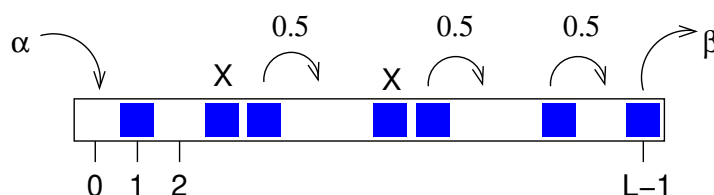


Übungen zur Computerorientierten Physik

1 Totally Asymmetric Exclusion Process (TASEP)

Das TASEP ist ein einfaches Modell für Transport im Nichtgleichgewicht.

Das System ist eine eindimensionale Kette von L Gitterplätzen $(0, \dots, L-1)$, die entweder leer (0) oder mit einem Teilchen besetzt sein können.



Die Dynamik des Systems wird durch folgende Regeln beschrieben, die jeweils innerhalb eines jeden “Sweeps” (Durchgangs) gelten, bei dem die Gitterplätze sinnvollerweise von $L-1$ startend bis 0 abgearbeitet werden:

1. Ist der letzte Gitterplatz besetzt, so verlässt das Teilchen das System mit Wahrscheinlichkeit β .
2. Teilchen innerhalb des Systems bewegen sich zum nächsten Gitterplatz (nur in positiver Richtung, daher “totally asymmetric”), sofern er frei ist, mit Wahrscheinlichkeit 0.5. Ist der nachfolgende Gitterplatz besetzt, bleibt ein Teilchen stehen.
3. Sofern der Gitterplatz 0 nicht besetzt ist, wird er mit Wahrscheinlichkeit α besetzt.

Wichtige Messgrößen sind die Dichte ρ (Anteil der besetzten Gitterplätze) und die Stromdichte j (mittlere Zahl der im System bewegten Teilchen pro Zeiteinheit und Gitterplatz.)

- Laden Sie sich das Programm `tasep_fragment.c` vom StudIP, das ein Skelett für das Programm bildet, das Sie vervollständigen müssen.
- Machen Sie sich mit den bisher implementierten Funktionen, Datenstrukturen und dem Hauptprogramm vertraut.

Hinweis: Alle Systemdaten sind in der Variable `sys` gespeichert, die vom selbstdefinierten Strukturtyp `tasep_sys_t` ist.

Das Programm können Sie mit `cc -o tasep tasep_fragment.c -lm -Wall` compilieren.

- Vervollständigen Sie die Funktion `tasep_sweep()`, so dass die oben angegebene Dynamik realisiert wird. Testen Sie Ihr Programm mit dem Debugger `gdb` indem Sie ein paar Schleifendurchläufe für ein kleines System schrittweise verfolgen (Argument in `gdb` mit `set args` angeben, Breakpoint mit `break` setzen, starten mit `run` und, nach Erreichen eines Breakpoints Schritte mit `step` machen, Variablen mit `print` ansehen).
- Entwerfen und implementieren Sie eine Funktion `tasep_density(...)`, die die Dichte des Systems (Anteil der besetzten Gitterplätze) zurück gibt. Übergeben Sie das System `sys` durch einen Pointer. Modifizieren Sie die Ausgabe im Hauptprogramm, so dass die Dichte mit ausgegeben werden.
- Erweitern Sie das Hauptprgramm, so dass nicht nur die jeweils aktuelle Dichte ρ und Stromdichte j , sondern auch die “laufenden” Mittelwerte $\bar{\rho}, \bar{j}$ über die jeweils bisherige Simulationszeit t ausgegeben wird. Führen Sie ggf. dazu weitere Variablen ein.
- Führen Sie Simulationen ($L = 100$ und “geeignete” (s.u.) Schrittzahl) für drei Sätze von Parameterwerten ($\alpha = 0.2, \beta = 0.1$), ($\alpha = 0.1, \beta = 0.2$) und ($\alpha = 0.7, \beta = 0.7$) durch und betrachten Sie $\rho(t)$, $j(t)$ bzw. die laufenden Mittelwerte (ggf. mit `gunplot` etc). Die Schrittzahl ist “geeignet”, wenn das System “equilibriert” ist, also laufende Mittelwerte verändern sich danach nicht mehr “wesentlich”. Was beobachten Sie für die stationären Werte $\lim_{t \rightarrow \infty} \bar{\rho}(t)$, $\lim_{t \rightarrow \infty} \bar{j}(t)$?
- Zusatzaufgaben:
Wie verhält sich die Equilibrierungszeit t_{eq} mit der Systemgröße L ?
Führend Sie systematisch Reihen von Simulationen mit Parametern α, β durch um ein Phasendiagramm zu bestimmen. Unterscheiden Sie dabei jeweils nach $\bar{\rho}, \bar{j}$ “klein” / “groß” (relativ).