



BOSCH

Technik fürs Leben

1. Bosch-Modellierungswettbewerb für Studierende der Universität Stuttgart

Die virtuelle Abbildung realer Vorgänge mit Hilfe geeigneter mathematischer Modelle und deren effiziente mathematische Lösung sind in der Entwicklung und Steuerung moderner Hochtechnologie essenziell. Die Modelle beschreiben komplexe physikalische Zusammenhänge, liefern Einblicke in messtechnisch nicht oder nur schwer zu erfassende Bereiche und Vorgänge und regeln Systeme basierend auf wichtigen Zustandsgrößen. Das ermöglicht die Entwicklung ressourcenschonender, langlebiger und effizienter Produkte. Eine besondere Herausforderung dabei ist, einen sicheren qualitativ hochwertigen Betrieb über eine große Produktstückzahl und die gesamte Produktlebensdauer zu leisten. Eine Steuerung, die besonders tolerant gegenüber Fehlern und Abweichungen von der idealen Vorgabe ist, ist also wichtig.

Wir laden euch ein, euch im Rahmen eines kleinen Wettbewerbs mit robusten und fehlertoleranten Steuerungen zu beschäftigen. Natürlich sind unsere realen Systeme ein bisschen zu komplex, um sie mal eben so nebenbei zu modellieren und eine robuste Steuerung zu entwickeln. Deswegen haben wir für euch einen fiktiven Roboter ausgewählt, den wir zu einem (ebenfalls fiktiven) Basketball-Wettkampf schicken wollen. Wir wollen natürlich gewinnen, obwohl unser Roboter ein paar Probleme mit der Genauigkeit der Steuerung hat. Findet die beste Steuerung für unseren

Roboter-Korbwurf und gewinnt einen tollen Preis!

Der Roboter wird an die Freiwurflinie gestellt und so ausgerichtet, dass er zentral vor dem Korb steht. Von dort aus wirft er den Ball mit seinem Teleskop-Wurfarm.

Bei dem Ball handelt es sich um einen offiziellen Spielball Größe 7 mit einem Umfang von $765\text{mm} \pm 15\text{mm}$ und einem Gewicht von $609\text{g} \pm 41\text{g}$.

Leider ist die Steuerung des Roboters nicht 100% zuverlässig und wird durch folgende Parameter maßgeblich beeinflusst:

- Der Teleskop-Wurfarm lässt sich auf eine beliebige Höhe zwischen 1m und 2m mit einer Toleranz von $\pm 15\text{cm}$ einstellen.
- Der Abwurfwinkel nach oben lässt sich ebenfalls mit einer Toleranz von $\pm 5^\circ$ einstellen.
- Die Beschleunigung des Wurfarms, sowie die Abwurfgeschwindigkeit lassen sich jeweils mit einer Toleranz von $\pm 5\%$ einstellen.

Wie sind die Parameter für den Ballwurf zu wählen, so dass der Roboter möglichst häufig trifft, der Wurf also möglichst robust gegenüber den gegebenen

Fehlertoleranzen ist? Welche Trefferwahrscheinlichkeit ergibt sich mit den vorgegebenen Daten und der angewandten Wurftechnik?

Hinweise zur Aufgabe:

Die Freiwurflinie befindet sich in einem Abstand von 4,30m vom Korbmittelpunkt. Hier befindet sich auch der Nullpunkt unseres Koordinatensystems.

Die genauen Abmessungen des Korbes und des Bretts können der schematischen Darstellung rechts entnommen werden.

Der Ballwurf soll ohne Drehimpuls modelliert werden. Der Ball wird als starrer Körper betrachtet, d.h. beim Aufprall auf das Brett oder den Korbrand gilt ein ideales Reflektionsgesetz. Würfe, die am Brett oder am Korb abprallen, danach aber den Korb passieren, sind zulässig.

Der Roboter kann sich nicht drehen. Der Wurf erfolgt immer geradeaus, unter Berücksichtigung der oben genannten Toleranzen.

Ein Wurf gilt als erfolgreich, wenn der Basketball den Korb vollständig innerhalb des Korbquerschnitts (in einer Höhe von 3,05m, siehe Skizze) passiert.

Hinweise zum Wettbewerb:

Die Abgabe der Lösung muss bis zum 22.09.2022 an Rng_CR_MathMeetsIndustry@bosch.com erfolgen.

Rückfragen zur Aufgabenstellung können auch vorab an diese Emailadresse gerichtet werden.

Eine Bearbeitung in Teams von bis zu 4 Personen ist zulässig.

Eine vollständige Lösung besteht aus einem (kommentierten!)

Python-Programm, das von der Jury gegen einen vorgegebenen

Satz an Testdaten geprüft wird, sowie einer mündlichen Präsentation des Vorgehens und der Ergebnisse im Rahmen eines Symposiums am 06.10.2022.

Das Python Programm muss eine Funktion bereitstellen, die wie folgt aufgerufen wird:

```
korbwurf(abweichung_wurfarmhoehe, abweichung_abwurfwinkel,
abweichung_beschleunigung, abweichung_geschwindigkeit, ballradius, ballgewicht)
```

und als Rückgabe einen Positionsvektor als numpy.array zur Verfügung stellt:

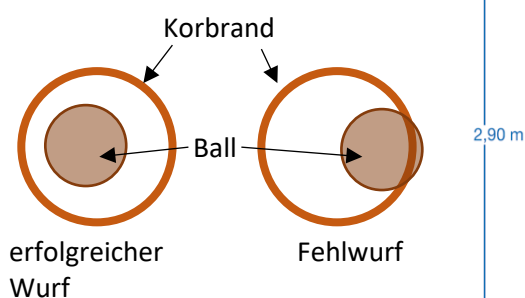
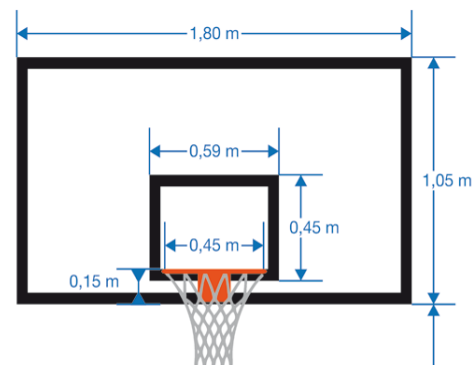
```
return Position des Ballmittelpunkts in einer Höhe von 3,05m (nach dem Scheitelpunkt
der Wurfparabel)
```

Die Abweichungen werden in den Einheiten übergeben, in denen sie oben im Text definiert sind (also wird z.B. die Abweichung von der Höhe des Wurfarmes in cm übergeben) Ballradius und Ballgewicht werden als Absolutwert übergeben und bewegen sich im Rahmen der oben genannten Toleranzen.

Die Funktion muss die Position des Ballmittelpunkts in einer Höhe von 3,05m zurückgeben, wobei der zweite Schnittpunkt mit der 3,05m-Ebene zu wählen ist.

In dem mündlichen Vortrag soll die Lösungsfindung und die verwendeten mathematischen Methoden im Vordergrund stehen. Die Jury soll nachvollziehen können, wie Sie die 4 Wurfparameter (Wurfarmhöhe, Abwurfgeschwindigkeit, -winkel, -beschleunigung) bestimmt haben und welche Grundideen dahinterstehen. Der Vortrag dauert 10min, anschließend sind noch 10min für Fragen der Jury vorgesehen.

Jedes Mitglied des Gewinnerteams des Wettbewerbs erhält **einen Bosch IXO Akkuschrauber** als Siegprämie.



© CCO

source: Wikipedia