Catalytic Reactions Tutorial

Henri Menke[1, a)] and Joost de Graaf[1, 2, b)]

[1)] *Institute for Computational Physics, Universität Stuttgart, Allmandring 3, D-70569 Stuttgart, Germany*

[2)] *School of Physics and Astronomy, University of Edinburgh, Scotland, Edinburgh EH9 3JL, United Kingdom*

(Dated: 16 August 2017)

In this tutorial we explore how to simulate catalytic reactions using ESPResSo for the purpose of modeling self-propelled particles (swimmers). We discuss the implemented reaction schemes and use one of them to study the enhanced translational diffusion of a catalytic swimmer. This should give a basic introduction on how to setup simulations involving catalytic reaction in ESPResSo and ideally enable you to conduct your own numerical experiments involving catalytic swimmers.

I. CATALTIC REACTIONS

Many chemical processes in nature do not occur at room temperature, because the energy penalty is too high to be overcome by thermal fluctuations. The activation energy of such processes can in some cases be lowered significantly in the presence of a catalyzer (a catalytic substance). Below, we discuss this concept in some detail, in order to prepare you for its application to catalytic swimmers.

We may write a general equilibrium reaction as

$$\underset{\text{reactants}}{\mathrm{R}} \rightleftharpoons \underset{\text{products}}{\mathrm{P}} \tag{1}$$

where R are the reactants and P are the products. We restrict ourselves to the situation where there is a single product and single reactant in the following. The double headed arrow is used to indicate that this reaction is in an equilibrium. The equilibrium reation (1) can be described by the equilbrium reaction constant

$$K_{\mathrm{eq}} = \frac{k_{\mathrm{eq},+}}{k_{\mathrm{eq},-}} = \frac{[\mathrm{P}]}{[\mathrm{R}]}, \tag{2}$$

with [P] and [R] the product and reactant concentration and $k_{\mathrm{eq},\pm}$ the forward and backward reaction rates. The chief effect of the reaction is a change of the concentrations of reactants and products over time, whenever the system is brought out of equilibrium. It pushes the system to maintain a steady-state concentration ratio between the products and reactants, governed by the equilibrium contstant K_{eq}. The associated differential equations can be written as

$$\frac{d[\mathrm{R}]}{dt} = +k_{\mathrm{eq},-}[\mathrm{P}] - k_{\mathrm{eq},+}[\mathrm{R}], \tag{3a}$$

$$\frac{d[\mathrm{P}]}{dt} = -k_{\mathrm{eq},-}[\mathrm{P}] + k_{\mathrm{eq},+}[\mathrm{R}]. \tag{3b}$$

A catalyzer shifts the free energy landscape in favour of one of the constituents R or P. We choose P, since it stands for the products and we want to enhance the forward reaction.

$$\underset{\text{reactants}}{\mathrm{R}} \xrightleftharpoons{\overset{\mathrm{C}}{\text{catalyzers}}} \underset{\text{products}}{\mathrm{P}} \quad . \tag{4}$$

If the back reaction rate is very small by comparison, it can be ignored and effectively the reaction only occurs in one direction.

$$\underset{\text{reactants}}{\mathrm{R}} \xrightarrow{\overset{\mathrm{C}}{\text{catalyzers}}} \underset{\text{products}}{\mathrm{P}} \quad . \tag{5}$$

[a)] `henri@icp.uni-stuttgart.de`

[b)] `jgraaf@icp.uni-stuttgart.de`

In the case of excessive catalysis (5) the back reaction rate $k_{\mathrm{eq},-}$ can be set to zero and the forward reaction rate is enhanced $k_{\mathrm{eq},+} \to k_{\mathrm{ct}}$. The rate equations (3) then reduce to

$$-\frac{d[\mathrm{R}]}{dt} = \frac{d[\mathrm{P}]}{dt} = k_{\mathrm{ct}}[\mathrm{R}], \tag{6}$$

which has a simple exponential solution $R(t) = R(0)\exp(-k_{\mathrm{ct}}t)$ and $P(t) = P(0) + R(0)(1 - \exp(-k_{\mathrm{ct}}t))$.

What we have discussed so far is a continuum model, as no actual particles are involved in the picture, only concentrations. To simulate such a simple catalytic reaction with molecular dynamics we need to come up with a minimal model. Phenomenologically, catalysis is triggered by a contact interaction, *i.e.*, the reactant has to make physical contact with the catalyzer. ESPResSo technically only simulates point particles — they obtain a size through the interaction potentials. To impose a finite extent, we define a *reaction range* around the catalyzer inside of which the conversion of a reactant to a product is feasible. This means that we have explicit catalyzer, reactant, and product particles, for which the inter-particle distance imposes whether a reaction takes place or not.

Next we have to think about how to implement (6) is this picture. Simply integration (6) yields an exponential decay as a function of time, see above. The reaction move is a stochastic process and takes place with a certain probability, depending on the reaction rate. This probability is inversely proportional to the exponential decay. We assign

$$P_{\mathrm{move}} = 1 - \mathrm{e}^{-k\,\Delta t}, \tag{7}$$

with the reaction rate $k \in \{k_{\mathrm{eq},+}, k_{\mathrm{eq},-}, k_{\mathrm{ct}}\}$ and the simulation time step Δt.

II. CATALTIC REACTIONS IN ESPRESSO

There are two implementations of the reaction scheme (3a), (3b), and (6), which we will refer to as *number conserving* and *non-number-conserving*. In the following we will discuss both of these schemes in detail. In all cases, there are only three species of the $N \geq 3$ possible species you can define in ESPResSo involved in the catalytic reaction. Multiple reactions involving a variety of species are not possible at this moment.

A. Non-Number-Conserving Scheme

In ESPResSo we assume the equilibrium constants in equations (3) to be equal, *i.e.*, $k_{\mathrm{eq},+} = k_{\mathrm{eq},-}$. Thus, according to (2), the reaction constant $K = 1$. The equilibrium constant is applied the reactant and product particles in the system and determines the conversion rate between these two. Essentially, this ensures that when there is a catalytic reaction, the fuel does not run out and the system is essentially coupled to a 'chemical bath'. For reactants in the vicinity of the catalyzer (as determined by the *reaction range*) the forward reaction is favoured and no back conversion will take place there. Also the forward reaction rate is enhanced to k_{ct} and can be specified as an input parameter. The catalyzer particles drive the system out of equilibrium.

Figure 1 shows the reaction scheme explained before pictorially. Within the reaction range, determined by r, the equilibrium reactions are also allowed to take place, but additionally the forward reaction is catalysed. Within one time step bulk reactions take place with the equilibrium constant. These might occur in both directions and push the the out-of-equilibrium concentrations produced by the catalyzer back towards the steady state. It is clear that it does not make sense to set $k_{\mathrm{eq}} > k_{\mathrm{ct}}$, as this would shadow the effect of the catalyzer compared to that of the equilibrium reactions in bulk.

The sketch in figure 1 and the above discussion explain why the scheme is called *non-number-conserving*. Namely, the number of products and reactants is not constant over time, even in the steady state there are short-time fluctuations. However, the overall number of particles is conserved. This is representative of the physical reality of a catalytic reaction, but frustrates the use of this algorithm in a number of situations. If, for instance, any of the species swapped around are charged, then the total charge of the simulation box need not be conserved, and by extention, the box need not be charge neutral. The electrostatic algorithms in ESPResSo break down for non-neutral boxes. In order to allow for the use of electrostatic algorithms, we introduce the *number-conserving* scheme next.

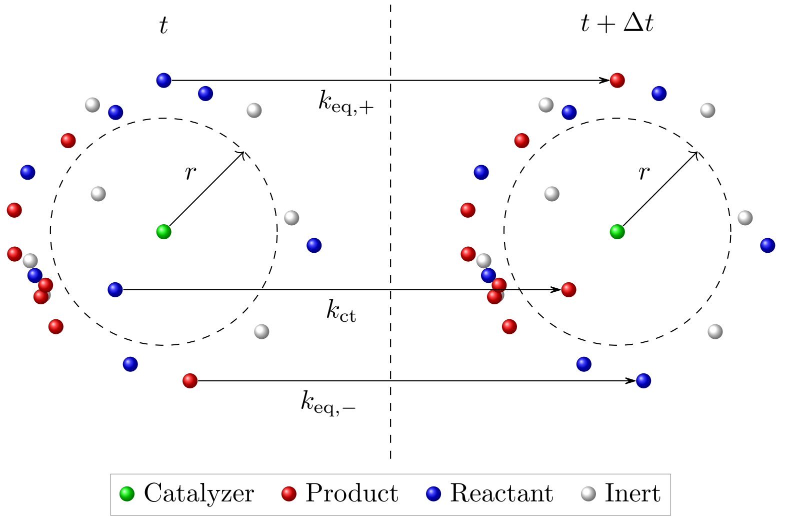

FIG. 1. Illustration of the *non-number-conserving* scheme. Within a cut-off radius r of the catalyzer (green), the reactant particles (blue) can be converted into products (red); inert particles (white) are left unaffected by the reactions taking place. The catalytic conversion is indicated using the arrow labeled k_{ct}, which shows the result of a conversion event taking place in one time step Δt (from left panel to right panel). Outside of this region a homogeneous equilibrium reaction takes place, that pushes the out-of-equilibrium concentrations produced by the catalyzer back towards equilibrium. This is indicated by the two arrows which cause conversion of products into reactants with $k_{\mathrm{eq},-}$ and reactants into products with $k_{\mathrm{eq},+}$. Even though $k_{\mathrm{eq},+} = k_{\mathrm{eq},-} = k_{\mathrm{eq}}$ in ESPResSo, we show them seperately to emphasise that one is a forward and one is a back reaction.

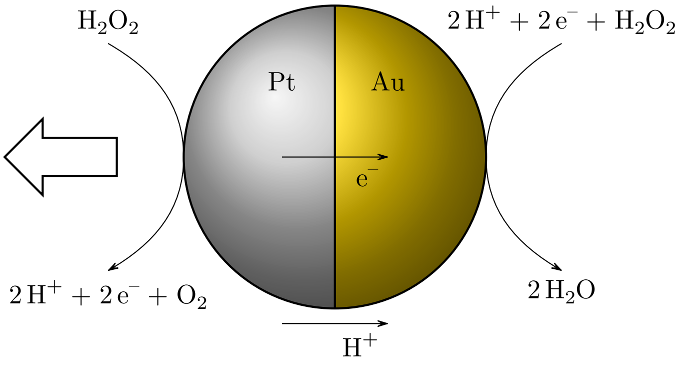

FIG. 2. A Janus swimmer, having a catalytic surface on both hemispheres, may support two different (redox) reactions. This results in breaking of reflection symmetry which propels the particle in the direction of the large arrow.

B. Number-Conserving Scheme

In this scheme, we disallow bulk reactions of single particles[a] and allow a reaction inside the reaction range involving only rectant-product pairs. Simply exchanging the particles in the pair would not yield an effect, when averaged over time, since the exchange occurs isotropically. We therefore introduced an additional symmetry breaking, namely of the spatial coordinate. This also conveniently allows us to model catalytic Janus swimmers.

Consider such a swimmer, see Fig. 2, consisting of Pt on one hemisphere and Au on the other hemisphere. Both surfaces enhance a reaction, here[?] [?]

$$H_2O_2 \xrightarrow{\mathrm{Pt}} 2\,H^+ + 2\,e^- + O_2 \tag{8}$$

$$2\,H^+ + 2\,e^- + H_2O_2 \xrightarrow{\mathrm{Au}} 2\,H_2O \tag{9}$$

[a] One can set the equilibrium reaction rate $k_{\mathrm{eq}} = 0$ to achieve this as well.

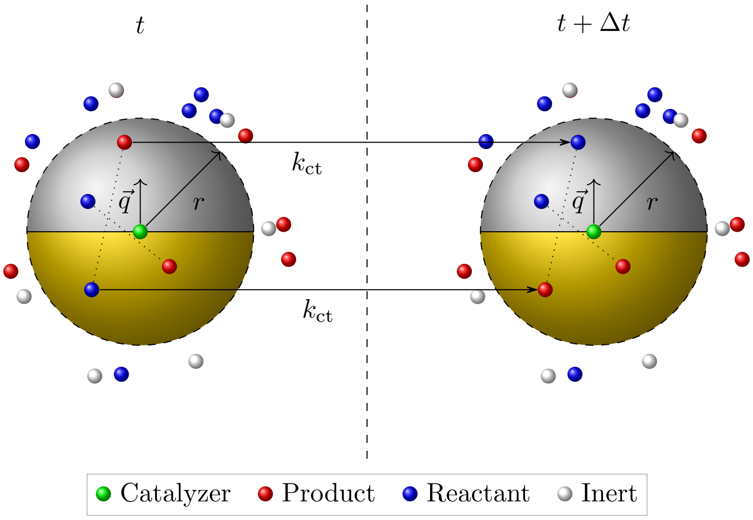

FIG. 3. Illustration of the *number-conserving* scheme. The cut-off range of r around the catalyzer (green) is subdivided into two half-spaces in analogy to figure 2. A reactant-product pair (blue and red connected by dotted line)is converted by swapping them if the pair is selected, and only if the product (red) resides in the upper half-space (silver background) and the corresponding reactant (blue) resides in the lower half-space (gold background). The catalytic reaction leads to a conversion of one reactant-product pair of particles in this example, denoted by the arrows annotated with k_{ct}. The second reactant-product pair within the reaction range is not viable for conversion. Additionally, particles in a pair may only undergo one exhange per time step.

The block arrow in the sketch points in the direction of movement induced by the reaction. We refer to upper and lower hemisphere (or half-space) in the following, as two regions that are determined with respect to the orientation of the particle and the plane through the center of the particle, with the orientation as its normal. In the present case one might choose Au as the upper hemisphere and Pt as the lower hemisphere. The reaction region is, as before, bounded by the *reaction range* r.

For simplicity, in the simulation we abstract away all the reaction mechanism including the electron transport through the Janus particle. The proton transport around the particle (through, diffusion, advection, and migration) is taken into account implictily via the thermalized MD simulation. We end up with a simple scheme, depicted in figure 3.

As already explained, reactions can only take place for reactant-product pairs. The conversion is such that a reactant is converted to a product and a product is converted to a reactant. This ensures conservation of the particle number for each type. To model a Janus particle as in Fig. 2, we allow an exchange move only when the following conditions are met:

1. Both partners of the reactant-product pair have to reside within the reaction range.

2. The product has to reside in the upper half-space of the reaction range.

3. The reactant has to reside in the lower half-space of the reaction range.

This is illustrated in Fig. 3.

C. Usage in ESPResSo

Catalytic reactions can be enabled in ESPResSo by compiling in the eponymous feature `CATALYTIC_REACTIONS`. The number-conserving method additionally requires the `ROTATION` feature to provide the particles with an orientation vector. In ESPResSo the orientation of the particle is defined by a quaternion; this in turn defines a rotation matrix that acts on the particle's initial orientation (along the z-axis), which then defines the particles current orientation[? ? ?] .

In ESPResSo particles are set up using the `system.part.add` command. It allows to set various options, such as the initial position (mandatory), the type, and the charge. To setup the reaction there is the ESPResSo command `Reaction`, which operates on particle types. The general syntax is

```
espressomd.reaction.Reaction(reactant_type=R,catalyzer_type=C,product_type=P,ct_range=r,ct_rate=k_ct,
                             eq_rate=k_eq, # optional
                             react_once=True/False, # optional
                             swap=True/False) # optional
```

The parameters in square brackets are optional and can be left out. They will then assume their default values which are `eq_rate=0.0`, `react_once=False`, and `swap=False`.

Important: Due to the method of implementation there can only be one reaction. You can alter the reaction parameters, but you may not change the reaction partners.

To set up a reaction between the types 1 and 2, where particles of type 3 act as catalyzers with a reaction range of 1.5 around them with a reaction rate of 20, one types

```
espressomd.reaction.Reaction(reactant_type=1,catalyzer_type=3,product_type=2,ct_range=1.5,ct_rate=20)
```

Here we have left out the optional parameters, but their meaning is nevertheless important. The first one, `eq_rate`, should be self explanatory; it sets the equilibrium reaction rate as detailed in the introductory sections. The `react_once` parameter determines whether a particle can take part in a only single reaction move per time step. In the case of multiple catalyzers in the system a particle might be tagged for reaction several times by different catalyzers because their reaction ranges overlap and the reactant is inside this overlap. This can be prevented by setting `react_once on`. That way the reaction rate is independent of the density of catalyzers. This can be convenient if one wants to model a surface using several catalyzer particles that are in close proximity to one another, but still have a uniform catalytic conversion rate for the entire surface. Finally, the parameter `swap` determines which scheme to use; `swap=False` uses the not number conserving scheme, `swap=True` uses the number-conserving scheme. The name 'swap' originates from the sketch figure 3, because it looks as if the particles are swapped.

However, the `swap` option does not really swap the particles, but only exchanges their type (and their charge, if ESPResSo was compiled with `ELECTROSTATICS`). This means that the particle numbers do not change. This allows one to track how frequently a particle undergoes a reaction, by tracking the properties of a particle via the particle number.

III. CONFIGURING ESPRESSO FOR CATALYTIC REACTIONS

For this tutorial to work you need a version of ESPResSo containing the catalytic reactions feature with the `swap` mechanism. Therefore check out the ESPResSo online repository at `https://github.com/espressomd/espresso`. If you have installed `git`, you can issue on the command line

```
$ git clone https://github.com/espressomd/espresso.git
```

Now you are ready to build ESPResSo. Change to the newly created directory and use the following commands to configure ESPResSo for compilation.

```
$ mkdir build
$ cd build
$ cmake ..
```

After this, you will need to copy the `myconfig-sample.hpp` file into `myconfig.hpp` and select the appropriate `FEATURES` in the latter.

```
$ cp myconfig-sample.hpp myconfig.hpp
```

To run all the tutorials you need to uncomment the following `FEATURES`:

```
#define ROTATION
#define ROTATIONAL_INERTIA
#define LANGEVIN_PER_PARTICLE
#define CATALYTIC_REACTIONS
#define LENNARD_JONES
```

Now you are ready to build ESPResSo. You have to run `cmake` again to process the modified `myconfig.hpp`.

```
$ cmake ..
$ make -j N
```

with N the number of processes used to build. Next you can unpack the archive with the tutorial files in this directory. You will find two folders, one called 'EXERCISES' and one called 'SOLUTIONS'.

IV. THE ENHANCED-DIFFUSION TUTORIAL

In the folder 'EXERCISES' you will find the `reaction.tcl` file. It is a tutorial to demonstrate that our approach to catalytic reactions leads to enhanced diffusion of the catalyzer. When you begin, the code is incomplete and will produce errors when evaluated in ESPResSo. It needs your input to function properly. A fully functional file exists in the 'SOLUTIONS' folder, but we recommend that you try solving the exercises on your own first.

To start the exercises, go into the 'EXERCISES' directory and invoke ESPResSo on the script

```
$ ./../build/Espresso reaction.py 0
```

where the parameter `0` determines whether the reaction is enabled. Here, 0/1 corresponds to reaction off/on, just like on your VCR. At this stage, executing the above line will cause an error, as the exercise has not yet been completed.

A. Structure of the Simulation

Let's walk through the script. It is best to open the file while reading this.

First, we read the activity parameter from the command line and verify it. Then we set up some general simulation parameters, such as box length, radius of the colloid, concentration of reactant and products, and the reaction rate. Next, we setup the thermostat parameters, but do not enable it yet.

Before we can set up the colloid and the small particles around, the first two exercises have to be completed.

The script continues to setup interactions between the colloid and the smaller particles and, most importantly, the reaction. The syntax for setting up a reaction is given above in the section "Usage in ESPResSo".

Warmup is performed by the `minimize_energy` routine. It has several advantages over traditional force-capping warmup, which you will learn about when completing the associated exercise.

Now the thermostat is enabled and the equilibration is performed. Finally, we perform five production runs in which we measure the mean-square displacement (MSD) and the angular velocity auto-correlation function (AVACF).

You should have learned about everything that does not have to with catalytic reactions in previous tutorials. If you are unfamiliar with any concept, it might be better to go back and complete the other tutorials first.

B. What to Expect

Once you have completed all the tasks in the exercise script, the time has come to run the actual simulation. Run the script twice, once with the activity parameter set to 0 and once set to 1. You will have two directories: `active-system` and `passive-system`. These contain MSD and AVACF data files numbered consecutively over the different runs.

It is now your job to average over these five files. You can do this in `gnuplot` or the scripting language of your choice. The MSD files have five columns each. The first two columns are time and samples, the last three are x, y, z. The AVACF files have only three columns, where the first two are also time and samples and the third is the AVACF averaged over all spatial components. For the MSD we still have to average over the three spatial dimensions.

When we have extracted the mean (and recommendably the standard error) we can plot it over time to achieve plots as in Figs. 4 and 5. We can clearly see, that the reaction facilitates enhanced translational diffusion while leaving the rotational diffusion unaffected.

Specifically, a passive particle shows a t^2 ballistic regime, typical for Langevin type simulations, followed by an intermediate regime, where the slope changes gradually to the long-time diffusive t. The active swimmer, shows similar behavior. However, the intermediate regime is stretched, with the long-time diffusion setting in only at around $t = 10$ (which corresponds to one rotational diffusion time). This is because the self-propulsion induces a persistent motion along the particle axis, leading to a more-or-less t^2 MSD; here a little less as the increase matches $t^{1.4}$ better. The only thing that breaks this persistence is rotational diffusion. Hence beyond a time for which the orientation of the particle randomizes, the behavior becomes diffusive again, albeit with a higher diffusion coefficient. It is therefore important to show that the rotational diffusion remains unaffected by the particle catalyzing solutes in its surrounding.

V. CONCLUDING REMARKS

With that, you have come to the end of this tutorial. We hope you found it informative and that you have a sufficient understanding of the way to deal with catalytic reactions in ESPResSo to set up simulations on your own.

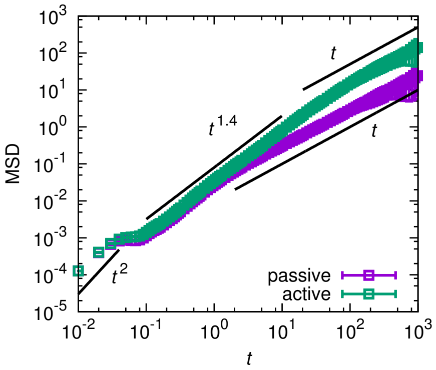

FIG. 4. Averaged MSD over five runs with standard error on the error bars for both, the active and the passive system. The black lines serve as a guide to the eye and indicate the dependence of the MSD on the time t.

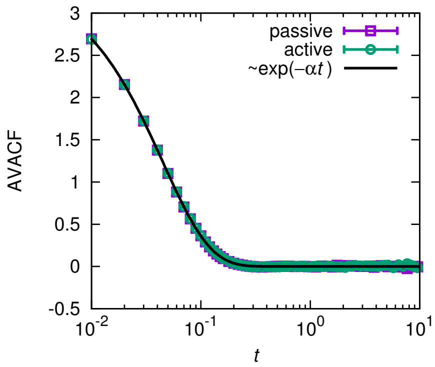

FIG. 5. The AVACF for the same system as in figure 4. Note that the activity does not influence the rotational behavior.

REFERENCES

[1] John G. Gibbs, Nicholas A. Fragnito, and Yiping Zhao. Asymmetric pt/au coated catalytic micromotors fabricated by dynamic shadowing growth. *Applied Physics Letters*, 97(25), 2010.

[2] Philip M. Wheat, Nathan A. Marine, Jeffrey L. Moran, and Jonathan D. Posner. Rapid fabrication of bimetallic spherical motors. *Langmuir*, 26(16):13052–13055, 2010. PMID: 20695541.

[3] A. Arnold et al. Espresso user guide. *User Guide: ESPResSo git repository*, 3.4-dev-1404-g32d3874:1, 2015.

[4] H. J. Limbach, A. Arnold, B. A. Mann, and C. Holm. ESPResSo – an extensible simulation package for research on soft matter systems. *Comp. Phys. Comm.*, 174:704, 2006.

[5] A. Arnold, O. Lenz, S. Kesselheim, R. Weeber, F. Fahrenberger, D. Roehm, P. Košovan, and C. Holm. ESPResSo 3.1 — Molecular Dynamics Software for Coarse-Grained Models. In M. Griebel and M. A. Schweitzer, editors, *Meshfree Methods for Partial Differential Equations VI*, volume 89 of *Lecture Notes in Computational Science and Engineering*, page 1. Springer, 2013.