Topology and geometry in chiral liquids

Zuiden, B.C. van; Zuiden B.C. van

Appendices

Appendix A Newton's laws of motion

In the seventeenth century Sir Isaac Newton published his famous work the *Philosophiæ Naturalis Principia Mathematica*♠ which laid the foundations of modern-day physics [106]. One of the most notable parts of the good book are the so-called *Newton's laws of motion*. Newton's laws of motion describe *any* classical system. Classical mechanics was born.

Newton's first law of motion is a steady-state condition. It states in layman terms that if a body is moving at a zero or constant velocity there are no net forces acting on that body. To quote:

> "*Every body perseveres in its state of rest, or of uniform motion in a right line, unless it is compelled to change that state by forces impressed thereon.*"

Which can be mathematically stated as:

$$\ddot{x}^{\mu} = 0 \quad \text{if and only if} \quad \sum_{i} F_i^{\mu} = 0. \tag{A.1}$$

Newton's second law of motion is a formulation of a differential equation. In layman terms it states that the acceleration of a body is proportional to the net forces on that body. Quote:

> "*The alteration of motion is ever proportional to the motive force impressed; and is made in the direction of the right line in which that force is impressed.*"

♠The original *Philosophiæ Naturalis Principia Mathematica* is written in Neo-Latin. The English translations cited here are found in Newton [168].

Or mathematically:

$$F^\mu = m\ddot{x}^\mu. \tag{A.2}$$

Equation (A.2) is colloquially called Newton's equations or Newton's equations of motion.

Finally, Newton's third law of motion tells us something about how bodies interact. In layman terms, if one body applies a force to another body, the other body will apply an equal and opposite force on the initial body. More colloquially: "action is reaction". To quote:

> *"To every action there is always opposed an equal reaction: or the mutual actions of two bodies upon each other are always equal, and directed to contrary parts."*

Mathematically stated as:

$$F_1^\mu = -F_2^\mu. \tag{A.3}$$

Appendix B The concept of energy

A useful concept in physics is the concept of energy. Energy is the ability to do work. Work is done when a force applied to a body changes the body's velocity. In classical mechanics there are two forms of energy. Firstly, there is the measure of energy due to velocity, which is called the kinetic energy $T(\dot{x}^\mu)$ and is defined as such:

$$T(\dot{x}^\mu) = \tfrac{1}{2}m\dot{x}^2. \tag{B.1}$$

Kinetic energy is basically the work needed to accelerate a body from zero motion to a given velocity. Secondly, there is a measure of energy due to position, which is called the potential energy[♠] $V(x^\mu)$ and typically depends on the interactions in a system. Colloquially, the potential energy can be described as the work that is needed for a body to obtain a certain position. Whatever the potential may be, the force generated due potential on the body is given by:

$$F^\mu\left(x^\rho\right) = -\delta^{\mu\nu}\partial_\nu V\left(x^\rho\right). \tag{B.2}$$

[♠] Sometimes referred to as just the *potential*.

Note that both kinetic and potential energy are defined per body. A system composed of multiple bodies has a total kinetic and potential energy given respectively by the sum of kinetic and potential energy of its individual bodies.

A useful concept to mention is that the sum of the total kinetic and potential energy of a system — often called the Hamiltonian or total energy and is denoted as $H = T + V$ — is always conserved if the system does not interact with another system. Mathematically, this can be shown by taking the time derivative of the Hamiltonian:

$$\frac{\mathrm{d}H}{\mathrm{d}t} = \frac{\mathrm{d}T}{\mathrm{d}t} + \frac{\mathrm{d}V}{\mathrm{d}t} = \dot{x}\left(m\ddot{x}^{\mu} + \delta^{\mu\nu}\partial_{\nu}V(x^{\rho})\right) = 0, \tag{B.3}$$

because we know from eqs. (B.2) and (A.2) that:

$$m\ddot{x}^{\mu} = -\delta^{\mu\nu}\partial_{\nu}V(x^{\rho}), \tag{B.4}$$

if and only if no non-conservative or external forces are present in the system. This means that the velocities in such a system increase if and only if the potential energy decreases. To change the value of the Hamiltonian for a system — in time — energy needs to be transfered from the system in question to another system.

Appendix C The stationary action principle

A century after Newton postulated his laws of motion, giving birth to classical mechanics, Joseph-Louis Lagrange reformulated classical mechanics for conservative systems. His reformulation is now often refered to *Lagrangian Mechanics*. Lagrange's reformulation is an example of a principle which is now know as the *least action principle* or more correctly *the stationary action principle*. We should stress that the stationary action principle is so powerful that it generates most of the equations in modern day advanced physics.

In order to explain the principle of stationary action we first have to define the action. The action is a functional♠ that takes any path the system can take. We will just impose that for the classical case the action $\mathcal{S}$ is defined by:

$$S(x^{\mu}) = \int_{t_i}^{t_f} L(x^{\mu}, \dot{x}^{\mu}, t)\mathrm{d}t, \tag{C.1}$$

♠Colloquially: a function of functions.

where L is called the *Lagrangian*. The principle of stationary actions then postulates that the equations of motion are found by finding the stationary points of the action. Mathematically, this is done by taking a functional derivative:

$$\delta \mathcal{S} = 0 \tag{C.2}$$

The stationary points of the action are found — or eq. (C.2) is satisfied — by applying the famous Euler–Lagrange equations:

$$\frac{\partial L}{\partial x^\mu} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{x}^\mu} = 0 \tag{C.3}$$

The Lagrangian L — implicitly defined in eq. (C.1) — depends on the physical problem. For conservative classical systems it turns out that the Lagrangian is always the difference between kinetic and potential energy:

$$L = T - V. \tag{C.4}$$

It is worth stating that the Lagrangian is coordinate-independent — it works in all coordinate systems — as it purely described by energetics which are coordinate-independent.

To summarize, applying the Euler–Lagrange to the classical Lagrangian will yield Newton's equations of motion for the particular problem. In the most general conservative classical case $L = \frac{1}{2}m\dot{x}^2 - V$, the Euler–Lagrange will yield:

$$m\ddot{x}^\mu = -\delta^{\mu\nu}\partial_\nu V. \tag{C.5}$$

Equation (C.5) is Newton's equation eq. (B.2) where all the forces are due to a potential — and thus conservative by definition — which can also be found by combining eqs. (A.2) and (B.2).

Appendix D Geodesic equation

In flat space♠ the *shortest* distance between two points, a^μ and b^μ, is known to be a straight line with a length found using the Pythagorean theorem:

$$s^2 = \delta_{\mu\nu}(b-a)^\mu(b-a)^\nu. \tag{D.1}$$

♠By flat space we mean space without curvature.

Another, more complicated way to find this path is done by integrating infinitesimal parts of it:

$$s^2 = \int_0^1 \delta_{\mu\nu} \dot{x}^\mu \dot{x}^\nu \mathrm{d}t, \tag{D.2}$$

where $x^\mu(t)$ is then given by:

$$x^\mu(t) = a^\mu + (b-a)^\mu t, \tag{D.3}$$

such that $x^\mu(0) = a^\mu$ and $x^\mu(1) = b^\mu$. The a priori unobvious benefit of writing it this way is that even though if we didn't know the path $x^\mu(t)$, it can always be found by minimizing the integral in eq. (D.2). We can minimize by using the Euler–Lagrange equations given in eq. (C.3) as we have previously done to find the stationary action in appendix C. The Euler–Lagrange equations will yield some set of differential equations, which once solved with the correct boundary conditions will yield the path that minimizes[♠] the distance between two points. This whole exercise seems a bit cumbersome, especially since in flat space it will always yield a straight line — the shortest path. Fortunately, it will, however, provide us with almost all the tools we need to start understanding curved space — or Riemannian geometry.

Let us, hence, enable curvature. Imagine you are walking on a flat field and suddenly a heavy earthquake smoothly deforms the surface. That is to say, the stresses produced by the earthquake stretches, compresses and bends the field — without any cuts or sharp creases. Although you will probably remain on the same spot on the field, your elevation might change. This is because the d-dimensional field is *embedded* in to a $(d+1)$-dimensional space.[♡] Suppose, our position on the field is given by x^μ and our position in the embedded space is given by $\vec{X}$. A infinitesimal distance between two points is given by:

$$\mathrm{d}s^2 = \partial_\mu \vec{X} \partial_\nu \vec{X} \mathrm{d}\dot{x}^\mu \mathrm{d}\dot{x}^\nu = g_{\mu\nu} \mathrm{d}\dot{x}^\mu \mathrm{d}\dot{x}^\nu. \tag{D.4}$$

Here, $g_{\mu\nu}$ is called the *metric tensor*. In in flat space it turns out $g_{\mu\nu} = \delta_{\mu\nu}$. In both eqs. (D.1) and (D.2) the $\delta_{\mu\nu}$ is the specific metric tensor in flat space. The metric tensor is a way to measure distances in our curved space but is also used to define the scalar product between two vectors on our curved space:

$$a^\mu b^\mu \equiv g_{\mu\nu} a^\mu b^\nu \tag{D.5}$$

[♠] There is no maximum for this problem, hence we can safely call the solution to Euler–Lagrange a minimum — although possibly degenerate.

[♡] If we want to imagine the earth's surface assume $d = 2$.

The metric tensor has a determinant often denoted as g and an inverse often denoted as $g^{\mu\nu}$. Although we have described what a tensor is for a two-dimensional surface embedded into three-dimensional space, all the equations we have used are valid in arbitrary dimensions.

To find the shortest distance between two points in our curved space we need to apply the Euler–Lagrange equations eq. (C.3) to $\mathrm{d}s^2$ in eq. (D.4). Doing so, results in the famous geodesic equations:

$$\ddot{x}^\mu + \Gamma^\mu_{\sigma\rho}\dot{x}^\sigma\dot{x}^\rho = 0. \tag{D.6}$$

Here $\Gamma^\mu_{\sigma\rho}$ are called the Christoffel symbols of the second kind, defined as such:

$$\Gamma^\mu_{\sigma\rho} = \tfrac{1}{2}g^{\mu\lambda}\left(\partial_\sigma g_{\lambda\rho} + \partial_\rho g_{\sigma\lambda} - \partial_\lambda g^{\sigma\rho}\right), \tag{D.7}$$

Christoffel symbols of the first kind $\Gamma_{\mu\sigma\rho}$ are defined as such:

$$\Gamma_{\mu\sigma\rho} = g_{\mu\lambda}\Gamma^\lambda_{\sigma\rho} = \tfrac{1}{2}\left(\partial_\sigma g_{\mu\rho} + \partial_\rho g_{\sigma\mu} - \partial_\mu g_{\sigma\rho}\right). \tag{D.8}$$

Christoffel symbols are *not* tensors — they change when the coordinate system changes. The geodesic equation generalizes our notion of a straight line on a curved space. In general a straight line is a line that minimizes the distance between two points.

Appendix E Geometry on a Monge patch

Suppose you are going for a stroll in some $(d+1)$-dimensional space confined on some d-dimensional hypersurface given by the at least $\mathcal{C}^2$ function $f : \mathbb{R}^d \to \mathbb{R}$. Your position is then given by $x^\mu \in \mathbb{R}^d$. This construction is called a *Monge patch* or *Monge gauge*, f is called the *Monge function*. The infinitesimal distance between two points is given by the Pythagorean theorem and is given by:

$$\begin{aligned}
\mathrm{d}s^2 &= \mathrm{d}f^2 + \sum_{i=1}^{d}\mathrm{d}x_i^2 \\
&= \left(\sum_{i=1}^{d}\partial_i f\,\mathrm{d}x_i\right)^2 + \sum_{i=1}^{d}\mathrm{d}x_i^2 \\
&= \sum_{i=1}^{d}(1+\partial_i f^2)\mathrm{d}x_i^2 + \sum_{i\neq j}\partial_i f\partial_j f\,\mathrm{d}x_i^2\mathrm{d}x_j^2.
\end{aligned} \tag{E.1}$$

From eq. (E.1) we conclude the metric tensor for such a Monge patch is given by:♠

$$g_{\mu\nu} = \delta_{\mu\nu} + f_\mu f_\nu. \tag{E.2}$$

Hence we can write:

$$\mathrm{d}s^2 = g_{\mu\nu}\mathrm{d}x^\mu \mathrm{d}x^\nu \tag{E.3}$$

Using Sylvester's determinant theorem, the determinant of the Monge metric is found to be:

$$g = 1 + \delta^{\mu\nu} f_\mu f_\nu, \tag{E.4}$$

where the repeating index is explicitly summed over. The metric inverse is found using the Sherman–Morrison formula and is given by:

$$g^{\mu\nu} = \delta^{\mu\nu} - \frac{1}{g}\delta^{\mu\rho} f_\rho \delta^{\nu\sigma} f_\sigma. \tag{E.5}$$

The Christoffel symbols can be found using the metric — and its inverse — using eq. (D.7) for eq. (E.3), or by solving the Euler–Lagrange equations eq. (C.3) with eq. (E.3) as the Lagrangian and then reading them of the geodesic equation given by eq. (D.6). In any case the Christoffel symbols of the first kind for a Monge patch are given by:

$$\Gamma_{\mu\sigma\rho} = f_\mu f_{\sigma\rho}. \tag{E.6}$$

Using Sylvester's determinant theorem again, we can find the eigenvalues of the metric by solving:♡

$$|(1-\lambda)\delta_{\mu\nu} + f_\mu f_\nu| = 0, \tag{E.7}$$

which yields:

$$\lambda_{1\ldots d-1} = 1 \quad \text{or} \quad \lambda_d = g, \tag{E.8}$$

which tells us that space can only be stretched on a Monge patch as compared to flat space, because $g \geq 1$, as one would expect.

♠Here, f_μ is shorthand notation for $f_\mu = \partial_\mu f$.
♡Here $|\ldots| = \det(\ldots)$ is the determinant.

Appendix F Fixed-point iterations

A common problem in mathematics is finding the zero values of some function $f : \mathbb{R} \to \mathbb{R}$, which is at least $\mathcal{C}^0$. More explicitly, for what values of x does $f(x) = 0$ hold? This is typically not a trivial task especially when x has a multiplicity.

A common, analytical, way to tackle the problem is to find or define some inverse function $f^{-1} : \mathbb{R} \to \mathbb{R}$ for which:

$$f^{-1}(f(x)) = x, \tag{F.9}$$

holds. It turns out, however, that f^{-1} does not always exist. Sometimes it is only valid for a subdomain of $\mathbb{R}$, or sometimes it does not exist at all. For instance, the function $f(x) = 0$ has no inverse, as any input value could produce zero. The function $f(x) = x^2$ has two inverse functions and from the answer — $f(x)$ — it is impossible to know which of the two to take:

$$f^{-1}(x) = \pm\sqrt{x}. \tag{F.10}$$

Fortunately, when studying most physics problems we usually know what solutions to expect from an equation. Even though sometimes the analytical ways are hard or impossible, numerical methods are a good alternative to find the answer. There are a multitude of numerical algorithms out there to solve this problem. Here, we will highlight one of these algorithms called the method of fixed-point iterations.

Suppose we have function $F : \mathbb{R} \to \mathbb{R}$ such that the x for which $F(x) = x$ also solves $f(x) = 0$. In that case we can use fixed-point iterations $x_{i+1} = F(x_i)$ to solve the problem given we guess x_0. Algorithmically, fixed-point iterations are implemented in the following way:

1. For $f(x) = 0$ find a corresponding contracting $F(x) = x$.
2. Make an initial guess x_0, close to x for which $f(x) = 0$.
3. Apply $x_{i+1} = F(x_i)$.
4. Repeat starting step 3 until $|x_{i+1} - x_i|$ is sufficiently small.

The fixed-point iteration converges if $F(x)$ is locally a *contraction mapping* in the neighborhood of x for which $F(x) = x$ or $f(x) = 0$, and where x_0 is sufficiently close to — in accordance to Banach's fixed-point theorem. To prove that a general $F(x)$ is a contraction mapping is a hard task. Therefore, fixed-points iterations are not necessarily a reliable way of solving the equation $f(x) = 0$, but when they work, they work very effectively.

As an example, consider the polynomial $f(x) = ax^2 + bx + c$ for which we would like to know the roots — the values of x for which $f(x) = 0$. If we rewrite $f(x) = 0$ as:

$$F(x) = x = -\frac{1}{a}\left(b + \frac{c}{x}\right), \tag{F.11}$$

we may perform fixed-points iterations on $F(x)$. A simple numerical test shows that this scheme converges for various arbitrary values of a, b, and c when $\sqrt{b^2 - 4ac} \in \mathbb{R}$, and when the initial value is chosen sufficiently close to a fixed point. An example program can be found in listing F.1.

It is worth pointing out that if x is multivalued — there are multiple distinct x's for which $f(x) = 0$ both depending on the attractiveness of the fixed point as well as where x_0 is chosen, the algorithm will converge to one of the solutions — provided the algorithm converges at all.

A benefit of fixed-point iterations is that it also work for higher dimensional problems of the form $f : \mathbb{R}^d \to \mathbb{R}^d$ where we have to find a corresponding $F^\mu(x^\mu) = x^\mu$, such that fixed-point iterations can be performed. For instance one could consider the function:

$$f^\mu(x^\mu) = \delta^{\mu\nu}\left(A_{\nu\sigma\rho}x^\sigma x^\rho - B_{\nu\sigma}x^\sigma + C_\nu\right) = 0, \tag{F.12}$$

with the fixed-point equation:

$$F^\mu(x^\mu) = \bar{B}^{\mu\sigma}\left(A_{\mu\sigma\rho}x^\sigma x^\rho + C_\mu\right) = x^\mu, \tag{F.13}$$

where $\bar{B}^{\mu\sigma}$ is the inverse of $B_{\mu\sigma}$.

Appendix G Numerical differentiation

A commonly overlooked tool in computational sciences is the method of *automatic differentiation*. Typically, numerical differentiation is done using the method of *finite differences*. Automatic differentiation, is an *exact* method of computing the derivative of a function for a given value. In order to understand how it works we first do a small recap of some high school math.

```cpp
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>
#include <limits>

using namespace std;
const double pi=acos(-1.0);
const double scale=5.0;
double a,b,c;

double f(double x)
{
    return a*pow(x,2)+b*x+c;
}

double F(double x)
{
    return -(b+c/x)/a;
}

double fxdpt(double x)
{
    int counter=0;
    double xp;
    do
    {
        x=F(xp=x);
        printf("%d: xp=%.17g, x=%.17g, F(x)=%.17g, f(x)=%.17g\n",counter,xp,x,F(x),f(x));
        counter++;
    }
    while(fabs(f(x))>numeric_limits<double>::epsilon() and counter<=0xFF);
    return x;
}

int main()
{
    srand48(time(NULL));
    double xpp,xmm;
    do
    {
        a=(2.0*drand48()-1.0)*scale;
        b=(2.0*drand48()-1.0)*scale;
        c=(2.0*drand48()-1.0)*scale;
        xpp=(-b+sqrt(pow(b,2)-4.0*a*c))/(2.0*a);
        xmm=(-b-sqrt(pow(b,2)-4.0*a*c))/(2.0*a);
    }
    while(isnan(xpp) or isnan(xmm));
    printf("a = %.17g, b = %.17g, c = %.17g\n",a,b,c);
    printf("x_+ = %.17g, x_- = %.17g\n",xpp,xmm);
    printf("f(x_+) = %.17g, f(x_-) = %.17g\n",f(xpp),f(xmm));
    const double miss=2.0*drand48();
    printf("x_0 = %.17g\n",miss*xpp);
    const double x=fxdpt(miss*xpp);
    printf("x=%.17g, f(x)=%.17g\n",x,f(x));
    if(fabs(x-xpp)<fabs(x-xmm)) printf("|x-x_+|=%.17g\n",fabs(x-xpp));
    else printf("|x-x_-|=%.17g\n",fabs(x-xmm));
    return EXIT_SUCCESS;
}
```

Listing F.1: An implementation for fixed-point iterations algorithm written in C++.

Suppose we have an at least $\mathcal{C}^1$ function $f : \mathbb{R} \to \mathbb{R}$. By definition the derivative $f'(x)$ is given by:

$$f'(x) \equiv \frac{\mathrm{d}f}{\mathrm{d}x} \equiv \lim_{h \to 0} \frac{1}{h} \left(f(x+h) - f(x) \right). \tag{G.1}$$

As an example, we can find the derivative of the function $f(x) = x^2$ by applying the definition in eq. (G.1):

$$\begin{aligned} f'(x) &= \lim_{h \to 0} \frac{1}{h} \left((x+h)^2 - x^2 \right) \\ &= \lim_{h \to 0} \frac{1}{h} \left(x^2 + h^2 + 2hx - x^2 \right) \\ &= \lim_{h \to 0} 2x + h \\ &= 2x. \end{aligned} \tag{G.2}$$

Higher-order derivatives can be found by re-applying eq. (G.1) up to the desired order. Say we are interested in the n^{th} order derivative, denoted by $f^n(x)$ or $\frac{\mathrm{d}^n f}{\mathrm{d}x^n}$, we just find the $(n-1)^{\text{th}}$ derivative, and then take its derivative. Since we know f, f^n can always be induced, if it exists. Hence, in our example we can find $f''(x)$ by taking the derivative of the previously found $f'(x) = 2x$ which yields: $f''(x) = 2$. One of the implications of eq. (G.1) is called Taylor's theorem which states that♠

$$f(x) = \sum_{n=0}^{\infty} \frac{1}{n!} \frac{\mathrm{d}^n f(a)}{dx^n} (x-a)^n, \tag{G.3}$$

where a is some arbitrary value in the domain of f.

It it worth mentioning that differentiation can be extended to work in higher dimensions as well.

G.1 Method of finite differences

Letting a computer find an exact derivative using eq. (G.1) is a very hard task. It is not a priori obvious whether a limit exists and what manipulations ought to be done to find the derivative. A straightforward solution to this problem is an approximate method called *finite difference*. The idea behind finite difference is roughly to omit the limit in eq. (G.1) and rather use an arbitrary small value for h instead. As this method is an approximate method it is always possible

♠As the following equation is purely a reminder, a proof will not be given here.

to improve the accuracy as is extensively done in Fornberg [158]. For brevity's sake we choose the most natural and trivial version of this method here. The simplest way of approximating a derivative numerically is done by:

$$f'(x) = \tfrac{1}{h}\left(f(x+h) - f(x)\right) + \mathcal{O}\left(h^2\right). \tag{G.4}$$

Now, let us take our previous example again. Suppose we want to know the value of the derivative in $x = 3$. We pick a small $h = 10^{-3}$ — for instance — and find:

$$\begin{aligned} f'(3) &\approx \frac{f(3+10^{-3}) - f(3)}{10^{-3}3} \\ &= \frac{9.006001 - 9}{10^{-3}} \\ &= \frac{0.006001}{10^{-3}} \\ &= 6.001. \end{aligned} \tag{G.5}$$

Obviously the value in eq. (G.5) is off by a little: the exact value should be $f'(3) = 6$ as found in eq. (G.2). Unfortunately, the finite difference method — even with higher accuracies — is a source of noise and artifacts in numerical simulations. Therefore an exact method would be very much preferred.

G.2 Symbolic differentiation

One way of achieving exact derivatives computationally is to hard code them into the computer code. This method becomes a somewhat tedious and error prone task when a lot of functions need to be differentiated. Another method would be to let the computer symbolically find all the derivatives of a function, and then generate and evaluate this function. Symbolic differentiation is done by taking standard operations and functions and applying eq. (G.1) to it. Typically for functions $f(x), g(x) : \mathbb{R} \to \mathbb{R}$, a table like table G.1 is found. This method is somewhat slow and hard to implement because a function needs to be parsed, then proper symbolic differentiation rules need to be applied, and then needs to be evaluated, which all end up being a complicated and computationally expensive task. Fortunately, there is yet another method that solves all these problems and this method is called automatic differentiation.

Name	Function	Derivative
Linear rule	af	af'
Sum rule	$f \pm g$	$f' \pm g'$
Product rule	fg	$f'g + fg'$
Quotient rule	$\frac{f}{g}$	$\frac{f'g - fg'}{g^2}$
Chain rule	$f(g)$	$f'(g)g'$
Constant rule	a	0
Power rule	x^n	nx^{n-1}
Exponential rule	a^x	$a^x \log a$
Logarithmic rule	$\log x$	$\frac{1}{x}$

Table G.1: Common rules for symbolic differentiation using the functions $f(x), g(x) : \mathbb{R} \to \mathbb{R}$.

Name	Operation	Result
Linear	$a(x + \varepsilon x')$	$ax + \varepsilon a x'$
Sum	$(x + \varepsilon x') \pm (y + \varepsilon y')$	$x \pm y + \varepsilon(x' \pm y')$
Product	$(x + \varepsilon x')(y + \varepsilon y')$	$xy + \varepsilon(x'y + xy')$
Quotient	$\frac{x+\varepsilon x'}{y+\varepsilon y'}$	$\frac{x}{y} + \varepsilon(\frac{x'y - xy'}{y^2})$

Table G.2: Basic mathematical operations for dual numbers.

G.3 Automatic differentiation

Automatic differentiation can be achieved by using *dual numbers*. We define dual numbers by introducing the following set[♠] $\mathbb{D} = \mathbb{R} \otimes \varepsilon\mathbb{R}$ such that $\varepsilon \neq 0$ and $\varepsilon^2 = 0$. Henceforth, we will write a that a dual number $\chi = x + \varepsilon x'$ where x is called the real part and x' is called the nilpotent part. Basic mathematical operations for dual numbers are listed in table G.2. Table G.2 reminds us somewhat of table G.1 and in fact it turns out that if we use Taylor's theorem,

[♠] This definition is similar to complex numbers where $\mathbb{C} = \mathbb{R} \otimes i\mathbb{R}$ such that $i^2 - 1$.

eq. (G.3), we find:

$$f(x+\varepsilon x') = \sum_{n=0}^{\infty} \frac{1}{n!} \frac{\mathrm{d}^n f(x)}{dx^n} (\varepsilon x')^n = f(x) + \varepsilon f'(x). \tag{G.6}$$

Meaning that whenever we take a function $f(x) : \mathbb{R} \to \mathbb{R}$ and promote it to $\bar{f}(x) : \mathbb{D} \to \mathbb{D}$ and evaluate it, the real part yields the function value, and the nilpotent value yields the function's derivative. We *automatically* obtain the function's derivative, just by evaluating the function for the appropriate dual number.

Computationally, the only thing that needs to be done is to *explain* to the computer how to work with dual numbers; explicitly, to define all the operators and special functions for dual numbers.♠

Automatic differentiation can be generalized to multivariable functions by promoting $f : \mathbb{R}^n \to \mathbb{R}^m$ to $f : \mathbb{D}^n \to \mathbb{D}^m$. Similarly, it can be used to find higher-order derivatives by introducing additional nilpotent variables.

Appendix H Numerical integration

Another commonly used tool of mathematics is *integration*. There are many ways to define integrals; we will use the Riemannian definition here:

$$\int_a^b f(x)\,\mathrm{d}x \equiv \lim_{h\to 0} h \sum_i^N f\left(a + hi\right), \tag{H.1}$$

where N is given by $N = \frac{b-a}{h}$. Informally, the integral is a measure of the sum of the areas between $f(x)$ and the x-axis. An often used concept is the *primitive* function $F(x)$ defined as such:

$$F(x) \equiv \int_0^x f(\xi)\,\mathrm{d}\xi. \tag{H.2}$$

The primitive is sometimes called the *antiderivative* due to the fundamental theorem of calculus which states:

$$f(x) = \int_0^x f'(\xi)\,\mathrm{d}\xi. \tag{H.3}$$

♠ Analogous to the way complex numbers are implemented on a computer.

An integral is linear in the following sense:

$$\int_a^b f(x)\,\mathrm{d}x + \int_b^c f(x)\,\mathrm{d}x + \int_a^c f(x)\,\mathrm{d}x, \tag{H.4}$$

and for a $\alpha \in \mathbb{R}$, constant:

$$\int_a^b \alpha f(x)\,\mathrm{d}x = \alpha \int_a^b f(x)\,\mathrm{d}x. \tag{H.5}$$

Numerically a generic integral can be approximated through various methods. We will highlight a few of these methods. The following methods assume that $t_{n+1} = t_n + h$ for small h. The methods can, however, be composited due to the linearity properties of the integral to integrate over larger intervals. First, the rectangle method, or rectangle rule:

$$\int_{t_n}^{t_{n+1}} f(t)\,\mathrm{d}t \approx h f(t_n), \tag{H.6}$$

which is a simple discretization of the definition of the integral, eq. (H.1). Second, the trapezoid method, or trapezoid rule:

$$\int_{t_n}^{t_{n+1}} f(t)\,\mathrm{d}t \approx \frac{h}{2}\left(f(t_{n+1}) + f(t_n)\right). \tag{H.7}$$

Finally, the Simpson method, or Simpson's rule:

$$\int_{t_n}^{t_{n+1}} f(t)\,\mathrm{d}t \approx \frac{h}{6}\left(f(t_{n+1}) + 4f\left(\tfrac{h}{2}\right) + f(t_n)\right). \tag{H.8}$$

Appendix I Forward Euler

One of the most trivial and common methods to solve differential equations is called *Forward Euler* [170]. Suppose we have the ordinary differential equation given by

$$\dot{x}^\mu(t) = f^\mu(x^\rho, t), \quad x(0) = 0, \tag{I.1}$$

for some arbitrary function $f(x) : \mathbb{R}^{d+1} \to \mathbb{R}^d$, using the definition of derivatives given in eq. (G.1) with a finite h — resembling the finite differences method discussed in appendix section G.1. Here, h is called the *time step* because we define the recurrent equation: $t_{i+1} = t_i + h$, where t_i is the time

series, the discrete variable representing t. Here, all series have an index i representing the number of time steps taken. The initial value of the index is zero, $i = 0$. We can then rewrite eq. (I.1) as such:

$$x^{\mu}_{i+1} = x^{\mu}_{i} + hf^{\mu}(x^{\rho}_{t}, t_i) + \mathcal{O}(h^2), \quad x^{\mu}_{0} = 0, \tag{I.2}$$

yielding a recurrent formula, which can be computed iteratively since we know the initial value x_0. As $h \to 0$ in eq. (I.2) the answer becomes more and more exact, although in principle corrections of order $\mathcal{O}(h^2)$ can always be applied.

When applying Forward Euler to a second order differential equation:

$$\ddot{x}^{\mu}(t) = f^{\mu}(\dot{x}^{\rho}, x^{\sigma}, t), \quad x^{\mu}(0) = 0, \quad \dot{x}^{\mu}(0) = 0 \tag{I.3}$$

for some arbitrary function $f(x) : \mathbb{R}^{2d+1} \to \mathbb{R}^{d}$, we first need to split one second-order differential equation to two first-order differential equations. We do so by introducing a *dummy* variable $y^{\mu} = \dot{x}^{\mu}$, substitution yields:

$$\begin{aligned} \dot{x}^{\mu}(t) &= y^{\mu}, & x^{\mu}(0) &= 0 \\ \dot{y}^{\mu}(t) &= f^{\mu}(y^{\rho}, x^{\sigma}, t), & y^{\mu}(0) &= 0, \end{aligned} \tag{I.4}$$

and we have turned our problem into a system of two first-order differential equations. Now we once again use the method of finite differences to find a recurrent solution — analogous to eq. (I.2) — which is now given by:

$$\begin{aligned} x^{\mu}_{i+1} &= x^{\mu}_{i} + hy^{\mu}_{i} + \mathcal{O}(h^2), & x^{\mu}_{0} &= 0 \\ y^{\mu}_{i+1} &= y^{\mu}_{i} + hf^{\mu}(y^{\rho}_{i}, x^{\sigma}_{i}, t_i), & y^{\mu}_{0} &= 0. \end{aligned} \tag{I.5}$$

Undoing the substation of $y^{\mu} = \dot{x}^{\mu}$ turn eq. (I.5) into:

$$\begin{aligned} x^{\mu}_{i+1} &= x^{\mu}_{i} + h\dot{x}^{\mu}_{i} + \mathcal{O}(h^2), & x^{\mu}_{0} &= 0 \\ \dot{x}^{\mu}_{i+1} &= \dot{x}^{\mu}_{i} + hf^{\mu}(\dot{x}^{\rho}_{i}, x^{\sigma}_{i}, t_i), & \dot{x}^{\mu}_{0} &= 0. \end{aligned} \tag{I.6}$$

As eqs. (I.2) and (I.6) are similarly derived, in both of the cases the answer is approximate up to order $\mathcal{O}(h^2)$. Equation (I.3) and therefore eq. (I.6) are equal to Newton's equations of motion, eq. (A.2), if $f^{\mu}(\dot{x}^{\rho}_{i}, x^{\sigma}_{i}, t_i) = \frac{1}{m}F^{\mu}_{i}$, where F^{μ}_{i} is the total force at timestep i on the body of interest with mass m. For conservative systems, $f^{\mu}(\dot{x}^{\rho}_{i}, x^{\sigma}_{i}, t_i) = -\frac{1}{m}\delta^{\mu\nu}\partial^{i}_{\nu}V(x^{\lambda}_{i})$, as found using eq. (B.4).

This method is called *forward* Euler because we have used forward differences to derive eqs. (I.2) and (I.6). Backward differences can also be used, this would be called *backward* Euler. The downside of backward Euler is that it typically yields an *implicit* algebraic equation that ought to be solved before continuing — often done using an iterative numerical scheme. Hence, backward Euler and forward Euler are respectively called implicit and explicit integrators. The upside of an explicit integrator is that it typically performs faster because no additional iterations are required to solve the implicit algebraic equation. The upside of an implicit integrator is that they are numerically more *stable* compared to explicit integrators. A numerical solution to an equation is unstable when it diverges — rapidly — from the analytic solution.

Appendix J Runge–Kutta's method

In appendix I we discovered how to solve a differential equation numerically, in a iterative fashion. We also discussed that the numerical is approximate up to first order, $\mathcal{O}(h)$. Since we solve the equation iteratively, that is to say for small time steps h, the error in our solution will accumulate over large time intervals. This is a problem both for the relevance and stability of a solution. An obvious solution is to decrease the time step h. The downside to this is, however, that for every decrement in the time step h we have to increase the number of iterations to yield the physical time as $t_{t+1} = t_t + h$. If we decrease h by a factor of 10, the computer has to do a factor of 10 more iterations and is hence a factor 10 slower.

Fortunately, we can do a higher-order approximation to the derivative, using the method of finite differences, and some additional tweaking. One popular way of doing that is called the Runge–Kutta method. For the equation:

$$\ddot{x}^\mu(t) = f^\mu(x^\mu, t), \quad x^\mu(0) = 0, \quad \dot{x}^\mu(0) = 0 \tag{J.1}$$

The Runge–Kutta method gives the solution:

$$\begin{aligned} x^\mu_{i+1} &= x^\mu_i + h\left(\dot{x}^\mu_i + \tfrac{1}{6}(k^\mu_1 + 2k^\mu_2)\right) + \mathcal{O}(h^4), & x^\mu_0 &= 0 \\ \dot{x}^\mu_{i+1} &= \dot{x}^\mu_i + \tfrac{1}{6}k^\mu_1 + \tfrac{1}{6}k^\mu_2 + \tfrac{1}{6}k^\mu_3, & \dot{x}^\mu_0 &= 0, \end{aligned} \tag{J.2}$$

with:

$$\begin{aligned} k^\mu_1 &= hf^\mu(x^\mu_i, t_i) \\ k^\mu_2 &= hf^\mu(x^\mu_i + \tfrac{h}{2}\dot{x}^\mu_i + \tfrac{h}{8}k^\mu_1, t_i + \tfrac{h}{2}) \\ k^\mu_3 &= hf^\mu(x^\mu_i + h\dot{x}^\mu_i + \tfrac{h}{2}k^\mu_2, t_i + h). \end{aligned} \tag{J.3}$$

The Runge–Kutta method solves Newton's equation for conservative forces eq. (B.2) up to third order accuracy $\mathcal{O}(h^3)$. Therefore this version of Runge–Kutta is also refered to as the Runge–Kutta 3 method.

Appendix K Symplectic Euler

Symplectic Euler is one of the simplest symplectic integrators that can be derived with the method of variational integration — as described in section 4.2. As discussed in appendix C, the Lagrangian for classical systems is given by $L = \frac{1}{2}m\dot{x}^2 - V$, see eq. (C.4). In order to discretize the Lagrangian we need to perform the integral in eq. (4.7). For the symplectic Euler method we will use the rectangle method for approximating integrals as discussed in appendix H, eq. (H.6). We will, additionally, discretize the velocity by approximating the numerical derivative as discussed in appendix section G.1, eq. (G.4), which yields:

$$\dot{x}^\mu = \tfrac{1}{h}\left(x^\mu_{i+1} - x^\mu_i\right). \tag{K.1}$$

Hence, we find the discrete Lagrangian for symplectic Euler to be given:

$$L_D^{\text{rect}}(x^\mu_i, x^\mu_{i+1}) = \tfrac{m}{2h}\left(\delta_{\mu\nu}x^\mu_{i+1}x^\nu_{i+1} - 2\delta_{\mu\nu}x^\mu_i x^\nu_{i+1} + \delta_{\mu\nu}x^\mu_i x^\nu_i\right) - hV_i. \tag{K.2}$$

Before we continue to apply the discrete Euler–Lagrange, we first establish the following identities:

$$\partial^i_\sigma \delta_{\mu\nu}x^\mu_j x^\nu_k = \delta_{\mu\nu}\left(\delta_{ji}\delta^\mu_\sigma x^\nu_k + \delta_{ki}x^\mu_j\delta^\nu_\sigma\right). \tag{K.3}$$

Applying the discrete Euler–Lagrange equations, eq. (4.11), to our discrete Lagrangian eq. (K.2) then yields:

$$\tfrac{m}{h}\left(2x^\mu_i - x^\mu_{i+1} - x^\mu_{i-1}\right) - h\delta^{\mu\sigma}\partial^i_\sigma V_i = 0. \tag{K.4}$$

When further simplyfing eq. (K.4), we find:

$$x^\mu_{i+1} = x^\mu_i + h\dot{x}^\mu_i - \tfrac{h^2}{m}\delta^{\sigma\mu}\partial^i_\sigma V_i. \tag{K.5}$$

Upon computing eq. (4.12), for this discrete Lagrangian, eq. (K.2) we obtain the following result:

$$\dot{x}^\mu_{i+1} = \tfrac{1}{h}\left(x^\mu_{i+1} - x^\mu_i\right), \tag{K.6}$$

which is somewhat remarkable as one might have expected it to be $\dot{x}_i^\mu$ instead. Upon substitution we find the following discrete equations of motion:

$$\dot{x}_{i+1}^\mu = \dot{x}_i^\mu - \tfrac{h}{m}\delta^{\sigma\mu}\partial_\sigma^i V_i, \tag{K.7}$$
$$x_{i+1}^\mu = x_i^\mu + h\dot{x}_{i+1}^\mu. \tag{K.8}$$

When invoking eq. (B.2), eq. (K.7) becomes:

$$\dot{x}_{i+1}^\mu = \dot{x}_i^\mu + \tfrac{h}{m}F_i^\mu, \tag{K.9}$$
$$x_{i+1}^\mu = x_i^\mu + h\dot{x}_{i+1}^\mu. \tag{K.10}$$

for non-conservative systems.

Appendix L Velocity Verlet

The velocity Verlet method is a symplectic method with higher order accuracy than Symplectic Euler. It can be found using the trapezoidal method eq. (H.7) when discretizing the Lagrangian — as we saw, when the rectangle method is invoked, we obtain symplectic Euler (see appendix K). For brevity we will impose the well known velocity Verlet equation here:

$$x_{i+1}^\mu = x_i^\mu + h\dot{x}_i^\mu + \tfrac{h^2}{2m}F^\mu(x_i^\mu), \tag{L.1}$$
$$\dot{x}_{i+1}^\mu = \dot{x}_i^\mu + \tfrac{h}{2m}\left(F^\mu(x_i^\mu) + F^\mu(x_{i+1}^\mu)\right), \tag{L.2}$$

For the velocity in eq. (L.2) the forces F_{i+1}^μ for the new positions have to be recomputed — this has a bad computational performance impact. In addition, note how the force $F^\mu(x_{i+1}^\mu)$ is solely a function of position x_{i+1}^μ, as we are calculating the new velocity $\dot{x}_{i+1}^\mu$. If the force would be dependent on the velocity $\dot{x}^\mu$ — which is the case for active or damped systems — eq. (L.2) would become implicit, which is yet another computational performance impact.

Appendix M The discrete conservation of energy

In this appendix we compare the energy increment per time step of forward Euler and symplectic Euler. The discrete Hamiltonian at timestep i, H_i, is given by:

$$H_i = \tfrac{1}{2}m\dot{x}_i^2 + V\left(x_i^\mu\right). \tag{M.1}$$

Therefore, H_{i+1} is given by:

$$H_{i+1} = \tfrac{1}{2} m \dot{x}_{i+1}^2 + V\left(x_{i+1}^\mu\right). \tag{M.2}$$

Upon computing the Hamiltonian difference $\Delta H = H_{i+1} - H_i$, we find the energy increment per timestep

$$\Delta H = \tfrac{1}{2} m \left(\dot{x}_{i+1}^2 - \dot{x}_i^2\right) + V\left(x_{i+1}^\mu\right) - V\left(x_i^\mu\right). \tag{M.3}$$

We do so for both forward Euler and symplectic Euler.

M.1 Forward Euler

The forward Euler equations for conservative systems are found in appendix I and are given by:

$$\dot{x}_{i+1}^\mu = \dot{x}_i^\mu - \tfrac{h}{m} \delta^{\mu\nu} \partial_\nu^i V\left(x_i^\lambda\right), \tag{M.4}$$

$$x_{i+1}^\mu = x_i^\mu + h \dot{x}_i^\mu. \tag{M.5}$$

We, hence, compute the Hamiltonian difference for forward Euler ΔH^f and initially find:

$$\begin{aligned} \Delta H^f = &- h \dot{x}_i^\lambda \delta_{\lambda\mu} \delta^{\mu\nu} \partial_\nu^i V\left(x_i^\lambda\right) + \tfrac{h^2}{2m} \left(\delta^{\mu\nu} \partial_\nu^i V\left(x_i^\lambda\right)\right)^2 \\ &+ V\left(x_i^\lambda + h \dot{x}_i^\lambda\right) - V\left(x_i^\lambda\right), \end{aligned} \tag{M.6}$$

since:

$$\tfrac{1}{2} m \left(\dot{x}_{i+1}^2 - x_i^2\right) = -h \dot{x}_i^\lambda \delta_{\lambda\mu} \delta^{\mu\nu} \partial_i^\mu V\left(x_i^\lambda\right) + \tfrac{h^2}{2m} \left(\delta^{\mu\nu} \partial_\nu V\left(x_i^\lambda\right)\right)^2. \tag{M.7}$$

Upon Taylor expanding the potential term up to second order in h, we find:

$$\Delta H^f = \tfrac{h^2}{2m} \left(\delta^{\mu\nu} \partial_\nu^i V\left(x_i^\lambda\right)\right)^2 + \tfrac{h^2}{2} \dot{x}_i^\mu \dot{x}_i^\nu \partial_\mu^i \partial_\nu^i V\left(x_i^\lambda\right) + \mathcal{O}\left(h^3\right), \tag{M.8}$$

$$= \tfrac{1}{2} m \left(\dot{x}_{i+1}^\mu - \dot{x}_i^\mu\right)^2 + \tfrac{h^2}{2} \dot{x}_i^\mu \dot{x}_i^\nu \partial_\mu^i \partial_\nu^i V\left(x_i^\lambda\right) + \mathcal{O}\left(h^3\right), \tag{M.9}$$

as the Taylor expansion yields:

$$\begin{aligned} V\left(x_i^\lambda + h \dot{x}_j^\lambda\right) = &V\left(x_i^\lambda\right) + h \dot{x}_j^\lambda \delta_{\lambda\mu} \delta^{\mu\nu} \partial_\nu V\left(x_i^\lambda\right) \\ &+ \tfrac{h^2}{2} \dot{x}_j^\mu \dot{x}_j^\nu \partial_\mu^i \partial_\nu^i V\left(x_i^\lambda\right) + \mathcal{O}\left(h^3\right). \end{aligned} \tag{M.10}$$

M.2 Symplectic Euler

The symplectic Euler equations for conservative systems are found in appendix K and are given by:

$$\dot{x}^{\mu}_{i+1} = \dot{x}^{\mu}_{i} - \tfrac{h}{m}\delta^{\mu\nu}\partial^{i}_{\nu}V\left(x^{\lambda}_{i}\right), \tag{M.11}$$

$$x^{\mu}_{i+1} = x^{\mu}_{i} + h\dot{x}^{\mu}_{i+1}. \tag{M.12}$$

We, hence, also compute the Hamiltonian difference for symplectic Euler ΔH^{s} and similarly eq. (M.6) initially find:

$$\begin{aligned}\Delta H^{s} = &- h\dot{x}^{\lambda}_{i}\delta_{\lambda\mu}\delta^{\mu\nu}\partial^{i}_{\nu}V\left(x^{\lambda}_{i}\right) + \tfrac{h^{2}}{2m}\left(\delta^{\mu\nu}\partial^{i}_{\nu}V\left(x^{\lambda}_{i}\right)\right)^{2} \\ &+ V\left(x^{\lambda}_{i} + h\dot{x}^{\lambda}_{i+1}\right) - V\left(x^{\lambda}_{i}\right).\end{aligned} \tag{M.13}$$

Upon Taylor expanding the potential term up to second order in h, eq. (M.10) and substituting $\dot{x}^{\mu}_{i+1}$ given by eq. (M.12), we find:

$$\Delta H^{s} = -\tfrac{h^{2}}{2m}\left(\delta^{\mu\nu}\partial^{i}_{\nu}V\left(x^{\lambda}_{i}\right)\right)^{2} + \tfrac{h^{2}}{2}\dot{x}^{\mu}_{i}\dot{x}^{\nu}_{i}\partial^{i}_{\mu}\partial^{i}_{\nu}V\left(x^{\lambda}_{i}\right) + \mathcal{O}\left(h^{3}\right), \tag{M.14}$$

$$= -\tfrac{1}{2}m\left(\dot{x}^{\mu}_{i+1} - \dot{x}^{\mu}_{i}\right)^{2} + \tfrac{h^{2}}{2}\dot{x}^{\mu}_{i}\dot{x}^{\nu}_{i}\partial^{i}_{\mu}\partial^{i}_{\nu}V\left(x^{\lambda}_{i}\right) + \mathcal{O}\left(h^{3}\right). \tag{M.15}$$

At this point we observe that the kinetic part of the equation vanishes when mechanical equilibrium is assumed and we indeed find energetic conservation [103, 129, 136].

Appendix N Conjugate Gradient

A widely used way to minimize a scalar function $f(x^{\mu}) : \mathbb{R}^{d} \to \mathbb{R}$ is by the so called conjugate gradient method. The conjugate gradient method relies on fact that

$$-\delta^{\mu\sigma}\partial_{\sigma}f(x^{\mu})$$

is the direction of steepest descent, hence, $f(x^{\mu})$ ought to be at least $\mathcal{C}^{1}$ now. Essentially, we could always walk in the direction of steepest descent until we reach the nearest valley in that direction and repeat until we are stuck at a point where the gradient is zero valued — the minimum.[♠] This is roughly

[♠] We are descending, we should not get stuck at any other stationary point.

how the conjugate method works. It improves, however, on the number of iterations we have to do before reaching the minimum. It does so by predicting the direction of the valley's descent instead of crisscrossing through it. This is achieved as follows:

1. Calculate the steepest descent vector $d_i^\mu = -\delta^{\mu\sigma}\partial_\sigma f(x_i^\mu)$.
2. Find the conjugate direction vector $\Delta_i^\mu = d_i^\mu + C_i\Delta_{i-1}^\mu$.
3. Perform a line minimization in the direction of Δ_i^μ and find the optimal λ in:

$$\lambda = \arg\min_\lambda f(x_i^\mu + \lambda\Delta_i^\mu), \tag{N.1}$$

   and $x^{i+1} = x_i^\mu + \lambda\Delta_i^\mu$ accordingly.
4. If $|\delta^{\mu\sigma}\partial_\sigma f(x_i^\mu)|$ is not sufficiently small go to step 1.

To kickstart the algorithm we assume $\Delta_0^\mu = d_0^\mu$. The C_i in step 2 is found by one of the following definitions:

- Fletcher–Reeves:

$$C_i^{\text{FR}} = \frac{\delta_{\mu\nu}x_i^\mu x_i^\nu}{\delta_{\mu\nu}x_{i-1}^\mu x_{i-1}^\nu}. \tag{N.2}$$

- Polak–Ribiére:

$$C_i^{\text{PR}} = \frac{\delta_{\mu\nu}x_i^\mu\left(x_i^\nu - x_{i-1}^\nu\right)}{\delta_{\mu\nu}x_{i-1}^\mu x_{i-1}^\nu}. \tag{N.3}$$

- Hestenes–Stiefel:

$$C_i^{\text{HS}} = \frac{\delta_{\mu\nu}x_i^\mu\left(x_i^\nu - x_{i-1}^\nu\right)}{\delta_{\mu\nu}\Delta_{i-1}^\mu\left(x_i^\nu - x_{i-1}^\nu\right)}. \tag{N.4}$$

- Dai–Yuan:

$$C_i^{\text{DY}} = \frac{\delta_{\mu\nu}x_i^\mu x_i^\nu}{\delta_{\mu\nu}\Delta_{i-1}^\mu\left(x_i^\nu - x_{i-1}^\nu\right)}. \tag{N.5}$$

For the first iteration we may use $C_0 = 1$. Each of these definitions has its own benefits; typically it is easy to implement a few and see which one performs better for your problem although in most cases the choice should not matter too much.

Appendix O Molecular dynamics

In the eyes of a physicist — possibly a theoretical soft matter physicist — the method of molecular dynamics [117, 144] is nothing but the generalized classical N-body problem solved computationally. The generalized classical N-body problem is basically N bodies interacting in some way. In the most general case we end up with a dynamical system of coupled equations:

$$m_n \ddot{x}_n^\mu = F_n^\mu \left(x_0^\mu, \dots, x_{N-1}^\mu \right), \tag{O.1}$$

where n denotes the n^{th} particle in the system and is the total force acting on the n^{th} particle is given by $F_n^\mu \left(x_0^\mu, \dots, x_{N-1}^\mu \right)$. In the presence of curvature, the equations of motion for the dynamical system are given by:

$$m_n \left(\ddot{x}_n^\mu + \Gamma_{\sigma\rho}^{\mu\, n} \dot{x}_n^\sigma \dot{x}_n^\rho \right) = F_n^\mu \left(x_0^\mu, \dots, x_{N-1}^\mu \right). \tag{O.2}$$

We have discussed symplectic methods for solving eq. (O.1) in chapter 4, particularly section 4.2 provided we discretize time. Depending on the type of system, we may choose Velocity Verlet as described in appendix L for conservative systems. For non-conservative we choose symplectic Euler appendix K, and in the presence of curvature we may use a modified variant of symplectic Euler as described in section 4.3. Obviously, in order to perform these methods initial conditions per particle are a necessity.

Although we are solving Newton's equations for every individual particle, we are usually not necessarily interested in the individual trajectories of these particles. Usually, we are interested in the statistical behavior of the system as a whole. Hence, molecular dynamics should be seen a method for solving statistical physics-type questions. As we are interested in statistical effects we choose N, the number, of particles to be large — canonically of the order one hundred to one hundred thousand.

Typically, interactions between particles are assumed to be pairwise, meaning a particle i interacts with a particle j yielding a force on particle i in the most general form given by:

$$\begin{aligned} F_{ij}^\mu \left(x_i^\mu, x_j^\mu \right) &= -g_i^{\mu\nu} \partial_\nu^i V_{ij} \left(\mathrm{d} \left(x_j^\mu, x_i^\mu \right) \right) \\ &= -g_i^{\mu\nu} \frac{V_{ij} \left(\mathrm{d} \left(x_j^\mu, x_i^\mu \right) \right)}{\mathrm{d} \left(x_j^\mu, x_i^\mu \right)} \frac{\mathrm{d}^\nu \left(x_j^\mu, x_i^\mu \right)}{\mathrm{d} \left(x_j^\mu, x_i^\mu \right)}, \end{aligned} \tag{O.3}$$

where $\mathrm{d}\left(x_j^\mu, x_i^\mu\right)$ is the distance between particle i and j and $\mathrm{d}^\mu\left(x_j^\mu, x_i^\mu\right)$ is the distance vector pointing from particle i towards particle j. Obviously, by Newton's third law, eq. (A.3) — which is only valid if $g_i^{\mu\nu} = g_j^{\mu\nu}$; which is the case in flat space — the force on particle j is given by:

$$F_{ji}^\mu\left(x_j^\mu, x_i^\mu\right) = -F_{ij}^\mu\left(x_i^\mu, x_j^\mu\right). \tag{O.4}$$

Finally, to find the total resulting forces on particle i we take the sum:

$$F_i^\mu = F_{\text{add}}^\mu - g_i^{\mu\nu}\partial_\nu^i \sum_{j\neq i}^{N} V_{ij}\left(\mathrm{d}\left(x_j^\mu, x_i^\mu\right)\right), \tag{O.5}$$

where F_{add}^μ are additional non-conservative forces.

As N increases we observe an computational performance problem. In order to find the resultant forces on particle i — given that there are interactions — we need to perform N^2 calculations. Hence, the complexity of solving Newton's equations of motion for N interaction particles scales as $\mathcal{O}\left(N^2\right)$. Fortunately, however, in many physical systems it turns out that the pairwise interaction potential $V_{ij}(r)$ is small as r increases. Such potentials are called short range. For short range potentials the sum in eq. (O.5) can be truncated to be only a sum over its neighbors:

$$\begin{aligned} F_i^\mu &= F_{\text{add}}^\mu - g_i^{\mu\nu}\partial_\nu^i \sum_{\langle ij\rangle}^{N} V_{ij}\left(\mathrm{d}\left(x_j^\mu, x_i^\mu\right)\right) \\ &= F_{\text{add}}^\mu - g_i^{\mu\nu}\partial_\nu^i \sum_{i\neq j}^{N} V_{ij}^{\text{tr}}\left(\mathrm{d}\left(x_j^\mu, x_i^\mu\right)\right), \end{aligned} \tag{O.6}$$

such that the truncated interaction potential $V_{ij}^{\text{tr}}(r_{ij})$ is given:

$$V_{ij}^{\text{tr}}(r_{ij}) = \begin{cases} V_{ij}(r_{ij}) - V_{ij}(r_c) & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c \end{cases}, \tag{O.7}$$

for some cutoff radius r_c. Defining the truncated potential as is done in eq. (O.7) ensures continuity of the potential energy function. Note how adding a constant $V_{ij}(r_c)$ to the potential does not affect force as we are taking the derivative. If r_c is chosen appropriately for the short range potential, the speed

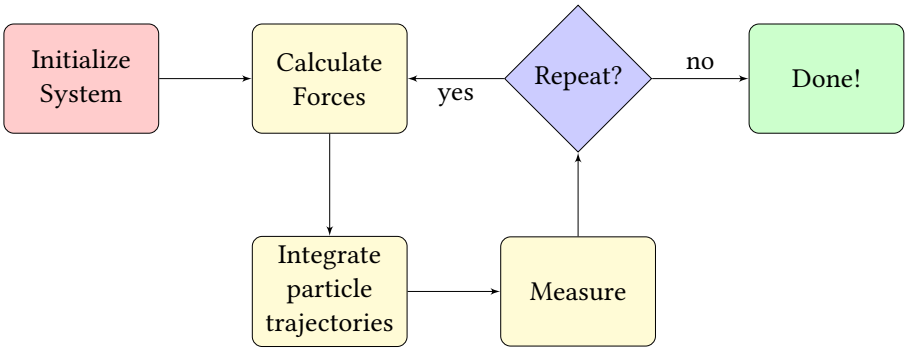

Figure O.1: A simple flowchart depicting how molecular dynamics is roughly done.

up could be significant, whereas the physical error is near insignificant — as interactions at large distances would have less influence than the numerical noise. For more about how to implement this efficiently, see appendix section O.2. For long range interactions with periodic boundaries Ewald summations are required.

Given the trajectory integration method with initial conditions, and the force calculation method, we can start implementing the Molecular dynamics algorithm as depicted in fig. O.1. The Molecular dynamics algorithm evaluates our system from $t_i \to t_{i+1} = t_i + h$ as an approximation to the real Newtonian dynamics. We can repeat this for as long as necessary, taking how many time steps as we want. As we are doing physics, we can measure certain — statistical — properties of interest in-between time steps — or possible more efficiently, just collect all the particles' positions and velocities for post-processing of the data later.

O.1 Boundary conditions

Another important ingredient of molecular dynamics is the boundary conditions. In order to perform the algorithm we have to define a coordinate system in d dimensions. To prevent values from overflowing — and particles from drifting off — boundary conditions are often required. There are occasions where boundary conditions do not matter. For instance, when the particles are attractive at longer ranges and there is no drift in the system, there is no reason to impose a boundary. When the particles are repulsive at longer ranges,

however, imposing a boundary is unavoidable. If no boundary is imposed the particles will scatter and possibly shoot off until a numerical overflow is encountered. Hence it is convenient to impose a simulation box in which the particles can be confined.

We may start by building the box in which the particles reside. The simulation box has a hypervolume:

$$V = \prod_{\delta}^{d} L_\delta \tag{O.8}$$

where d is the dimension of the system and L_δ is the size of the box in each component. The origin is placed right in the middle of the box such that the boundaries of the box are present at every $\pm\frac{1}{2}L_\delta$. A two-dimensional box is depicted in fig. O.2. The boundaries can have boundary conditions per dimension. Canonical boundary conditions are either periodic, where the box is tiled and wrapped, or hard, where the particles perfectly bounce off the boundary. Additionally, we may change the shape of the simulation box itself by applying linear transformations, making it shrink, rotate, expand and shear. Should we, however, want hard boundaries of an arbitrary shape, there is yet another way of imposing them. Consider a particle confined inside the hatched blob in fig. O.2. We may approximately compose such a shape from straight lines and circles; circles and straight lines have very well-defined regular tangents. Once we know the lines and circles — with their tangents — we can apply a spring-like force to a particle in case it tries to transcend the boundary. Choosing a sufficiently stiff spring constant will mimic a perfect hard boundary quite well.

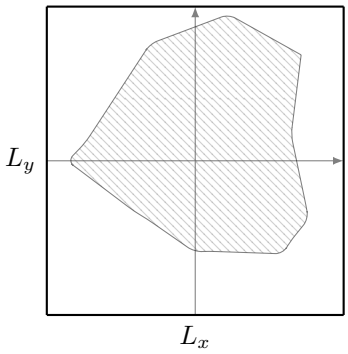

Figure O.2: A two-dimensional box with dimensions $L_x \times L_y$. The origin of the box is where the grey axes cross. The hatched blob represents an arbitrary boundary imposed on the particles within the box.

It is worth mentioning that if periodic boundary conditions are applied, distances ought to be measured through the boundary wrapping to the other side. This will ensure that particles will interact with other particles through the boundary if sufficiently close.

O.2 Neighbor search algorithms

In order to truly speed up the molecular dynamics algorithm we truncated our potential in eq. (O.6). For the implementation of the molecular dynamics algorithm this means we have to find all the neighbors of particle i within a distance r_c.

In order to do so, let us introduce $r_s > r_c$, which is called the skin radius. In order then to generate a list of all particles within a distance r_c of particle i, all we have to do is consider the list of particles within the skin list — particles within a distance r_s of particle i. We may wonder why this is useful at all; we simply translated the problem of generating the neighbor list to generating the skin list. Why is this an improvement? The answer lies in the dynamics of the problem. Generating the neighbor list from the skin list is of order $\mathcal{O}(NK)$, where $K \leq N$ is the average number of particles in a skin list. Naively, however, generating the skin list of order $\mathcal{O}\left(N^2\right)$ — we will improve on this shortly. Of course we have to generate the skin list, but only if a particle has traveled a distance larger than $r_s - r_c$, which is typically much less frequent than every time step, provided r_s is chosen sufficiently large.

The naive way of finding the nearest neighbors of particle i is to test whether all other particles are within a radius r_s. If positive the particles are added to the skin list. This method is sometimes called a brute force neighbor search.

A more sophisticated method is the *cell algorithm*. As the name suggests, the cell algorithm subdivides space into square, cubic or hypercubic cells — depending on the dimensionality — much like a d-dimensional histogram. Once done, the skin list of particle i is generated by considering particles in the same and nearest neighboring cells. In the case of periodic boundary conditions, the cells will wrap around.

Yet another method of generating the skin list is the *k-d tree*. A k-d tree is a binary tree that stores the particles' positions in a ordered fashion. The ordering is done in such a way that every node is a k-dimensional point where one of the dimensions of space is split in half, generating two subtrees. This way, particles close to each other remain in the same subtree whereas particles far away from each other are in the other subtree. On every level of the tree this process is repeated for a different dimension of the k-dimensional space. Due to the computational time complexity properties of a tree, locating, updating, adding and removing particles from the tree can be at most logarithmic order depending on implementation — for N particles time complexity is of order $\mathcal{O}(\log N)$. The initial build of the tree is of order $\mathcal{O}(N \log N)$.

Appendix P `libmd`

In order to perform molecular dynamics simulations optimized for soft matter physics, we have created a computer library called `libmd`. `libmd` is an in-house molecular dynamics computer library written in `C++11`. The library was initially written in 2013 by dr. J. Paulose, (now dr.) T. H. Beuman, MSc and B. C. van Zuiden, MSc (maintainer) under the supervision of (now prof.) dr. Vincenzo Vitelli.

`libmd` implements molecular dynamics as described in appendix O for arbitrary dimensions. It combines the computational techniques like automatic differentiation (appendix section G.3), state of the art neighbor search algorithms (appendix section O.2) and symplectic integration using the described integrators (appendices K and L and section 4.3) to solve for scenarios with and without curvature. The curvature part of the code assumes a Monge patch (appendix E), where the user only has to provide the Monge function. In addition the library can cope with arbitrary potentials or force terms. It is designed to be extremely flexible and hackable, it supports hooks and is fairly well documented.

Appendix Q Finite discrete lattices

In this appendix we describe some finite discrete two-dimensional lattices, for fun and for profit. These discrete lattices have been used in computational implementations of this thesis. Throughout this appendix we will use the modulus operator denoted $\%$ and defined as $a\%b \equiv a - b\left\lfloor \frac{a}{b} \right\rfloor$ where $\lfloor \ldots \rfloor$ is the floor function[♠] and $a, b \in \mathbb{Z}$. The modulus operator is assumed to have the same precedence as multiplication and devision.

Q.1 Square lattice

The square lattice is a somewhat trivial lattice. Its $N \in \mathbb{N}$ sites are composed of $q_x \in \mathbb{N}$ rows and $q_y \in \mathbb{N}$ columns such that $N = q_x q_y$. Given a lattice spacing a, site $i \in \{0, \ldots, N-1\}$ is positioned at:

$$x_i = a\left(i\%q_x\right), \tag{Q.1}$$

$$y_i = a\left\lfloor \frac{i}{q_x} \right\rfloor. \tag{Q.2}$$

[♠] One can define the floor function as $\lfloor x \rfloor = \max_{z \in \mathbb{Z}} (z \leq x)$.

The four neighbors j_n^i of site i are found using the following formulas:

$$j_0^i = (i+1)\,\%q_x + q_x\left\lfloor \tfrac{i}{q_x} \right\rfloor, \tag{Q.3}$$

$$j_1^i = \left(i + q_x\left(\left\lfloor \tfrac{i}{q_x} \right\rfloor + 1\right)\right)\%N, \tag{Q.4}$$

$$j_2^i = (i-1+q_x)\,\%q_x + q_x\left\lfloor \tfrac{i}{q_x} \right\rfloor, \tag{Q.5}$$

$$j_3^i = \left(i + q_x\left(\left\lfloor \tfrac{i}{q_x} \right\rfloor - 1\right) + N\right)\%N. \tag{Q.6}$$

Q.2 Triangular lattice

A somewhat more complicated lattice is the triangular lattice. Its $N \in \mathbb{N}$ sites are composed of $q_x \in \mathbb{N}$ rows and $q_y \in 2\mathbb{N}$ columns such that $N = q_x q_y$. Given a lattice spacing a, site $i \in \{0, \ldots, N-1\}$ is positioned at:

$$x_i = a\,(i\%q_x) + \tfrac{a}{2}\left(\left\lfloor \tfrac{i}{q_x} \right\rfloor \%2\right), \tag{Q.7}$$

$$y_i = \tfrac{\sqrt{3}a}{2}\left\lfloor \tfrac{i}{q_x} \right\rfloor. \tag{Q.8}$$

The six neighbors j_n^i of site i are found using the following formulas:

$$j_0^i = (i+1)\,\%q_x + q_x\left\lfloor \tfrac{i}{q_x} \right\rfloor, \tag{Q.9}$$

$$j_1^i = \left(\left(i + \left\lfloor \tfrac{i}{q_x} \right\rfloor \%2\right)\%q_x + q_x\left(\left\lfloor \frac{i}{q_x} \right\rfloor + 1\right)\right)\%N, \tag{Q.10}$$

$$j_2^i = \left(\left(i - 1 + q_x + \left\lfloor \tfrac{i}{q_x} \right\rfloor \%2\right)\%q_x + q_x\left(\left\lfloor \frac{i}{q_x} \right\rfloor + 1\right)\right)\%N, \tag{Q.11}$$

$$j_3^i = (i-1+q_x)\,\%q_x + q_x\left\lfloor \tfrac{i}{q_x} \right\rfloor, \tag{Q.12}$$

$$j_4^i = \left(\left(i - 1 + q_x + \left\lfloor \tfrac{i}{q_x} \right\rfloor \%2\right)\%q_x + q_x\left(\left\lfloor \tfrac{i}{q_x} \right\rfloor - 1\right) + N\right)\%N, \tag{Q.13}$$

$$j_5^i = \left(\left(i + \left\lfloor \tfrac{i}{q_x} \right\rfloor \%2\right)\%q_x + q_x\left(\left\lfloor \tfrac{i}{q_x} \right\rfloor - 1\right) + N\right)\%N. \tag{Q.14}$$

Q.3 Honeycomb lattice

An even more complicated lattice is the honeycomb lattice. Its $N \in \mathbb{N}$ sites are composed of $q_x \in 2\mathbb{N}$ rows and $q_y \in 2\mathbb{N}+1$ columns such that $N = q_x q_y$. Given a lattice spacing a, site $i \in \{0, \dots, N-1\}$ is positioned at:

$$x_i = \tfrac{\sqrt{3}a}{2} i \% q_x, \tag{Q.15}$$

$$y_i = \tfrac{3a}{2} \left\lfloor \tfrac{i}{q_x} \right\rfloor \% q_y + \tfrac{a}{2} \left| (i \% q_x) \,\%2 - \left(\left\lfloor \tfrac{i}{q_x} \right\rfloor \% q_y \right) \%2 \right| \tag{Q.16}$$

The three neighbors j_n^i of site i are found using the following formulas:

$$j_0^i = (i \% q_x + 1) + q_x \left\lfloor \tfrac{i}{q_x} \right\rfloor, \tag{Q.17}$$

$$j_1^i = (i \% q_x - 1 + qx) + q_x \left\lfloor \tfrac{i}{q_x} \right\rfloor, \tag{Q.18}$$

$$j_2^i = \left(i + q_x \left(2 \left| (i \% q_x) \,\%2 - \left(\left\lfloor \tfrac{i}{q_x} \right\rfloor \% q_y \right) \%2 \right| - 1 \right) + N \right) \% N. \tag{Q.19}$$

Q.4 Lieb lattice

A possibly more complicated lattice is the Lieb lattice. Its $N \in \mathbb{N}$ sites are composed of $q_x \in \mathbb{N}$ rows and $q_y \in 2\mathbb{N}$ columns such that $N = q_x q_y$. Given a lattice spacing a, site $i \in \{0, \dots, N-1\}$ let us first define:

$$n = \tfrac{3q_x}{2} + q_x \%2, \tag{Q.20}$$

$$f_i = i \% n, \tag{Q.21}$$

site i is then positioned at:

$$x_i = \begin{cases} a f_i & f_i < q_x \\ 2a \left(f_i - q_x \right) & f_i \geq q_x \end{cases}, \tag{Q.22}$$

$$y_i = \begin{cases} 2a \left(\left\lfloor \tfrac{i}{n} \right\rfloor \% q_x \right) & f < q_x \\ 2a \left(\left\lfloor \tfrac{i}{n} \right\rfloor \% q_x + \tfrac{1}{2} \right) & f \geq q_x \end{cases}. \tag{Q.23}$$

Sites with three types of sites. First there is $f_i < q_x$ and $f_i \% 2 = 0$ with four neighbors:

$$j_0^i = \left(f_i + 1 \right) \% q_x + n \left\lfloor \tfrac{i}{n} \right\rfloor, \tag{Q.24}$$

$$j_1^i = \left(\left\lfloor \tfrac{f_i}{2} \right\rfloor + q_x + n \left\lfloor \tfrac{i}{n} \right\rfloor \right) \% N, \tag{Q.25}$$

$$j_2^i = \left(f_i - 1 + q_x \right) \% q_x + n \left\lfloor \tfrac{i}{n} \right\rfloor, \tag{Q.26}$$

$$j_3^i = \left(\left\lfloor \tfrac{f_i}{2} \right\rfloor + q_x + n \left(\left\lfloor \tfrac{i}{n} \right\rfloor + 1 \right) + N \right) \% N. \tag{Q.27}$$

Second, there is $f_i < q_x$ and $f_i \% 2 = 1$ with two neighbors:

$$j_0^i = (f_i + 1) \,\%q_x + n\left\lfloor \tfrac{i}{n} \right\rfloor, \tag{Q.28}$$

$$j_1^i = (f_i - 1 + q_x) \,\%q_x + n\left\lfloor \tfrac{i}{n} \right\rfloor. \tag{Q.29}$$

Finally, there is $f_i \geq q_x$ also with two neighbors:

$$j_0^i = \left(2\,(f - q_x) + n\left(\left\lfloor \tfrac{i}{n} \right\rfloor + 1\right)\right) \%N, \tag{Q.30}$$

$$j_1^i = \left(2\,(f - q_x) + n\left(\left\lfloor \tfrac{i}{n} \right\rfloor\right)\right) \%N. \tag{Q.31}$$

References

[1] J. L. Aragones, J. P. Steimel, and A. Alexander-Katz. *Elasticity-induced force reversal between active spinning particles in dense passive media.* **Nat. Comm.** *7.* ***2016*** (*54*).

[2] J. Denk, L. Huber, E. Reithmann, and E. Frey. *Active Curved Polymers Form Vortex Patterns on Membranes.* **Phys. Rev. L** *116* 17. ***2016*** (*53*).

[3] S. J. Ebbens. *Active colloids: Progress and challenges towards realising autonomous applications.* **Curr. Opin. Coll. I. Sci** *21.* ***2016*** (*75*).

[4] C. He, X. Ni, H. Ge, Y.-B. Sun X.-C. Chen, M.-H. Lu, X.-P. Liu, and Y.-F. Chen. *Acoustic topological insulator and robust one-way sound transport.* **Nat. Phys.** ***2016*** (*34*).

[5] V. Koning and V. Vitelli. *Crystals and liquid crystals confined to curved geometries.* Fluids, Colloids, and Soft Materials: An Introduction to Soft Matter Physics *7.* ***2016*** (*5, 77*).

[6] A. Snezhko. *Complex collective dynamics of active torque-driven colloids at interfaces.* **Curr. Opin. Coll. I. Sci** *21.* ***2016*** (*53*).

[7] H. Wioland, F. G. Woodhouse, J. Dunkel, and R. E. Goldstein. *Ferromagnetic and antiferromagnetic order in bacterial vortex lattices.* **Nat. Phys.** ***2016*** (*34, 35*).

[8] J. Yan, M. Han, J. Zhang, C. Xu, E. Luijten, and S. Granick. *Reconfiguring active particles by electrostatic imbalance.* **Nat. Mater.** ***2016*** (*34*).

[9] A. Bricard, J.-B. Caussin, D. Das, C. Savoie, V. Chikkadi, K. Shitara, O. Chepizhko, F. Peruani, D. Saintillan, and D. Bartolo. *Emergent vortices in populations of colloidal rollers.* **Nat. Comm.** *6.* ***2015*** (*34*).

[10] D. R. Brumley, M. Polin, T. J. Pedley, and R. E. Goldstein. *Metachronal waves in the flagellar beating of Volvox and their hydrodynamic origin.* **Jour. of The Roy. Soc. I** *12* 108. ***2015*** (*54*).

[11] F. Ginot, I. Theurkauff, D. Levis, C. Ybert, L. Bocquet, L. Berthier, and C. Cottin-Bizonne. *Nonequilibrium Equation of State in Suspensions of Active Colloids.* **Phys. Rev. X** *5.* ***2015*** (*33, 53*).

[12] Y. Goto and H. Tanaka. *Purely hydrodynamic ordering of rotating disks at a finite Reynolds number.* **Nat. Comm.** *6.* ***2015*** (*54*).

[13] T. Kariyado and Y. Hatsugai. *Manipulation of Dirac Cones in Mechanical Graphene.* **Sci. Rep.** *5.* ***2015*** (*34*).

[14] A. B. Khanikaev, R. Fleury, S. H. Mousavi, and A. Alu. *Topologically robust sound propagation in an angular-momentum-biased graphene-like resonator lattice.* **Nat. Comm.** *6.* ***2015*** (*34, 36, 40*).

[15] C. Maggi, F. Saglimbeni, M. Dipalo, F. De Angelis, and R. D. Leonardo. *Micromotors With Asymmetric Shape That Efficiently Convert Light Into Work By Thermocapillary Effect.* **Nat. Comm.** *6.* ***2015*** (*75*).

[16] L. M. Nash, D. Kleckner, A. Read, V. Vitelli, A. M. Turner, and W. T. M. Irvine. *Topological mechanics of gyroscopic metamaterials.* **Proc. Nat. Acad. Sci.** *112* 47. ***2015*** (*5, 34*).

[17] P. Oswald and G. Poy. *Lehmann rotation of cholesteric droplets: Role of the sample thickness and of the concentration of chiral molecules.* **Phys. Rev. E** *91* 3. ***2015*** (*53*).

[18] J. Paulose, B. G. Chen, and V. Vitelli. *Topological modes bound to dislocations in mechanical metamaterials.* **Nat. Phys.** ***2015*** (*1, 34*).

[19] J. Paulose, A. S. Meeussen, and V. Vitelli. *Selective buckling via states of self-stress in topological metamaterials.* **Proc. Nat. Acad. Sci.** *112* 25. ***2015*** (*1, 34*).

[20] D. J. G. Pearce and M. S. Turner. *Emergent behavioural phenotypes of swarming models revealed by mimicking a frustrated anti-ferromagnet.* **Roy. Soc. I** *12* 111. ***2015*** (*35*).

[21] A. P. Petroff, X.-L. Wu, and A. Libchaber. *Fast-Moving Bacteria Self-Organize into Active Two-Dimensional Crystals of Rotating Cells.* **Phys. Rev. L** *114.* ***2015*** (*53*).

[22] S. Sabrina, M. Spellings, S. C. Glotzer, and K. J. M. Bishop. *Coarsening dynamics of binary liquids with active rotation.* ***2015*** (*54*).

[23] F. Saglimbeni, M. Dipalo, F. De Angelis, R. Di Leonardo, and C. Maggi. *Micromotors with asymmetric shape that efficiently convert light into work by thermocapillary effects.* **Nat. Comm.** *6.* ***2015*** (*53*).

[24] M. Spellings, M. Engel, D. Klotsa, S. Sabrina, A. M. Drews, N. H. P. Nguyen, K. J. M. Bishop, and S. C. Glotzer. *Shape control and compartmentalization in active colloidal cells.* **Proc. Natl. Acad. Sci. USA** *112* 34. ***2015*** (*54*).

[25] R. Susstrunk and S. D. Huber. *Observation of phononic helical edge states in a mechanical topological insulator.* **Sci.** *349* 6243. ***2015*** (*34*).

[26] R. Suzuki, C. A. Weber, E. Frey, and A. R. Bausch. *Polar pattern formation in driven filament systems requires non-binary particle collisions.* **Nat. Phys.** ***2015*** (*35*).

[27] J. Tailleur and M. E. Cates. *Motility-Induced Phase Separation.* **Ann. Rev. Cond. Matt. Phys.** *6.* ***2015*** (*33, 53*).

[28] P. Wang, L. Lu, and K. Bertoldi. *Topological Phononic Crystals with One-Way Elastic Edge Waves.* **Phys. Rev. L** *115* 10. ***2015*** (*34*).

[29] I. Williams, E. C. Oğuz, T. Speck, P. Bartlett, H. Löwen, and C. P. Royall. *Transmission of torque at the nanoscale.* **Nat. Phys.** *12* 1. ***2015*** (*75*).

[30] J. Yan, S. C. Bae, and S. Granick. *Rotating crystals of magnetic Janus colloids.* **Soft Matt.** *11* 1. ***2015*** (*55*).

[31] Z. Yang, F. Gao, X. Shi, X. Lin, Z. Gao, Y. Chong, and B. Zhang. *Topological Acoustics.* **Phys. Rev. L** *114* 11. ***2015*** (*5, 34, 36, 39, 40, 51*).

[32] K. Yeo, E. Lushi, and P. M. Vlahovska. *Collective Dynamics in a Binary Mixture of Hydrodynamically Coupled Microrotors.* **Phys. Rev. L** *114*. ***2015*** (*54*).

[33] A. Cavagna and I. Giardina. *Bird Flocks as Condensed Matter.* **Ann. Rev. Cond. Matt. Phys.** *5* 1. ***2014*** (*33, 53*).

[34] B. G. Chen, N. Upadhyaya, and V. Vitelli. *Nonlinear conduction via solitons in a topological mechanical insulator.* **Proc. Nat. Acad. Sci.** *111* 36. ***2014*** (*1, 34*).

[35] E. Efrati and W. T. M. Irvine. *Orientation-dependent handedness and chiral design.* **Phys. Rev. X** *4* 1. ***2014*** (*75*).

[36] Y. Fily, S. Henkes, and M. C. Marchetti. *Freezing and phase separation of self-propelled disks.* **Soft Matt.** *10* 13. ***2014*** (*66*).

[37] B. Florijn, C. Coulais, and M. van Hecke. *Programmable mechanical metamaterials.* **Phys. Rev. L** *113* 17. ***2014*** (*1*).

[38] V. Koning. *On the geometry of fracture and frustration.* Casimir PhD Series, Delft-Leiden; 2014-25. ***2014*** (*5*).

[39] N. Kumar, H. Soni, S. Ramaswamy, and A. K. Sood. *Flocking at a distance in active granular matter. 5 SP -.* ***2014*** (*34, 48*).

[40] E. Lushi, H. Wioland, and R. E. Goldstein. *Fluid flows created by swimming bacteria drive self-organization in confined suspensions.* **Proc. Nat. Acad. Sci.** *111* 27. ***2014*** (*67*).

[41] N. H. P. Nguyen, D. Klotsa, M. Engel, and S. C. Glotzer. *Emergent Collective Phenomena in a Mixture of Hard Shapes through Active Rotation.* **Phys. Rev. L** *112* 7. ***2014*** (*54*).

[42] Ariel Amir, Jayson Paulose, and David R. Nelson. *Theory of interacting dislocations on cylinders.* **Phys. Rev. E** *87.* ***2013*** (*87*).

[43] A. Bricard, J.-B. Caussin, N. Desreumaux, O. Dauchot, and D. Bartolo. *Emergence of macroscopic directed motion in populations of motile colloids.* **Nat.** *503* 7474. ***2013*** (*33–35, 44, 48, 53, 75*).

[44] T. Brotto, J.-B. Caussin, E. Lauga, and D. Bartolo. *Hydrodynamics of confined active fluids.* **Phys. Rev. L** *110* 3. ***2013*** (*48*).

[45] M. Cavallaro Jr, M. A. Gharbi, D. A. Beller, S. Copar, Z. Shi, R. D. Kamien, S. Yang, T. Baumgart, and K. J. Stebe. *Ring around the colloid.* **Soft Matter** *9*. ***2013*** (*13*).

[46] T. Fukui, Y. Hatsugai, and H. Suzuki. *Chern Numbers in Discretized Brillouin Zone: Efficient Method of Computing (Spin) Hall Conductances.* **Phys. Soc. of Japan**. ***2013*** (*41*).

[47] S. Fürthauer, M. Strempel, S. W. Grill, and F. Jülicher. *Active Chiral Processes in Thin Films.* **Phys. Rev. L** *110* 4. ***2013*** (*53*).

[48] A. Kaiser and Löwen, H. *Vortex arrays as emergent collective phenomena for circle swimmers.* **Phys. Rev. E** *87* 3. ***2013*** (*54*).

[49] C. L. Kane and T. C. Lubensky. *Topological boundary modes in isostatic lattices.* **Nat. Phys.** *10* 1. ***2013*** (*34*).

[50] V. Koning, T. Lopez-Leon, A. Fernandez-Nieves, and V. Vitelli. *Bivalent defect configurations in inhomogeneous nematic shells.* **Soft Matter** *9*. ***2013*** (*13*).

[51] Q. Liu, B. Senyuk, M. Tasinkevych, and I. I. Smalyukh. *Nematic liquid crystal boojums with handles on colloidal handlebodies.* **Proc. Nat. Acad. Sci.** *110* 23. ***2013*** (*13*).

[52] M. C. Marchetti, J. F. Joanny, S. Ramaswamy, T. B. Liverpool, J. Prost, M. Rao, and R. A. Simha. *Hydrodynamics of soft active matter.* **Rev. Mod. Phys.** *85* 3. ***2013*** (*4, 33–36, 48, 53*).

[53] D. Marenduzzo, C. Micheletti, E. Orlandini, and D. Sumners. *Topological friction strongly affects viral DNA ejection.* **Proc. Nat. Acad. Sci.** *110* 50. ***2013*** (*14*).

[54] P. Nitoń, A. Żywociński, M. Fiałkowski, and R. Hołyst. *A "nano-windmill" driven by a flux of water vapour: a comparison to the rotating ATPase.* **Nanoscale** *5* 20. ***2013*** (*75*).

[55] E. Pairam, J. Vallamkondu, V. Koning, B. C. van Zuiden, P. W. Ellis, M. A. Bates, V. Vitelli, and A. Fernandez-Nieves. *Stable nematic droplets with handles.* **Proc. Nat. Acad. Sci.** *110* 23. ***2013*** (*13, 24, 25, 27, 28, 31*).

[56] J. Palacci, S. Sacanna, A. P. Steinberg, D. J. Pine, and P. M. Chaikin. *Living crystals of light-activated colloidal surfers.* **Sci.** *339* 6122. ***2013*** (*33, 53*).

[57] M. C. Rechtsman, J. M. Zeuner, Y. Plotnik, Y. Lumer, D. Podolsky, F. Dreisow, S. Nolte, M. Segev, and A. Szameit. *Photonic Floquet topological insulators.* **Nat.** *496* 7444. ***2013*** (*34*).

[58] S. Sacanna, D. J. Pine, and G.-R. Yi. *Engineering shape: the novel geometries of colloidal self-assembly.* **Soft Matt.** *9.* ***2013*** (*75*).

[59] T. Sanchez, D. T. N. Chen, S. J. DeCamp, M. Heymann, and Z. Dogic. *Spontaneous motion in hierarchically assembled active matter.* **Nat.** *491* 7424. ***2013*** (*33, 53*).

[60] B. Senyuk, Q. Liu, S. He, R. D. Kamien, R. B. Kusner, T. C. Lubensky, and I. I. Smalyukh. *Topological colloids.* **Nat.** *493* 7431. ***2013*** (*13*).

[61] A. P. Solon and J. Tailleur. *Revisiting the Flocking Transition Using Active Spins.* **Phys. Rev. L** *111* 7. ***2013*** (*35*).

[62] H. Wioland, F. G. Woodhouse, J. Dunkel, J. O. Kessler, and R. E. Goldstein. *Confinement Stabilizes a Bacterial Suspension into a Spiral Vortex.* **Phys. Rev. L** *110* 26. ***2013*** (*67*).

[63] Z. Yao and M. O. de la Cruz. *Packing of charged chains on toroidal geometries.* **Phys. Rev. E** *87.* ***2013*** (*13*).

[64] B. Button, L.-H. Cai, C. Ehre, M. Kesimer, D. B. Hill, J. K. Sheehan, R. C. Boucher, and M. Rubinstein. *A periciliary brush promotes the lung health by separating the mucus layer from airway epithelia.* **Sci.** *337* 6097. ***2012*** (*54*).

[65] F. Farrell, M. Marchetti, D. Marenduzzo, and J. Tailleur. *Pattern Formation in Self-Propelled Particles with Density-Dependent Motility.* **Phys. Rev. L** *108* 24. ***2012*** (*35, 48*).

[66] S. Fürthauer, M. Strempel, S. W. Grill, and F. Jülicher. *Active chiral fluids.* **Eur. Phys. JE** *35* 9. ***2012*** (*68*).

[67] Y. Sumino, K. H. Nagai, Y. Shitaka, D. Tanaka, K. Yoshikawa, H. Chaté, and K. Oiwa. *Large-scale vortex lattice emerging from collectively moving microtubules.* **Nat.** *483* 7390. ***2012*** (*53*).

[68] D. Svenšek and R. Podgornik. *Confined chiral polymer nematics: Ordering and spontaneous condensation.* **Europhys. Lett.** *100.* ***2012*** (*14*).

[69] T. Vicsek and A. Zafeiris. *Collective motion.* **Phys. Rep.** *517* 3-4. ***2012*** (*33, 34, 53*).

[70] N. Coq, A. Bricard, F.-D. Delapierre, L. Malaquin, O. du Roure, M. Fermigier, and D. Bartolo. *Collective Beating of Artificial Microcilia.* **Phys. Rev. L** *107.* ***2011*** (*55*).

[71] V. S. R. Jampani, M. Škarabot, M. Ravnik, S. Čopar, S. Žumer, and I. Mu ševi č. *Colloidal entanglement in highly twisted chiral nematic colloids: Twisted loops, Hopf links, and trefoil knots.* **Phys. Rev. E** *84.* ***2011*** (*13*).

[72] T. Lopez-Leon and A. Fernandez-Nieves. *Drops and shells of liquid crystal.* **Coll. and Poly. Sci.** *289.* ***2011*** (*13*).

[73] T. Lopez-Leon, V. Koning, K. B. S. Devaiah, V. Vitelli, and A. Fernandez-Nieves. *Frustrated nematic order in spherical geometries.* **Nat. Phys.** *7.* ***2011*** (*13*).

[74] R. L. Blumberg Selinger, A. Konya, A. Travesset, and J. V. Selinger. *Monte Carlo Studies of the XY Model on Two-Dimensional Curved Surfaces.* **Phys. Chem. B** *115* 48. ***2011*** (*13*).

[75] H. Shin and G. M. Grason. *Filling the void in confined polymer nematics: Phase transitions in a minimal model of dsDNA packing.* **Europhy. Lett.** *96* 3. ***2011*** (*14*).

[76] S. Thutupalli, R. Seemann, and S. Herminghaus. *Swarming behavior of simple model squirmers.* **New J. Phys.** *13.* ***2011*** (*33, 53*).

[77] U. Tkalec, M. Ravnik, S. Čopar, S. Žumer, and I. Muševič. *Reconfigurable Knots and Links in Chiral Nematic Colloids.* **Sci.** *333* 6038. ***2011*** (*13*).

[78] L. Tortora and O. D. Lavrentovich. *Chiral symmetry breaking by spatial confinement in tactoidal droplets of lyotropic chromonic liquid crystals.* **Proc. Nat. Acad. Sci.** *108* 13. ***2011*** (*30*).

[79] J. Deseigne, O. Dauchot, and H. Chaté. *Collective Motion of Vibrated Polar Disks.* **Phys. Rev. L** *105.* ***2010*** (*33, 34, 53*).

[80] U. Gasser, C. Eisenmann, G. Maret, and P. Keim. *Melting of crystals in two dimensions.* **Chem. Phys. Chem.** *11* 5. ***2010*** (*83*).

[81] B. Guirao, A. Meunier, S. Mortaud, A. Aguilar, J.-M. Corsi, L. Strehl, Y. Hirota, A. Desoeuvre, C. Boutin, Y.-G. Han, Z. Mirzadeh, H. Cremer, M. Montcouquiol, K. Sawamoto, and N. Spassky. *Coupling between hydrodynamic forces and planar cell polarity orients mammalian motile cilia.* **Nat. Cell Bio.** *12* 4. ***2010*** (*54*).

[82] M. Z. Hasan and C. L. Kane. *Colloquium: Topological insulators.* **Rev. Mod. Phys.** *82* 4. ***2010*** (*5, 33, 34, 39, 41*).

[83] Y. Hatwalne and M. Muthukumar. *Chiral Symmetry Breaking in Crystals of Achiral Polymers.* **Phys. Rev. L** *105.* ***2010*** (*14*).

[84] A. Kudrolli. *Concentration Dependent Diffusion of Self-Propelled Rods.* **Phys. Rev. L** *104.* ***2010*** (*33, 53*).

[85] V. Schaller, C. Weber, C. Semmrich, E. Frey, and A. R. Bausch. *Polar patterns of driven filaments.* **Nat.** *467* 7311. ***2010*** (*33, 34, 44, 53*).

[86] I. I. Smalyukh, Y. Lansac, N. A. Clark, and R. P. Trivedi. *Three-dimensional structure and multistable optical switching of triple-twisted particle-like excitations in anisotropic fluids.* **Nat. Mater.** *9* 2. ***2010*** (*13*).

[87] N. Uchida and R. Golestanian. *Synchronization and Collective Dynamics in a Carpet of Microfluidic Rotors.* **Phys. Rev. L** *104.* ***2010*** (*53*).

[88] N. Uchida and R. Golestanian. *Synchronization in a carpet of hydrodynamically coupled rotors with random intrinsic frequency.* **Europhys. Lett.** *89* 5. ***2010*** (*53*).

[89] P. Yeh and C. Gu. *Optics of Liquid Crystal Displays.* Wiley Series in Pure and Applied Optics. ***2010*** (*29, 31*).

[90] M. Bowick and L. Giomi. *Two-dimensional matter: order, curvature and defects.* **Adv. in Phys.** *58.* ***2009*** (*13, 77*).

[91] K. Drescher, K. C. Leptos, I. Tuval, T. Ishikawa, T. J. Pedley, and R. E. Goldstein. *Dancing Volvox : Hydrodynamic Bound States of Swimming Algae.* **Phys. Rev. L** *102* 16. ***2009*** (*53*).

[92] A. Leforestier and F. Livolant. *Structure of toroidal DNA collapsed inside the phage capsid.* **Proc. Nat. Acad. Sci.** *106* 23. ***2009*** (*14*).

[93] D. Marenduzzo, E. Orlandini, A. Stasiak, D. W. Sumners, L. Tubiana, and C. Micheletti. *DNA–DNA interactions in bacteriophage capsids are responsible for the observed DNA knotting.* **Proc. Nat. Acad. Sci.** *106* 52. ***2009*** (*14*).

[94] E. Pairam and A. Fernández-Nieves. *Generation and stability of toroidal droplets in a viscous liquid.* **Phys. Rev. L** *102* 23. ***2009*** (*27*).

[95] E. Prodan and C. Prodan. *Topological Phonon Modes and Their Role in Dynamic Instability of Microtubules.* **Phys. Rev. L** *103.* ***2009*** (*34*).

[96] L. Giomi and M. J. Bowick. *Defective ground states of toroidal crystals.* **Phys. Rev. E** *78.* ***2008*** (*13*).

[97] L. Giomi and M. J. Bowick. *Elastic Theory of Defects in Toroidal Crystals.* **Europhys. J. E** *27.* ***2008*** (*13*).

[98] E. Lemaire, L. Lobry, N. Pannacci, and F. Peters. *Viscosity of an electro-rheological suspension with internal rotations.* **Jour. of Rheo.** ***2008*** (*53*).

[99] A. Hexemer, V. Vitelli, E. J. Kramer, and G. H. Fredrickson. *Monte Carlo study of crystalline order and defects on weakly curved surfaces.* **Phys. Rev. E** *76* 5. ***2007*** (*88*).

[100] M. Ravnik, M. Škarabot, S. Žumer, U. Tkalec, I. Poberaj, D. Babi č, N. Osterman, and I. Mu ševi č. *Entangled Nematic Colloidal Dimers and Wires.* **Phys. Rev. L** *99.* ***2007*** (*13*).

[101] T. Araki and H. Tanaka. *Colloidal Aggregation in a Nematic Liquid Crystal: Topological Arrest of Particles by a Single-Stroke Disclination Line.* **Phys. Rev. L** *97.* ***2006*** (*13*).

[102] E. Bertin, M. Droz, and G. Gregoire. *Boltzmann and hydrodynamic description for self-propelled particles.* **Phys. Rev. E** *74* 2. ***2006*** (*35, 48*).

[103] A. Stern and M. Desbrun. *Discrete geometric mechanics for variational time integrators.* ***2006*** (*77, 79, 125*).

[104] V. Vitelli, J. B. Lucks, and D. R. Nelson. *Crystallography on curved surfaces.* **Proc. Nat. Acad. Sci.** *103* 33. ***2006*** (*77, 85, 86*).

[105] J. A. Acebrón, L. L. Bonilla, C. J. Pérez Vicente, F. Ritort, and R. Spigler. *The Kuramoto model: A simple paradigm for synchronization phenomena.* **Rev. Mod. Phys.** *77*. ***2005*** (*53*).

[106] G. R. Fowles and G. L. Cassiday. *Analytical Mechanics.* International student edition. ***2005*** (*105*).

[107] R. Kirchhoff and Löwen, H. *T-structured fluid and jamming in driven Brownian rotators.* **Europhys. Lett.** *69* 2. ***2005*** (*54*).

[108] I. H. Riedel, K. Kruse, and J. Howard. *A self-organized vortex array of hydrodynamically entrained sperm cells.* **Sci.** *309* 5732. ***2005*** (*53*).

[109] H. Stark and T. C. Lubensky. *Poisson bracket approach to the dynamics of nematic liquid crystals: The role of spin angular momentum.* **Phys. Rev. E** *72* 5. ***2005*** (*68*).

[110] J. Toner, Y. Tu, and S. Ramaswamy. *Hydrodynamics and phases of flocks.* **Ann. of Phys.** *318* 1. ***2005*** (*4, 33–36, 48, 53*).

[111] J.-C. Tsai, F. Ye, J. Rodriguez, J. P. Gollub, and T. C. Lubensky. *A Chiral Granular Gas.* **Phys. Rev. L** *94* 21. ***2005*** (*53, 68, 69, 75*).

[112] M. Bowick, D. R. Nelson, and A. Travesset. *Curvature-induced defect unbinding in toroidal geometries.* **Phys. Rev. E** *69*. ***2004*** (*13*).

[113] S. M. Carroll. *Spacetime and Geometry: An Introduction to General Relativity.* ***2004*** (*77*).

[114] I. M. Kulić, D. Andrienko, and M. Deserno. *Twist-bend instability for toroidal DNA condensates.* **Europhys. Lett.** *67* 3. ***2004*** (*13, 14, 17*).

[115] D. R. Nelson, T. Piran, and S. Weinberg. *Statistical Mechanics of Membranes and Surfaces.* ***2004*** (*5, 77*).

[116] P. Prinsen and P. van der Schoot. *Parity breaking in nematic tactoids.* **Jour. of Phys. Cond. Matt.** *16* 49. ***2004*** (*30*).

[117] D. C. Rapaport. *The Art of Molecular Dynamics Simulation.* ***2004*** (*127*).

[118] C. C. Conwell, I. D. Vilfan, and N. V. Hud. *Controlling the size of nanoscale toroidal DNA condensates with static curvature and ionic strength.* **Proc. Nat. Acad. Sci.** *100*. ***2003*** (*14*).

[119] M. Kleman and O. D. Lavrentovich. *Soft Matter Physics: An Introduction.* ***2003*** (*11, 12, 14, 18, 25, 31*).

[120] P. Lenz, J.-F. Joanny, F. Jülicher, and J. Prost. *Membranes with Rotating Motors.* **Phys. Rev. L** *91.* ***2003*** (*54*).

[121] V. N. Manoharan, M. T. Elsesser, and D. J. Pine. *Dense packing and symmetry in small clusters of microspheres.* **Sci.** *301* 5632. ***2003*** (*75*).

[122] M. R. Stukan, V. A. Ivanov, A. Y. Grosberg, W. Paul, and K. Binder. *Chain length dependence of the state diagram of a single stiff-chain macromolecule: Theory and Monte Carlo simulation.* **Journ. Chem. Phys.** *118.* ***2003*** (*14*).

[123] Y. Tabe and H. Yokoyama. *Coherent collective precession of molecular rotors with chiral propellers.* **Nat. Mater.** *2* 12. ***2003*** (*53*).

[124] A. Travesset. *Universality in the screening cloud of dislocations surrounding a disclination.* **Phys. Rev. B** *68* 11. ***2003*** (*77, 91*).

[125] M. D. Contreras and R. Sanchez. *Application of a factorial design to the study of the flow behavior, spreadability and transparency of a Carbopol ETD 2020 gel. Part II.* **Int. Jour. Pharma.** *234* 1. ***2002*** (*27*).

[126] R. D. Kamien. *The geometry of soft materials: a primer.* **Rev. Mod. Phys.** *74.* ***2002*** (*15*).

[127] S. Nonaka, H. Shiratori, Y. Saijoh, and H. Hamada. *Determination of left-right patterning of the mouse embryo by artificial nodal flow.* **Nat.** *418* 6893. ***2002*** (*54*).

[128] H. Levine, W. J. Rappel, and I. Cohen. *Self-organization in systems of self-propelled particles.* **Phys. Rev. E** *63* 1 Pt 2. ***2001*** (*42*).

[129] J. E. Marsden and M. West. *Discrete mechanics and variational integrators.* **Acta Num. 2001** *10.* ***2001*** (*77, 79, 125*).

[130] H. Stark. *Physics of colloidal dispersions in nematic liquid crystals.* **Phys. Rep.** *351* 6. ***2001*** (*13*).

[131] M. Stevens. *Simple Simulations of DNA Condensation.* **Biophys. Journ.** *80.* ***2001*** (*14*).

[132] P. M. Chaikin and T. C. Lubensky. *Principles of Condensed Matter Physics.* ***2000*** (*5, 11*).

[133] H. Levine, W.-J. Rappel, and I. Cohen. *Self-organization in systems of self-propelled particles.* **Phys. Rev. E** *63* 1. ***2000*** (*4*).

[134] R. K. Soong, G. D. Bachand, H. P. Neves, A. G. Olkhovets, H. G. Craighead, and C. D. Montemagno. *Powering an inorganic nanodevice with a biomolecular motor.* **Sci.** *290* 5496. ***2000*** (*75*).

[135] J. Stelzer and R. Bernhard. *Director configurations of a nematic liquid crystal confined in a toroidal geometry. A finite element study.* **arXiv**. ***2000*** (*13*).

[136] C. Kane, J. E. Marsden, and M. Ortiz. *Symplectic-energy-momentum preserving variational integrators.* **Jour. Math. Phys.** *40* 7. ***1999*** (*77, 79, 125*).

[137] Philippe P. *Novel phases and colloidal assemblies in liquid crystals.* **Cur. Opin. Coll. IS** *4* 1. ***1999*** (*13*).

[138] O. D. Lavrentovich. *Topological defects in dispersed words and worlds around liquid crystals, or liquid crystal drops.* **Liquid Crystals** *24* 1. ***1998*** (*13*).

[139] T. C. Lubensky, D. Pettey, N. Currier, and H. Stark. *Topological defects and interactions in nematic emulsions.* **Phys. Rev. E** *57*. ***1998*** (*13*).

[140] S. Pafka, J. Cserti, and G. Tichy. *Monte Carlo studies of two-dimensional dislocation systems in applied external stress.* **Mod. and Sim. in Mat. Sci. and Eng.** *6* 5. ***1998*** (*101*).

[141] P. Poulin and D. A. Weitz. *Inverted and multiple nematic emulsions.* **Phys. Rev. E** *57*. ***1998*** (*13*).

[142] J. Toner and Y. Tu. *Flocks, herds, and schools: A quantitative theory of flocking.* **Phys. Rev. E** *58* 4. ***1998*** (*4*).

[143] V. A. Bloomfield. *DNA condensation by multivalent cations.* **Biopol.** *44* 3. ***1997*** (*14*).

[144] F. Ercolessi. *A molecular dynamics primer.* **Spring college in computational physics, ICTP, Trieste** *19*. ***1997*** (*127*).

[145] P. Poulin, H. Stark, T. C. Lubensky, and D. A. Weitz. *Novel Colloidal Interactions in Anisotropic Fluids.* **Sci.** *275* 5307. ***1997*** (*13*).

[146] P. S. Drzaic. *Liquid Crystal Dispersions.* Liquid Crystals Series,VoL 1. ***1995*** (*13, 27*).

[147] O. D. Lavrentovich and V. M. Pergamenshchik. *Patterns in thin liquid crystal films and the divergence ("surface like") elasticity.* **Jour. Mod. Phys. B** *09* 18n19. ***1995*** (*25*).

[148] J. Toner and Y. Tu. *Long-Range Order in a Two-Dimensional Dynamical XY-Model: How Birds Fly Together.* **Phys. Rev. L** *75* 23. ***1995*** (*4, 33, 35, 48, 53*).

[149] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet. *Novel Type of Phase Transition in a System of Self-Driven Particles.* **Phys. Rev. L** *75* 6. ***1995*** (*4, 42*).

[150] J. C. Crocker and D. G. Grier. *Microscopic measurement of the pair interaction potential of charge-stabilized colloid.* **Phys. Rev. L** *73*. ***1994*** (*101*).

[151] O. D. Lavrentovich and V. M. Pergamenshchik. *Stripe Domain Phase of a Thin Nematic Film and the K_{13} Divergence Term.* **Phys. Rev. L** *73.* ***1994*** (*25*).

[152] R. D. Polak, G. P. Crawford, B. C. Kostival, J. W. Doane, and S. Žumer. *Optical determination of the saddle-splay elastic constant K_{24} in nematic liquid crystals.* **Phys. Rev. E** *49.* ***1994*** (*25*).

[153] A. Sparavigna, O. D. Lavrentovich, and A. Strigazzi. *Periodic stripe domains and hybrid-alignment regime in nematic liquid crystals: Threshold analysis.* **Phys. Rev. E** *49.* ***1994*** (*25*).

[154] P. G. de Gennes and J. Prost. *The Physics of Liquid Crystals.* ***1993*** (*11, 14, 18, 23*).

[155] D. W. Allender, G. P. Crawford, and J. W. Doane. *Determination of the liquid-crystal surface elastic constant K_{24}.* **Phys. Rev. L** *67.* ***1991*** (*25*).

[156] K. W. Wojciechowski, D. Frenkel, and A. C. Bra ńka. *Nonperiodic solid phase in a two-dimensional hard-dimer system.* **Phys. Rev. L** *66.* ***1991*** (*66*).

[157] O. D. Lavrentovich and V. V. Sergan. *Parity-breaking phase transition in tangentially anchored nematic drops.* **Il Nuovo Cimento D** *12* 9. ***1990*** (*27, 32*).

[158] B. Fornberg. *Generation of finite difference formulas on arbitrarily spaced grids.* **Math. of Comput.** *51* 184. ***1988*** (*25, 116*).

[159] K. J. Strandburg. *Two-dimensional melting.* **Rev. Mod. Phys.** *60.* ***1988*** (*83*).

[160] C. A. Murray and D. H. Van Winkle. *Experimental observation of two-stage melting in a classical two-dimensional screened Coulomb system.* **Phys. Rev. L** *58.* ***1987*** (*83*).

[161] R. D. Williams. *Two transitions in tangentially anchored nematic droplets.* **Journal of Physics A** *19* 16. ***1986*** (*27, 30*).

[162] D. H. Lee, J. D. Joannopoulos, J. W. Negele, and D. P. Landau. *Discrete-Symmetry Breaking and Novel Critical Phenomena in an Antiferromagnetic Planar XY Model in Two Dimensions.* **Phys. Rev. L** *52* 6. ***1984*** (*61*).

[163] G. E. Volovik and O. D. Lavrentovich. *Topological dynamics of defects: boojums in nematic drops.* **Sov. Phys. JETP** *58* 6. ***1983*** (*27, 30*).

[164] O. G. Mouritsen and A. J. Berlinsky. *Fluctuation-Induced First-Order Phase Transition in an Anisotropic Planar Model of N2 on Graphite.* **Phys. Rev. L** *48* 3. ***1982*** (*60*).

[165] Y. Saito. *Monte Carlo studies of two-dimensional melting: Dislocation vector systems.* **Phys. Rev. B** *26.* ***1982*** (*101*).

[166] I. M. Lifshitz, A. Yu. Grosberg, and A. R. Khokhlov. *Some problems of the statistical physics of polymer chains with volume interaction.* **Rev. Mod. Phys.** *50.* ***1978*** (*14*).

[167] M. Schick and R. B. Griffiths. *Antiferromagnetic ordering in the three-state Potts model.* **J. Phys. A: Math Gen** *10* 12. ***1977*** (*60*).

[168] I. Newton. *Principia mathematica.* **Harvard University Press**. ***1972*** (*105*).

[169] Robert B. Meyer. *Piezoelectric Effects in Liquid Crystals.* **Phys. Rev. L** *22.* ***1969*** (*23*).

[170] M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables. 55.* ***1964*** (*119*).

[171] L. E. Dahler and J. S. Scriven. *Angular momentum of continua.* **Nat.** *192* 4797. ***1961*** (*68*).

[172] L. D. Landau, E. M. Lifshitz, J. B. Sykes, J. S. Bell, and M. E. Rose. *Quantum Mechanics, Non-Relativistic Theory. 11* 12. ***1958*** (*52*).

[173] J. J. Thomson. *On the structure of the atom: an investigation of the stability and periods of oscillation of a number of corpuscles arranged at equal intervals around the circumference of a circle; with application of the results to the theory of atomic structure.* **Lond. Edinb. Dubl. Phil. Mag.** *7* 39. ***1904*** (*91*).