Supporting Information

A Thermodynamic Tank Model for Studying the Effect of Higher

Hydrocarbons on Natural Gas Storage in Metal-Organic Frameworks

Hongda Zhang,[a] Pravas Deria,[b] Omar K Farha,[b, c] Joseph T Hupp,[b]
and Randall Q Snurr*[a]

[a] *Department of Chemical and Biological Engineering, Northwestern University,
2145 Sheridan Road, Evanston, Illinois 60208, United States*
[b] *Department of Chemistry and International Institute for Nanotechnology,
Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, United States*
[c] *Department of Chemistry, Faculty of Science, King Abdulaziz University,
Jeddah, Saudi Arabia*

Simulation details

1. Single-component adsorption isotherms

To obtain the single-component isotherms of methane, ethane and propane in all 120 selected MOFs (except for HKUST-1), we performed grand canonical Monte Carlo (GCMC) simulations using a classical force field with the RASPA code.[1] The van der Waals interactions were modeled using a truncated Lennard-Jones potential without tail correction terms and a cutoff of 12.8 Å. MOF frameworks were treated as rigid and the adsorbate molecules were allowed to be flexible (except for methane). United atom models were used for three alkanes. Universal force field[2] and TraPPE force field[3] parameters were used for MOF atoms and methane, ethane and propane molecules, respectively. Lorentz-Berthelot mixing rules were used to obtain the guest-host Lennard-Jones interaction parameters. Each system was equilibrated for 2000 Monte Carlo (MC) cycles (each cycle contains a number of Monte Carlo steps equal to the number of adsorbed guest molecules in the current system) followed by another 3000 MC cycles to calculate the ensemble averages. Heats of adsorption of methane, ethane and propane were calculated at 0.001 bar and 298 K using the fluctuation method during the GCMC simulations.[4]

2. Ternary mixture adsorption isotherms

As a validation of the IAST prediction used in the ANG tank model, ternary mixture GCMC simulations were also carried out in some of the MOF structures. The mixture simulations used the same method and force field parameters as introduced above. The gas phase composition was fixed at 96% methane, 3.3% ethane, and 0.7% propane. 10000 MC cycles were used to reach equilibration and 15000 MC cycles were used for the production phase. The comparison between the mixture GCMC simulation result and the IAST prediction in NU-125 is shown in **Fig. 1**.

3. Calculations of MOF properties

Helium void fractions were calculated via Widom insertions of a helium probe and the void fraction was equal to the average Widom factor.[5] The pore size distributions (PSD) were calculated using the method of Gelb and Gubbins.[6] Volumetric and gravimetric surface areas were calculated geometrically with a probe, whose radii was 1.86 Å, using the Zeo++ software package.[7] The radii of the framework atoms are from the data set provided by the Cambridge Crystallography Data Center (CCDC).

Experiment details

1. Synthesis of HKUST-1

Due to the poor prediction of methane adsorption isotherm in HKUST-1 that we observed in **Table 2**, we decided to use the experimentally measured adsorption isotherms instead of GCMC simulation results for this particular structure. A high-quality HKUST-1 sample was prepared via solvothermal reaction of cupric nitrate and benzene-tri-carboxylic acid following a literature method[8] in a solvent mixture of water and ethanol. This as-synthesized HKUST-1 material contains both H_2O and EtOH as coordinating ligands. To prepare pure HKUST-1 used in this study, we first exchanged this material with ethanol (3x over 8 h each) and then subjected it to supercritical CO_2 activation[9], after which this solid was transferred in a drying tube inside a glove box and activated at the activation port of a Micromeritics 2020 ASAP sorption instrument with a stepwise heating (30 min each at 70 and 90 °C, and then 12 h at 110 °C) under dynamic vacuum. This sample was handled under inert conditions where the BET surface area of this sample was then measured to be 1860 m^2/g with a total pore volume of 0.77 cm^3/g.

2. Gas adsorption measurement in HKUST-1

The methane adsorption data in HKUST-1 were obtained from ref 10. The reason why we did not test it again in our lab is that the instrument we currently possess has a pressure limit of 20 bar, which is far below the pressure point we are interested in (65 bar). Gravimetric ethane and propane adsorption isotherms were collected in our lab on an IGA-200 (Hiden Isochema) instrument at 25 °C (298 K) below their corresponding saturation pressures up to ca. 8 and 4 bar, respectively. The HKUST-1 sample used in the measurement was 19.70 (+/- 0.012) mg (fully activated and evacuated). The instrument directly measured the absolute gravimetric uptake and the unit of the adsorption uptake has been converted into number of molecules per unit cell using the crystallographic density. The results are shown in **Figure S1**.
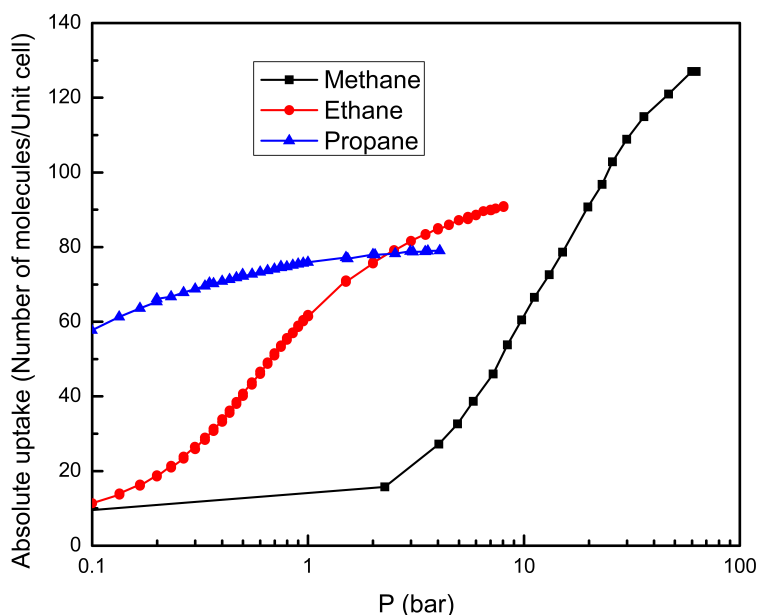
Figure S1. Experimental methane, ethane, and propane single-component adsorption isotherms in HKUST-1 at 298 K. The methane isotherm is from ref. 10. Ethane and propane adsorption measurements were measured in this work as described above.

Validation of IAST

Ideal adsorbed solution theory (IAST) is the most widely used method for estimating mixture adsorption from single-component isotherms and has been shown to work well for many systems in MOFs. In particular for alkane adsorption, IAST has been validated in the literature for hexane isomer mixtures and for adsorption of some other alkane isomer mixtures (C4, C5, C7) in a series of zeolites and MOFs.[11, 12] In **Figure S2**, we show the selectivity predicted from IAST (using the simulated single-component isotherms as input) compared to the selectivity predicted from full mixture simulations for an equi-molar hexane isomer mixture in ten zeolites and MOFs at 433 K and 1 bar obtained from the literature[11, 12]. The definition of the selectivity is:

$$Selectivity = [(q_1 + q_2 + q_3) / (q_4 + q_5)] / [(f_1 + f_2 + f_3) / (f_4 + f_5)]$$

where f_i and q_i are the fugacity and corresponding loading of component i in the mixture, and i is the component number (1 = n-hexane, 2 = 2-methylpentane, 3 = 3-methylpentane, 4 = 2,3-dimethylbutane, 5 = 2,2-dimethylbutane). From the figure we can see that the correlation is quite promising. In addition, we also directly compared the loadings of each component obtained from mixture simulations and IAST calculations at 433 K and 1 bar. In **Table S1**, we report the relative differences of loadings (relative difference = $(q_{i,simulation} - q_{i,IAST})/q_{i,simulation}$) for each component in each material. In general, the differences are small. Note that a few large numbers in the table are for cases where the absolute loadings are low. To summarize, IAST works very well for mixtures of hexane isomers, which are chemically quite similar to the alkane mixtures that we study in this work. Furthermore, IAST is known to work well for 1) homogeneous adsorbents and 2) molecules that are similar in size and chemical nature. The second condition is clearly met by methane/ethane/propane mixtures, and the 120 MOFs screened in this work do not possess strong binding sites such as open metal sites, which would introduce strong

heterogeneity in the adsorbent. Accordingly, IAST should be reliable for application in the ANG tank model.
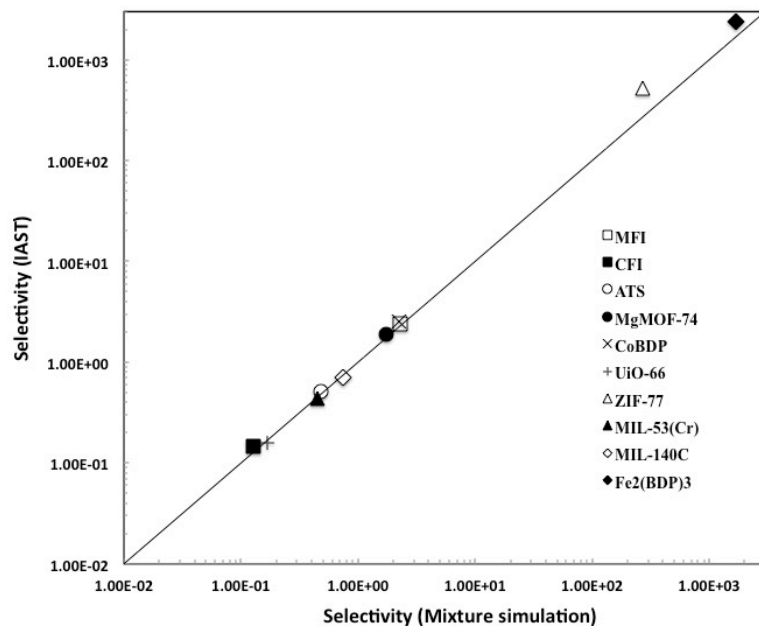
Figure S2. Correlation between the selectivity of hexane isomers calculated from full mixture simulations and from IAST at 433 K and 1 bar in zeolites MFI, CFI, ATS and MOFs MgMOF-74, CoBDP, UiO-66, ZIF-77,[11] MIL-53(Cr), MIL-140C and Fe$_2$(BDP)$_3$.[12]

Table S1. Relative differences of loadings from mixture simulations and IAST calculations at 433 K and 1 bar.

Materials	Relative difference of loadings (%)				
	Component 1	Component 2	Component 3	Component 4	Component 5
MFI	39.1	10.2	0.9	10.8	3.0
CFI	60.0	21.6	1.2	2.5	3.4
ATS	11.3	7.4	0.6	4.0	2.3
MgMOF-74	0.4	0.0	2.2	3.2	25.3
CoBDP	0.4	3.7	5.9	6.7	12.0
UiO-66	13.6	11.9	0.0	0.9	2.2
ZIF-77	5.6	6.0	21.2	50.0	63.6
MIL-53(Cr)	0.0	2.1	3.3	0.5	0.8
MIL-140C	12.8	2.9	9.6	3.8	2.8
$Fe_2(BDP)_3$	0.3	32.7	37.7	56.8	8.4

# Assumptions adopted in the model

Here we provide detailed justification for the three main assumptions in the ANG tank model. The first assumption is to ignore heat effects and assume isothermal filling and releasing processes. We make this approximation because we propose to use the ANG tank system with home-filling stations where the tank is filled slowly overnight. Also, when the vehicle is working on the road, the tank releases the gaseous fuel to the engine over a period of many hours. In this slow filling and releasing scenario, it is valid to assume that the heat effects can be ignored. Additionally, a slow filling and releasing scenario has been adopted in experimental investigations of ANG tanks, and the

temperature could be successfully controlled at ambient temperature (298 K).[13]

The second assumption is to neglect pressure and concentration gradients inside the tank. Again, for a slow filling and releasing process, properties of the system inside the tank should be quite uniform because the system is not strongly perturbed. Thus, the pressure and concentration gradients should be extremely small.

The last assumption is that the gas phase and the adsorbed phase reach equilibrium instantaneously during the adsorption or desorption process. In previous work[14], we conducted molecular dynamics simulations to predict the diffusivities of these alkanes in some selected MOFs. We estimated that it takes only a few tens of milliseconds for the alkane to diffuse through a 1 μm particle, which is quite fast compared with the time scale of the filling or releasing process (hours).

Additional results
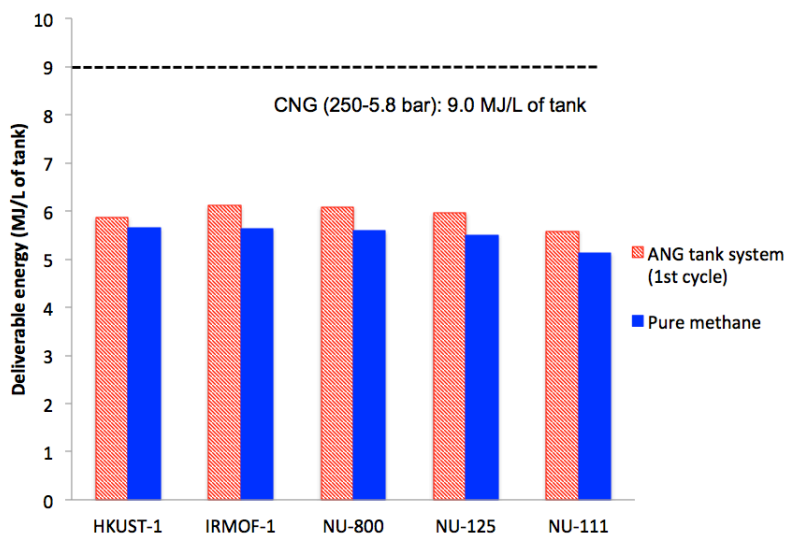
Figure S3. Comparison between the deliverable energy (65-5.8 bar) of the first cycle for natural gas mixtures and for pure methane in the five selected MOFs.
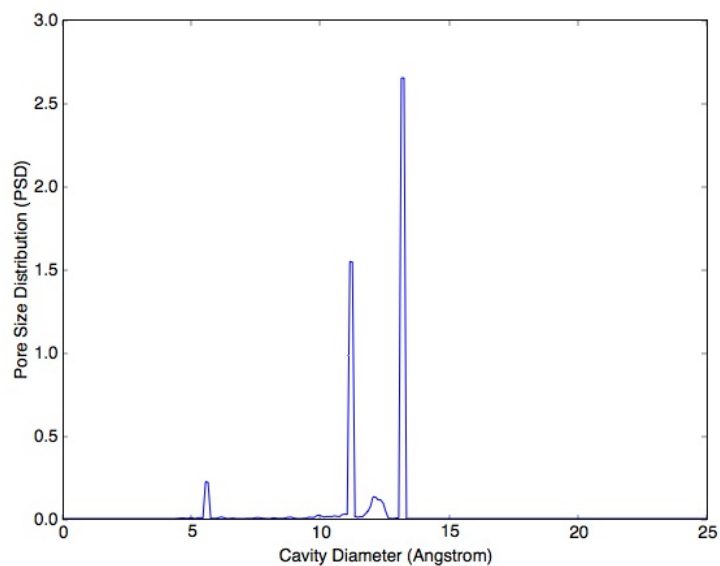
Figure S4. Calculated pore size distribution of HKUST-1.
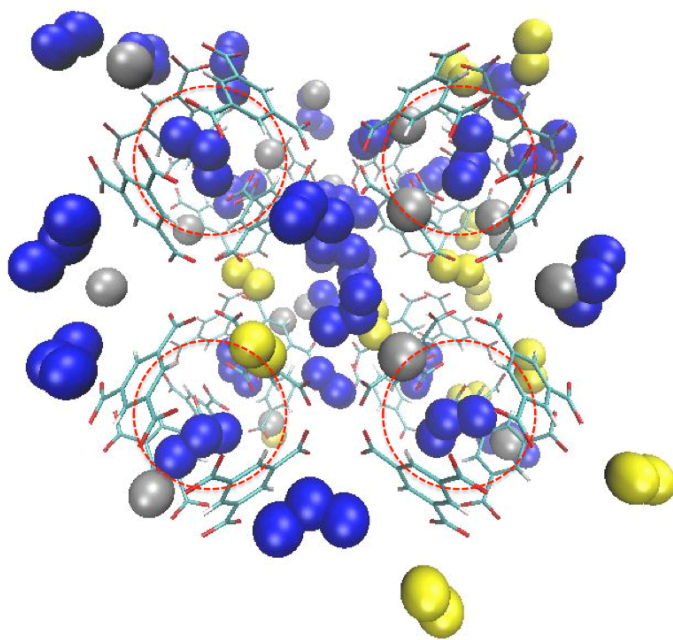
Figure S5. Snapshot of adsorbed methane (gray), ethane (yellow) and propane (blue) molecules in HKUST-1 during GCMC mixture simulation at 5.8 bar and 298 K. The 4 small cages in the front are labeled with red circles. There are also 4 small cages in the back of this unit cell. It is observed that each small cavity traps a propane molecule.
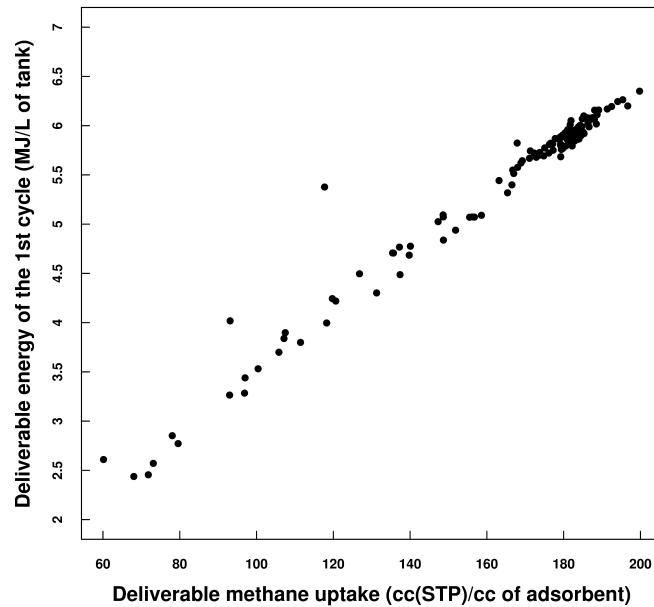
Figure S6. Correlation between the deliverable energy at the first cycle and the deliverable capacity of pure methane (65-5.8 bar).
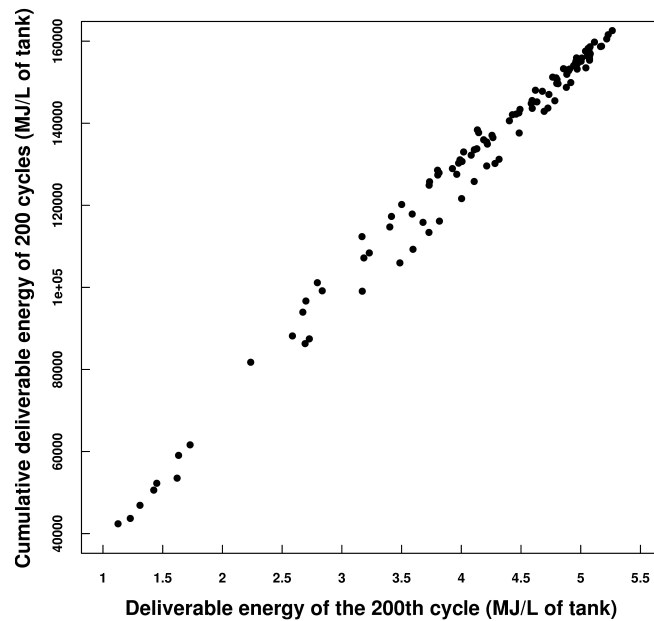
Figure S7. Correlation between the cumulative deliverable energy of 200 cycles and the deliverable energy at the 200th cycle.
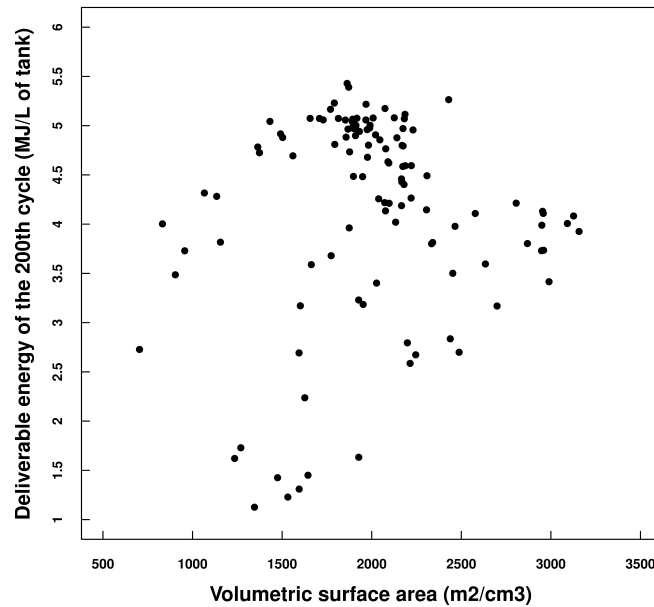
Figure S8. Correlation between the deliverable energy at the 200th cycle and the geometrical volumetric surface area.
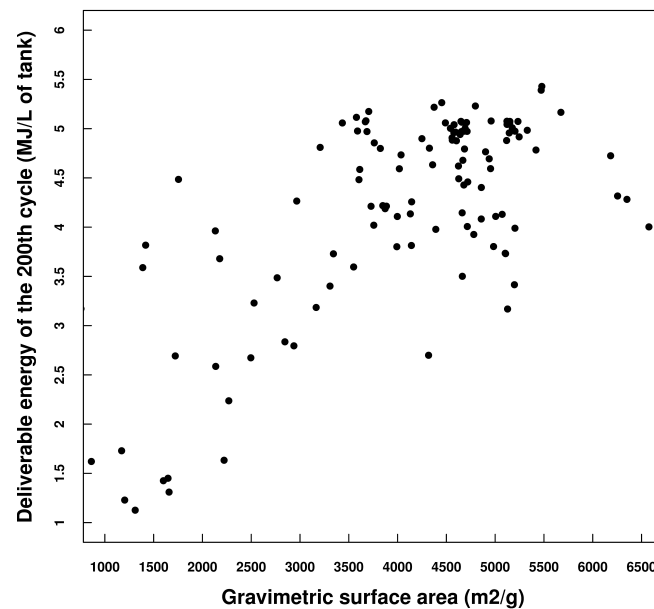
Figure S9. Correlation between the deliverable energy at the 200th cycle and the geometrical gravimetric surface area.

Table S2. The methane uptake and deliverable energy data of 120 Screened MOFs.

CCDC structure ID	CH$_4$ uptake 5.8 bar (cc/cc of adsorbent)[a]	CH$_4$ uptake 65 bar (cc/cc of adsorbent)[a]	Del. energy of the 1st cycle (MJ/L of tank)	Del. energy of the 200th cycle (MJ/L of tank)	Miles per tank (steady state)
ACUFEK	55	237	5.91	4.19	123
ALUKIC	51	234	5.84	3.99	117
ALUKOI	52	233	5.86	3.73	110
ALUKUO	51	234	5.79	4.13	121
ALULAV	53	235	5.83	4.11	121
AMODUC	61	245	5.90	3.80	112
AMOFAK	63	243	5.79	3.73	110
ANUGEW	36	218	5.90	4.91	144
ANUGIA	58	247	6.02	4.02	118
ANUGOG	66	254	6.06	3.80	112
BAZFUF	26	208	6.05	5.43	159
BAZGAM	12	119	3.90	4.00	118
BICPUA	46	242	6.20	4.86	143
BIWYAH10	22	157	4.71	4.21	124
CAVPEW	36	216	5.94	4.97	146
CAVPIA	37	220	5.93	4.94	145
CAVPUM	36	218	5.96	5.00	147
CAXSUR	75	241	5.40	3.40	100
COJHIT	81	218	4.49	2.24	66
DIDDOK	62	242	5.77	3.50	103
DUBWON	128	196	2.44	1.23	36
EDUVOO	29	210	5.93	5.23	154
EFAYIU	33	217	5.97	4.96	146

EHIJAH	31	202	5.74	5.06	149
EPOTAF	68	247	5.68	3.42	100
FAHPOV	33	200	5.51	4.73	139
FAPVAV	133	206	2.57	1.31	38
FEBXIV	25	173	5.03	4.88	143
FEFDEB	48	233	5.93	4.48	132
GAQYIH	22	191	5.62	5.17	152
GUPCAW	39	179	4.68	3.68	108
HAFTOZ	39	223	5.99	4.98	146
HAWWIN	67	198	4.30	2.59	76
HEXVEM	19	156	4.77	4.78	140
HKUST-1	83	267	5.86	3.96	116
HOGLEV	80	235	5.07	3.23	95
HOHMEX	25	165	4.78	4.69	138
HOHMIB	60	233	5.70	4.22	124
ICAQIO	39	230	6.17	5.06	149
ICAQOU	40	228	6.16	5.04	148
ICAREL	75	231	5.07	3.18	93
ICAROV	40	229	6.16	5.07	149
ILITUT	50	229	5.81	4.26	125
IYOWID	37	218	5.96	4.97	146
JEJWEB	47	234	5.99	4.59	135
KANMIX	94	190	3.28	1.73	51
LEHXUT	34	208	5.73	4.88	143
LEJCIO	27	194	5.55	5.07	149
LELDUD	41	223	5.91	4.80	141
LETQEI	25	132	3.84	3.82	112

LUFNAC01	123	195	2.46	1.13	33
LURRIA	39	224	6.10	4.96	146
MOCKAR	85	250	5.32	2.79	82
NAYZOE	45	229	5.93	4.49	132
NIBHOW	20	169	5.09	5.04	148
NIBHUC	35	214	5.86	4.90	144
NIZZIG	82	200	5.38	3.59	105
NOFRAC	108	201	4.02	3.17	93
NU-111	32	200	5.58	4.97	146
NU-125	56	240	5.96	5.08	149
NU-800	30	217	6.08	5.26	155
OHUKIM	48	233	6.00	4.68	137
OWITAQ	54	233	5.80	4.01	118
OWITUK	53	233	5.79	4.08	120
OWIVAS	56	237	5.81	3.93	115
PETVEQ	23	150	4.50	4.11	121
PEVQEO	37	232	6.26	5.11	150
POCPAZ	98	177	2.77	1.43	42
PUZLOM	46	215	5.64	4.59	135
QOWRAV01	27	209	6.01	5.39	158
QUSBIP	57	242	5.92	3.81	112
RAPYUE	67	219	4.94	2.84	83
RAVKEG	85	196	3.80	2.69	79
RAVXAP	31	137	3.70	3.73	109
RAVXET	22	119	3.44	3.49	102
RAYMIP	58	258	6.35	4.13	121
RAYMOV	44	237	6.19	4.76	140

REGREC	84	232	4.84	2.67	78
RUVKAV	49	230	5.89	4.59	135
SEMNIJ	21	157	4.71	4.72	139
SUKYON	83	266	5.86	3.17	93
SUNJER	138	198	2.61	1.62	48
TASFAU	47	222	5.69	3.98	117
TEQPEM	25	193	5.82	5.06	148
TOHSAL	87	246	5.09	2.70	79
UPOZAB	34	207	5.68	4.81	141
UWUQEJ	116	209	3.26	1.63	48
VAGMAT	31	210	5.88	4.97	146
VAGMEX	30	206	5.82	5.07	149
VAGMIB	51	234	5.91	4.46	131
VAGMIB01	53	237	5.99	4.43	130
VAGMOH	58	239	5.87	4.15	122
VAZTOG	31	210	5.89	4.97	146
VEBHUG	24	199	5.77	5.22	153
VEHJOJ	51	231	5.76	4.26	125
VETMIS	28	199	5.67	5.17	152
VUJBEI	53	229	5.72	4.21	124
VUSKEA	33	222	6.11	5.07	149
WAGYUA	33	210	5.82	5.00	147
XAFFAN	31	211	5.90	4.98	146
XAFFER	30	210	5.89	5.06	148
XAFFIV	30	206	5.81	4.98	146
XAFFOB	30	208	5.87	5.01	147
XAHPED	31	203	5.72	5.08	149

XAHPIH	26	189	5.44	5.07	149
XAHPON	25	174	5.07	4.92	144
XAHPUT	16	136	4.22	4.28	126
XAHQAA	15	135	4.24	4.32	127
XALMOO	46	240	6.24	4.62	136
XANLUV	39	226	6.06	4.80	141
XAWVUN	42	228	6.08	4.79	141
XAZGEK	99	217	4.00	2.73	80
XEBHOC	41	228	6.08	4.88	143
XEDGIY	91	169	2.85	1.45	43
XOVPUU	31	216	6.07	5.08	149
XOVQEF	35	221	6.02	4.96	146
XUTQEI	50	227	5.75	4.63	136
YEQRIV	13	114	3.53	3.60	106
YUGZIK	48	231	5.87	4.40	129
ZUQVIQ	42	199	5.07	4.48	132

[a] The methane uptake data are from reference 15.

Matlab routine for the ANG tank model

```
%Specify the input data.
Pm; Uptake_m; Pe; Uptake_e; Pp; Uptake_p; %Pressures (bar) and uptakes
(number of molecules/unit cell) from the single-component isotherms.
Vuc; %Volume of the unit cell of the adsorbent material (m^3).
T; %Temperature (K) of the calculated ANG system.
Acfactor1; Acfactor2; Acfactor3; Tc1; Tc2; Tc3; Pc1; Pc2; Pc3; k12; k23;
k31; %Peng-Robinson equation of state parameters (Tc: K, Pc: Pa).
Vtank=0.152; %Volume of the tank (m^3).
Voidspacefrac=0.25; %The packing loss (volume fraction).
Nc=200; %Number of operation cycles needed to be calculated.


%Define constants.
GasolineRate=0.0009; %Gasoline consumption rate (L/s).
Etank=34.2*GasolineRate*1000; %Energy flow rate from the ANG tank into the
engine (kJ/s).
Euc=Etank*(Vuc/(1-Voidspacefrac))/Vtank; %Energy flow rate scaled for one
unit cell volume of the adsorbent material (kJ/s).
Muc=(16*0.96+30*0.033+44*0.007)*(Euc/(0.96*890.35+0.033*1559.9+0.007*22
04)); %Mass flow rate scaled for one unit cell volume of the adsorbent material
(g/s).
Pf=65; %Final pressure of the tank filling process.
t0=33000;
ts=0:1:t0; %Set up the time intervals that will be calculated during the
releasing process.



%Define and initialize the properties that will be calculated in the model
during operation cycles.
%Filling part:
y1f=zeros(1,Nc); %y1f: mole fraction of methane in the gas phase;
y2f=zeros(1,Nc); %y2f: mole fraction of ethane in the gas phase;
y3f=zeros(1,Nc); %y3f: mole fraction of propane in the gas phase;
f1f=zeros(1,Nc); %f1f: fugacity coefficient of methane in the gas phase;
f2f=zeros(1,Nc); %f2f: fugacity coefficient of ethane in the gas phase;
f3f=zeros(1,Nc); %f3f: fugacity coefficient of propane in the gas phase;
x1f=zeros(1,Nc); %x1f: mole fraction of methane in the adsorbed phase;
x2f=zeros(1,Nc); %x2f: mole fraction of ethane in the adsorbed phase;
x3f=zeros(1,Nc); %x3f: mole fraction of propane in the adsorbed phase;
```

```matlab
naf=zeros(1,Nc); %naf: total number of molecules per unit cell in the adsorbed
phase;
nbf=zeros(1,Nc); %nbf: total number of molecules per unit cell in the gas
phase;
na1f=zeros(1,Nc); %na1f: number of methane molecules per unit cell in
adsorbed phase;
na2f=zeros(1,Nc); %na2f: number of ethane molecules per unit cell in adsorbed
phase;
na3f=zeros(1,Nc); %na3f: number of propane molecules per unit cell in
adsorbed phase;
n1f=zeros(1,Nc); %n1f: number of all methane molecules inside the tank;
n2f=zeros(1,Nc); %n2f: number of all ethane molecules inside the tank;
n3f=zeros(1,Nc); %n3f: number of all propane molecules inside the tank;

%Releasing part:
tr=zeros(t0+1,Nc); %tr: time (s) during the releasing process;
Pr=zeros(t0+1,Nc); %Pr: pressure (bar) inside the tank;
y1r=zeros(t0+1,Nc); %y1r: mole fraction of methane in the gas phase;
y2r=zeros(t0+1,Nc); %y2r: mole fraction of ethane in the gas phase;
y3r=zeros(t0+1,Nc); %y3r: mole fraction of propane in the gas phase;
f1r=zeros(t0+1,Nc); %f1r: fugacity coefficient of methane in the gas phase;
f2r=zeros(t0+1,Nc); %f2r: fugacity coefficient of ethane in the gas phase;
f3r=zeros(t0+1,Nc); %f3r: fugacity coefficient of propane in the gas phase;
x1r=zeros(t0+1,Nc); %x1r: mole fraction of methane in the adsorbed phase;
x2r=zeros(t0+1,Nc); %x2r: mole fraction of ethane in the adsorbed phase;
x3r=zeros(t0+1,Nc); %x3r: mole fraction of propane in the adsorbed phase;
nar=zeros(t0+1,Nc); %nar: total number of molecules per unit cell in the
adsorbed phase;
nbr=zeros(t0+1,Nc); %nbr: total number of molecules per unit cell in the
gas phase;
nfr=zeros(t0+1,Nc); %nfr: molar flow rate (molecules/uc*s) of all three
components;
na1r=zeros(t0+1,Nc); %na1r: number of methane molecules per unit cell in
adsorbed phase;
na2r=zeros(t0+1,Nc); %na2r: number of ethane molecules per unit cell in
adsorbed phase;
na3r=zeros(t0+1,Nc); %na3r: number of propane molecules per unit cell in
adsorbed phase;
n1r=zeros(t0+1,Nc); %n1r: number of all methane molecules inside the tank;
n2r=zeros(t0+1,Nc); %n2r: number of all ethane molecules inside the tank;
n3r=zeros(t0+1,Nc); %n3r: number of all propane molecules inside the tank;
```

```matlab
%Peng-Robinson equation of state for pure component:
function y=Compressionfactor_pure(x,A,B)
y=x^3-(1-B)*x^2+(A-3*B^2-2*B)*x-(A*B-B^2-B^3);
end

function y=PREOS_pure(Acfactor,Tc,Pc,T,P)
Tr=T/Tc;
Pr=P/Pc;
alphaT=(1+(0.37646+Acfactor*1.54226-0.26992*(Acfactor^2))*(1-Tr^0.5))^2
;
A=0.457235*alphaT*Pr/(Tr^2);
B=0.077796*Pr/Tr;
opt=optimset('TolFun',1e-20);
Z=fzero(@Compressionfactor_pure,opt,A,B);
y=exp((Z-1)-log(Z-B)-A/(2*2^0.5*B)*log((Z+(1+2^0.5)*B)/(Z+(1-2^0.5)*B))
);
end

%Peng-Robinson equation of state for mixture:
function y=Compressionfactor_mix(x,alpha,beta,gamma)
y=x^3+alpha*x^2+beta*x+gamma;
end

function
y=PREOS_mix(Acfactor1,Acfactor2,Acfactor3,Tc1,Tc2,Tc3,Pc1,Pc2,Pc3,k12,k
23,k31,P,y1,y2)
y3=1-y1-y2;
Accfactor=[Acfactor1,Acfactor2,Acfactor3];
Tc=[Tc1,Tc2,Tc3];
Pc=[Pc1,Pc2,Pc3];
k=[0,k12,k31;k12,0,k23;k31,k23,0];
Tr=298./Tc;
Pr=1e5*P./Pc;
alphaT=(1+(0.37646+Accfactor.*1.54226-0.26992.*(Accfactor.^2)).*(1-Tr.^
0.5)).^2;
A=0.457235.*alphaT.*Pr./(Tr.^2);
B=0.077796.*Pr./Tr;
Amix=y1^2*A(1)+y2^2*A(2)+y3^2*A(3)+2*y1*y2*((A(1)*A(2))^0.5)*(1-k12)+2*
y2*y3*((A(2)*A(3))^0.5)*(1-k23)+2*y3*y1*((A(3)*A(1))^0.5)*(1-k31);
Bmix=y1*B(1)+y2*B(2)+y3*B(3);
```

```matlab
alpha=-1+Bmix;
beta=Amix-3*Bmix^2-2*Bmix;
gamma=-Amix*Bmix+Bmix^2+Bmix^3;
opt=optimset('TolFun',1e-20);
Zmix=fzero(@Compressionfactor_mix,opt,alpha,beta,gamma);
f=[1,1,1];
for i=1:3
    f(i)=exp(B(i)./Bmix*(Zmix-1)-log(Zmix-Bmix)-Amix/(2*sqrt(2)*Bmix)*((
    2*(y1*(A(i).*A(1))^0.5*(1-k(1,i))+y2*(A(i).*A(2))^0.5*(1-k(2,i))+y3*
    (A(i).*A(3))^0.5*(1-k(3,i))))/Amix-B(i)./Bmix)*log((Zmix+(1+sqrt(2))
    *Bmix)/(Zmix+(1-sqrt(2))*Bmix)));
end
v=Zmix*8.314*298/(P*1e5);
y=[f(1),f(2),f(3),v];
end


%Convert the pressures in the single-component isotherms into fugacities:
fm=Pm; fe=Pe; fp=Pp;
for i=1:length(Pm)
    fm(i)=PREOS_pure(Acfactor1,Tc1,Pc1,T,Pm(i)*1e5)*Pm(i);
end
for i=1:length(Pe)
    fe(i)=PREOS_pure(Acfactor2,Tc2,Pc2,T,Pe(i)*1e5)*Pe(i);
end
for i=1:length(Pp)
    fp(i)=PREOS_pure(Acfactor3,Tc3,Pc3,T,Pp(i)*1e5)*Pp(i);
end


%Fit the isotherm data into Langmuir isotherm equations:
function y=Langmuir(b,x)
y=b(1)*b(2)*x./(1+b(2)*x);
end


b=nlinfit(fm,Uptake_m,'Langmuir',b01);
N1=b(1); b1=b(2);
b=nlinfit(fe,Uptake_e,'Langmuir',b02);
N2=b(1); b2=b(2);
b=nlinfit(fp,Uptake_p,'Langmuir',b03);
N3=b(1); b3=b(2);
%b01, b02, b03 are the initial guesses of the fitting parameters.
```

```matlab
%Main part of the ANG tank model.
%Initialization before solving the equation set:
x0(1,:)=[0.3,0.3,0.8,0.8,0.8,0.3,0.3,300,30]; %Initial guesses of the
roots of the equation set for the filling process in the 1st cycle:
M=zeros(14,14); %Mass matrix for solving the DAEs in tank releasing process.
M(9,9)=1;
M(10,10)=1;
M(11,11)=1;
opt_filling=optimset('MaxFunEvals',50000,'MaxIter',5000);
opt_releasing=odeset('mass',M);
x=zeros(1,9);
n58=zeros(1,Nc+1); %A vector whose components are the number of time
intervals at which the pressure inside the tank reaches 5.8 bar during
releasing process in each cycle.
n58(1)=1;
n1r(1,1)=0;
n2r(1,1)=0;
n3r(1,1)=0;
y0=zeros(Nc,14);

%Equation set for the filling process.
function
y=Tank_filling(x,N1,b1,N2,b2,N3,b3,Acfactor1,Acfactor2,Acfactor3,Tc1,Tc
2,Tc3,Pc1,Pc2,Pc3,k12,k23,k31,Vuc,Voidspacefrac,P,n1r,n2r,n3r,m,n)
%x(1)=y1f x(2)=y2f x(3)=f1f x(4)=f2f x(5)=f3f x(6)=x1f x(7)=x2f x(8)=naf
x(9)=nbf
f=PREOS_mix(Acfactor1,Acfactor2,Acfactor3,Tc1,Tc2,Tc3,Pc1,Pc2,Pc3,k12,k
23,k31,P,x(1),x(2));
y(1)=x(3)-f(1);
y(2)=x(4)-f(2);
y(3)=x(5)-f(3);
y(4)=N1*log(1+b1*(P*x(1)*x(3)/x(6)))-N2*log(1+b2*(P*x(2)*x(4)/x(7)));
y(5)=N1*log(1+b1*(P*x(1)*x(3)/x(6)))-N3*log(1+b3*(P*(1-x(1)-x(2))*x(5)/
(1-x(6)-x(7))));
y(6)=x(8)-1/(x(6)/(N1*b1*(P*x(1)*x(3)/x(6))/(1+b1*(P*x(1)*x(3)/x(6))))+
x(7)/(N2*b2*(P*x(2)*x(4)/x(7))/(1+b2*(P*x(2)*x(4)/x(7))))+(1-x(6)-x(7))
/(N3*b3*(P*(1-x(1)-x(2))*x(5)/(1-x(6)-x(7)))/(1+b3*(P*(1-x(1)-x(2))*x(5
)/(1-x(6)-x(7))))));
y(7)=x(9)-(Vuc*Voidspacefrac/(1-Voidspacefrac))/f(4)*6.02e23;
y(8)=(x(6)*x(8)+x(1)*x(9)-n1r(m,n))/(x(7)*x(8)+x(2)*x(9)-n2r(m,n))-0.96
/0.033;
```

```matlab
y(9)=(x(6)*x(8)+x(1)*x(9)-n1r(m,n))/((1-x(6)-x(7))*x(8)+(1-x(1)-x(2))*x(9)-n3r(m,n))-0.96/0.007;
end


%Equation set for the releasing process.
function
dx=Tank_releasing(t,x,N1,b1,N2,b2,N3,b3,Acfactor1,Acfactor2,Acfactor3,Tc1,Tc2,Tc3,Pc1,Pc2,Pc3,k12,k23,k31,Vuc,Voidspacefrac,Muc)
%x(1)=y1,x(2)=y2,x(3)=x1,x(4)=x2,x(5)=na,x(6)=nb,x(7)=nf,x(8)=P,x(9)=n1,x(10)=n2,x(11)=n3,x(12)=f1,x(13)=f2,x(14)=f3
dx=zeros(14,1);
f=PREOS_mix(Acfactor1,Acfactor2,Acfactor3,Tc1,Tc2,Tc3,Pc1,Pc2,Pc3,k12,k23,k31,x(8),x(1),x(2));
dx(1)=x(9)-x(3)*x(5)-x(1)*x(6);
dx(2)=x(10)-x(4)*x(5)-x(2)*x(6);
dx(3)=x(11)-(1-x(3)-x(4))*x(5)-(1-x(1)-x(2))*x(6);
dx(4)=(16*x(1)+30*x(2)+44*(1-x(1)-x(2)))*x(7)-Muc*6.02e23;
dx(5)=N1*log(1+b1*(x(8)*x(1)*x(12)/x(3)))-N2*log(1+b2*(x(8)*x(2)*x(13)/x(4)));
dx(6)=N1*log(1+b1*(x(8)*x(1)*x(12)/x(3)))-N3*log(1+b3*(x(8)*(1-x(1)-x(2))*x(14)/(1-x(3)-x(4))));
dx(7)=x(5)-1/(x(3)/(N1*b1*(x(8)*x(1)*x(12)/x(3))/(1+b1*(x(8)*x(1)*x(12)/x(3))))+x(4)/(N2*b2*(x(8)*x(2)*x(13)/x(4))/(1+b2*(x(8)*x(2)*x(13)/x(4))))+(1-x(3)-x(4))/(N3*b3*(x(8)*(1-x(1)-x(2))*x(14)/(1-x(3)-x(4)))/(1+b3*(x(8)*(1-x(1)-x(2))*x(14)/(1-x(3)-x(4))))));
dx(8)=x(6)-(Vuc*Voidspacefrac/(1-Voidspacefrac))/f(4)*6.02e23;
dx(9)=-x(1)*x(7);
dx(10)=-x(2)*x(7);
dx(11)=-(1-x(1)-x(2))*x(7);
dx(12)=x(12)-f(1);
dx(13)=x(13)-f(2);
dx(14)=x(14)-f(3);
end


%Loop over operation cycles:
for i=2:Nc+1
    x=fsolve(@Tank_filling,x0(i-1,:),opt_filling,N1,b1,N2,b2,N3,b3,Acfactor1,Acfactor2,Acfactor3,Tc1,Tc2,Tc3,Pc1,Pc2,Pc3,k12,k23,k31,Vuc,Voidspacefrac,Pf,n1r,n2r,n3r,n58(i-1),i-1);
    %Save the result at 65 bar of the previous cycle as the initial guesses
    of the roots for filling process in the next cycle.
```

```
x0(i,:)=x;
%Save the output data from the filling process.
y1f(:,i-1)=x(:,1);
y2f(:,i-1)=x(:,2);
y3f(:,i-1)=1-x(:,1)-x(:,2);
f1f(:,i-1)=x(:,3);
f2f(:,i-1)=x(:,4);
f3f(:,i-1)=x(:,5);
x1f(:,i-1)=x(:,6);
x2f(:,i-1)=x(:,7);
x3f(:,i-1)=1-x(:,6)-x(:,7);
naf(:,i-1)=x(:,8);
nbf(:,i-1)=x(:,9);
na1f(:,i-1)=x1f(:,i-1).*naf(:,i-1);
na2f(:,i-1)=x2f(:,i-1).*naf(:,i-1);
na3f(:,i-1)=x3f(:,i-1).*naf(:,i-1);
n1f(:,i-1)=na1f(:,i-1)+y1f(:,i-1).*nbf(:,i-1);
n2f(:,i-1)=na2f(:,i-1)+y2f(:,i-1).*nbf(:,i-1);
n3f(:,i-1)=na3f(:,i-1)+y3f(:,i-1).*nbf(:,i-1);


%Take the result from the filling process at 65bar as the initial
conditions for the releasing part.
y0(i-1,:)=[y1f(1,i-1),y2f(1,i-1),x1f(1,i-1),x2f(1,i-1),naf(1,i-1),nb
f(1,i-1),Muc*6.02e23/(16*y1f(1,i-1)+30*y2f(1,i-1)+44*y3f(1,i-1)),Pf,
n1f(1,i-1),n2f(1,i-1),n3f(1,i-1),f1f(1,i-1),f2f(1,i-1),f3f(1,i-1)];
[t,y]=ode15s(@Tank_releasing,ts,y0(i-1,:),opt_releasing,N1,b1,N2,b2,
N3,b3,Acfactor1,Acfactor2,Acfactor3,Tc1,Tc2,Tc3,Pc1,Pc2,Pc3,k12,k23,
k31,Vuc,Voidspacefrac,Muc);
tr(1:t0+1,i-1)=t;
Pr(1:t0+1,i-1)=y(:,8);
y1r(1:t0+1,i-1)=y(:,1);
y2r(1:t0+1,i-1)=y(:,2);
y3r(1:t0+1,i-1)=1-y(:,1)-y(:,2);
f1r(1:t0+1,i-1)=y(:,12);
f2r(1:t0+1,i-1)=y(:,13);
f3r(1:t0+1,i-1)=y(:,14);
x1r(1:t0+1,i-1)=y(:,3);
x2r(1:t0+1,i-1)=y(:,4);
x3r(1:t0+1,i-1)=1-y(:,3)-y(:,4);
nar(1:t0+1,i-1)=y(:,5);
nbr(1:t0+1,i-1)=y(:,6);
```

```matlab
    nfr(1:t0+1,i-1)=y(:,7);
    na1r(:,i-1)=x1r(:,i-1).*nar(:,i-1);
    na2r(:,i-1)=x2r(:,i-1).*nar(:,i-1);
    na3r(:,i-1)=x3r(:,i-1).*nar(:,i-1);
    n1r(1:t0+1,i)=y(:,9);
    n2r(1:t0+1,i)=y(:,10);
    n3r(1:t0+1,i)=y(:,11);
    %Determine at which time point in the releasing process the pressure
    inside the tank reaches 5.8 bar.
    for k=1:t0
        if (Pr(k,i-1)-2)*(Pr(k+1,i-1)-2)<=0
            n58(i)=k;
        end
    end
    t0=t0;
    ts=0:1:t0;
end

%Analysis of the final result.
tr58=zeros(1,Nc);
y1r58=zeros(1,Nc);
y2r58=zeros(1,Nc);
y3r58=zeros(1,Nc);
x1r58=zeros(1,Nc);
x2r58=zeros(1,Nc);
x3r58=zeros(1,Nc);
nar58=zeros(1,Nc);
nbr58=zeros(1,Nc);
na1r58=zeros(1,Nc);
na2r58=zeros(1,Nc);
na3r58=zeros(1,Nc);
n1r58=zeros(1,Nc);
n2r58=zeros(1,Nc);
n3r58=zeros(1,Nc);
for i=1:Nc
    tr58(i)=n58(i+1)./3600;
    y1r58(i)=y1r(n58(i+1),i);
    y2r58(i)=y2r(n58(i+1),i);
    y3r58(i)=y3r(n58(i+1),i);
    x1r58(i)=x1r(n58(i+1),i);
    x2r58(i)=x2r(n58(i+1),i);
```

```matlab
    x3r58(i)=x3r(n58(i+1),i);
    nar58(i)=nar(n58(i+1),i);
    nbr58(i)=nbr(n58(i+1),i);
    na1r58(i)=na1r(n58(i+1),i);
    na2r58(i)=na2r(n58(i+1),i);
    na3r58(i)=na3r(n58(i+1),i);
    n1r58(i)=n1r(n58(i+1),i+1); %Number of methane molecules in the whole
tank at the end of each releasing process (at 5.8 bar).
    n2r58(i)=n2r(n58(i+1),i+1); %Number of ethane molecules in the whole tank
at the end of each releasing process (at 5.8 bar).
    n3r58(i)=n3r(n58(i+1),i+1); %Number of propane molecules in the whole
tank at the end of each releasing process (at 5.8 bar).
end
for i=1:Nc
    Delta_E(i)=((n1f(i)-n1r58(i))*890.35+(n2f(i)-n2r58(i))*1559.9+(n3f(i
    )-n3r58(i))*2204.0)/6.02e23/Vuc/1e6*(1-Voidspacefrac); %Calculate the
    deliverable energy (MJ/L of tank) of each cycle.
end
MPT=21.4.*(tr58); %Calculate the miles one can drive with one tank of ANG
of each cycle.
```

References

1 D. Dubbeldam, S. Calero, D. E. Ellis and R. Q. Snurr, *Molecular Simulation*, 2015, DOI: 10.1080/08927022.2015.1010082.

2 A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard III and W. M. Skiff, *J. Am. Chem. Soc.*, 1992, **114**, 10024-10035.

3 M. G. Martin and J. I. Siepmann, *J. Phys. Chem. B*, 1998, **102**, 2569-2577.

4 D. Nicholson and N. G. Parsonage, *Computer simulation and the statistical mechanics of adsorption*. London: Academic Press, 1982.

5 O. Talu and A. L. Myers, *A. I. Ch. E. Journal*, 2001, **47**, 1160-1168.

6 L. D. Gelb and K. E. Gubbins, *Langmuir*, 1999, **15**, 305-308.

7 T. F. Willems, C. H. Rycroft, M. Kazi, J. C. Meza and M. Haranczyk, *Microporous and Mesoporous Materials*, 2012, **149**, 134-141.

8 N. C. Jeong, B. Samanta, C. Y. Lee, O. K. Farha and J. T. Hupp, *J. Am. Chem. Soc.*, 2012, **134**, 51-54.

9 J. E. Mondloch, O. Karagiaridi, O. K. Farha and J. T. Hupp, *CrystEngComm*, 2013, **15**, 9258-9264.

10 Y. Peng, V. Krungleviciute, I. Eryazici, J. T. Hupp, O. K. Farha and T. Yildirim, *J. Am. Chem. Soc.*, 2013, **135**, 11887-11894.

11 D. Dubbeldam, R. Krishna, S. Calero and A. O. Yazaydin, *Angew. Chem., Int. Ed.*, 2012, **51**, 11867-11871.

12 Z. R. Herm, B. M. Wiers, J. A. Mason, J. M. van Baten, M. R. Hudson, P. Zajdel, C. M. Brown, N. Masciocchi, R. Krishna and J. R. Long, *Science*, 2013, **340**, 960-964.

13 R. B. Rios, M. Bastos-Neto, M. R. Amora Jr., A. E. B. Torres, D. C. S. Azevedo and C. L. Cavalcante Jr., *Fuel*, 2011, **90**, 113-119.

14 B. Borah, H. Zhang and R. Q. Snurr, *Chem. Eng. Sci.*, 2015, **124**, 135-143.

15 Y. G. Chung, J. Camp, M. Haranczyk, B. J. Sikora, W. Bury, V. Krungleviciute, T. Yildirim, O. K. Farha, D. S. Sholl and R. Q. Snurr, *Chem. Mater.*, 2014, **26**, 6185-6192.