

Lösungsblätter zur Klausur

Robotik I: Einführung in die Robotik

am 19. Juli 2016, 18:00 – 19:00 Uhr

Name:	Vorname:	Matrikelnummer:
-------	----------	-----------------

Aufgabe 1	von 5 Punkten
Aufgabe 2	von 4 Punkten
Aufgabe 3	von 9 Punkten
Aufgabe 4	von 7 Punkten
Aufgabe 5	von 3 Punkten
Aufgabe 6	von 4 Punkten
Aufgabe 7	von 5 Punkten
Aufgabe 8	von 8 Punkten

Gesamtpunktzahl:	
------------------	--

	Note:
--	-------

Name:

Vorname:

Matr.-Nr.:

2

Aufgabe 1

1. Das Quaternion \mathbf{q} :

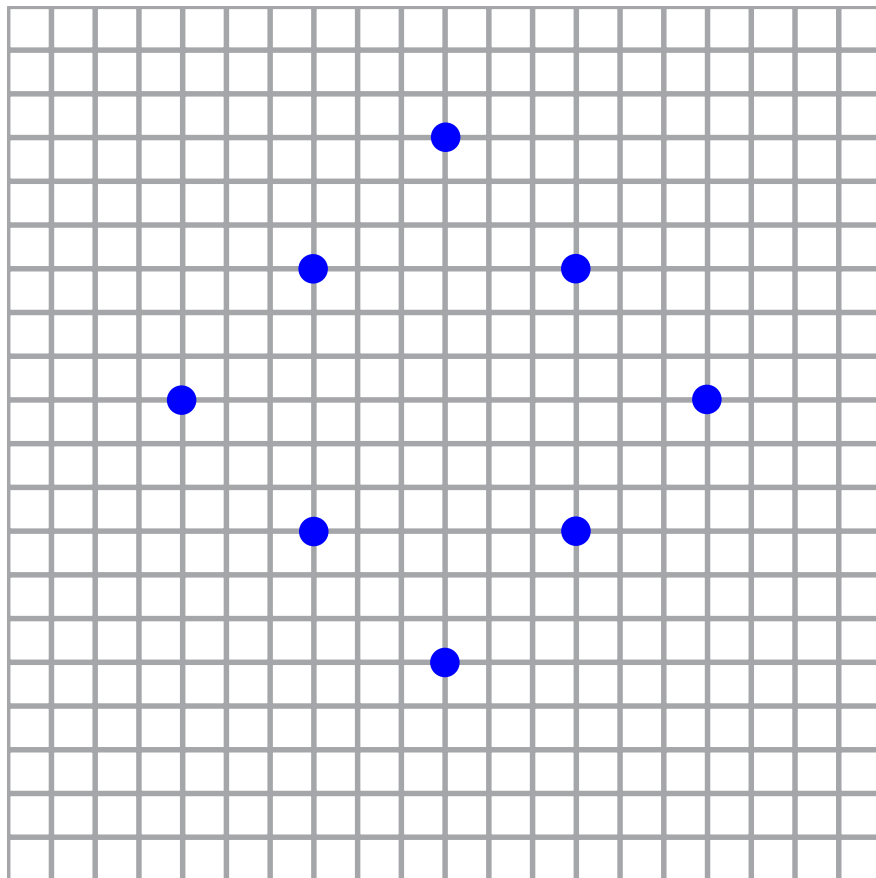
2. Das konjugierte Quaternion \mathbf{q}^* :

3. Rotation des Vektors \mathbf{v} :

Aufgabe 2

1. Was versteht man unter einem Voronoi Diagramm?

2. Zeichnen Sie in die nachfolgende Abbildung das fertige Voronoi Diagramm ein.



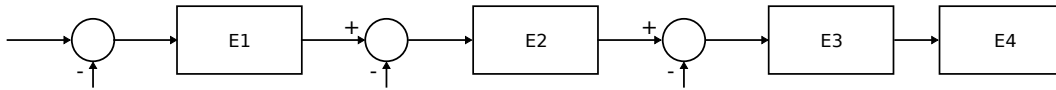
Aufgabe 3

1.
 -
 -
2.
 - Regler A:
 - Regler B:
 - Regler C:

3. Tragen Sie die Lösung in die nachfolgende Tabelle ein.

Übertragungsglied	Funktionalbeziehung	Symbol
M-Glied/Multiplizierglied		
S-Glied/Summierglied		
KL-Glied/Kennlinienglied		
P-Glied/Proportionalglied		

4. Ergänzen Sie den folgenden Wirkungsplan



a) • E1:

• E2:

• E3:

• E4:

b) • R1:

• R2:

• R3:

Aufgabe 4

1. Die vier Schritte des RANSAC Algorithmus:

2. Tragen Sie die ersten drei RANSAC Iterationen in die nachfolgenden Abbildungen ein.

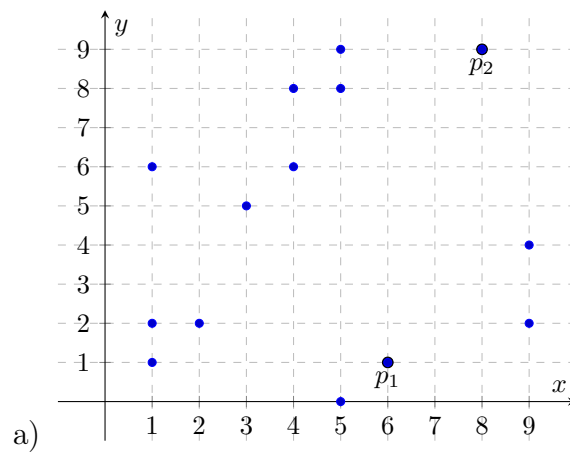


Abbildung 1: Modell für 1. Iteration

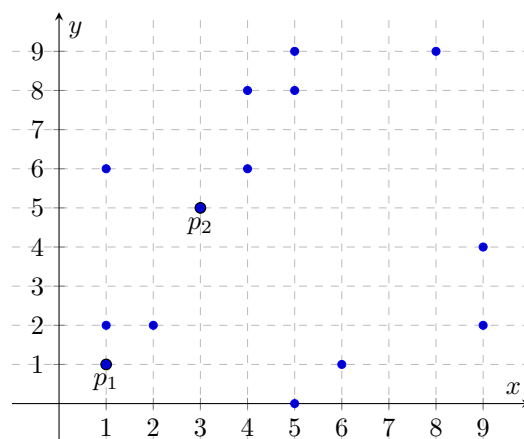


Abbildung 2: Modell für 2. Iteration

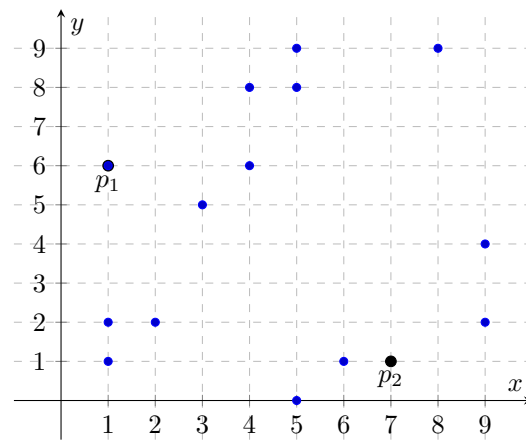


Abbildung 3: Modell für 3. Iteration

b) Tragen Sie die Anzahl der Inlier für jede Iteration ein.

- Iteration 1:
- Iteration 2:
- Iteration 3:

3.

Name:

Vorname:

Matr.-Nr.:

8

Aufgabe 5

1.

2.

Aufgabe 6

1. •

•

•

•

2. •

•

•

•

Aufgabe 7

1. Beispiel für inverse Kinematik:

2. Jacobi-Matrix:

3. Transformationsmatrix:

Aufgabe 8

Beantworten Sie die folgenden Fragen, indem sie entweder richtig oder falsch ankreuzen. Für jede korrekte Antwort erhalten Sie 0,5 Punkte. Jede nicht oder falsch beantwortete Frage wird mit 0 Punkten bewertet.

a)

<i>Roboterprogrammierung</i>	<i>richtig</i>	<i>falsch</i>
Ein Vorteil der dynamikbasierten interaktiven Programmierung ist die Möglichkeit lokale Hindernisse zu vermeiden.		
Ein Nachteil der dynamikbasierten interaktiven Programmierung ist, dass die erstellten Programme nicht generalisieren.		
Planungsbasierte interaktive Programmierverfahren können mehr als eine Lösung zur Verfügung stellen.		
Play-Back Programmierung ist nicht besonders gut für schwere Roboter geeignet.		

b)

<i>Regelung</i>	<i>richtig</i>	<i>falsch</i>
In einem Regelkreis wirkt die Störgröße direkt auf das dynamische System ein.		
Bei einer Kraftregelung ist die Reibung im dynamischen System vernachlässigbar.		
Parameter einer Impedanzregelung sind Steifigkeit, Dämpfung und Trägheit.		
Eine Regelung im kartesischen Raum ist weniger Aufwendig als im Gelenkwinkelraum.		

c)

<i>Gelenke</i>	<i>richtig</i>	<i>falsch</i>
Die Drehachse eines Torsionsgelenks bildet einen rechten Winkel mit den Achsen der beiden angeschlossenen Glieder.		
Die Steward-Plattform ist ein paralleler Roboter.		
Das Lineargelenk ist ein Spezialfall des Torsionsgelenks.		
Das menschliche Ellenbogengelenk ist ein Beispiel für ein Revolvergelenk.		

d)

<i>Modelle</i>	<i>richtig</i>	<i>falsch</i>
Das geometrische Kantenmodell wird zur schnellen Kollisionsberechnung verwendet.		
Das geometrische Modell wird zur Berechnung der Roboterdynamik benötigt.		
Bei der Lösung des direkten kinematischen Problems werden Gelenkwinkel berechnet.		
Das dynamische Modell beschreibt die Bewegung von Objekten auf Grund von wirkenden Kräften und Momenten.		