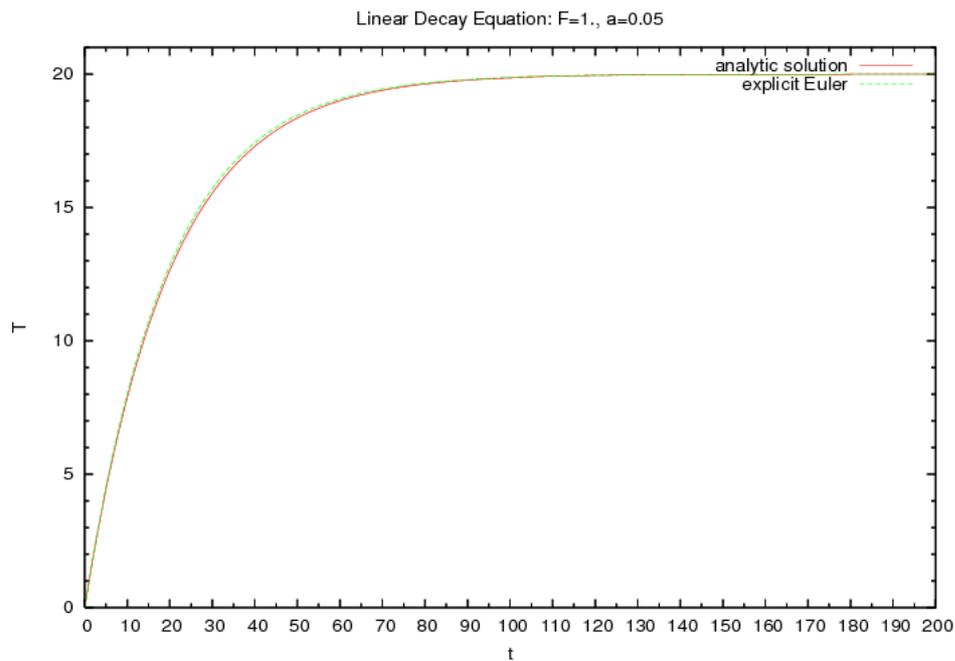


Lösungen zur Übung 0/1: 'Evolutionsgleichung'

Aufgabe 0/1:

Der Code zur Berechnung der analytischen Lösung der Evolutionsgleichung findet sich im file **evolution.f90**, derjenige zur Berechnung der numerischen Lösung mit dem expliziten Euler Verfahren im file **evolution_numerics.f90** (zusammen mit anderen numerischen Verfahren; s.u.).

Mit den angegebenen Parametern ergeben sich folgend Kurven (analytische und numerische Lösung)



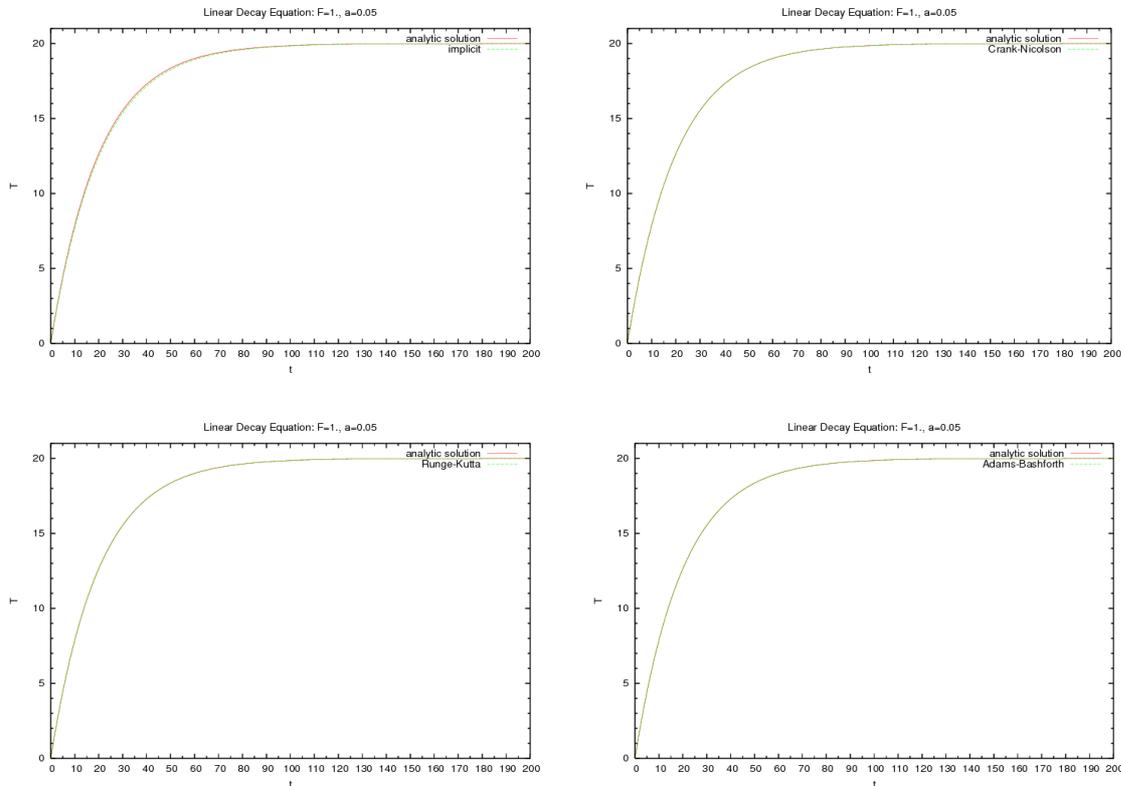
Aufgabe 2:

Für $\Delta t = 1$. ergibt sich: RMS = 8.02402869E-02 und MAX. = 0.18786907
Für $\Delta t = 0.5$: RMS = 3.98211181E-02 MAX. = 9.29384232E-02

Beide Fehler halbieren sich in etwa. Die Fehlerordnung ist also 1. (Note: Die Ergebnisse können, je nach Computer, variieren)

Aufgabe 3:

Auch die Codes für die anderen numerischen Verfahren finden sich in **evolution_numerics.f90**. Für die unterschiedlichen Verfahren ergeben sich, mit den gegebenen Parametern, folgende Kurven:



Wie zu sehen ist, liegen die numerischen Lösungen für das Crank-Nicolson, das Runge-Kutta und das Adams-Bashforth Verfahren praktisch genau auf der analytischen Lösung. als Fehler ergeben sich:

Für $\Delta t = 1$:

implizit:	RMS = 7.79263377E-02	MAX. = 0.18019581
Crank-Nicolson:	RMS = 6.62017846E-04	MAX. = 1.53636932E-03
Runge-Kutta:	RMS = 1.94818767E-06	MAX. = 3.81469727E-06
Adams-Bashforth:	RMS = 3.22923600E-03	MAX. = 7.51113892E-03

Für $\Delta t = 0.5$:

implizit:	RMS = 3.92376296E-02	MAX. = 9.10186768E-02
Crank-Nicolson:	RMS = 1.55181027E-04	MAX. = 3.72886658E-04
Runge-Kutta:	RMS = 2.98837995E-06	MAX. = 7.62939453E-06
Adams-Bashforth:	RMS = 8.15307198E-04	MAX. = 1.89781189E-03

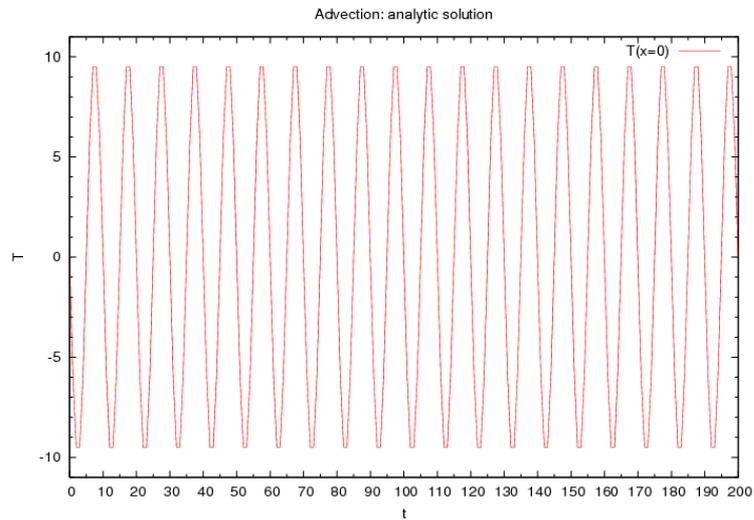
Beim Runge-Kutta erlaubt die Rechengenauigkeit keine Ermittlung der Fehlerordnung. Ansonsten ergeben sich die im Skript angegebenen Werte.

Lösungen zur Übung 0/2: 'Lineare Advektionsgleichung'

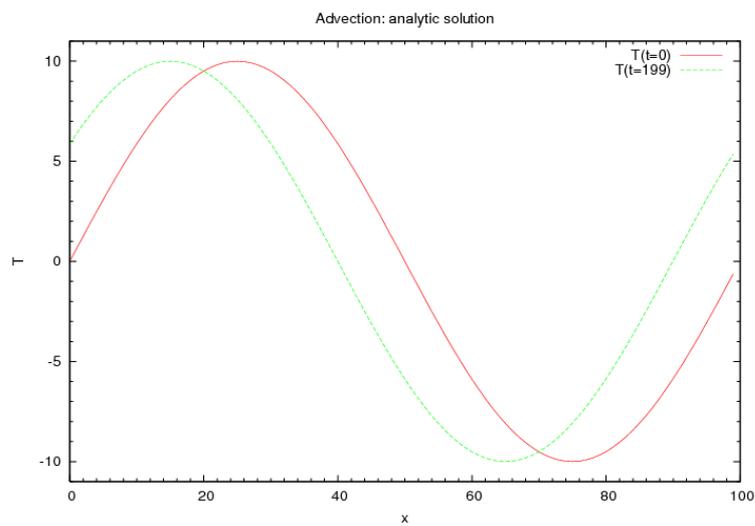
Aufgabe 0/1:

Der Code zur Berechnung der analytischen Lösung der linearen Advektionsgleichung findet sich im file **advection.f90**, Mit den angegebenen Parametern ergeben sich folgend Kurven:

a) Punkt $x=0$ für alle t



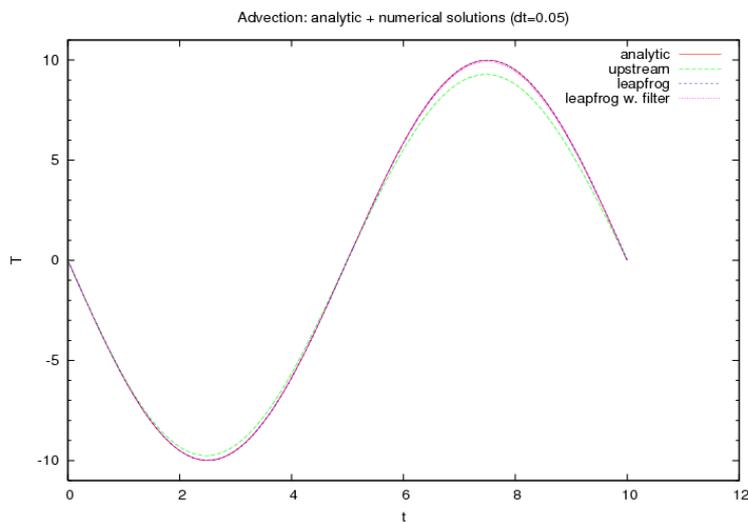
b) Alle x für $t=0$ und $t=199$



Aufgabe 2:

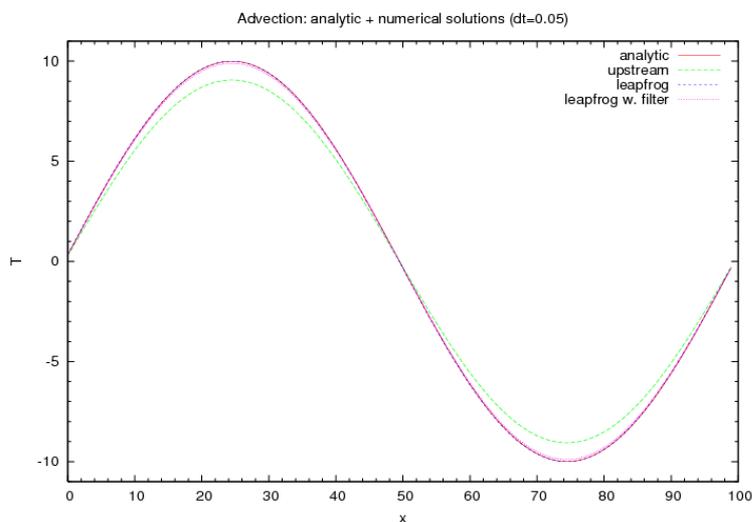
Um numerische Stabilität zu erhalten, muss die Courant-Zahl (Cr) klein genug gewählt werden. D.h., bei festgehaltenen $\Delta x (=1)$ und $u (=10)$ muss Δt angepasst werden. Für das Upstream Verfahren (bei positivem u) und dem Leapfrog ohne Filter muss gelten: $Cr \leq 1$ (d.h. $\Delta t \leq 0.1$), für Leapfrog mit Filter: $Cr \leq (1-\gamma)^2$ (mit $\gamma = \text{Filterkonst.}$; s. Skript). Um bei allen Verfahren auf der sicheren Seite zu sein, wird hier $\Delta t=0.05$ verwendet (und γ wird auf 0.1 gesetzt). File **advection_numerics.f90** enthält den code für alle hier gezeigten numerischen Lösungen (sowie die analytische für die Advektion der Welle). Im Programm müssen ggf. Parameter wie Wellenzahl, Zeitschritt, etc. angepasst werden. Für die hier gewählten Parameter ergeben sich folgende Kurven (Note: t ist die Zeit. Da auch hier 200 Zeitschritte gerechnet wurden und ist natürlich die Gesamtzeit kürzer als bei Aufgabe 1, da Δt deutlich kleiner ist):

a) Punkt $x=0$ für alle t



(Note: die Analytische Lösung und die beiden Leapfrog Lösungen liegen fast übereinander)

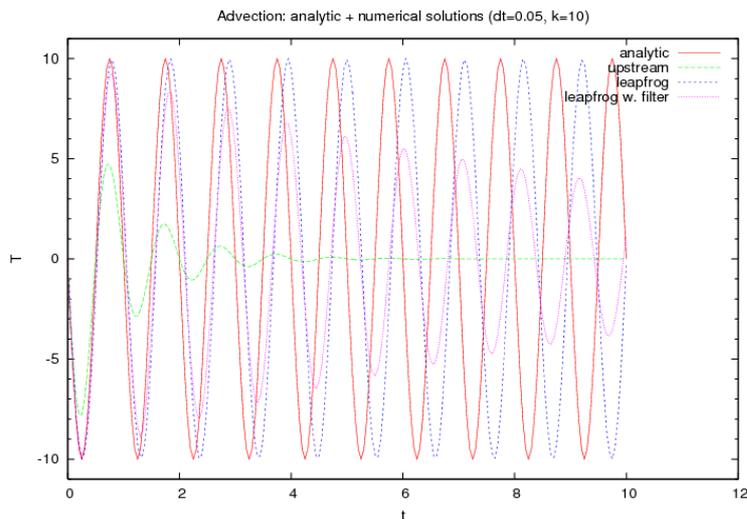
b) Alle x für $t=199\Delta t$



Aufgabe 3:

Wie aus der Vorlesung bekannt, ist der Fehler (Amplitude und Phase) abhängig von der Courant-Zahl (Cr) und der Wellenzahl der eingesetzten Anfangswelle. Je kürzer die Welle und je kleiner die Courant-Zahl, desto größer wird der Fehler (bei $Cr=1$ bzw. $(1-\gamma)^2$ verschwinden die Fehler). Hier ist exemplarisch das Ergebnis für $Cr=0.5$ (wie in Aufgabe 2) und für Wellenzahl 10 gezeigt:

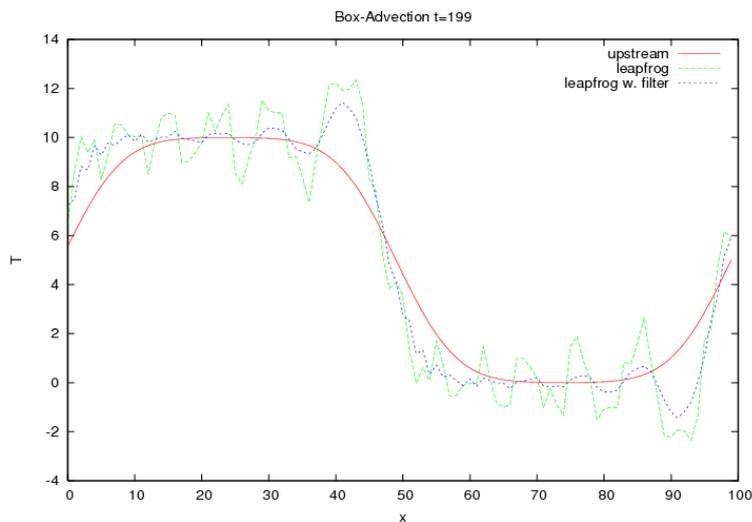
Punkt $x=0$ für alle t



Wie zu erkennen ist (und aus der Vorlesung bekannt), ist das upstream Verfahren stark diffusiv und die Welle ist dort nach kurzer Zeit vollständig verschwunden. Auch das Leapfrog Verfahren mit Filter zeigt numerische Diffusion. Die Amplitude geht auf etwa ein Drittel des Anfangswerts zurück. Nur das original Leapfrog zeigt keine numerische Diffusion, allerdings ist, wie im Verfahren mit Filter, starke numerische Dispersion zu bemerken: am Ende der Integration sind die numerischen Wellen um ca. 180° zur analytischen Lösung phasenverschoben.

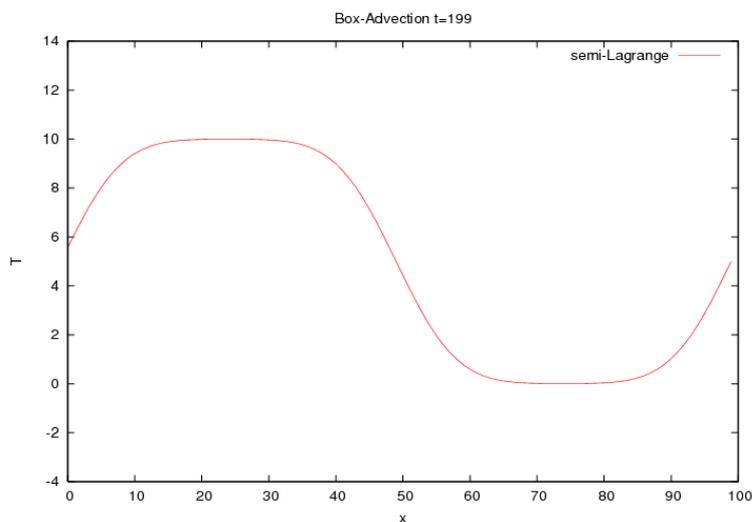
Aufgabe 4:

Zur Lösung dieser Aufgabe muss im Programm zur numerischen Lösung zunächst der Anfangswert angepasst werden. Da diesmal für die im Leapfrog Verfahren benötigte Lösung für $t = \Delta t$ keine analytische Lösung bereitsteht, muss diese anderweitig berechnet werden. In **advection_box.f90** geschied dies (einfach) durch Verwendung der Werte des upstream Verfahrens. Als Ergebnis sind hier die horizontale Verteilungen von T nach 199 Zeitschritten gezeigt (gleiche Einstellungen wie in Aufgabe 2). Wie zu sehen ist, zeigt das upstream Verfahren numerische Dispersion und Diffusion (Abflachen der Kanten) während das Leapfrog Verfahren (mit und ohne Filter), numerisches Rauchen zeigt (zu einfache Berechnung von $t=1$; numerischer Mode, der im gefilterten Fall kleiner ist), die Kanten aber etwas besser erhält.



Aufgabe 5:

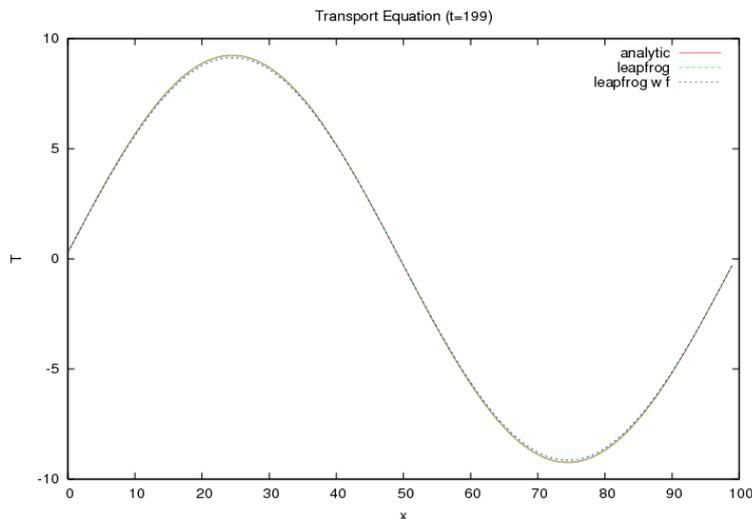
Das Muster-Programm ist **advection_sl.f90**. Als Beispiel ist hier die Advektion des Kastens aus Aufgabe 4 gezeigt (nach 199 Zeitschritten). Wie zu sehen ist, ist auch das Semi-Lagrange Verfahren diffusiv, was auf die räumliche (lineare) Interpolation in jedem Zeitschritt zurückzuführen ist. Genauere Interpolationsverfahren (z.B. Splines) liefern i. A. bessere Ergebnisse.



Lösungen zur Übung 3: 'Lineare Transportgleichung'

Aufgabe 1:

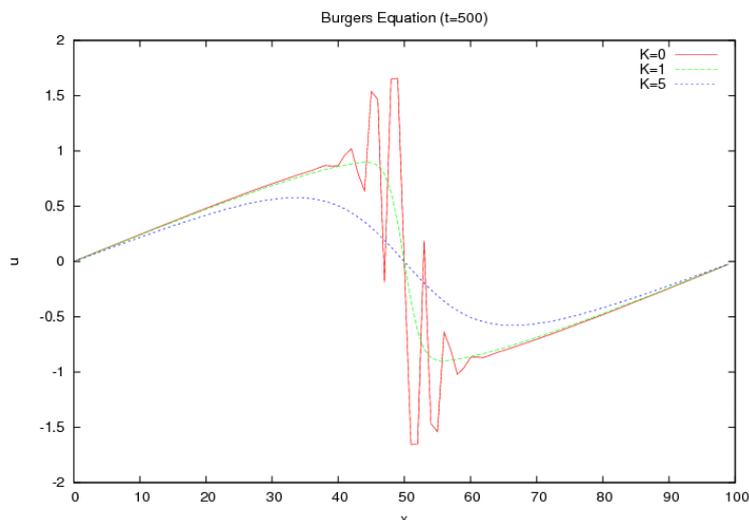
Um die Transportgleichung zu lösen, kann das Programm für die lineare Advektion (Beispiel: Sinus-Welle) um den Diffusiven Teil erweitert werden. Im Muster Programm **transport.f90** wird die time-splitting Methode, bestehend aus Leapfrog für die Advektion und FTCS für die Diffusion, verwendet. Als Ergebnis für den Diffusionskoeffizienten $K=2$ ergeben sich folgende (fast identische) Kurven nach 199 Zeitschritten für die drei Lösungen (analytisch, mit und ohne Filter):



Lösungen zur Übung 4: 'Burgers Gleichung (Nichtlineare Advektion/Transport)'

Aufgabe 1:

Hier kann die numerische Lösung der Transportgleichung als Vorlage genommen werden. Im Muster Programm **burgers.f90** wird die Burgers Gleichung mit dem Leapfrog/FTCS Verfahren für $K=0$ (unviskos), $K=1$ und $K=5$ gelöst. Hierbei wird eine energieerhaltende Diskretisierung für den Advektionsterm verwendet (s. Skript). Nach 500 Zeitschritten ergibt sich folgende Verteilung:



Während der unviskose Fall bereits im Bereich der unphysikalischen Lösung ist (s. Skript), zeigen die beiden viskosen Fälle ein unterschiedlich starkes Aufsteilen der Schockwelle.