Converting from simulation to SI

symbol	meaning	unit	simulation value	SI value
M	cell mass	mass	N_v simulation mass	$3 \cdot 10^{-12}$ kg
V_0	starting cell volume	length^3	1 simulation volume	$23^3 \cdot 10^{-18}$ m^3
P	volume oscillation period(?)	time	$\sqrt{\frac{M}{LB}} = \sqrt{\frac{N_v V_0}{V_0^{1/3} \epsilon_v}} = \sqrt{N_v}$ simulation time	$1.14 \cdot 10^{-8}$ s

symbol	meaning	unit	value
M'	mass re-scaling factor	$\frac{\text{kg}}{\text{simulation mass}}$	$\frac{3 \cdot 10^{-12}\ \text{kg}}{N_v\ \text{simulation mass}}$
L'	length re-scaling factor	$\frac{\text{m}}{\text{simulation length}}$	$\frac{23 \cdot 10^{-6}\ \text{m}}{1\ \text{simulation length}}$
T'	time re-scaling factor	$\frac{\text{s}}{\text{simulation time}}$	$\frac{1.14 \cdot 10^{-8}\ \text{s}}{\sqrt{N_v}\ \text{simulation time}}$

Converting other things:

symbol	meaning	unit	value in simulation units	SI value
ϵ_v	volume spring constant	Energy Length^{-6}	1	10^{23} J m^{-6}
ϵ_a	area spring constant	Energy Length^{-4}	$\frac{1}{2}$	10^{13} J m^{-4}
ϵ_c	contact spring constant	Energy Length^{-2}	$\frac{1}{2}$	10^{4} J m^{-2}
ϵ_b	bending spring constant	Energy radian^{-2}	10^{-4}	10^{-9} J Radian^{-2}
$\epsilon_v \bar{V_0}$	cell volume modulus	Energy Length^{-3}	1	10^{9} J m^{-3}
$\epsilon_a \bar{a_0}$	face area modulus	Energy Length^{-2}	10^{-3}	10^{2} J m^{-2}
$\epsilon_c \sigma$	contact spring modulus	Energy Length^{-1}	10^{-4}	10^{-8} J m^{-1}
$\epsilon_b \bar{\theta_0}$	edge bending spring modulus	Energy Radian^{-1}	10^{-7}	10^{-13} J Radian^{-1}
$\epsilon_v \bar{V_0}^2$	cell volume energy scale	Energy	1	10^{-5} J
$\epsilon_a \bar{a_0}^2$	face area energy scale	Energy	10^{-4}	10^{-10} J
$\epsilon_c \sigma^2$	contact spring energy scale	Energy	10^{-3}	10^{-8} J
$\epsilon_b \bar{\theta_0}^2$	edge bending spring energy scale	Energy	10^{-10}	10^{-16} J

13 Creating Jammed Packings

13.1 Binary Search

Binary search for jammed packings. The packing must begin at a low density with pressure less than target pressure P_t. A range of acceptable pressures is set from the high value of pressure $P_{t,h}$ to the low value of pressure $P_{t,l}$. Usually they are one percent away from each other. A binary search like algorithm is used to find the a box size L that satisfies the target pressure.

At first $P < P_{t,l}$, and in each iteration L is shrunk by a fixed amount r_{scale}. Each time the algorithm finds an unjammed state it saves this state.

The loop: If the pressure is less than the lower bound of the target pressure, the state is saved, L_h is noted. If, currently a jammed state is known L is set to be in the middle of the known over-pressure and under-pressure states. Since we are at lower than threshold pressure – rearrangements are a concern and so L_l is unset as this may no longer correspond to a jammed packing. If L_l was no known at the beginning of this iteration, L is instead rescalled by a fixed amount r_{scale}.

If $P > P_{t,l}$ this is an over-pressure state. First the state is returned to the most recent under-pressure state. Then L_l is set to be the current L, and the new L is set to be the mean of the new L_l and the previously known L_h.

If $P_{t,l} < P < P_{t,h}$ its a success and the loop breaks. The loop also breaks if the lower and upper bounds are too close to one another and it looks impossible to find a target pressure state.

When there is a known over-pressure state (L_l is set) and we keep finding over-pressure states this works as a binary search algorithm. However due to the possibility of rearrangements we can't use the binary search method when we have an under-pressure state, hence the unsetting of L_l when we find under pressure states. In pseudo code the algorithm is as follows:

```
Set $P_{t,l}$ ;                                  /* lower pressure bound for the jammed state */
Set $P_{t,h}$ ;   /* upper pressure bound for the jammed state, often $P_{t,h} = 1.01 * P_{t,l}$ */
Set $E_{\text{thresh}}, N_{\text{minimization}}$ ;   /* energy minimization sensitivity - could replace with force as appropriate, and
  max number of minimization steps */
$success = 0$;
$r_{\text{scale}} = 0.999$ ;                                                  /* rate of shrinking */
$L_l = -1$; $L_h = -1$;                                    /* Lower and upper bound of box size */
Set $L_{tol}$ ;                                      /* Smallest space searched usually Ĩe-10 */
Initialize System, state.Positions, state.Velocities, box size $L$ ;   /* Initialize state with positions and
  velocities */
Save, OldState = state;
while true do
    state = EnergyMinimize(state, $L$, $E_{\text{thresh}}$, $N_{\text{minimization}}$);
    $L_{\text{prior}} = L$ ;                               /* save prior $L$ for rescalling purposes. */
    $P$ = Pressure(state, $L$);
    if $P < P_{t,l}$ then
        OldState = state ;                                          /* save known unjammed state */
        $L_{\text{old}} = L$;
        $L_h = L$ ;                      /* when $L_h > 0$ it is the box length for a known unjammed state */
        if $L_l > 0$ then
            $L = (L_h + L_l)/2$ ;  /* split the difference between known jammed state and known unjammed state */
            $L_l = -1$ ; /* If $P < P_{t,l}$ rearrangment possible, $L_l$ may no longer correspond to unjammed state */
        else
            $L = L \cdot r_{\text{scale}}$ ;          /* shrink the box by a fixed ammount if no jammed state is known */
        end
    else
        if $P > P_{t,h}$ then
            $L_l = L$ ;                      /* when $L_l > 0$ is is the box length for a known jammed state */
            state = OldState ;                          /* reset to prior state and shrink a smaller amount */
            $L_{\text{prior}} = L_{\text{old}}$;
            $L = (L_h + L_l)/2$ ;  /* split the difference between known jammed state and known unjammed state */
        else
            success = $true$;
            break;
        end
    end
    state.Positions = state.Positions $\cdot L/L_{\text{prior}}$ ; /* Rescale system to avoid boundary effects.  NB: be aware of
      added presress esp.  with many body potentials */
    if $L_l > 0$ and $L_h > 0$ and $|L_h/L_l - 1| < L_{tol}$ then
        error("No Jammed State Found");                  /* no jammed state at this pressure could be found */
    end
end
```

Algorithm 5: Jamming Threshold