

DriftingDroids

noch ein weiteres Ricochet Robots Lösungsprogramm

Version 1.3.5 (2015-09-13)

Geschrieben von Michael Henke <smack42@googlemail.com>

Homepage <https://github.com/smack42/DriftingDroids>

Inhaltsverzeichnis

Willkommen.....	1
Installation und Start.....	2
Spielen.....	3
Spiel vorbereiten.....	5
Einstellungen.....	7
Danksagungen.....	8
Lizenz.....	8

Willkommen

Ricochet Robots (auch bekannt als Rasende Roboter) ist ein Brettspiel von Alex Randolph. Wenn Du es noch nicht kennen solltest dann kannst Du hier etwas darüber lesen:

Wikipedia (de) https://de.wikipedia.org/wiki/Rasende_Roboter
Wikipedia (en) https://en.wikipedia.org/wiki/Ricochet_Robot
BoardGameGeek (en) <http://boardgamegeek.com/boardgame/51/ricochet-robots>

DriftingDroids ist eine Computervariante des Brettspiels Ricochet Robots. Es enthält einen **Lösungsalgorithmus** der die **optimalen Lösungen** für jedes Spielproblem findet. Du kannst das Programm als persönlichen **Trainer** verwenden, der Dir die besten Lösungen zeigt, oder als **Schiedsrichter** bzw. Besserwisser in geselligen Spielrunden mit dem echten Brettspiel.

Installation und Start

Entpacke das heruntergeladene Zip-Archiv in ein Verzeichnis Deiner Wahl.

Doppelklicke die Datei **start.jar** um DriftingDroids zu starten.

Java SE (JRE Version **6 oder neuer**) wird benötigt um das Programm zu benutzen.

Windows Nutzer können die aktuelle Version von Java hier herunterladen:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Auf anderen Systemen, zum Beispiel **Mac OS X** oder **Linux**, verwende einfach das Java, das bereits vorinstalliert ist oder das sich leicht installieren lässt, zum Beispiel Java für Mac OS X oder IcedTea / OpenJDK.

Für technisch versierte Nutzer

Per **Kommandozeile** lässt sich das Programm wie folgt starten:

cd <Pfad> (hier hattest Du das heruntergeladene Zip-Archiv entpackt)

cd DriftingDroids_1.3.5

java -jar start.jar

Der integrierte Programmstarter (AppStart) wertet die mitgelieferte Datei **appstart.properties** aus, welche Du mit einem Texteditor bearbeiten kannst, um zum Beispiel

- den vom Programm verwendeten **Speicher** zu verändern mittels **-Xmx**
- eine bestimmte **Sprache** einzustellen mittels **user.language**
- die Option **UseSlowSearchMoreSolutions** zu aktivieren und damit den besonders gründlichen, langsamen Modus des Lösungsalgorithmus zu verwenden, der für manche Spielkonfigurationen mehr unterschiedliche, gleich gute (optimale) Lösungen findet.

Der Programmstarter kann auch umgangen und das Programm direkt gestartet werden, was die Verwendung erweiterter Optionen beim Start der Java VM ermöglicht:

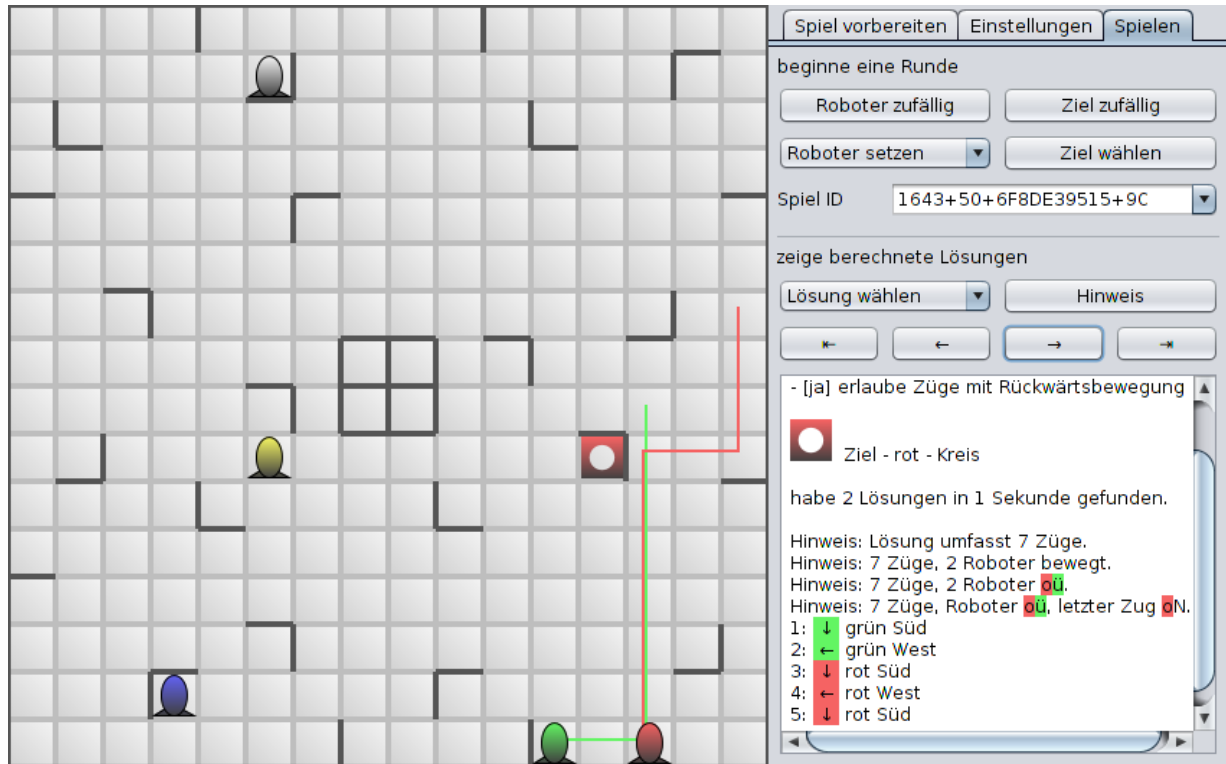
cd DriftingDroids_1.3.5

cd lib

java [Java VM Optionen] -jar driftingdroids.jar

Spielen





Das Programm startet im Modus **Spielen**. Du kannst eine (weitere) **Runde beginnen**, indem Du die Positionen der Roboter und / oder des Ziels neu setzt.



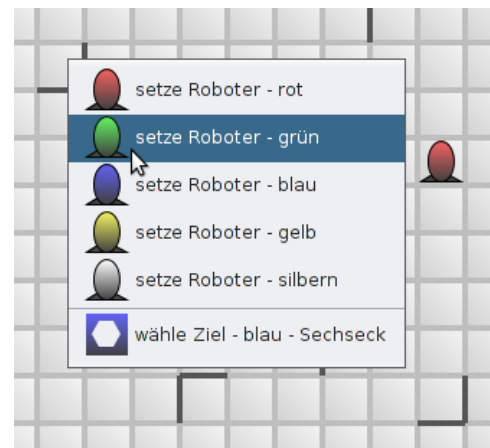
- **Roboter zufällig** Das Programm setzt alle Roboter auf zufällig gewählte Positionen.
- **Roboter setzen** Wähle zuerst eine Farbe aus der Liste und klicke dann auf die neue Position des Roboters auf dem Spielplan. Wiederhole dies für alle zu setzenden Roboter.
- **Ziel zufällig** Das Programm wählt aus den vorhandenen Zielen eines per Zufall aus.
- **Ziel wählen** Klicke zuerst hier und dann auf eines der Ziele, die auf dem Spielplan angezeigt werden.
- **Spiel ID** Das Programm erzeugt eine eindeutige ID für jede Spielkonfiguration, welche aus der Anordnung der 4 Teile des Spielplans und den Positionen der Roboter und des Ziels besteht. Wähle eine ID aus der Liste oder tippe eine ID ein (kopieren & einfügen). Du kannst auch die ID eines interessanten Spielproblems kopieren und in eine Textdatei einfügen und abspeichern oder an einen Freund senden, der sie in seiner Kopie von DriftingDroids verwenden kann.

Der Lösungsalgorithmus arbeitet im Hintergrund wenn eine neue Runde begonnen wird (wie oben beschrieben) und zeigt seine Ergebnisse im Ausgabebereich rechts unten an.

Benutze die folgenden Eingaben, um die **berechneten Lösungen** anzusehen:

- **Lösung wählen** Für manche Spielkonfigurationen findet der Lösungsalgorithmus mehrere optimale Lösungen. Diese Liste von Lösungen ist sortiert gemäß der Einstellung **bevorzuge Lösung mit minimaler / maximaler Anzahl von Robotern in Bewegung** (siehe Kapitel Einstellungen). Die beste Lösung befindet sich ganz oben in der Liste und ist standardmäßig ausgewählt.
- **Hinweis** Erster Hinweis: wie viele Züge umfasst die Lösung. Klicke noch einmal für den zweiten Hinweis: wie viele Züge und wie viele bewegte Roboter hat die Lösung. Dritter Hinweis: wie viele Züge und welche bewegten Roboter. Vierter Hinweis: wie viele Züge, welche bewegten Roboter und was ist der letzte Zug der Lösung.
-  alle gezeigten Züge rückgängig machen.
-  den letzten gezeigten Zug rückgängig machen.
-  zeige den nächsten Zug.
-  zeige alle Züge.

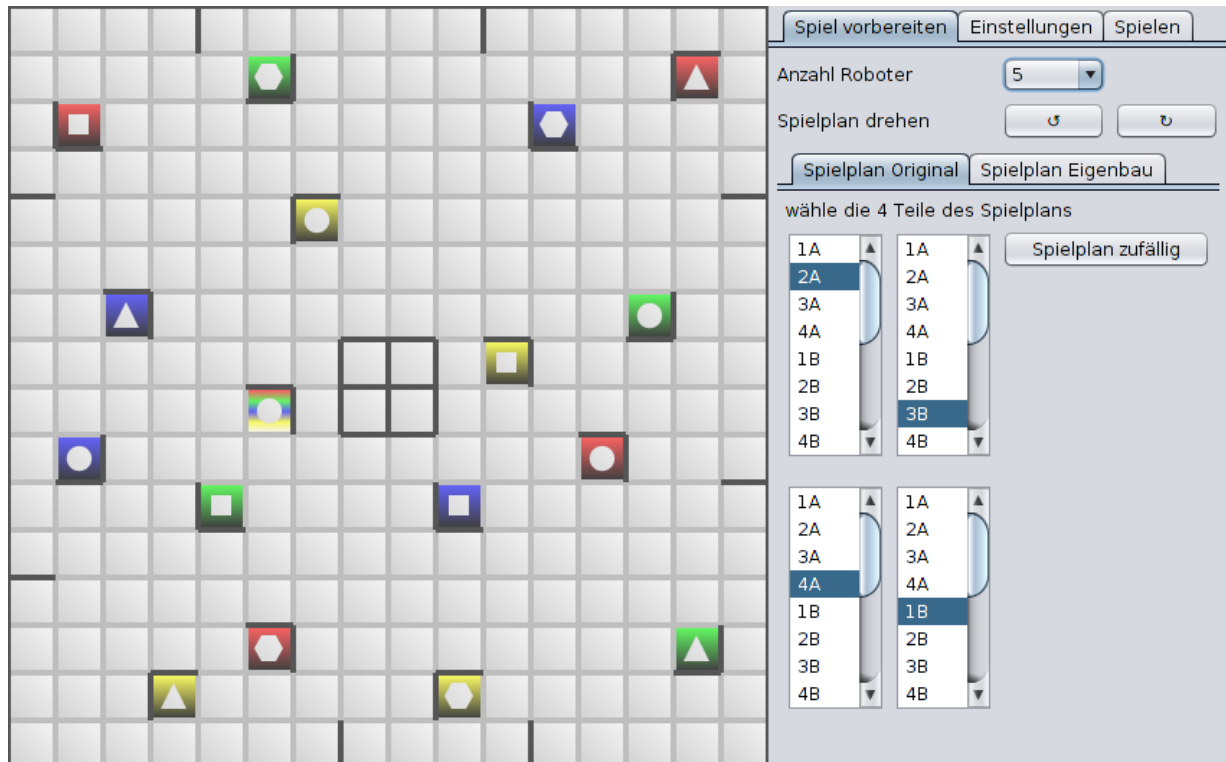
Die Funktionen **Roboter setzen** und **Ziel wählen** lassen sich auch mittels **Kontextmenü** ausführen. Dazu klickst Du mit der entsprechenden Maustaste (üblicherweise der rechten) auf die gewünschte Stelle auf dem Spielplan.



Die meisten der genannten Funktionen lassen sich auch mittels **Tastenkombinationen** (Hotkeys) ausführen. Diese werden im Tooltip angezeigt, wenn der Mauspfel über einem Bedienelement verweilt. (zum Beispiel **Hinweis** mit **Strg-H**)

Spiel vorbereiten

Hier kannst Du die 4 Teile des Spielplans zusammensetzen und die Anzahl der verwendeten Roboter ändern. Der Spielplan auf der linken Seite zeigt alle Ziele an, damit Du die Einzelteile des Spielplans leichter erkennst.



Wähle die **Anzahl der Roboter**, die auf dem Spielplan stehen sollen. Die erste Ausgabe von Rasende Roboter enthielt 4 Roboter (rot, grün, blau, gelb), während neuere Ausgaben von Ricochet Robots noch einen fünften Roboter (silbern) mitbringen.

Der aktuelle Spielplan kann durch Klick auf  oder  jeweils um 90 Grad gedreht werden.

Spielplan Original

Die Teile des Spielplans in DriftingDroids entsprechen denen des Brettspiels Ricochet Robots. Sie wurden ganz einfach durchnummeriert (1, 2, 3, 4) und die Vorder- und Rückseiten wurden A, B, C, D genannt. (siehe Fotos in Verzeichnis doc)

Mit Klick auf **Spielplan zufällig** setzt das Programm den Spielplan zusammen: es werden alle 4 verschiedenen Teile verwendet, sie werden auf zufällige Positionen gelegt mit einer zufälligen Seite (A, B, C oder D) nach oben.

Du kannst die Teile des Spielplans auch manuell anordnen, indem Du aus den vier Listen jeweils einen Eintrag 1A ... 4D auswählst. Das Programm beschränkt die Auswahl dabei nicht, so dass Du zum Beispiel auch einen Spielplan aus vier gleichen Teilen bauen könntest.

Spielplan Eigenbau

Hier kannst Du einen Spielplan erstellen, den es so bei Ricochet Robots nicht gibt. Jede Wand und jedes Ziel können einzeln gesetzt oder entfernt werden. Dazu wird einfach mit der Maus auf die gewünschten Stellen des Spielplans geklickt.

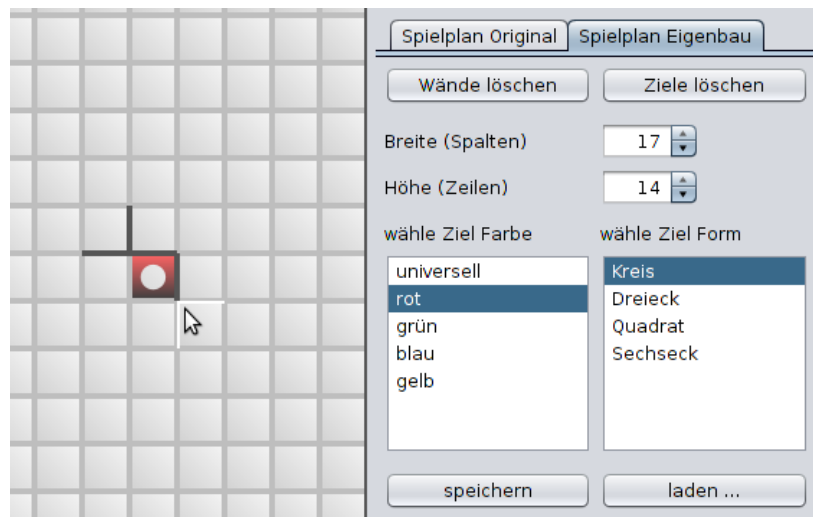
Durch Klick mit der **ersten Maustaste** (üblicherweise links) werden die weiß hervorgehobenen **Wände gesetzt** (1 bis 2) oder, falls sich der Mausfeil über der Mitte eines Feldes befindet und deshalb keine Wand weiß hervorgehoben ist, das aktuell **ausgewählte Ziel platziert**. (Farbe und Form des Ziels wird in den zwei Listen festgelegt)

Durch Klick mit der **zweiten Maustaste** (üblicherweise rechts) oder mit **Umschalttaste + erste Maustaste** werden die weiß hervorgehobenen **Wände** (1 bis 2) oder das **Ziel** unter dem Mausfeil **gelöscht**.

Mittels **wände löschen** und **ziele löschen** kann der Spielplan komplett geleert werden.

Die **Breite (Spalten)** und die **Höhe (Zeilen)** des Spielplans können frei eingestellt werden, jedoch beträgt die maximale Größe des Spielplans **4096** Felder, also zum Beispiel 64 x 64 oder 100 x 40 Felder.

Hinweis: Die Außenwände an den Rändern des Spielplans können nicht entfernt werden.



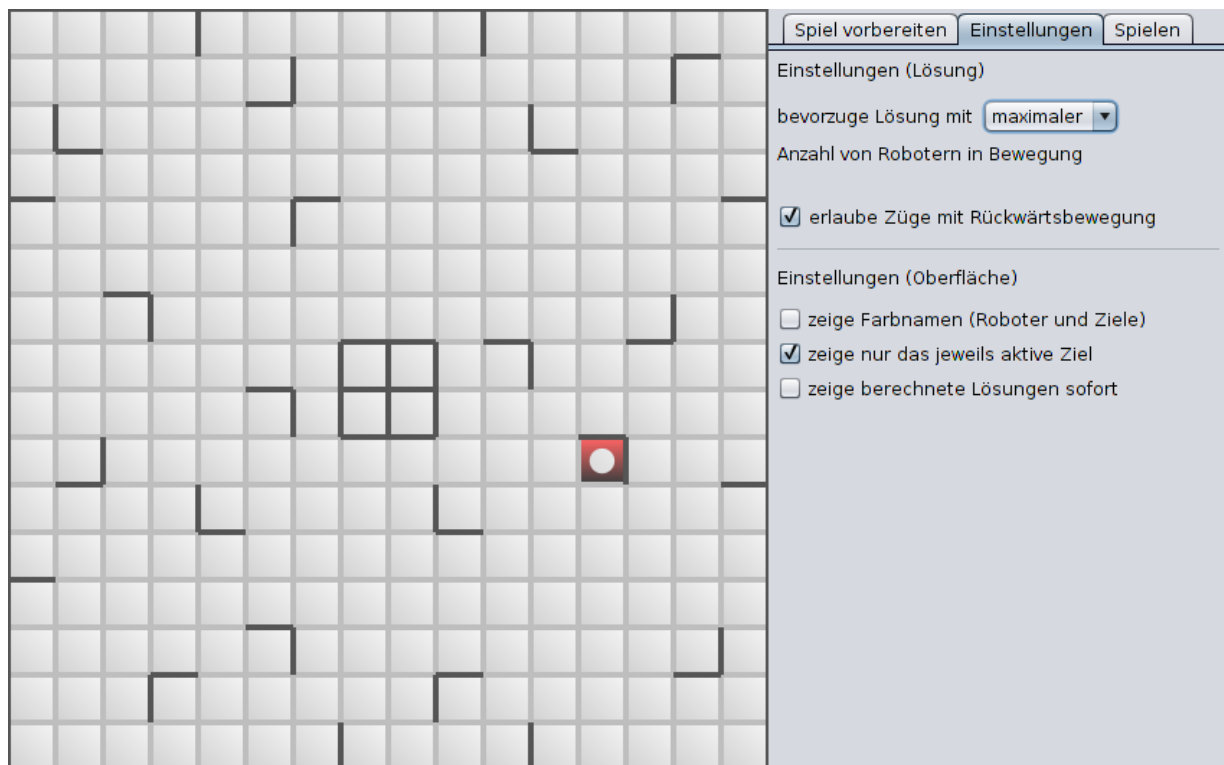
Durch Klick auf **speichern** wird der aktuelle Spielplan als **Text** in die **Zwischenablage** (Clipboard) **kopiert**, den Du dann in ein Textdokument oder eine E-Mail einfügen und so speichern oder verschicken kannst.

Nach Klick auf **laden...** öffnet sich ein Eingabefenster, in das Du einen als **Text** gespeicherten Spielplan aus der Zwischenablage **einfügen** kannst.


Einstellungen

Der Lösungsalgorithmus besitzt zwei Einstellungen:

- **bevorzuge Lösung mit minimaler / maximaler Anzahl von Robotern in Bewegung**
Nachdem alle optimalen Lösungen gefunden wurden, wird die Liste der Lösungen entsprechend sortiert. (betrifft die Liste **Lösung wählen** auf der Seite **Spiele**)
- **erlaube Züge mit Rückwärtsbewegung** Manche Spieler spielen Ricochet Robots nach strengerer Regeln, nach denen die Roboter nicht in die Richtung zurück bewegt werden dürfen, aus der sie in einem früheren Zug gekommen sind. Mittels DriftingDroids kann man zeigen, dass manche Lösungen durch solche Rückwärts-Züge kürzer werden. (Beispiel: Spiel ID **0765+42+2E21BD0F+93** hat 24-Züge optimale Lösungen mit Rückwärts-Zügen und 26-Züge optimale Lösungen ohne Rückwärts-Züge) Siehe auch Diskussionen in Online-Foren wie diesem: <http://boardgamegeek.com/forum/22741/ricochet-robots/rules>



Für die Oberfläche des Programms gibt es drei Einstellungen:

- **zeige Farbnamen** Wenn aktiv, dann werden die Roboter und Ziele mit Buchstaben markiert, die zur jeweiligen Farbe gehören (**r**ot, **g**rün, **b**lau, **g**elb, **s**ilbern).
- **zeige nur das jeweils aktive Ziel** Wenn aktiv, dann wird immer nur ein Ziel auf dem Spielplan angezeigt.
- **zeige berechnete Lösungen sofort** Wenn aktiv, dann werden die berechneten Lösungen automatisch angezeigt, so als hättest Du auf  (zeige alle Züge) geklickt.

Danksagungen

Dieses Programm verwendet die folgenden Java Bibliotheken:

- DesignGridLayout <http://designgridlayout.java.net/>
- AppStart <http://code.google.com/p/appstart/>

Diese Werkzeuge werden verwendet um dieses Programm zu erstellen:

- Eclipse <http://www.eclipse.org/>
- ProGuard <http://proguard.sourceforge.net/>

Dank für Ideen und Inspiration geht an:

- meine Schwester Doro, die mir das Spiel Ricochet Robots gezeigt hat. Sie hat auch die Übersetzung nach Esperanto beigetragen.
- alle Leute die Artikel, wissenschaftliche Aufsätze oder Computerprogramme zu Ricochet Robots veröffentlich haben.
- David Hansel für den Quellcode seines schnellen Lösungsprogramms.
- Michael Fogleman für sein Programm mit neuen Ideen für einen besseren Lösungsalgorithmus.
- Jay (x11x@pisem.net) für die Übersetzung ins Russische.
- Raphaël Huck für die Übersetzung ins Französische.

Lizenz

DriftingDroids - yet another Ricochet Robots solver program.
Copyright (C) 2011-2015 Michael Henke <smack42@googlemail.com>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.