg der Länge $d(p_j, x)$ zurücklegen. Für uns lohnt sich das nur dann, wenn das Verhältnis $\frac{d(p_j,x)}{w_i}$ für $i = 1, \ldots n$ am kleinsten ist. Wenn wir für alle Punkte $x \in g$ den optimalen Ort p_j bestimmen wollen, was hat das mit der klassischen Anwendung der Davenport-Schinzel-Sequenzen zu tun?
Hinweis: Betrachten Sie die Funktionen $\frac{d(p_j,x)}{w_i}$.

Postanschrift:
Universität Bonn
Institut für Informatik I
Römerstr. 164, D-53117 Bonn

Telefon: +49 (2 28) 73–4322   73-4135
Fax: +49 (2 28) 73–4321
Email: Elmar.Langetepe@informatik.uni-bonn.de   ebbers@informatik.uni-bonn.de

Aufgabe 3: Bei der Behandlung von Translationsbewegungen eines beliebigen (nichtkonvexen) Roboters mit m Ecken unter beliebigen (nichtkonvexen) Hindernissen mit insgesamt n Ecken kann der Raum der freien Konfigurationen $\mathcal{C}_{frei}$ von der Komplexität $O(m^2n^2)$ sein. Deshalb wird für die effiziente Bewegungsplanung nur die Zusammenhangskomponente $\mathcal{Z}_s$ von $\mathcal{C}_{frei}$ berechnet, die den Startpunkt s enthält. Dies kann durch einen Divide & Conquer-Algorithmus erfolgen, bei dem der Merge-Schritt mittels des in der Vorlesung vorgestellten Sweep-Algorithmus, der Scouts verwendet, durchgeführt wird.
Spielen Sie diesen speziellen Divide & Conquer-Algorithmus an dem folgenden Beispiel zur Berechnung der "lila" Zellen, die den Punkt s enthalten, durch. Teilen und mergen Sie dabei wie folgt: (((12)(34))((56)(78)))