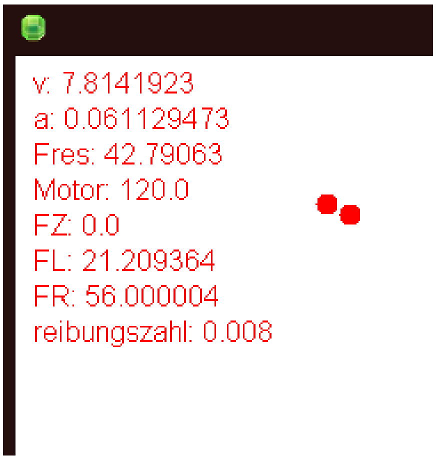


Modellieren von physikalischen Kräften

Von Johan Stettler

Betreut von Stefan Rothe

Gymnasium Kirchenfeld Bern

Abteillung MN

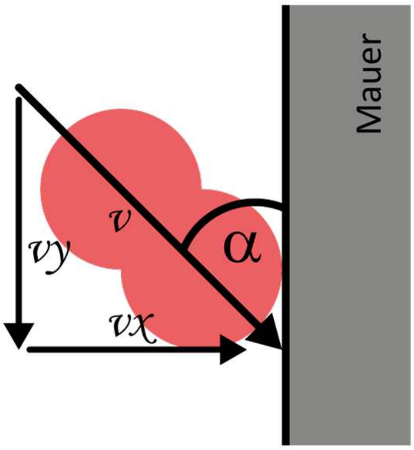
2014

```
//Anpassen der Positionen
xPos = xPos + vx * Time + Fzx * Time;
yPos = yPos + vy * Time + Fzy * Time;
controllX  = controllX + vx * Time + Fzx * Time;
controllY  = controllY + vy * Time + Fzy * Time;
linksX = xPos + deltaY * rcar;
rechtsX = xPos - deltaY * rcar;
linksY = yPos - deltaX * rcar;
rechtsY = yPos + deltaX * rcar;

//Reverse
if(reverse && v <= 0){
vx = Time * vr * deltaX;
vy = Time * vr * deltaY;
xPos = xPos + vx;
yPos = yPos + vy;
controllX  = controllX + vx;
controllY  = controllY + vy;

if(rightturn){
    controllX = controllX - xPos;
    controllY = controllY - yPos;
    zwischenX = controllX;
    controllX = (float) cos(thetaReverse) * controllX + (float) sin(thetaReverse) * controllY;
    controllY = (float) cos(thetaReverse) * controllY - (float) sin(thetaReverse) * zwischenX;
    controllX = controllX + xPos;
    controllY = controllY + yPos;
}

if(leftturn){
    controllX = controllX - xPos;
    controllY = controllY - yPos;
    zwischenX = controllX;
    controllX = (float) cos(thetaReverse) * controllX - (float) sin(thetaReverse) * controllY;
    controllY = (float) sin(thetaReverse) * zwischenX + (float) cos(thetaReverse) * controllY;
    controllX = controllX + xPos;
    controllY = controllY + yPos;
}
```



Inhaltsverzeichnis

Einleitung

In der Wissenschaft wie auch in der Wirtschaft wird viel mit Modellen gearbeitet. Sie dienen dem Zweck, gewisse Sachverhalte zu verdeutlichen. Es gibt viele verschiedene Arten von Modellen, z.B. die Darstellung eines Moleküls, der Bauplan eines Gebäudes oder ein Wettermodell. Sie können auf verschiedene Weise dargestellt werden: gezeichnet auf Papier, angefertigt aus Kunststoff oder simuliert an einem Computer.

Als Maturaarbeit habe ich mich entschieden, ein Modell einer zweidimensionalen Autosimulation am Computer zu realisieren. In einem ersten Schritt mache ich mich vertraut mit den physikalischen Grundlagen der Kräfte, welchen ein Fahrzeug beim Fahren unterliegt. Dann schaue ich, wie und welche der einzelnen Kraftkomponenten, Formeln und Konstanten anpasst, weglassen und neu hinzufügt werden müssen, damit diese simulierten Kräfte im Programm umgesetzt werden können. Damit versuche ich herauszufinden, wie nahe ich an eine realitätsgetreue Simulation kommen kann. Ich werde in meiner Maturaarbeit jede physikalische Komponente einzeln, Schritt für Schritt, durcharbeiten. Jede dieser Komponenten wird kurz vorgestellt. Danach beschreibe ich, wie diese schliesslich im Programm eingeführt werden.

Für einen Schüler der Abteilung MN, mit Schwerpunktfach Physik und angewandte Mathematik und mit Ergänzungsfach Informatik, ist dies eine optimale Arbeit. Ich bin sehr interessiert an der Mathematik und Physik, besonders an deren Problemlöseverfahren, welches das zentrale Thema meiner Arbeit ist. Ausserdem wollte ich schon immer an einem grösseren Projekt im Bereich der Programmierung arbeiten, und dies in Kombination mit anderen Fächern, als eine interdisziplinäre Arbeit, was das Ganze noch spannender macht. Deswegen habe ich mich für diese Maturaarbeit entschieden.

Geschwindigkeit

Die Geschwindigkeit, oft mit $\vec{v}$ abgekürzt, ist das Grundlegendste bei einer Autosimulation. Sie steht für die Veränderung der Position in einem Zeitintervall in einer bestimmten Richtung. Da es sich bei $\vec{v}$ um einen Vektor handelt, braucht es dafür im Programm eine Richtungsangabe und einen Betrag. Für den Zeitparameter werden die Berechnungen in regelmässigen Zeitabständen durchgeführt. Dafür gibt es im Programm eine Methode, die den Zeitparameter, hier τ, liefert. Im Programm wurde noch eine Konstante von 30 hinzumultipliziert. Diese Konstante ist notwendig, da nicht alle Komponenten SI-Einheiten besitzen, was zu falschen Einheiten und Werten, und dies wiederum zu einer fehlerhaften Simulation führen würde. Dieser Wert wurde durch Ausprobieren im Simulator selbst bestimmt.

Neben τ sind am Anfang noch zwei Punkte im Programm gegeben; der eine soll die Front des Fahrzeuges darstellen, abgekürzt mit Index F, der andere das Heck, abgekürzt mit Index H. Dazu gibt es eine Geschwindigkeit $\vec{v}$, die vorerst konstant bleibt. Diese wird erst mit der Beschleunigung variierbar. Das Fahrzeug sollte sich nun in Richtung der Front fortbewegen. Dafür werden die folgenden Deltas aus den Koordinaten der beiden Punkte berechnet:

$$\Delta x = x_F - x_H \qquad \Delta y = y_F - y_H$$

Diese Deltas dienen zur Orientierung. Falls z.B. das Δx negativ ist, so bewegt sich das Fahrzeug auf der x-Achse nach links, sonst nach rechts. Analog beim Δy: Ist es negativ, bewegt sich das Auto nach oben, sonst nach unten.

Die Bewegung erfolgt nun in zwei Schritten. $\vec{v}$ wird in $\overrightarrow{vx}$ und $\overrightarrow{vy}$ aufgeteilt, damit die Translation einfacher zu berechnen ist. Durch das so genannte Superpositionsprinzip kann ein Vektor in andere Vektoren aufgeteilt werden (DMK DPK DCK, 2011, S. 34).

$$\vec{v} = \overrightarrow{vx} + \overrightarrow{vy}$$

Ein Beispiel: Würde $\vec{v}$ 5 betragen, darf sich das Auto in der gegebenen Richtung nur um 5 verschieben. Dies würde, laut dem Satz des Pythagoras, in einem Idealfall einem $\overrightarrow{vx}$ von 3 und einem $\overrightarrow{vy}$ von 4 entsprechen (Abb. 1). In den meisten Fällen sind $\overrightarrow{vx}$ und $\overrightarrow{vy}$ nicht ganzzahlig, sondern haben einen Betrag einer rationalen Zahl. Damit diese in jedem Fall berechnet werden können, wird eine Konstante z eingeführt, welche den Abstand zwischen den beiden vorherigen Punkten darstellt (Abb. 2).

$$z = \sqrt{\Delta x^2 + \Delta y^2}$$

Daraus ergeben sich nun zwei rechtwinklige und ähnliche Dreiecke. Weil die Dreiecke durch ihre Winkel ähnlich sind, gelten die gleichen Verhältnisse bei den Seitenlängen (DMK DPK DCK, 2011, S. 88).

$$\Delta x \div z = |vx| \div |v| \qquad \Delta y \div z = |vy| \div |v|$$

Und daraus ergibt sich:

$$|vx| = \Delta x \div z \cdot |v| \qquad |vy| = \Delta y \div z \cdot |v|$$

Zum Schluss werden nun die Koordinaten der beiden Punkte entsprechend angepasst. Hierbei darf die Zeitkomponente nicht fehlen. Die zurückgelegte Strecke bei konstanter Geschwindigkeit berechnet man wie folgt (DMK DPK DCK, 2011, S. 155):

$$\Delta s = \vec{v} \cdot \Delta t$$

Und daraus ergibt sich im Programm:

$$x_F := x_F + \overrightarrow{vx} \cdot \tau \qquad y_F := y_F + \overrightarrow{vy} \cdot \tau$$

$$x_H := x_H + \overrightarrow{vx} \cdot \tau \qquad y_H := y_H + \overrightarrow{vy} \cdot \tau$$

Abb.1

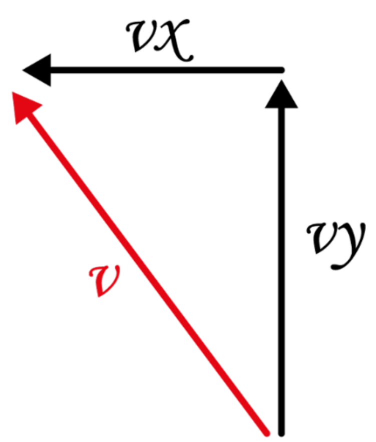
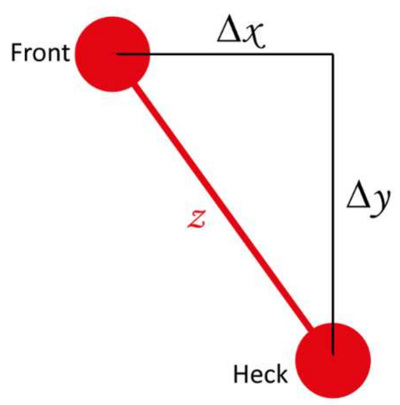


Abb.2



Beschleunigung

Beschleunigung, oft mit $\vec{a}$ abgekürzt, ist die Veränderung der Geschwindigkeit pro Zeit, oder anders ausgedrückt: Gibt man Gas, nimmt die Geschwindigkeit zu. Ohne Beschleunigung kommt das Fahrzeug wegen der Reibungskräfte einmal zum Stehen. Dies wird später behandelt. Durch die Beschleunigung ist die Geschwindigkeit auch nicht mehr konstant. Diese muss nun auch regelmässig neu berechnet werden. So wird die Beschleunigung in der Simulation eingeführt:

$$\vec{v} := \vec{v} + \vec{a} \cdot \tau$$

Auch hier ist der Zeitparameter wieder notwendig, weil die Geschwindigkeit abhängig von der Beschleunigung in einem bestimmten Zeitintervall ist. Da es sich nicht um eine konstante Beschleunigung handelt, müssen τ und $\vec{a}$ immer wieder neu berechnet werden. τ ist schon existent und $\vec{a}$ muss noch gefunden werden. Dafür wird hier das zweite newtonsche Gesetz verwendet (frustfrei-lernen.de, 2014). Es beschreibt den Zusammenhang zwischen der resultierenden Kraft, der Masse und der Beschleunigung eines Körpers.

$$\overrightarrow{F_{RES}} = m \cdot \vec{a} \quad => \quad \overrightarrow{F_{RES}} \div \mathrm{m} = \vec{a}$$

m steht für die Masse des Körpers oder in diesem Fall für die Masse des Fahrzeuges. Ihr wurde ein Betrag von 700kg zugewiesen. Die durchschnittliche Masse eines PKWs ist ungefähr 1400kg (statista, 2014), also das Doppelte der gegebenen Masse. Auch hier wurde eine Konstante von ½ hinzumultipliziert, um das Gleichgewicht der Grössen auszubalancieren. $\overrightarrow{F_{RES}}$ ist die resultierende Kraft, die sich aus allen wirkenden Kräften zusammensetzt.

Jeder Körper braucht eine Kraft, welche ihn in Bewegung versetzt. Dies beschreibt das erste newtonsche Gesetz, das so genannte Trägheitsgesetz. Solange keine äussere Kräfte auf den Körper wirken, bleiben seine Richtung und Geschwindigkeit unverändert. (frustfrei-lernen.de, 2014). Dies kann auch bedeuten, dass $\vec{v}$ einfach null bleibt. Momentan gib es nur eine Kraft, und zwar die Motorenkraft $\overrightarrow{F_M}$, welche das Fahrzeug nach vorne antreibt (Auch diese gilt als äussere Kraft).

$$\overrightarrow{F_{RES}} = \overrightarrow{F_M} \qquad \overrightarrow{F_M} = [0\,;\ 120]\ \mathrm{Newton}$$

Da es sehr aufwändig ist, die ganze Mechanik eines Motors zu simulieren, wurde ihm einfach ein Betrag von max. 120 Newton gegeben. Für diese Simulation ist diese Kraft verhältnismässig am geeignetsten, um die Reibungskräfte zu kompensieren und um die Geschwindigkeit in einem vernünftigen Mass zu halten. So kommt nun die Beschleunigung zum Zuge: Wird Gas gegeben, nimmt der Betrag von $\overrightarrow{F_M}$ zu, sonst nimmt er ab. Dies beeinflusst dann $\overrightarrow{F_{RES}}$, dies wiederum $\vec{a}$ und dies wiederum $\vec{v}$.

Reibungskräfte

Wie schon im vorangegangenen Kapitel thematisiert, könnte ein Fahrzeug theoretisch, laut dem Trägheitsprinzip, durch eine Anfangsbeschleunigung, mit konstanter Geschwindigkeit, sich unendlich lange bewegen, ohne weiter zu beschleunigen, solange keine äusseren Kräfte auf den Wagen einwirken. In der Praxis herrschen jedoch solche äusseren Kräfte, und zwar die Reibungskräfte. Diese sorgen dafür, dass ein Fahrzeug zum Stillstand kommt, wenn man diesen nicht mit einer anderen Kraft entgegenwirkt. Unterschieden wird in dieser Arbeit zwischen dem Luftwiderstand, welcher später behandelt wird, und den Reibungskräften $\overrightarrow{F_R}$, welche nochmals unterteilt sind in Haft- und Rollreibung.

Im Normalfall ist das Fahrzeug am Rollen, und somit wirkt dann auch die Rollreibung. Aus der Physik kommt die Formel (DMK DPK DCK, 2011, S. 157):

$$\overrightarrow{F_R} = \overrightarrow{F_N} \cdot fr$$

fr ist der Rollreibungskoeffizient (eine Konstante aus der Formelsammlung), mit dem die Rollreibung berechnet wird. $\overrightarrow{F_N}$ ist die Normalkraft. Sie ist eine Art tragende Kraft. Wenn man zum Beispiel eine Tasse Kaffee auf einen Tisch stellt, trägt der Tisch sozusagen die Tasse. Diese wird von der Gravitationskraft zum Erdmittelpunkt bewegt, doch die Tasse scheint ja nicht durch den Tisch durchzubrechen. Es muss also etwas dieser Gewichtskraft entgegenwirken, und dies ist die Normalkraft (DITTMAR-ILGEN, 2014).

$$\overrightarrow{F_N} = \overrightarrow{F_G} \cdot \cos\alpha$$

$\overrightarrow{F_G}$ ist die Gravitationskraft, die sich wie folgt berechnen lässt (DMK DPK DCK, 2011, S. 157):

$$\overrightarrow{F_G} = m \cdot g$$

α ist der Neigungswinkel der Ebene. Ist die Ebene geneigt, so ist $\overrightarrow{F_N} < \overrightarrow{F_G}$. Ist die Ebene um 90° geneigt, so ist $\overrightarrow{F_N}$ null und der Gegenstand wird nicht mehr getragen und fällt deswegen. Im Idealfall ist die tragende Ebene nicht geneigt, also α = 0°, dann gilt:

$$\overrightarrow{F_N} = \overrightarrow{F_G}$$

Und daraus folgt:

$$\overrightarrow{F_R} = m \cdot g \cdot fr$$

Die Rollreibungskraft ist die einzige, die in der Simulation nicht angepasst, sondern mit den richtigen Konstanten berechnet worden ist. Sie ist der Kern der Kräfte, nach welcher alle anderen Kraftkomponenten angepasst worden sind, so wie auch der Motor.

Nun gilt es, diese Rollreibung zu kompensieren, und zwar mit dem Motor, um eine Beschleunigung zu bewirken. Neu:

$$\overrightarrow{F_{RES}} = \overrightarrow{F_M} - \overrightarrow{F_R}$$

Allerding gibt es da ein kleines Problem: Würde der Betrag negativ ausfallen, so würde sich das Fahrzeug rückwärts bewegen, was in der Natur ja nicht geschieht. Das Problem wurde wie folgt gelöst: Wenn $\overrightarrow{F_{RES}}$ negativ ist, so bleibt $\vec{v}$ einfach null.

Leider dauert es eine geringe Zeit, bis $\overrightarrow{F_M}$ $\overrightarrow{F_R}$ kompensieren kann, damit das Fahrzeug in Bewegung kommt. Dies ist ein falscher Effekt, der in der Physik nicht existiert, jedoch einen anderen Existenten ersetzen kann, der nicht simulierbar schien.

Eigentlich wechselt ein Fahrzeug, wenn es ins Rollen kommt, zwischen der Haftreibung und der Rollreibung. Die Haftreibung lässt sich genau gleich berechnen wie die Rollreibung, doch besitzt sie einen viel grösseren Reibungskoeffizienten und ist demnach auch viel grösser. Damit der Wechsel in die Rollreibung von statten geht, muss die antreibende Kraft, in diesem Fall der Motor, die Haftreibung überwinden, sprich gleich gross oder grösser im Betrag sein. Erst dann rollt das Fahrzeug und die Rollreibung setzt ein.

Dieser Wechsel konnte nicht simulieren werden. Es wurde versucht, die Haftreibung, beziehungsweise ihren Reibungskoeffizienten, so anzupassen, dass der Motor die Haftreibung zu überwinden vermag, aber der Motor nicht zu stark ist im Verhältnis mit der Rollreibung, damit das Fahrzeug noch eine vernünftige Geschwindigkeit besitzt und nicht zu schnell fährt. Da ohnehin schon ein idealisierter Motor existiert, wie vorhin beschrieben, war dies nur möglich, wenn die Haft- und die Rollreibung nahezu gleich gross waren, und deswegen konnte die Haftreibung genauso gut weggelassen werden. Der Wechsel würde sowieso nur einen kurzen Augenblick dauern. Im Gegenzug sorgt die negative resultierende Kraft, die man erst ins Positive drehen muss, für einen ähnlichen Effekt.

Die Haftreibung spielt dafür wieder beim Bremsen eine Rolle. Wird die Bremse betätigt, werden die Räder beim Fahrzeug blockiert und die Rollreibung wird durch Haftreibung ersetzt, beziehungsweise der Rollreibungskoeffizient fr durch den Haftreibungskoeffizient fh. fh ist in der Regel hundert Mal grösser, doch damit die Verhältnisse in der Simulation stimmen, wurde fh nur ein doppelt so grosser Betrag zugewiesen als für fr.

Nun kommt noch der Luftwiderstand. Der Luftwiderstand $\overrightarrow{F_L}$ ist ein recht einfaches Thema. $\overrightarrow{F_L}$ sorgt für ein Gleichgewicht, damit eine bestimmte Maximalgeschwindigkeit nicht überboten werden kann, da $\overrightarrow{F_L}$ selber von $\vec{v}$ abhängt, und zwar quadratisch. Berechnet wird dieser wie folgt (DMK DPK DCK, 2011, S. 165):

$$\overrightarrow{F_L} = \frac{1}{2} \cdot A \cdot \rho_L \cdot cw \cdot \vec{v}^2$$

A ist die Stirnfläche des Fahrzeuges und beträgt ca. $1.8m^2$, ρ_L ist die Dichte von Luft, welche das Medium der Umgebung ist, und cw ist eine Konstante, die von der Form der Stirnfläche abhängt und ca. 0.32 beträgt (kfz-tech, 2014).

Neu:

$$\overrightarrow{F_{Res}} = \overrightarrow{F_M} - \overrightarrow{F_R} - \overrightarrow{F_L}$$

Zentripetalkraft

Die Zentripetalkraft $\overrightarrow{F_z}$ ist die Kraft, welche für die Kreisbewegung des Fahrzeuges in einer Kurve verantwortlich ist.

$$\overrightarrow{F_Z} = \frac{m \cdot v^2}{r}$$

Die Grösse r ist der Radius der Kreisbahn (Abb. 3). Die Kreisbahn resultiert aus den Vektoren $\vec{v}\ und\ \overrightarrow{F_Z}$. $\overrightarrow{F_Z}$ steht immer senkrecht zu $\vec{v}$ und $\vec{v}$ ist immer tangential zum Kreis. So resultieren die Kräfte genau in diese Kreisbahn (DMK DPK DCK, 2011, S. 159).

Abb.3

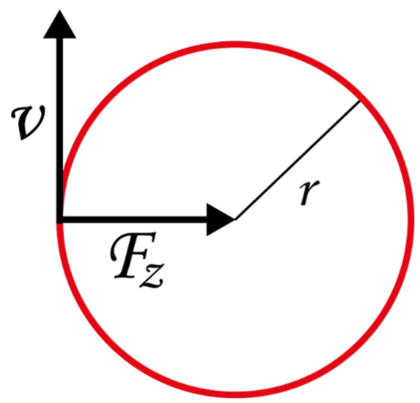
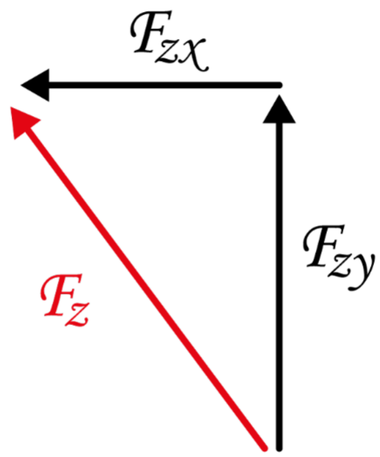


Abb.4



Der Betrag für $\overrightarrow{F_Z}$ ist schnell und einfach berechnet, doch auch hier gibt es wieder eine Anpassung vorzunehmen. Der Betrag sollte nicht zu gross werden. Darum wurde der Radius als Konstante von 200000 festgelegt, damit die Kraft in einem vernünftigen Mass bleibt. Diese Konstante ist notwendig, da es sich wieder um das Problem der Grössen und Einheiten handelt, wie schon im ersten Kapitel erklärt worden ist. Die 200000 wurden deswegen zugewiesen, da sich diese Konstante beim Ausprobieren im Simulator am besten bewährt hat. Der Radius ist aber nur im Betrag konstant, doch in der Bewegung variiert er natürlich. So sind kleinere und grössere Kurven möglich. Dies hängt nun von der Geschwindigkeit des Fahrzeuges ab.

$\overrightarrow{F_Z}$ kann auch wie $\vec{v}$ irgendwie im Raum ausgerichtet sein und wird auch in zwei neue Vektoren aufgeteilt, $\overrightarrow{F_{ZX}}$ und $\overrightarrow{F_{ZY}}$ (Abb. 4). Auch hier sind die beiden Dreiecke wieder ähnlich, und so können die Beträge für die einzelnen Vektoren berechnet werden:

$$|F_{ZX}| = |vy| \div |v| \cdot |F_Z| \qquad |F_{ZY}| = |vx| \div |v| \cdot |F_Z|$$

Jetzt stimmen die Beträge für die beiden Vektoren. Nun muss nur noch die Richtung stimmen. Falls es sich um eine Rechtsdrehung handelt, muss sich das Vorzeichen von $\overrightarrow{F_{zx}}$ ändern, bei einer Linksdrehung das Vorzeichen von $\overrightarrow{F_{zy}}$ (Abb. 5 und 6). Wichtig zu merken ist, dass sich $\overrightarrow{F_{zx}}$ aus $\overrightarrow{vy}$ berechnen lässt und $\overrightarrow{F_{zy}}\ aus\ \overrightarrow{vx}$.

Abb.5

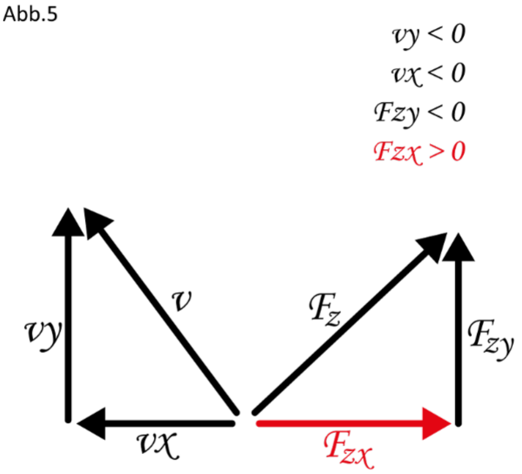
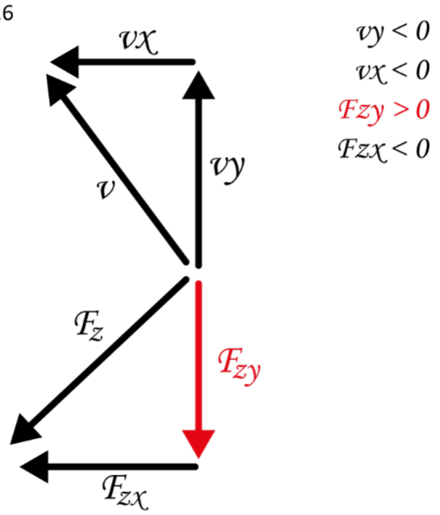


Abb.6



Nun bewegt sich das Fahrzeug nach links und nach rechts, aber es braucht noch eine Drehmatrix, mit welcher der Frontkreis um den Heckkreis rotiert wird, damit sich auch eine Kreisbewegung ergibt. So ist die ganze Bewegung in drei Teile aufgeteilt: Translation durch $\vec{v}$, Translation durch $\overrightarrow{F_Z}$ und Rotation vom Frontkreis. Die ersten beiden Schritte sind in einem zusammengefasst worden:

$$x_F := x_F + \overrightarrow{vx} + \overrightarrow{F_{ZX}} \qquad y_F := y_F + \overrightarrow{vy} + \overrightarrow{F_{ZY}}$$

$$x_H := x_H + \overrightarrow{vx} + \overrightarrow{F_{ZX}} \qquad y_H := y_H + \overrightarrow{vy} + \overrightarrow{F_{ZY}}$$

Für die Rotation wird eine Drehmatrix benötigt. Bekannt ist allerdings nur eine Drehung um den Nullpunkt. So muss der Frontpunkt zuerst um die Position des Hecks zum Nullpunkt verschoben werden, rotieren und dann zurückspringen. Dafür benötigt es noch einen Winkel θ. θ wurde ein Wert von 2.4 (im Bogenmass) zugewiesen. Im Simulator verhält sich dieser Wert am realistischsten von der Bewegung her und deswegen wurde dieser genommen.

In einer Formelsammlung findet man die Drehmatrizen (DMK DPK DCK, 2011, S. 113):

Drehung nach rechts:

$$\begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} := \begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} - \begin{pmatrix} Heck\ x \\ Heck\ y \end{pmatrix}$$

$$\begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} := \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix}$$

$$\begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} := \begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} + \begin{pmatrix} Heck\ x \\ Heck\ y \end{pmatrix}$$

Drehung nach links:

$$\begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} := \begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} - \begin{pmatrix} Heck\ x \\ Heck\ y \end{pmatrix}$$

$$\begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} := \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix}$$

$$\begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} := \begin{pmatrix} Front\ x \\ Front\ y \end{pmatrix} + \begin{pmatrix} Heck\ x \\ Heck\ y \end{pmatrix}$$

Nun sollte $\overrightarrow{F_Z}$ auch Teil von $\overrightarrow{F_{Res}}$ sein. Bislang waren alle Kräfte kollinear und konnten so einfach addiert werden. Für die Richtung kam es nur auf das Vorzeichen an. Da nun $\overrightarrow{F_Z}$ nicht auf der gleichen Linie liegt, darf diese eigentlich nicht hinzuaddiert werden. Da aber $\overrightarrow{F_Z}$ rechtwinklig zum Rest steht, wurde das Ganze durch einen Pythagoras berechnet.

$$|F_{RES}| = \sqrt{|(F_M - F_R - F_L)|^2 + |F_Z|^2}$$

$\overrightarrow{F_{Res}}$ darf so nur berechnet werden, wenn $\overrightarrow{F_M}$ > ($\overrightarrow{F_R} + \overrightarrow{F_L}$), sonst gäbe es eine resultierende Kraft und Beschleunigung, ohne dass der Motor überhaupt läuft. $\overrightarrow{F_{Res}}$ wäre positiv durch das Quadrieren und es gäbe dadurch eine nicht physikalisch logische Beschleunigung.

Nun muss noch ein Problem in der Kurvenbewegung gelöst werden. Da sich das Fahrzeug nicht mehr gerade in Radrichtung bewegt, gilt technisch die Rollreibung nicht mehr gleich. In der Realität verformen sich die Reifen durch die Kurvenbewegung mehr und die Rollreibung kann um bis zu 60% zunehmen (schuelerlexikon, 2014). Dafür braucht es nun extra eine Reibungskraft für die Kurvenbewegung und diese ist in der Simulation 160% von der Rollreibung.

Rückwärtsgang

Nun kommt ein kleines Zwischenkapitel, der Rückwärtsgang. Damit sich das Fahrzeug rückwärts bewegt, muss $\vec{v}$ entgegen der Fahrtrichtung zeigen. Dafür wird $\vec{v}$ einfach ein konstanter Betrag von (-1) zugewiesen. Durch das Verrechnen mit den anderen Komponenten werden einfach alle Geschwindigkeitsvektoren in ihrer Richtung gewendet und so fährt das Auto statt vorwärts, rückwärts.

Kollision

Wenn zwei Körper kollidieren, spricht man in der Physik von einem so genannten Stoss. Bei einem Stoss werden Impulse übertragen, und Energien werden übertragen und umgewandelt. Ein Impuls ist auch wieder ein Vektor, welcher oft mit $\vec{p}$ abkürzt wird. Der Impuls lässt sich von der Masse des Fahrzeuges wie auch von seiner Geschwindigkeit berechnen (DMK DPK DCK, 2011, S. 157):

$$\vec{p} = \vec{v} \cdot m$$

Bei den Fahrzeugen spielt vor allem die kinetische Energie (Bewegungsenergie) eine Rolle. Diese wird mit E abgekürzt. Die kinetische Energie lässt sich, ähnlich wie der Impuls, aus der Masse und der Geschwindigkeit des Fahrzeuges berechnen (DMK DPK DCK, 2011, S. 158):

$$E = \frac{1}{2} \cdot m \cdot \vec{v}^2$$

Wichtig zu merken ist, dass die vektorielle Summe der Impulse aller beteiligten Körper in einem abgeschlossenen System vor dem Stoss immer genau gleich gross ist wie nach dem Stoss. Dies nennt man den Impulserhaltungssatz (DMK DPK DCK, 2011, S. 157):

$$\vec{p} = \vec{p}'$$

Vor dem Stoss wird in der Gleichung ohne besonderes Merkmal dargestellt*, nach dem Stoss* wird dafür mit einem Apostroph gekennzeichnet.

Ein kleines Beispiel: Zwei Fahrzeuge kollidieren. Vor dem Stoss hat das erste den Impuls $\overrightarrow{p_1}$ und das zweite den Impuls $\overrightarrow{p_2}$. Nach dem Stoss haben die Fahrzeuge entsprechend $\overrightarrow{p_1}'$ und $\overrightarrow{p_2}'$. Nun gilt:

$$\vec{p} = \overrightarrow{p_1} + \overrightarrow{p_2} \qquad \vec{p}' = \overrightarrow{p_1}' + \overrightarrow{p_2}'$$

Und mit dem Impulserhaltungssatz gilt:

$$\vec{p} = \vec{p}' \qquad \overrightarrow{p_1} + \overrightarrow{p_2} = \overrightarrow{p_1}' + \overrightarrow{p_2}'$$

Jedoch sind $\overrightarrow{p_1}$ und $\overrightarrow{p_1}'$ nicht gleich gross. Deshalb kann man allgemein sagen:

$$\overrightarrow{p_1} \neq \overrightarrow{p_1}' \qquad \overrightarrow{p_2} \neq \overrightarrow{p_2}'$$

Dies ist natürlich verständlich. Wenn zwei Körper kollidieren, bewegt sich der einzelne vor dem Stoss anders, sowohl in seiner Richtung als auch in seiner Geschwindigkeit, als er es nach dem Stoss tut.

So bleibt also nur der gesamte Impuls erhalten. Bei der Energie ist dies anders. In der Physik braucht man zum Berechnen der Energie zwei extreme Fälle. Diese kommen zwar in Wirklichkeit nie vor. Jeder Stoss ist aber eine Mischform der beiden Extreme (Wikipedia, 2014). Diese Fälle nennt man vollkommen elastischer Stoss und vollkommen unelastischer Stoss. Bei jedem Stoss gilt folgende Gesetzmässigkeit (DMK DPK DCK, 2011, S. 159):

$$E = E' + U$$

U ist ein Teil der umgewandelten kinetischen Energie, z.B. Wärme oder Verformung der Körper, E‘ ist das, was von der Bewegung nach dem Stoss übrig geblieben ist. Beim vollkommen elastischen Stoss soll eben keine Energie verloren gehen. Dies bedeutet, das U gleich null ist. Die kinetische Energie bleibt also in der Summe aller beteiligten Körper erhalten (Wie beim Impuls, ist die Energie des einzelnen vor und nach dem Stoss nicht dieselbe.). Die Körper prallen dann voneinander weg. Bei einem vollkommen unelastischen Stoss wird das grösstmögliche von E in U umgewandelt, sodass die Körper aneinander haften bleiben.

Wie schon erwähnt, handelt es sich bei jedem Stoss um eine Mischform der beiden Fälle. Stösst jetzt aber ein sehr elastischer Springball an eine Wand, springt dieser mit der fast gleichen Geschwindigkeit zurück, die er vor dem Stoss hatte. So kann man als sehr gute Näherung den vollkommen elastischen Stoss zum Rechnen nehmen, da U in dem Falle fast null ist. Als ähnlich gute Näherung werden im Programm die Stösse als vollkommen unelastisch angesehen, um die Bewegung und Geschwindigkeiten der Fahrzeuge zu berechnen.

Die Energie spielt daher auch keine Rolle für die Berechnungen und aus diesem Grund benötigt es nur den Impulserhaltungssatz. Dafür werden wieder die vorherigen Formeln genommen und umgeformt:

$$\vec{p} = \vec{p}' \qquad \overrightarrow{p_1} + \overrightarrow{p_2} = \vec{p} \qquad \vec{p} = \vec{v} \cdot m$$

$$\Rightarrow \; m \cdot \overrightarrow{v_1} + m \cdot \overrightarrow{v_2} = 2 \cdot m \cdot \vec{v}'$$

Alle Fahrzeuge haben im Programm die gleiche Masse. Deswegen kann diese aus der Gleichung gekürzt werden. Nun werden die Impulse, wie es auch $\vec{v}$ ist, in X- und Y-Komponenten unterteilt. Die neue Geschwindigkeit beider Fahrzeuge lässt sich nun wie folgt berechnen:

$$\overrightarrow{vx} = \frac{\overrightarrow{vx_1} + \overrightarrow{vx_2}}{2} \qquad \overrightarrow{vy} = \frac{\overrightarrow{vy_1} + \overrightarrow{vy_2}}{2}$$

Falls bei der Kollision eine Wand beteiligt ist, wird $\overrightarrow{v_2}$ gleich $\overrightarrow{v_1}$ sein, einfach mit einem anderen Vorzeichen, um die Geschwindigkeit des Fahrzeuges aufzuheben, allerdings nur in einer Komponente. Kollidiert ein Fahrzeug mit einer Wand, welche östlich oder westlich orientiert ist, wird $\overrightarrow{vx_2}$ gleich minus $\overrightarrow{vx_1}$ gesetzt und auf diese Weise wird $\overrightarrow{vx}$ auf null gesetzt. Ist die Wand nördlich oder südlich ausgerichtet, wird $\overrightarrow{vy}$ auf null gesetzt, und zwar mit der gleichen Methode.

Zusätzlich muss der Motor bei einem Stoss auf null gesetzt werden, da sonst der Impulserhaltungssatz nicht mehr gilt. Dieser gilt nur, wenn es sich um ein abgeschlossenes System handelt (DMK DPK DCK, 2011, S. 157). Das bedeutet, dass keine anderen Kräfte oder zusätzliche Beschleunigung wirken dürfen, und der Motor sorgt eben für eine zusätzliche Beschleunigung. Darum wird dieser bei einem Stoss auf null gesetzt.

Für eine etwas realistischere Darstellung eines Stosses wurde das Programm noch etwas genauer bearbeitet. Wird ein Fahrzeug von der Seite getroffen, dreht dieses sich noch ein wenig in die entgegengesetzte Richtung. Dies wird mit einer Drehmatrix realisiert, wie im Kapitel der Zentripetalkraft beschrieben. Bei einem Frontalstoss wird die Drehung ausgelassen. Dazu muss das Programm wissen, wann es sich um eine Kollision handelt und ob diese frontal ist oder nicht.

Zuerst wird die Kollision mit einer Wand behandelt. Das Auto besteht aus zwei Kreisen und wenn einer der Kreise überlappt, also mit seiner Position plus dem Kreisradius hinter oder gleich der Position der Wand ist, handelt es sich um eine Kollision. Es ist möglich, dass ein Fahrzeug hinter die Mauer kommt, da die Geschwindigkeit grösser sein kann als der Abstand zur Mauer und da die Abfrage der Kollision in einem unabhängigen Schritt erfolgt wie die Translation. Deswegen braucht es noch eine Korrektur. Wenn also ein Kreis überlappt, wird seine Position minus seinem Radius gleich der Position der Mauer gesetzt. Das Fahrzeug berührt dann korrekterweise die Wand und der Stoss kann ausgeführt werden. (Abb. 7).

Abb.7

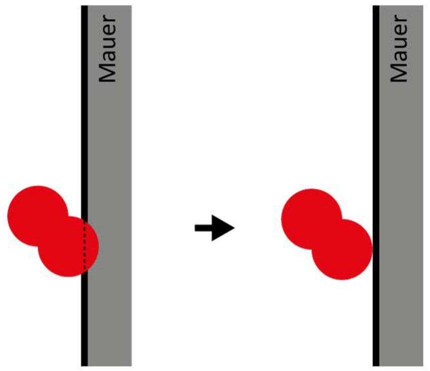
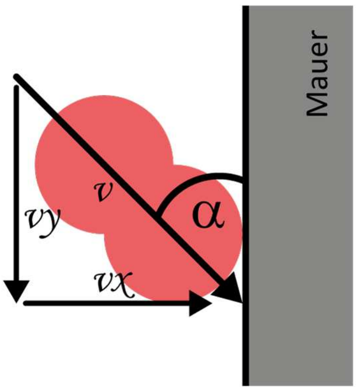


Abb.8



Jetzt muss noch kontrolliert werden, ob der Stoss frontal ist oder nicht. Ist der Winkel α zwischen 81° und 99°, betrachtet dies das Programm als eine frontale Kollision und das Fahrzeug prallt nicht mit einem Drehwinkel von der Wand weg. Ist der Winkel aber verschieden von diesem Intervall, handelt es sich um eine seitliche Kollision und die Drehmatrix wird gebraucht (Abb. 8). Für das Überprüfen des Winkels wird einfach der Kosinus von den entsprechenden Seiten berechnet und berechnet, ob dieser innerhalb dieses Intervalls liegt. Je nachdem, ob der Kosinus positiv oder negativ ist, handelt es sich um eine Links- bzw. um eine Rechtsdrehung. Für den abgebildeten Fall braucht es folgende Abfrage:

$$\frac{|v|}{|vy|} = \cos \propto$$

Bei der Kollision zwischen den Fahrzeugen wird es etwas komplexer. Die Abfrage, ob zwei Kreise sich treffen, ist einfach. Auch hier wird wieder geschaut, ob sich die Position plus dem Kreisradius der beiden Fahrzeuge überschneiden (Abb.9). Auch hier ist eine Überschneidung möglich. Die Korrektur ist aber etwas schwieriger. Die Positionen müssen so korrigiert werden, dass sich die Kreise angrenzen und der Abstand zwei Mal dem Kreisradius entspricht. Dies wird wie folgt gemacht:

$$\Delta Korrektur = Front_1 - Fornt_2$$

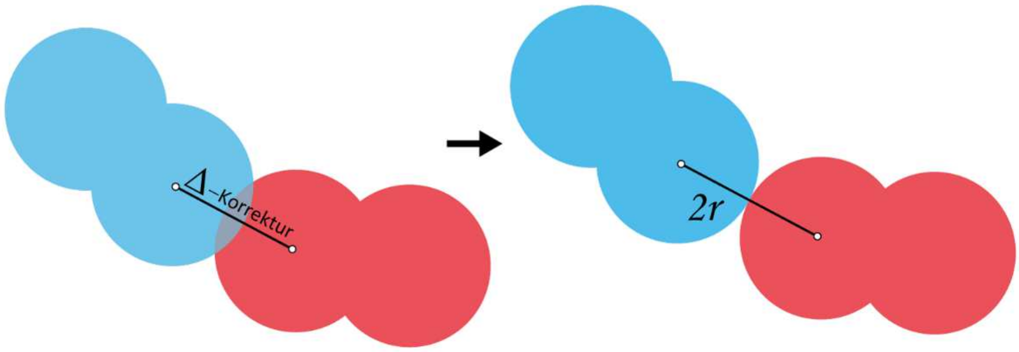
$$\Delta Korrektur := \frac{2 \cdot Radius - \Delta Korrektur}{2}$$

$$Front_1 \coloneqq Front_1 \pm \Delta Korrektur$$

$$Fornt_2 \coloneqq Fornt_2 \pm \Delta Korrektur$$

Je nachdem, welches Fahrzeug sich wo befindet, wird die Korrektur addiert oder subtrahiert.

Abb.9



Sind die Positionen korrigiert, kommt nun der Stoss. Dafür einen kurzen Ausflug in die Geometrie. Um die Kollision der Fahrzeuge zu optimieren, muss das Programm wissen, wie viele und welche Kreise sich in einer Linie befinden. Berühren sich zwei Kriese, werden diese mit einer imaginären Geraden g verbunden (Abb. 10). Diese hat folgende Gleichung (DMK DPK DCK, 2011, S. 106):

$$g : \vec{l} = \binom{x_1}{y_1} + k \cdot \binom{x_2}{y_2}, \; k \in \mathrm{R}$$

Nun wird überprüft, ob ein weiterer Punkt kollinear ist. Zuerst wird ein *k* mit den X-Koordinaten berechnet. Dann wird mit diesem *k* kontrolliert, ob die Y-Koordinate auch auf dieser Geraden liegt. Geht die Gleichung auf, so liegt der Punkt auf derselben Geraden.

$$k = \frac{x_3 - x_1}{x_2}$$

$$y_3 = y_1 + y_2 \cdot k$$

So findet das Programm heraus, ob das Fahrzeug einen frontalen Crash hat oder nicht. Liegen Heck und Front desselben Wagens auf der Geraden, hat dieses Fahrzeug eine frontale Kollision und muss nicht gedreht werden. Es können auch alle vier Kreise auf der Geraden sein und so haben beide Fahrzeuge eine frontale Kollision. Liegen Heck und Front vom selben Auto nicht auf derselben Geraden, ist die Kollision von der Seite und der Wagen erhält einen Spin.

Für den Spin muss man noch wissen, ob das Fahrzeug links oder rechts, und ob es an der Front oder am Heck getroffen wird. Ob Front oder Heck ist nun bekannt, aber für links oder rechts werden zwei neue Punkte eingeführt, die sich immer auf der entsprechenden Seite des Fahrzeuges befinden (Abb. 11). Ist der Abstand vom linken Punkt zum anderen Fahrzeug kürzer, wird das abfragende Auto logischerweise auf der linken Seite getroffen. Umgekehrt wird es auf der rechten Seite getroffen. Ist der Treffer Front und links, so dreht sich das Fahrzeug nach rechts. Ist der Treffer Heck und links, so dreht sich der Wagen nach links.

Abb.10

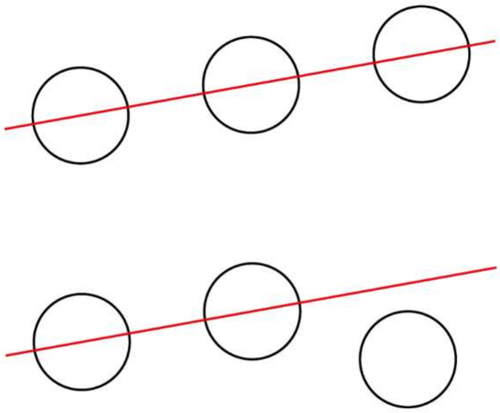
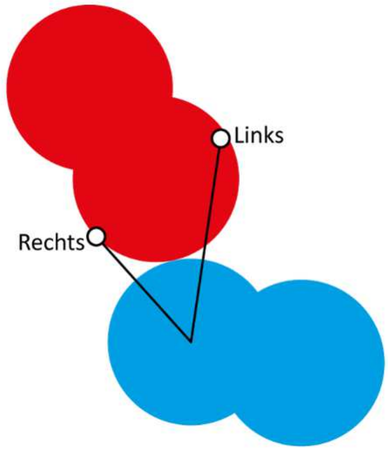


Abb.11



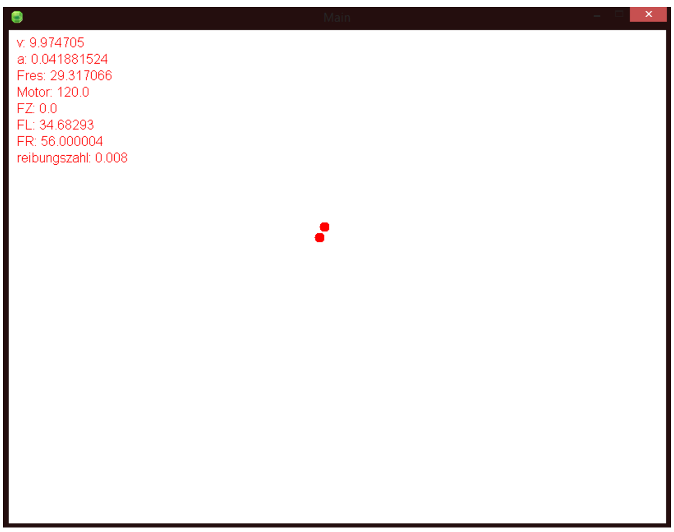
Abschluss/Erweiterung der Simulation

Alle Kräfte und Komponenten sind in einem Simulator impliziert und mehrfach neu aufeinander abgestimmt worden. Dies war nötig, da es bei der Einführung einer neuen Komponente zu falschen Resultaten gekommen war; schliesslich stehen alle Kräfte in Verbindung und sind voneinander abhängig.

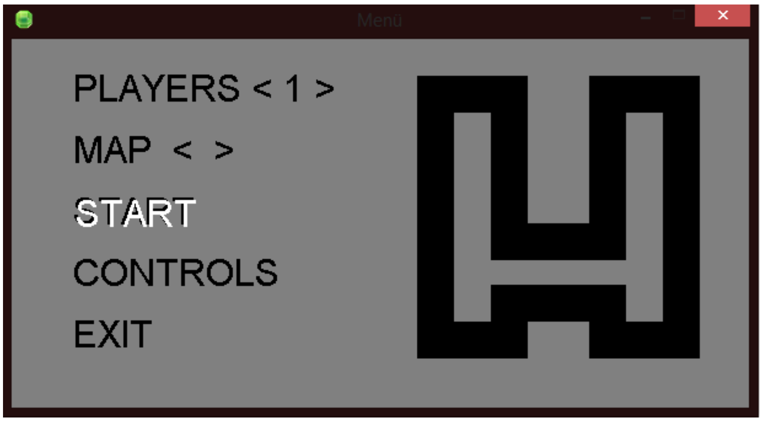
Nachdem alle Komponenten in der Simulation umgesetzt worden sind, wurde das Programm noch nebenbei zu einem kleinen Rennspiel erweitert. Das Spiel ist nicht sehr ausgearbeitet, da es nicht der Kern der Maturaarbeit ist. Dennoch gibt es ein Menü, neun verschiedene Karten und einen bis zu Vier-Spieler-Modus. Die Kräfte sind dieselben wie in der Simulation. Für das Spiel wurden noch zusätzlich Mauern und Checkpoints eingebaut. Diese Checkpoints verhindern, dass ein Spieler mogeln oder abkürzen kann. Jeder Spieler muss diese in der korrekten Reihenfolge passieren, sonst wird der Rundenzähler nicht erhöht.

Die Karten sind klein und statisch und haben alle je nur ein besonderes Merkmal. Dies verleiht dem Spiel eine gute Übersicht und eine spannende Dynamik. Da jeder Level nur ein signifikantes Merkmal hat, ist das Spiel an und für sich recht simpel.

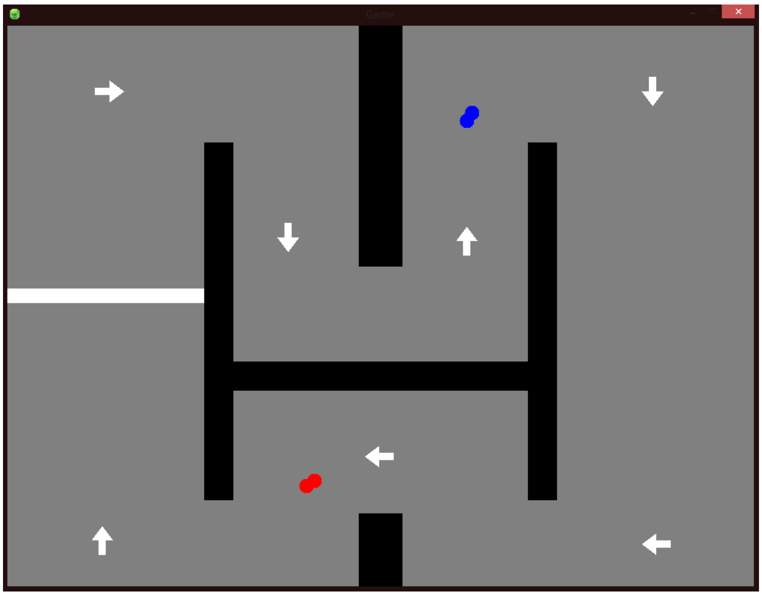
Simulator:



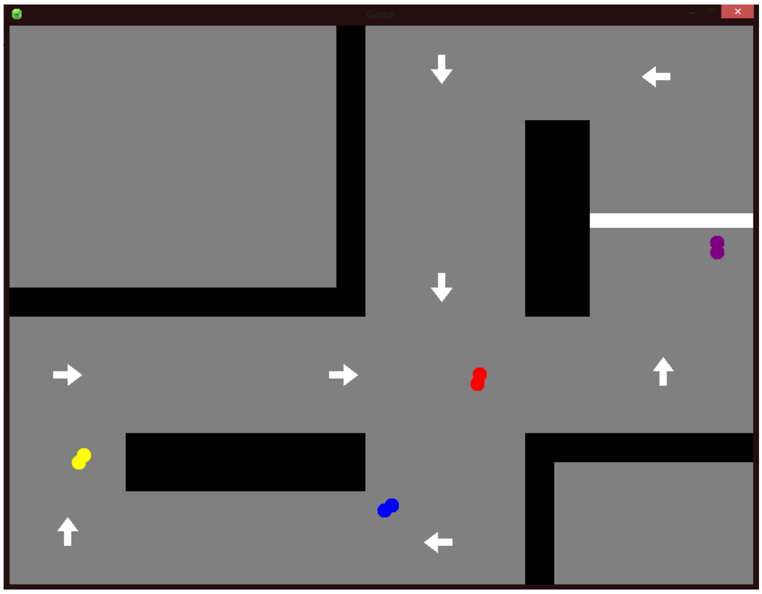
Menü:



Map 3:



Map 4:



Fazit

Mein Ziel war es, physikalische Kräfte, welchen ein Fahrzeug beim Fahren unterliegt, in einer Simulation darzustellen. Dabei wollte ich herauszufinden, wie nahe die Simulation an eine realistischen Darstellung herankommt. Eine Herausforderung dabei war, herauszufinden, welche Formeln oder Konstanten ich verwenden bzw. modifizieren muss, um diese Kräfte in einem Programm überhaupt simulieren zu können.

Diese Simulation ist zweidimensional konzipiert und deswegen sind die Sachverhalte, beziehungsweise die physikalischen Kräfte, abstrakt. Formeln mussten nur mit einigen Konstanten modifiziert werden, damit die Einheiten zueinander passen. Einige andere Komponenten konnten durch ihre Komplexität nicht umgesetzt werden und sind vereinfacht wiedergegeben, so wie der Motor oder auch das Anfahren. Es wurden ausschliesslich diese Kräfte simuliert, da diese mit zweidimensionalen Komponenten sinnvoll impliziert werden können. Andere Kräfte, wie beispielsweise das Drehmoment in der Kurvenbewegung, konnten nicht berücksichtigt werden, da diese noch eine dreidimensionale Komponente benötigen.

Unter den vorgegebenen Rahmenbedingungen (Zweidimensionalität, einfaches Modell ohne Anspruch auf vollständige Realitätstreue) ist ein plausibles Modell entstanden; Bewegung, Beschleunigung und Bremsen eines Fahrzeugs sind klar erkennbar und die Kräfte stehen in gutem Verhältnis zueinander. Insofern konnte ich die Arbeit so ausführen, wie ich sie mir vorgestellt hatte.

Anhang

Quellen:

DITTMAR-ILGEN, D. H. (24. 10 2014). helpster. Von Der Unterschied zwischen Normalkraft und Gewichtskraft - einfach erklärt: http://www.helpster.de/der-unterschied-zwischen-normalkraft-und-gewichtskraft-einfach-erklaert_170157 abgerufen

DMK DPK DCK. (2011). Formeln Tabellen Begriffe. Zürich: orell füssli.

frustfrei-lernen.de. (24. 10 2014). Von Newtonsche Gesetze: http://www.frustfrei-lernen.de/mechanik/newtonsche-gesetze.html abgerufen

kfz-tech. (24. 10 2014). Von Luftwiderstand: http://www.kfz-tech.de/Formelsammlung/Luftwiderstand.htm abgerufen

schuelerlexikon. (25. 10 2014). Von Physik Abitur: http://m.schuelerlexikon.de/phy_abi2011/Fahrphysik.htm abgerufen

statista. (24. 10 2014). Von Durchschnittliches Gewicht von Personenkraftwagen ausgewählter Hersteller im Jahr 2010 (in Kilogramm): http://de.statista.com/statistik/daten/studie/238004/umfrage/gewicht-von-pkw-nach-autoherstellern/ abgerufen

Wikipedia. (26. 10 2014). Von Stoß (Physik): http://de.wikipedia.org/wiki/Sto%C3%9F_(Physik) abgerufen

Bilder:

Gymnasium Kirchenfeld Logo: http://www.kinet.ch/gymkirchenfeld/logo_kreator.html

Alle anderen Bilder und Grafiken wurden vom Autor dieser Maturaarbeit selber hergestellt.