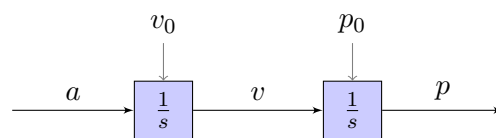


## Labor-Bericht

### Matlab-Simulation eines PID-kontrollierten Gleichstrommotors

Noah Hüsler und Almar Suter

14. Juni 2017



## Inhaltsverzeichnis

<b>1</b>	<b>Motivation</b>	<b>2</b>
<b>2</b>	<b>Simulation</b>	<b>2</b>
<b>3</b>	<b>Ergebnisse</b>	<b>3</b>
<b>4</b>	<b>Zusammenfassung</b>	<b>3</b>

## 1 Motivation

Einen Regelkreis zu simulieren ist ein sehr interessanter Anwendungsfall für Matlab. In diesem Versuch ging es insbesondere darum zwei physikalische Systeme in Simscape zu überbrücken und gemeinsam zu verwenden. Dazu wurde die Regelung eines Elektromotors gewählt.

Im Abbildung TODO: kann man den Regelkreis vom Aufbau her erkennen. Ziel ist es, dass eine gewünschte Drehzahl für einen Elektromotor eingestellt werden kann und der PID-Regler dafür sorgt, dass dieser Fall auch eintritt. Da der Motor natürlich nur begrenzt Kraft hat und der Regler nicht perfekt ausregeln kann wird das Verhalten untersucht wenn am Motor eine Last angehängt wird (Schritt-Antwort).

Das Ziel dieses Versuches ist es einen Gleichstrommotor mit angeschlossener Last durch einen PID-Regler zu regeln.

## 2 Simulation

Zu Anfang wurde ein einfacher Antrieb für den Motor wie in Abbildung ?? TODO: gezeigt aufgebaut.

Wichtig hier war dass die HBrücke und der PWM-Regler dasselbe Referenzpotential besitzen und dass ein 'Solver Configuration'-Block mit dem System verbunden ist. Dieser wird benötigt damit Matlab weiss wie er das System ausrechnen soll. Der Motor lief danach wie gewünscht an. Die Simulation dauerte schon für 3 Sekunden Simulationszeit extrem lange. Die Ursache hierfür wurde wie im PWM-Signal wie in Abbildung TODO: ersichtlich gefunden. Hier müssen natürlich extremkleine Simulationsschritte gemacht werden um das Signal korrekt wiedergeben zu können. Um das Problem zu beheben wurde der PWM-Mode auf 'Averaged' gestellt, da die HBrücke sich sowieso in diesem Modus befindet. Nach dieser Optimierung lief die Simulation jeweils in Nullzeit und der Regelkreis konnte ungehindert konstruiert werden.

In einem nächsten Schritt wurde ein PID-Regler ins System eingefügt. Zudem wurde die Konstantspannungsquelle durch eine Verstellbare Spannungsquelle ersetzt. Dieser Schritt ist in Abbildung TODO: zu sehen. In Abbildung TODO: kann die geregelte Drehzahl erkannt werden. Wie man unschwer erkennt oszilliert die Drehzahl extrem stark. Dies deutete auf ein viel zu hohes P hin. Da der Wertebereich des PWM-Regler-Inputs zwischen 0 und 5 Volt liegt aber der Drehzahl-Output des Motors bei 0 bis 4000 Umdrehungen hat der PID-Regler den PWM-Regler massiv übersteuert. Dafür wurde dann ein Gain von  $\frac{1}{800}$  wie in Abbildung TODO: zu sehen in den Regelkreis eingefügt. Zudem wurde der maximale Wertebereich des PID-Reglers auf zwischen 0 und 1 eingestellt um eine Überspannung am PWM-Regler zu vermeiden.

Nun zeigte der Regler schon ein ziemlich gutes Verhalten. In nur 3 Sekunden konnte er von 0 auf stabile 2000 Umdrehungen pro Minute regeln. Da ein Motor aber sogut wie nie ohne Last verwendet wird, wurde ein Gegenmoment an den Motor angehängt, welches nach einer Totzeit von 2 Sekunden ein Moment von 0.1Nm auf den Motor wirken lässt und ihn belastet.

Dieser finale Regelkreis ist in Abbildung TODO: zu erkennen.

### **3 Ergebnisse**

### **4 Zusammenfassung**