

# 1 Einsatz von Simulationen in der Physikausbildung

Die Simulation hat sich inzwischen neben den klassischen Säulen Experiment und Theorie als weitere Erkenntnismethode der Physik etabliert.<sup>1</sup> Sie verwendet numerische Methoden, um die Lösungen komplizierter Gleichungen zu erforschen, für die analytische Verfahren nicht oder nur mit sehr hohem Aufwand weiterführen. Auf diese Weise schlägt sie Brücken zwischen der Theorie und der Phänomenologie, die das Verständnis für die Konsequenzen der zugrundeliegenden Theorien verbessern oder manchmal auch erst ermöglichen.

In der Physikausbildung von Ingenieuren sind Theorie und Experiment seit jeher vertreten: Die klassische Vorlesung gibt einen Überblick über die Phänomene und die Basiselemente der Theorie, in Praktika oder Laboren werden experimentelle Grunderfahrungen und -techniken vermittelt. Doch auch hier lässt sich die Simulation nutzbringend einsetzen: Sie kann als didaktisches Werkzeug dienen, das die Konsequenzen der Theorie zu untersuchen erlaubt, auch wenn den Studenten die dazu benötigten mathematischen Werkzeuge fehlen. Sie lässt sich aber auch zur Nachbildung von Experimenten verwenden, an denen in idealisierter Weise am Computer quantitative Ergebnisse gewonnen werden können.

Die didaktischen Vorteile solcher »virtuellen Experimente« sind offensichtlich: Anders als im wirklichen Laborversuch sind alle störenden Faktoren von vornherein eliminiert, die »Versuche« gelingen immer und der zu vermittelnde Zusammenhang tritt deutlich hervor. Darüber hinaus garantiert die völlige Reproduzierbarkeit für alle Studenten das gleiche Ergebnis. Gerade für kleine Hochschulen kommt noch ein finanzieller Aspekt dazu: Die Ausstattung der Physiklabore ist häufig auf reine Grundexperimente beschränkt, oft gibt es auch nicht genügend Arbeitsplätze, so dass die Versuche der Studenten durch Demonstrationsexperimente in der Vorlesung ersetzt werden. Hier können vorhandene Computer genutzt werden, um die Studenten aktiv an der Gewinnung von physikalischen Zusammenhängen zu beteiligen.

Selbstverständlich können Simulationen echte Experimente auch in der Ausbildung nicht ersetzen: Gerade die Schwierigkeiten, denen jeder Experimentator gegenübersteht, und die Sorgfalt und die Methoden, die nötig sind, um sie zu überwinden, sind

---

<sup>1</sup> Der Text dieses Abschnitts basiert weitgehend auf [22] und wird hier verwendet mit freundlicher Genehmigung der Herausgeber des Global Journal of Engineering Education.

ein wichtiger Teil der zu vermittelnden Erfahrungen. Der echte Kontakt mit dem zu untersuchenden System führt darüber hinaus durch seine ganz andersartigen Aktivitäten und Sinneseindrücke als eine Arbeit am Computer zu einer besonderen Verbesserung des Lernerfolgs. Aber die größte Gefahr beim Einsatz simulierter Experimente besteht darin, dass die Studenten die Simulation mit der Realität verwechseln. Gerade angesichts einer zunehmenden gesellschaftlichen Tendenz zur Virtualisierung, in der die Grenzen zwischen »Egoshootern« und »virtuellem Krieg« absichtlich verwischt werden, muss der Lehrende besonders darauf hinweisen, welche Erkenntnisse mit Simulationen gewonnen werden können und welche nicht!

## 1.1 Simulationen mit Java-Applets

Zur Implementierung der für eine Simulation benötigten numerischen Verfahren können alle gängigen Programmiersprachen sowie eine Vielzahl vorhandener Problemlösungsumgebungen verwendet werden. Im PhysBeans-Projekt werden auf der Programmiersprache Java [16, 8] basierende Applets erstellt, die für die Umsetzung virtueller Experimente eine Reihe von Vorteilen bieten:

- ▶ einfache Einbindung in HTML-Seiten, die Handlungsanweisungen oder Hintergrundinformationen enthalten,
- ▶ Unabhängigkeit vom speziellen Rechnertyp oder Betriebssystem,
- ▶ allgemeine Verfügbarkeit entsprechender Betrachter (Browser-Plugins) [31],
- ▶ sowohl direkt im Internet als auch auf CD-ROM einsetzbar.

Auch aus der Sicht des Entwicklers spricht einiges für Java-Applets:

- ▶ Sprachkonstruktionen für zeitgesteuerte Aktionen [27],
- ▶ standardisierte Bibliotheken für graphische Oberflächen [47],
- ▶ frei verfügbare Numerik-Klassen, z. B. für lineare Algebra, komplexe Zahlen und spezielle Funktionen [42],
- ▶ frei verfügbare Entwicklungsumgebungen für graphisches Programmieren [4].

Virtuelle Experimente in Form von Applets lassen sich in ganz verschiedener Weise in die Physikausbildung integrieren: Sie können als Demonstrationen in einer Vorlesung eingesetzt werden oder als Werkzeug zum angeleiteten, gezielten »Experimentieren« der Studenten in Übungen. Darüber hinaus können sie aber auch der spielerischen Erforschung im Selbstunterricht oder der Vorlesungsnachbearbeitung und -vertiefung dienen. Studenten werden dadurch zu aktivem Lernen angeregt, was das Verständnis abstrakter Konzepte erleichtert und zu besseren mentalen Modellen physikalischer Zusammenhänge führt [14].

Zunehmend erscheinen physikalische Lehrbücher, die auf einer ergänzenden CD-ROM entsprechende Applets einsetzen [1, 44, 9]. Auch im Internet lassen sich eine ganze Reihe von Applets zu vielen Themen der Physik finden, ausgehend etwa von den Linksammlungen von Physicsweb [36] oder dem Projekt »physik multimedial« [39].

Nicht verschweigen sollte man allerdings den grundsätzlich beträchtlichen Aufwand, der zum Erstellen von Applets wie den hier vorgestellten nötig ist. Abhilfe schafft nur der konsequente Einsatz wiederverwendbarer Basisbausteine, der sich aber natürlich erst bei einer genügend großen Anzahl verschiedener Applets rentiert. Ein verwandter Aspekt ist die Anpassbarkeit an eigene Wünsche: Häufig ließe sich ein im Internet gefundenes Applet in eigenen Lehrveranstaltungen besser nutzen, wenn man einige Parameter (Beschriftungen, einstellbare Größen usw.) ändern könnte. Da aber das Einarbeiten in den Quellcode eines anderen Programmierers – wenn er denn überhaupt freigegeben wird – eine schwierige Angelegenheit ist, findet man eine Vielzahl von ähnlichen Applets, die alle von Grund auf neu programmiert worden sind.

Eine Lösung des Problems, Applets zur Verfügung zu stellen, die in vielfältiger Weise an unterschiedliche Anforderungen angepasst werden können, ohne dass vertiefte Java-Programmierkenntnisse nötig sind, bieten die Physlets [9, 2]. Es handelt sich dabei um eine Sammlung von Applets zu verschiedenen Gebieten der Physik, die jeweils durch JavaScript-Kommandos konfiguriert werden können. Der Anwender muss auf seinen Webseiten nur die benötigten Applets einfügen und sie mit JavaScript-Befehlen an seine Bedürfnisse anpassen. Er kann sogar mehrere Applets miteinander verknüpfen und Daten zwischen ihnen austauschen, z. B. um Ergebnisse der Simulationsprogramme mit Hilfe spezieller Präsentations-Applets graphisch darzustellen. Auf diese Weise können aus wenigen komplexen Applets ganz verschiedene Anwendungen entstehen, die jeweils unterschiedliche Aspekte des untersuchten physikalischen Systems in den Vordergrund stellen.

Zwar wird auf diese Weise der Anwender vom Java-Code der Applets völlig abgeschirmt, aber die Vielzahl an Möglichkeiten fordert dennoch einen erheblichen Einarbeitungsaufwand: Neben den Grundzügen von JavaScript muss man auch die zahlreichen besonderen Merkmale der eingesetzten Applets kennen. Diese stellen sich als monolithische Blackboxen dar, deren Source-Code von den Physlet-Entwicklern nicht herausgegeben wird. Dadurch ist es nicht möglich, die Sammlung der Physlets selbst um weitere Physik-Bereiche zu erweitern.

Das PhysBeans-Projekt verfolgt eine andere Strategie: Es werden eine Menge von Bausteinen zur Verfügung gestellt, aus denen Applets zur Simulation physikalischer Systeme in graphischer Weise, ohne explizite Verwendung von Java-Code, zusammengesetzt werden können. Da die verwendeten Techniken auf dem JavaBeans-Modell [6] basieren, lassen sich hierfür verbreitete Java-Entwicklungsumgebungen einsetzen. Außerdem wird der Source-Code der Bausteine (der eigentlichen »PhysBeans«) als

Open Source zur Verfügung gestellt. Ein PhysBeans-Anwender kann dann – je nach Vorkenntnissen – auf drei verschiedenen Ebenen arbeiten, indem er

- ▶ vorhandene Applets durch Verändern von Parameterwerten an eigene Bedürfnisse anpasst oder graphisch umgestaltet,
- ▶ die vorhandenen Bausteine im Rahmen der abgedeckten physikalischen Gebiete zu neuen Anwendungen verknüpft oder
- ▶ eigene Bausteine programmiert, um neue physikalische Bereiche zu erschließen.

Wie die folgenden Kapitel zeigen werden, lassen sich auf den beiden ersten Ebenen Simulationsprogramme für eine Vielzahl verschiedener Anwendungsfälle erstellen, ohne dass Programmierkenntnisse dazu nötig sind. Darüber hinaus liefern die PhysBeans-Quellen viele Beispiele, die das Erstellen eigener Beans erleichtern.

## 1.2 Grundaufbau der PhysBeans-Applets

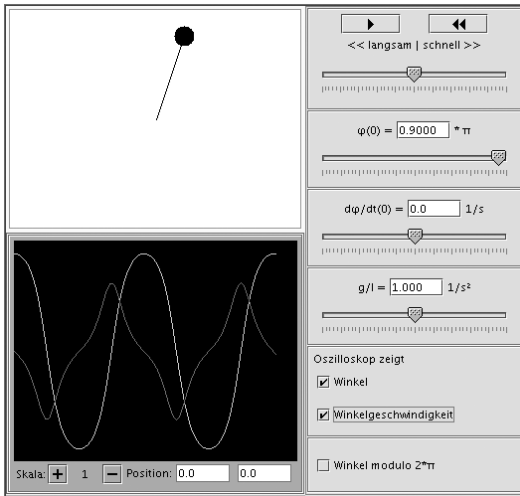
Simulationsprogramme, die in der Lehre eingesetzt werden sollen, müssen nicht nur die physikalischen Zusammenhänge mit numerischen Methoden nachbilden, sondern auch didaktischen Kriterien genügen. Dazu muss vor allem auf die Erstellung der graphischen Benutzeroberfläche besondere Sorgfalt verwendet werden. Da dies einen recht erheblichen Programmieraufwand erfordert, findet man viele Applets im Internet, deren optische Gestaltung eine intuitive Benutzung erschwert. Beim Entwurf der PhysBeans-Applets und ihrer Bausteine wurden dagegen folgende Anforderungen zugrundegelegt:

- ▶ Die Applets sind möglichst ohne weitere Erklärungen benutzbar,
- ▶ alle veränderbaren Größen sind unmittelbar erkennbar,
- ▶ Parameter können nur innerhalb sinnvoller Bereiche eingestellt werden,<sup>2</sup>
- ▶ Startwerte sind so gewählt, dass sich sofort aussagekräftige Ergebnisse einstellen.

Zu diesem Zweck haben alle PhysBeans-Applets einen sehr ähnlichen Aufbau: Sie bestehen in der Regel aus einem Eingabebereich, der sämtliche Eingabe-Elemente enthält, einem Objekt-Bereich, der das zu simulierende Objekt schematisch darstellt, und einem Ausgabe-Bereich, in dem Messergebnisse graphisch oder als Zahlen angezeigt werden. Zur besseren Wiedererkennung werden die einzelnen Bereiche jeweils in einer charakteristischen Farbe dargestellt, außerdem sind sie häufig in gleicher Weise zueinander angeordnet, soweit dies möglich und sinnvoll ist. Das *Applet zum mathematischen Pendel* (Abschnitt 2.2.4) zeigt diesen Aufbau in prototypischer Weise:

---

<sup>2</sup> sofern nicht der Lerneffekt eines Applets gerade in der Untersuchung von Gültigkeitsgrenzen besteht,



Applet zum mathematischen Pendel

Sämtliche Änderungen am System werden über spezielle, immer wiederkehrende Eingabe-Elemente durchgeführt. In diesem Beispiel kommen aus der PhysBeans-Bibliothek Komponenten zur Zeitsteuerung und zur Werteingabe sowie Auswahlboxen zum Einsatz. Die Zeiteingabe dient zur Steuerung des Zeitablaufs der Simulation. Sie erlaubt die Änderung der Geschwindigkeit, das Anhalten bzw. Weiterführen der Simulation sowie einen Neustart. Zur Kennzeichnung werden die bei Audio-/Video-Recordern üblichen Symbole verwendet, dadurch ist die Bedienungsweise unmittelbar klar. Parameterwerte werden in jeweils identisch aufgebauten Feldern eingegeben, wobei sowohl die schnelle qualitative Eingabe über einen Schieberegler als auch die Angabe eines genau reproduzierbaren Zahlenwertes möglich ist. Natürlich sind beide Eingabemethoden gekoppelt, außerdem werden bei der Texteingabe nur Werte im durch den Regler bestimmten Bereich zugelassen. Ein kurzer Text erläutert jeweils die zu verändernde Größe und gibt ihre Einheit an. Mit Hilfe der Auswahlboxen kann man festlegen, welche Kurven im Oszilloskop angezeigt werden, darüber hinaus sind gelegentlich weitere Darstellungsparameter angegeben. Diese Boxen ersetzen auch eine Legende: Durch einfaches An- und Ausschalten wird sofort deutlich, welche Farbe zu welcher physikalischen Größe gehört.

Der Ausgabe-Bereich besteht häufig aus einem simulierten Oszilloskop, das entweder die physikalisch interessierenden Größen als Funktionen der Zeit darstellt oder jeweils eine Größe in Abhängigkeit von einer anderen (XY-Oszilloskop). Zur Anpassung an

stark veränderliche Wertebereiche kann die Skala über Schaltknöpfe in mehreren Stufen jeweils um einen Faktor zwei vergrößert oder verkleinert werden. Quantitative Messungen an den Kurven sind leicht möglich: Klickt man eine beliebige Stelle im Oszilloskop-Fenster an, werden die zugehörigen Koordinaten angezeigt, wobei alle Werte jeweils direkt in den entsprechenden SI-Basiseinheiten angegeben werden, unabhängig vom gewählten Skalierungsfaktor. Im Anzeigebereich des Oszilloskops verwandelt sich der Cursor in ein Fadenkreuz, um auf diese Möglichkeit hinzuweisen. Damit sich verschiedene Kurven in einer – bis auf die Einheit – gemeinsamen Skala darstellen lassen, sind die Anfangswerte aller Eingabeparameter so aufeinander abgestimmt, dass die physikalischen Größen, in SI-Einheiten ausgedrückt, vergleichbare Größenordnungen haben. Bei Änderungen der Systemgrößen laufen die Kurven unter Umständen auseinander, hier muss ggf. bei Messungen verschiedener Größen zur genaueren Darstellung jeweils eine andere Skalierung eingesetzt werden.

Der Objektbereich enthält eine schematische Darstellung des zu untersuchenden Systems, im Beispielapplet zeigt es etwa die Bewegung des mathematischen Pendels. Hier wäre eine realistischere Darstellung, etwa dreidimensional und mit Beleuchtungseffekten, vielleicht wünschenswert. Sehr schöne Beispiele dafür findet man etwa in [34, 35]. Für die hier angesprochene Zielgruppe – Studenten oder Schüler an gymnasialen Oberstufen – steht der erzielbare didaktische Nutzen aber in keinem Verhältnis zum dafür notwendigen Programmieraufwand. Außerdem erinnert die stilisierte Darstellung an den virtuellen Charakter des Experiments. In einigen Applets fällt der Objektbereich weg, wenn er keine weiteren Erkenntnisse liefert und der Platz für die Anzeige benötigt wird. Gelegentlich wird ihm durch entsprechende Farbgebung das Oszilloskop zugeordnet. Dies soll ausdrücken, dass die angezeigten Kurven – in einer höheren Abstraktionsstufe – hier das eigentliche Untersuchungsobjekt darstellen.

Durch die Verwendung gleichartiger Elemente in ähnlicher, klar aufgebauter Anordnung ist die Bedienung der Applets ohne zusätzliche Erklärungen intuitiv verständlich, ein spielerischer Umgang führt sofort zum Erfolg. Einzige Ausnahme ist wohl die Aufnahme von Messwerten im Oszilloskop: Hier muss einmal auf den grundsätzlichen Mechanismus hingewiesen werden. Diesem Primat der unmittelbaren Verständlichkeit fiel auch die Möglichkeit zum Opfer, Objekte direkt mit der Maus manipulieren zu können. Beispielsweise ließe sich die Anfangsauslenkung des Pendels durch direktes Bewegen des Pendels mit der Maus eingeben, auch alternativ zum hier vorgesehenen Eingabe-Element. Diese Technik wird in vielen Applets angewandt. Die hier verwendete Methode, das Objekt selbst nicht unmittelbar zugreifbar zu machen, entspricht aber auch den tatsächlichen Gegebenheiten heutiger Experimente: Nach dem Aufbau wird man in der Regel keinen direkten Kontakt zum eigentlichen Messobjekt mehr haben. Die im nächsten Abschnitt beschriebenen ferngesteuerten Experimente entsprechen naturgemäß ebenfalls dieser Vorgehensweise.

## 1.3 Vergleich mit anderen Ansätzen

Einfacher als das Erstellen eigener Simulationsprogramme ist der Einsatz von Numerikpaketen wie Matlab [20], die mit wenigen Befehlen das Bestimmen von Lösungen – etwa durch Integration nichtlinearer Differentialgleichungen – und ihre graphische Darstellung ermöglichen. In [46] wird an einigen Beispielen aus der Mathematik- und Physikausbildung gezeigt, wie Matlab bei mathematischen Fragestellungen aus ganz unterschiedlichen Bereichen schnell zu numerischen Lösungen führt. Dass sich auch Algebrasysteme wie Maple [30] für diesen Zweck einsetzen lassen, wird in [41] am Fallbeispiel des mathematischen Pendels ausführlich dargestellt. Der Schwerpunkt liegt hier zwar in der Analyse der Gleichungen mittels analytischer Verfahren wie Reihenentwicklungen, aber es lassen sich zusätzlich auch numerische Lösungen bestimmen.

Das Verwenden solcher Programme setzt bei den Studenten nicht nur Grundkenntnisse ihrer Bedienung voraus, sondern erfordert auch ein Verständnis der mathematischen Problemstellung. Nicht die physikalischen Phänomene stehen hier im Vordergrund, sondern die zu ihrer Beschreibung nötigen Gleichungen sowie die Verfahren zu ihrer Lösung. Daher bieten sich solche Ansätze eher für fortgeschrittene Studenten an, die über die nötigen Vorkenntnisse aus der Mathematik und Informatik verfügen. Dann aber sind sie besonders geeignet, in interdisziplinärer Weise Aspekte von Mathematik, Physik und Informatik zusammenzubringen.

Im Gegensatz dazu sollen die hier beschriebenen Applets mit Hilfe ihrer graphischen Oberfläche eine experimentelle Situation nachbilden und den Studenten von der zugrundeliegenden Mathematik abschirmen. Sie eignen sich daher schon für die Anfängerausbildung, sind aber naturgemäß auf die jeweilige physikalische Situation beschränkt.

Zwar lassen sich auch in Matlab graphische Oberflächen programmieren, allerdings nimmt der Erstellungsaufwand dann sehr stark zu. Darüber hinaus erfordern solche Matlab-Programme das Vorhandensein einer entsprechenden Matlab-Umgebung, die für die Studenten in der Regel nicht verfügbar ist. Schließlich lassen sie sich nicht in einfacher Weise in eine HTML-Umgebung einbinden.

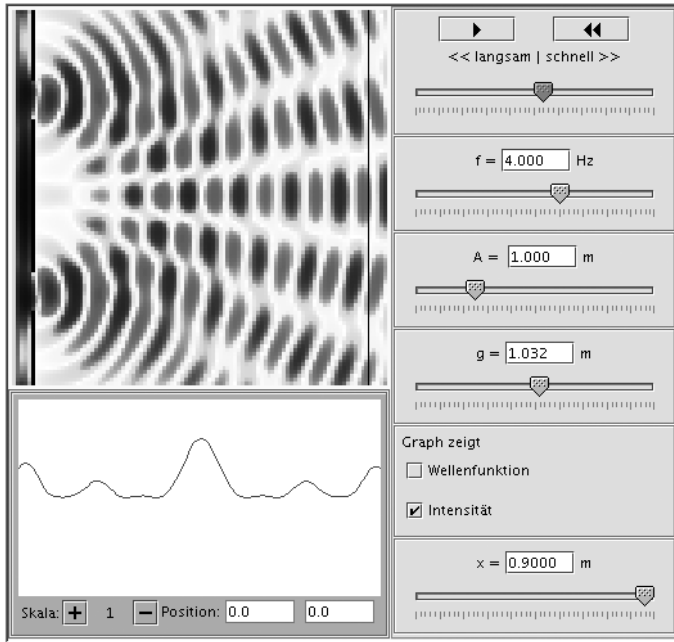
Einen ganz anderen Weg gehen die »virtuellen Labore«, die in [15] vorgestellt werden: Es handelt sich hier um reale Experimente, die über das Internet ferngesteuert werden können und deren Ergebnisse in Echtzeit übertragen werden. Ein ausgeklügeltes Reservierungssystem verhindert dabei den gleichzeitigen Zugriff durch mehrere Anwender. Auch wenn durch den vorgegebenen Aufbau die Möglichkeiten auf die vorgegebene Situation eingeschränkt sind (auch die Möglichkeiten, Fehler zu machen!), entsteht doch ein recht unmittelbarer Eindruck des Experimentierens. Dazu tragen auch simultan übertragene Videoaufnahmen des Versuchsaufbaus bei.

Der technische Aufwand für ein solches ferngesteuertes Labor ist hoch, er wird sich weniger für physikalische Standardversuche als eher für kompliziertere Versuchsaufbauten

lohlen. Dann aber kann die gute zeitliche Ausnutzung solcher Anlagen sicher den Aufwand rechtfertigen. Im in [40] beschriebenen Experiment kommt noch hinzu, dass im Fach Mechatronik die für die Fernsteuerung nötigen Aktuatoren und Sensoren ja selber Gegenstand der Untersuchung sind, so dass nicht nur die Durchführung, sondern schon der Aufbau eines »virtuellen Labors« ein nützliches Projekt im Rahmen dieses Studiengangs darstellt.

Das sich ergebende Beugungsmuster hat sehr starke Ähnlichkeit mit dem bei der *Überlagerung von Wellen zweier Punktquellen* (Abschnitt 2.3.13).

Messen Sie die Position der Minima am Schirm in Abhängigkeit des Spaltabstands  $g$  und der Frequenz  $f$  und vergleichen Sie diese mit den entsprechenden Ergebnissen bei der Interferenz von Punktquellen.



Doppelspalt (Applet im Abschnitt 2.3.16)

Bei hohen Frequenzen wird das Muster deutlich komplizierter. Gleichzeitig stimmen die Positionen der Minima schlechter mit den für großen Schirmabstand geltenden Beziehungen überein, obwohl der Schirmabstand sich nicht ändert. Warum?

Messen Sie die Intensität des zentralen Maximums als Funktion der Frequenz. Können Sie das Ergebnis erklären?

### 2.3.17 Reflexion einer Welle

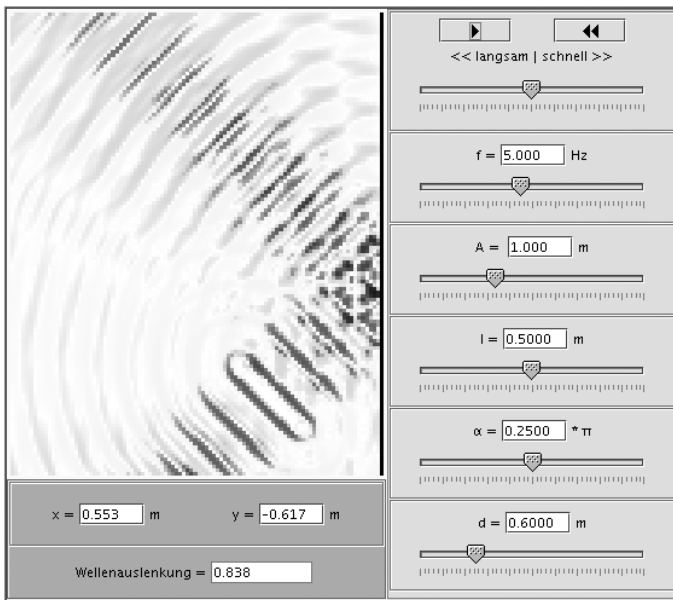
Im Rahmen der geometrischen Optik werden Wellen als einfache Strahlen (senkrecht zu den Wellenfronten) genähert. Die Reflexion einer Welle an einem Hindernis wird

dann mit dem bekannten »Einfallswinkel = Ausfallswinkel« für ein- und ausfallenden Strahl beschrieben. Diese Näherung ist gut, solange die Wellenlänge klein ist gegenüber den sonst auftretenden Längen.

Damit sich in der kleinen simulierten Wellenwanne die Reflexion gut erkennen lässt, muss die Wellenlänge möglichst klein sein (soweit es die Simulationsgenauigkeit zulässt!) und die Größe der Quelle möglichst groß (um ebene Wellen zu erreichen). Außerdem sollte man versuchen, das Auftreten von stehenden Wellen zu vermeiden, indem man die Quelle so klein macht und/oder so platziert, dass die reflektierte Welle nicht in sie hineinläuft.

Messen Sie nun bei verschiedenen Einfallswinkeln den Ausfallswinkel der reflektierten Welle. Inwieweit macht die Strahl-Näherung Sinn? Vergleichen Sie dabei auch das Verhalten bei verschiedenen Anregungsfrequenzen.

Betrachten Sie das Bild, das sich bei der Anfangskonfiguration und einer Frequenz von 1 Hz ergibt. Die hier benutzte Quelle erzeugt nur nach oben ausgelenkte Pulse, wie Sie durch direktes Messen der Wellenauslenkung oder direkt an der Färbung erkennen können. Nach der Reflexion werden daraus nach unten ausgelenkte Pulse. Dies ist der übliche Phasensprung um  $\pi$ , der bei Reflexion an einem festen Ende auftritt.



Reflexion einer Welle (Applet im Abschnitt 2.3.17)

Auch ein ganz anderes Phänomen lässt sich mit diesem Applet untersuchen: Stellen Sie dazu die Länge der Quelle möglichst klein ein, die Amplitude möglichst groß. Der Abstand von der Wand liege im mittleren Bereich, die Frequenz etwa bei 3 Hz. Lassen Sie die Simulation eine Weile laufen, bis sich ein festes Muster eingestellt hat.

Wenn Sie genau hinschauen, erkennen Sie mehrere Minima (weiße Linien) in Hyperbelform, wie sie auch bei der Interferenz zweier Punktquellen zu finden sind (vgl. *Überlagerung von Wellen zweier Punktquellen*, Abschnitt 2.3.13). Wie ist das zu erklären?

Antwort: Die Wand wirkt auf die Welle wie ein Spiegel. Statt eine Punktquelle vor dem Spiegel zu betrachten, könnte man genauso gut das Interferenzmuster der Punktquelle und ihres Spiegelbildes untersuchen. Diese Idee wird auch in der Elektrostatik unter dem Begriff »Spiegelladung« zur Berechnung elektrischer Felder verwendet.

### 2.3.18 Brechung einer Welle

Die Frequenz einer Welle ist durch die Quelle vorgegeben. Ändert sich die Ausbreitungsgeschwindigkeit  $c$ , etwa beim Übergang in ein anderes Medium, so muss sich auch die Wellenlänge  $\lambda$  ändern gemäß der fundamentalen Beziehung

$$\lambda f = c$$

Im Applet läuft die Welle aus einem Medium mit  $c_1 \approx 1$  m/s in ein anderes mit vorgebarem  $c_2$ . Messen Sie die Wellenlänge inner- und außerhalb des Mediums bei verschiedenem  $c_2$  und verschiedenem  $f$ . Vergleichen Sie sie mit dem theoretischen Wert.

Bestimmen Sie den Brechungswinkel  $\varepsilon'$  bei verschiedenem Einfallswinkel  $\varepsilon$  und für verschiedene  $c_2$ . Vergleichen Sie die Ergebnisse mit dem Brechungsgesetz

$$\frac{\sin \varepsilon}{\sin \varepsilon'} = n$$

Was ist also der Zusammenhang zwischen dem Brechungsindex  $n$  und den Ausbreitungsgeschwindigkeiten der beiden Media? Ein Teil der einfallenden Welle wird auch reflektiert. Messen Sie die Amplituden der einzelnen Wellen und bestimmen Sie daraus Reflektions- und Transmissionsgrad. Beachten Sie dabei, dass die Quelle nur positive Wellenberge erzeugt.

### 2.3.19 Totalreflexion

Trifft eine ebene Welle unter einem Einfallswinkel  $\varepsilon$  auf die Grenze zu einem Medium mit niedrigerer Ausbreitungsgeschwindigkeit, ist der Austrittswinkel  $\varepsilon'$  größer als der