SUPPORTING INFORMATION FOR:

Cao, Z., L. Shen, S. Zhong, L.T. Liu, H.X. Kong, and Y.Z. Sun. 2017. A probabilistic dynamic MFA model for Chinese urban housing stock. *Journal of Industrial Ecology.*

Summary

This supporting information contains detailed information on models and data sources. Results for parameters of material intensity's PDF are presented in Section S7. Results for material flows of various material types are presented in Section S10. Spearman's rank correlation coefficients between the total population and the urban housing stock are presented in Section S12.

Content

Section S1. Projection for housing floor area per capita

The parameters for logistic and Gompertz combined function are 0.423 and 0.019. To obtain the stochastic errors of the logistic growth pattern, the forecast package is employed here to select the best ARIMA model. The Akaike information criterion with a correction for finite sample sizes (AICc) is the default information criterion to be used in model selection (Hyndman, 2016). According to this criterion, a zero mean ARIMA (1, 0, 0) model with the minimum AICc value (AICc = 38.11) is adopted here. The coefficient of AR is 0.690.

Combining the fitted floor area per capita and the stochastic errors of the logistic growth pattern, 5,000 time series of the floor area per capita are obtained. The R codes are listed as followings:

```
#'area' is the historical floor area per capita and 'year' is the corresponding time series
nls1<-nls(area~50/(1+(50/2-1)*exp(a*(1-exp(b*(year-1900))))),start=list(a=0.2,b=0.026),trace=TRUE) #define the logistic and Gompertz combined function for per capita floor area
A<-summary(nls1)$parameter[1,1] #get the value for a
B<-summary(nls1)$parameter[2,1] #get the value for b
year_sim<-1900:2100 #define the simulation duration
fitted_area<-50/(1+(50/2-1)*exp(A*(1-exp(B*(year_sim-1900))))) #compute the fitted floor area per capita without stochastic errors
library(forecast) #load the forecast package
rate <- ts(summary(nls1)$residuals, start = 1985, frequency = 1) #get the residuals of the Nonlinear Least Squares and set the residuals as a time series
arima1<-auto.arima(rate,trace=T) #run the auto.arima command to get the parameters of ARIMA
r.sims <- matrix(nrow=88, ncol=5000) #define a matrix to store the 5,000 simulations of time series from 2013-2100
M<-5000 #simulation times
for(i in 1:M)
{
  set.seed(i)
  ts <- simulate(arima1, nsim =88) #simulate the residuals of the floor area per capita
  r.sims[,i]<-as.numeric(ts+ fitted_area[114:201]) #get the floor area per capita
}
#get the median and 95% CIs of the floor area per capita
median<-apply(r.sims,1,quantile,probs=c(0.025,0.5,0.975))[2,]
quantile2.5<-apply(r.sims,1,quantile,probs=c(0.025,0.5,0.975))[1,]
quantile97.5<-apply(r.sims,1,quantile,probs=c(0.025,0.5,0.975))[3,]
floor_u<-rbind(r.sims[1,],r.sims[2,],r.sims[3,],r.sims[8,],r.sims[13,],r.sims[18,],r.sims[23,],r.sims[28,],r.sims[33,],r.sims[38,],r.sims[43,],r.sims[48,],r.sims[53,],r.sims[58,],r.sims[63,],r.sims[68,],r.sims[73,],r.sims[78,],r.sims[83,],r.sims[88,]) #get the 5,000 simulations of floor area per capita in urban region in 2013, 2014, 2015, 2020, 2025, 2030, 2035, …, 2095, 2100
```

Section S2. Projection for building lifetime

The nlstools package is employed here to perform bootstrapping of the building lifetime. This package provides a non-parameter bootstrap command. The R codes are listed as followings:

```
#'new' is the newly completed floor area every year; 'demolish' is the demolished floor area at every year; 'time' is time span that each newly completed floor area has survived for from the year being built to one certain year; 'a' and 'b' are respectively scale and shape parameter of the Weibull distribution function
library(nlstools) #load the nlstools package
data_weibull<-data.frame(x=time,y=demolish) #define the empirical data for regression
#define the formula for the Nonlinear Least Squares regression; some lines have been omitted for brevity
formulaExp<-as.formula(
  demolish~
      (exp(-((time-1)/a)^b)-exp(-((time)/a)^b))*new[1]+
    (exp(-((time-2)/a)^b)-exp(-((time-1)/a)^b))*new[2]+
    (exp(-((time-3)/a)^b)-exp(-((time-2)/a)^b))*new[3]+
    ......+
      (exp(-((time-31)/a)^b)-exp(-((time-30)/a)^b))*new[31]
    (exp(-((time-32)/a)^b)-exp(-((time-31)/a)^b))*new[32]
)
nls1<- nls(formulaExp,start=list(a=30,b=2),data=data_weibull) #get the scale and shape parameter by the Nonlinear Least Squares regression
set.seed(1); nls_boot<-nlsBoot(nls1,niter=5000) #perform the bootstrapping and get the 5,000 simulations for building lifetime
```

Section S3. Projection for population

The probabilistic population projection method is closely related to the time series approach (Raftery et al., 2012). In this method, the future population trajectories are produced by trajectories of age-specific fertility rates and age-specific mortality rates. The age-specific fertility rates are generated by a total fertility rate projection model (Alkema et al., 2011). The age-specific mortality rates are generated by a life expectancy projection model (Raftery et al., 2013). These two models are developed based on several consensuses and empirical regularities.

The evolution of fertility includes three broad phases: first starts at a high value, and declines to a level below the replacement level, and then increases gradually again, but eventually fluctuates around the replacement level. The transition of fertility is represented by a random walk model with drift. The drift term is defined as the sum of two logistic functions. The details of the two logistic functions can be found in the related work (Alkema et al., 2011).

The evolution of life expectancy is also captured by a double-logistic function. The expected life expectancy gain is a double-logistic function of the current level of life expectancy (Raftery et al., 2013).

To estimate the parameters in the two double-logistic functions and to incorporate prior information, the Markov chain Monte Carlo (MCMC) method is adopted. The MCMC method could estimate the posterior distribution of the parameters in double-logistic functions.

The population projection is implemented by three R packages bayesPop (Ševčíková and

Raftery, 2012), bayesTFR (Ševčíková et al., 2011) and bayesLife (Ševčíková and Raftery, 2011). The R codes are listed as followings: (Note: the '#' is the annotation symbol)

(1) Life Expectancy

```
# Projection for Life Expectancy
library(bayesLife) #load the bayesLife package
sim.dir <- file.path(getwd(), 'e0') #set the storage path for the MCMC prediction parameters for Life Expectancy
m <- run.e0.mcmc(sex='F', nr.chains=5, thin=1, iter=7000, output.dir=sim.dir) # run the MCMC simulation for Life Expectancy parameters
diag <- e0.diagnose(sim.dir, thin=5, burnin = 2000, express = FALSE, country.sampling.prop = NULL, keep.thin.mcmc=FALSE, verbose = TRUE) #run convergence diagnostics of existing Life Expectancy MCMCs
has.mcmc.converged(diag) #check if the existing diagnostics converged
pred <- e0.predict(m, burnin=2000, nr.traj=5000, verbose=TRUE) # predict the Life Expectancy
e0.trajectories.plot(pred, country="China", both.sexes=TRUE, pi=c(95,80), main="Life Expectancy of China (Female and Male)") #plot the Life Expectancy trajectories of China
```

(2) Total Fertility Rate

```
#Projection for Total Fertility Rate (TFR)
library(bayesTFR) #load the bayesTFR package
simulation.dir <- file.path(getwd(), 'mylongrun') #set the storage path for the MCMC prediction parameters for TFR
m1 <- run.tfr.mcmc(nr.chains=5, iter=7000, output.dir=simulation.dir) #run the MCMC simulation for TFR parameters
diag1 <- tfr.diagnose(simulation.dir, thin=1, burnin=2000) #run convergence diagnostics of existing TFR MCMCs
has.mcmc.converged(diag1) #check if the existing diagnostics converged
pred1 <- tfr.predict(sim.dir=simulation.dir, end.year=2100, burnin=2000, nr.traj=5000, verbose=TRUE) #predict the Total Fertility Rate
tfr.trajectories.plot(pred2, half.child.variant=FALSE, country='China', pi=c(95, 80), typical.trajectory = TRUE, nr.traj=5000, main="Total Fertility Rate of China (%)") #plot the Total Fertility Rate trajectories of China
```

(3) Population

```
#Projection for Chinese population
library(bayesPop) #load the bayesPop package
pop.dir <- file.path(getwd(), 'pop') #set the storage path for the population prediction
pop.pred<-pop.predict(countries=c("China"), output.dir=pop.dir, nr.traj=10000, inputs=list(tfr.sim.dir= simulation.dir, e0F.sim.dir=sim.dir, e0M.sim.dir="joint_")) #predict the population of China
pop.trajectories.plot(pop.pred, country="China", pi=c(95,80), typical.trajectory = TRUE, sum.over.ages=TRUE, main="Population Projection of China") #plot the Population trajectories of China
population<-pop.trajectories(pop.pred,country="China") #get the 5,000 simulations of Chinese population
population<-rbind(rep(1362514.26),rep(1369435.67),population) #add population in 2013 and 2014 to population
```

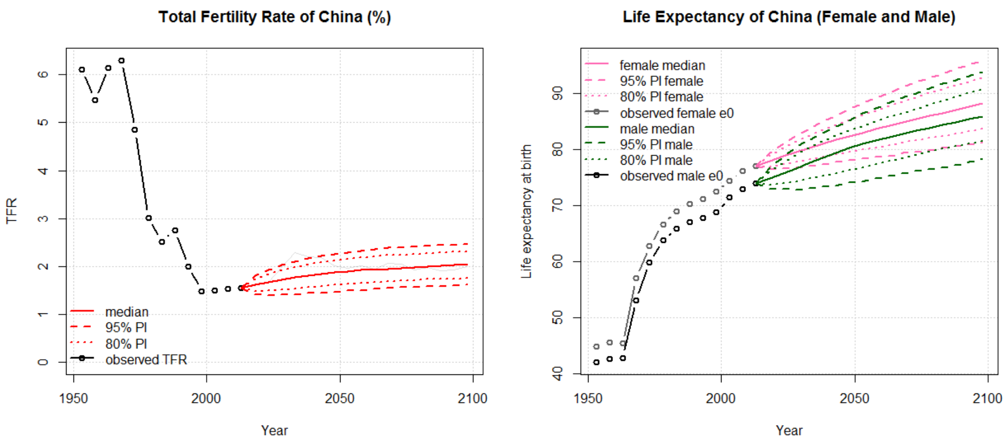


Figure S1 Prediction for the total fertility rate (left) and life expectancy (right) of China during 2015-2100.

Section S4. Projection for urbanization rate

The shape parameters of S-curve are 2.465 and 0.056. To obtain the stochastic errors of the S-curve, the forecast package is employed here to select the best ARIMA model. The Akaike information criterion with a correction for finite sample sizes (AICc) is the default information criterion to be used in model selection. According to this criterion, a zero mean ARIMA (1, 0, 3) model with the minimum AICc value (AICc = -316.65) is adopted here. The coefficients of AR, MA1, MA2, MA3 are 0.9188, 0.5627, 0.4075, 0.4226, respectively.

Combining the fitted urbanization rate and the stochastic errors of the S-curve, 5,000 time series of the urbanization rate are obtained. The R codes are listed as followings:

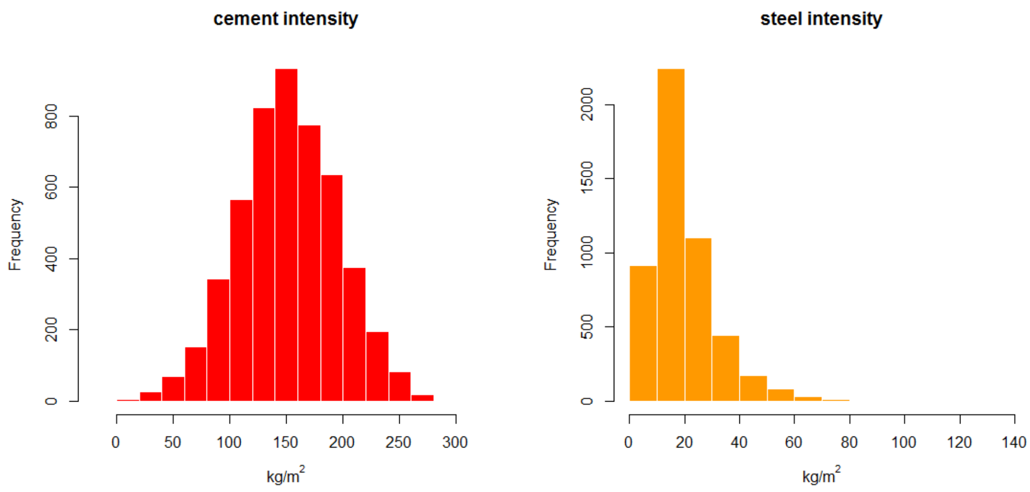
```
#' urban_rate' is the historical urbanization rate and 'year' is the time series
nls1 <- nls(urban_rate ~ 0.8/(1+a*exp(-b*(year-1986))), start = list(a=2,b=0.005), trace = TRUE) #define the S-curve for the urbanization rate
A<-summary(nls1)$parameter[1,1] #get the value for a
B<-summary(nls1)$parameter[2,1] #get the value for b
year_sim<-1900:2100 #define the simulation duration
fitted_ur<-0.8/(1+A*exp(-B*(year_sim-1986))) #compute the fitted urbanization rate without stochastic errors
rate <- ts(summary(nls1)$residuals, start = 1986, frequency = 1) #get the residuals of the Nonlinear Least Squares (nls) and set the residuals as a time series
library(forecast) #load the forecast package
arima1<-auto.arima(rate,trace=T) #run the auto.arima command to get the parameters of ARIMA
r.sims <- matrix(nrow=86, ncol=5000) #define a matrix to store the 5,000 simulations of time series from 2015-2100
M<-5000 #simulation times
for(i in 1:M)
{
  set.seed(i)
  ts <- simulate(arima1, nsim =86) #simulate the residuals of the urbanization rate
  r.sims[,i]<-as.numeric(ts+ fitted_ur [116:201]) #get the urbanization rate
}
#get the median and 95% CIs of the urbanization rate
```

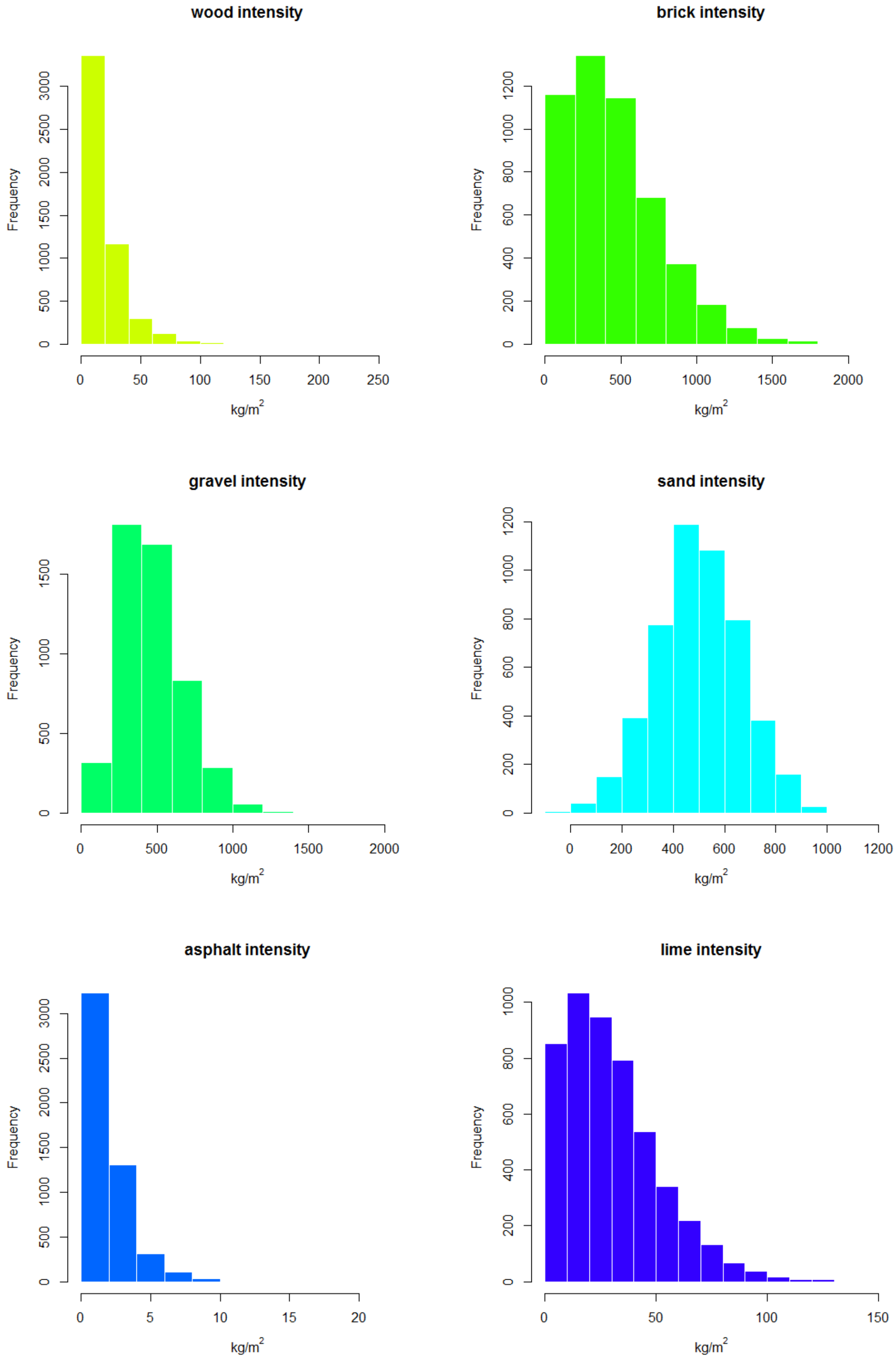
```
median<-apply(r.sims,1,quantile,probs=c(0.025,0.5,0.975))[2,]
quantile2.5<-apply(r.sims,1,quantile,probs=c(0.025,0.5,0.975))[1,]
quantile97.5<-apply(r.sims,1,quantile,probs=c(0.025,0.5,0.975))[3,]
urbanization<-rbind(r.sims[1,],r.sims[6,],r.sims[11,],r.sims[16,],r.sims[21,],
r.sims[26,],r.sims[31,],r.sims[36,],r.sims[41,],r.sims[46,],r.sims[51,],r.sims[56,],
r.sims[61,],r.sims[66,],r.sims[71,],r.sims[76,],r.sims[81,],r.sims[86,]) #get the 5,000 simulations in 2015,2020, 2025, 2030, 2035,.., 2095, 2100
urbanization<-rbind(rep(area_bj[36]),rep(area_bj[37]),urbanization) #add urbanization rate in 2013 and 2014 to 2015-2100
```

Section S5. Uncertainties of material intensity

The fitdistrplus package is employed here to fit the sample data with possible distribution functions and to pick out the best distribution for material intensity. After that, the bootstrap method is employed to obtain to parameters of the distribution function. The R codes are listed as followings:

```
library("fitdistrplus") #load the fitdistrplus package
#'data' contains 146 samples
fw <- fitdist(data, "weibull") #fit the Weibull distribution
fg <- fitdist(data, "gamma") #estimate the distribution parameters of Gamma distribution
fln <- fitdist(data, "lnorm") #estimate the distribution parameters of Lognormal distribution
fn <-fitdist(data,"norm") #estimate the distribution parameters of Normal distribution
gofstat(list(fw,fg,fln,fn)) #compare the distribution functions
#select the best distribution function and estimate parameters for the distribution function
set.seed(1); fw.B <- bootdist(fw, niter = 1000)
set.seed(1); fln.B <- bootdist(fln, niter = 1000)
set.seed(1); fn.B <- bootdist(fn, niter = 1000)
set.seed(1); fg.B<-bootdist(fg, niter = 1000)
#get 5,000 simulations for the material intensity
rweibull(5000,shape=summary(fw.B)$CI[1,1],scale=summary(fw.B)$CI[2,1])
rlnorm(5000,meanlog=summary(fln.B)$CI[1,1],sdlog=summary(fln.B)$CI[2,1])
rnorm(5000,mean= summary(fn.B)$CI[1,1],sd= summary(fn.B)$CI[2,1])
rgamma(5000,shape=summary(fg.B)$CI[1,1],rate=summary(fg.B)$CI[2,1])
```




wood intensity
Frequency
0
500
1000
1500
2000
2500
3000
0
50
100
150
200
250
kg/m²
brick intensity
Frequency
0
200
400
600
800
1000
1200
0
500
1000
1500
2000
kg/m²
gravel intensity
Frequency
0
500
1000
1500
0
500
1000
1500
2000
kg/m²
sand intensity
Frequency
0
200
400
600
800
1000
1200
0
200
400
600
800
1000
1200
kg/m²
asphalt intensity
Frequency
0
500
1000
1500
2000
2500
3000
0
5
10
15
20
kg/m²
lime intensity
Frequency
0
200
400
600
800
1000
0
50
100
150
kg/m²

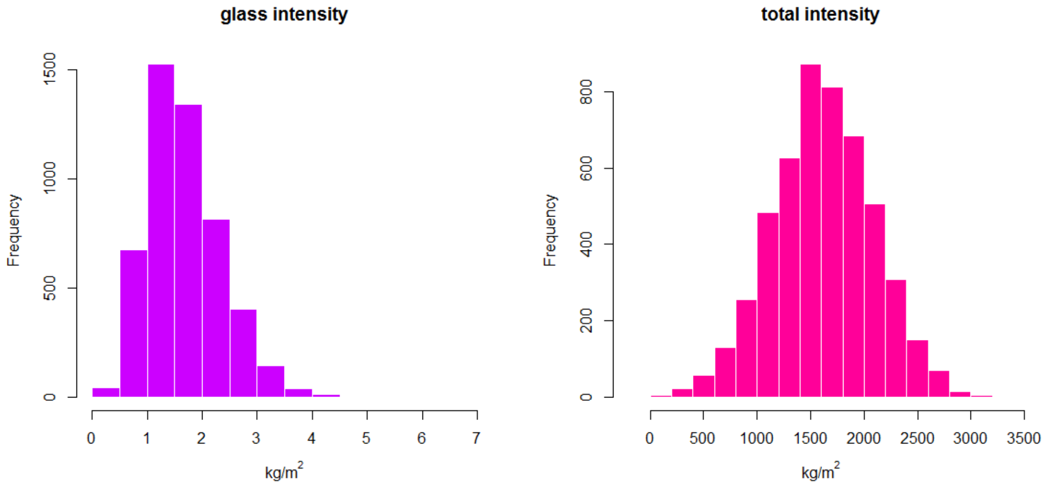


Figure S2 Histograms of material intensity simulations.

Section S6. Selection of good distribution for material intensity

The goodness-of-fit statistics and criteria of material intensity for different materials are displayed in Table.A1-10.

The selection of good candidate among the four distributions are mainly based on Aikake's Information Criterion (AIC), Bayesian Information Criterion (BIC), and Anderson-Darling statistic (Delignette-Mullerand Dutang, 2014). The two classical penalized criteria, i.e., AIC and BIC, are indicators in priority. When these two indicators are extremely close, the Anderson-Darling statistic should be taken into consideration. The remaining two statistics are auxiliary indicators.

The AIC and BIC reward goodness of fit (as assessed by the likelihood function) and penalize increasing number of estimated parameters. To evaluate the goodness-of-fit and to discourage overfitting, the goodness-of-fit criteria (AIC and BIC) should be considered as the indicators in priority.

The goodness-of-fit statistics aims to measure the distance between the fitted parametric distribution and the empirical distribution. The Anderson-Darling statistic has taken the complexity of the model while the other two statistics have not (Delignette-Mullerand Dutang, 2014). Thus, the remaining two statistics are auxiliary indicators.

(1) Steel intensity

All of goodness-of-fit statistics and criteria for steel intensity are giving preference to the Lognormal distribution.

Table S1 Goodness-of-fit statistics and criteria for steel intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.086	0.088	0.055	0.136
Cramer-von Mises statistic	0.380	0.173	0.039	0.941
Anderson-Darling statistic	2.636	1.125	0.257	5.846
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	1093.727	1075.436	1063.925	1141.011
Bayesian Information Criterion	1099.694	1081.403	1069.892	1146.978

(2) Cement intensity

All of goodness-of-fit statistics and criteria for cement intensity are giving preference to the Normal distribution.

Table S2 Goodness-of-fit statistics and criteria for cement intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.080	0.079	0.105	0.077
Cramer-von Mises statistic	0.348	0.208	0.416	0.187
Anderson-Darling statistic	2.430	1.510	2.803	1.331
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	1524.865	1524.893	1547.648	1515.090
Bayesian Information Criterion	1532.832	1530.860	1553.615	1521.057

(3) Wood intensity

All of goodness-of-fit statistics and criteria for wood intensity are giving preference to the Lognormal distribution.

Table S3 Goodness-of-fit statistics and criteria for wood intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.149	0.132	0.097	0.256
Cramer-von Mises statistic	0.697	0.437	0.271	2.899
Anderson-Darling statistic	4.462	2.767	1.497	15.892
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	1135.295	1122.672	1101.249	1297.248
Bayesian Information Criterion	1141.235	1128.611	1107.189	1303.187

(4) Brick intensity

The AIC and BIC values for brick intensity are giving preference to the Weibull distribution.

Table S4 Goodness-of-fit statistics and criteria for brick intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.164	0.178	0.191	0.103
Cramer-von Mises statistic	0.826	1.030	1.555	0.396
Anderson-Darling statistic	5.159	5.765	8.697	2.441
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	2010.020	2021.541	2063.594	2013.143
Bayesian Information Criterion	2015.946	2027.466	2069.520	2019.069

(5) Gravel intensity

The AIC, BIC and Anderson-Darling statistic for gravel intensity are giving preference to the Gamma distribution.

Table S5 Goodness-of-fit statistics and criteria for gravel intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.103	0.068	0.063	0.126
Cramer-von Mises statistic	0.437	0.107	0.139	0.567
Anderson-Darling statistic	2.866	0.721	0.852	3.533
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	1972.214	1949.861	1950.971	1995.553
Bayesian Information Criterion	1978.182	1955.828	1956.938	2001.520

(6) Sand intensity

The AIC and BIC for sand intensity are giving preference to the Gamma distribution and Normal distribution and the two criterions are extremely close. The Anderson-Darling statistic is giving preference to the Normal distribution.

Table S6 Goodness-of-fit statistics and criteria for sand intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.105	0.094	0.116	0.094
Cramer-von Mises statistic	0.415	0.277	0.487	0.237
Anderson-Darling statistic	2.570	1.760	2.971	1.532
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	1899.208	1893.002	1908.953	1893.882
Bayesian Information Criterion	1905.161	1898.956	1914.906	1899.836

(7) Asphalt intensity

All of goodness-of-fit statistics and criteria for asphalt intensity are giving preference to the Lognormal distribution.

Table S7 Goodness-of-fit statistics and criteria for asphalt intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.144	0.133	0.128	0.220
Cramer-von Mises statistic	0.837	0.556	0.408	1.956
Anderson-Darling statistic	4.795	3.225	2.466	10.866
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	402.610	390.368	389.114	481.495
Bayesian Information Criterion	408.314	396.072	394.818	487.199

(8) Lime intensity

The AIC, BIC and Anderson-Darling statistic for lime intensity are giving preference to the Weibull distribution.

Table S8 Goodness-of-fit statistics and criteria for lime intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.078	0.081	0.140	0.148
Cramer-von Mises statistic	0.223	0.221	0.793	0.790
Anderson-Darling statistic	1.249	1.281	4.646	4.481
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	1259.552	1260.691	1307.571	1314.628
Bayesian Information Criterion	1265.519	1266.659	1313.538	1320.595

(9) Glass intenisty

All of goodness-of-fit statistics and criteria for glass intensity are giving preference to the Gamma distribution.

Table S9 Goodness-of-fit statistics and criteria for glass intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.108	0.076	0.090	0.125
Cramer-von Mises statistic	0.561	0.249	0.363	0.633
Anderson-Darling statistic	3.451	1.618	2.353	3.785
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	287.872	278.517	300.504	292.658
Bayesian Information Criterion	293.784	284.429	306.416	298.569

(10) Total intenisty

The AIC and BIC for total intensity are giving preference to the Gamma distribution and Normal distribution and the two criterions are extremely close. The Anderson-Darling statistic is giving preference to the Normal distribution.

Table S10 Goodness-of-fit statistics and criteria for total intensity

	Weibull	Gamma	Lognormal	Normal
(a) Goodness-of-fit statistics				
Kolmogorov-Smirnov statistic	0.091	0.082	0.104	0.074
Cramer-von Mises statistic	0.387	0.214	0.371	0.145
Anderson-Darling statistic	2.478	1.382	2.322	0.950
(b) Goodness-of-fit criteria				
Aikake's Information Criterion	2219.358	2207.855	2219.721	2207.960
Bayesian Information Criterion	2225.325	2213.823	2225.688	2213.927

Section S7. Parameters of material intensity's PDF

Table S11 Parameters of PDF and medians and 95% CIs of random samples (n=5000) for various building material (Unit: kg/m^2)

Material type	Distribution	Parameters for distribution functions		Medians and 95% CIs of random samples
		Parameters A	Parameters B	
Cement	Normal	151.67	42.66	151.01 [64.03, 237.81]
Steel	Lognormal	2.81	0.55	16.52 [5.44, 50.08]
Wood	Lognormal	2.64	0.78	13.83 [2.82, 67.64]
Brick	Weibull	496.96	1.53	399.37 [42.09, 1182.67]
Gravel	Gamma	0.01	5.11	434.07 [154.02, 939.74]
Sand	Normal	498.73	162.97	496.20 [163.90, 827.82]
Asphalt	Lognormal	0.42	0.71	1.51 [0.35, 6.41]
Lime	Weibull	33.04	1.46	26.28 [2.49, 81.98]
Glass	Gamma	3.73	6.24	1.58 [0.64, 3.20]
Total	Normal	1616.98	457.62	1609.89 [676.78, 2541.09]

Note: a. For normal distribution, parameter A and parameter B refer to mean and standard deviation, respectively.

b. For lognormal distribution, parameter A and parameter B refer to mean and standard deviation, respectively.

c. For Weibull distribution, parameter A and parameter B refer to scale and shape, respectively.

d. For gamma distribution, parameter A and parameter B refer to rate and shape, respectively.

Section S8. Uncertainty propagation for the stock, inflow and outflow of floor area

The essence of the uncertainty propagation in the present paper is to repeat stock-driven model for 5,000 times. The R codes are listed as followings:

(1) Urban population

```
population_u <- matrix(nrow=20, ncol=5000) #define a matrix to store the urban population
for(i in 1:5000)
{
  population_u[,i]<-population[,i]*1000*urbanization[,i]
}
```

(2) Urban housing stock

```
stock_u<- matrix(nrow=20, ncol=5000) #define a matrix to store the urban housing stock
for(i in 1:5000)
{
  stock_u[,i]<-population_u[,i]*floor_u[,i]
}
```

(3) Inflow and outflow

```
#'data' contains the newly completed floor area
area_complete<-data$V2 #define the newly completed floor area
year_complete<-data$V1 #define the year vector
stock_u<-rbind(stock_u_2012,stock_u) #add the urban housing stock in 2012 into the 'stock_u' matrix
complete.sim<-matrix(nrow=119, ncol=5000) #define a matrix to store the newly complete floor area
output.sim<-matrix(nrow=88, ncol=5000) #define a matrix to store the outflow of floor area
for(i in 1:5000)
{
  year_complete<-data$V1 #initialize the 'year_complete' for every loop
  area_complete<-data$V2 #initialize the 'area_complete' for every loop
  A<-nls_boot$coefboot[i,1] #get the scale parameter of building lifetime function
  B<-nls_boot$coefboot[i,2] #get the shape parameter of building lifetime function
  for(j in 2013:2015)
  {
    output<-sum(area_complete*(pweibull(j-year_complete,scale=A,shape=B)-pweibull(j-1-year_complete,scale = A,shape = B))) #compute the outflow at year j
    input<-stock_u[j-2011,i]-stock_u[j-2012,i]+output # compute the newly completed floor area at year j
    area_complete<-c(area_adjust,input_2013) #add the 'input' into the 'area_complete' vector
    year_complete<-c(year_complete,j) #add the 'j' into the 'year_complete' vector
    output.sim[j-2012,i]<-output #store the outflow at year j into the 'output.sim' matrix
  }
  for(j in 2016:2100)
  {
   Y<-trunc((j-2011)/5)+4 #because the prediction of population can only provide with 5-year interval time series
    output<-sum(area_complete*(pweibull(j-year_complete,scale=A,shape=B)-pweibull(j-1-year_complete,scale = A,shape = B))) #compute the outflow at year j
    input<-(stock_u[Y,i]-stock_u[Y-1,i])/5+output #compute the newly completed floor area at year j
    area_complete<-c(area_complete,input) #add the 'input' into the 'area_complete' vector
```

```
            year_complete<-c(year_complete,j) #add the 'j' into the 'year_complete' vector
            output.sim[j-2012,i]<-output #store the outflow at year j into the 'output.sim' matrix
        }
        complete.sim[,i]<-area_complete #store the inflow from 2013-2100 into the 'input.sim'
matrix
    }
```

Section S9. Uncertainty propagation for the stock, inflow and outflow of materials

The material stock, inflow and outflow are simply multiplying the counterparts by the material intensity. The R codes are listed as followings:

(1) Material stock

```
    #obtain 5,000 simulations for the material intensity
    set.seed(1); cement_u<-rnorm(5000,mean=151.671,sd=42.656)
    set.seed(1); steel_u<-rlnorm(5000,meanlog=2.813,sdlog=0.545)
    set.seed(1); wood_u<-rlnorm(5000,meanlog=2.639,sdlog=0.780)
    set.seed(1); brick_u<-rweibull(5000,scale=496.964,shape=1.527)
    set.seed(1); gravel_u<-rgamma(5000,rate=0.011,shape=5.113)
    set.seed(1); sand_u<-rnorm(5000,mean=498.727,sd=162.968)
    set.seed(1); asphalt_u<-rlnorm(5000,meanlog=0.422,sd=0.711)
    set.seed(1); lime_u<-rweibull(5000,scale=33.041,shape=1.457)
    set.seed(1); glass_u<-rgamma(5000,rate=3.733,shape=6.243)
    set.seed(1); total_u<-rnorm(5000,mean=1616.980,sd=457.618)
    #material stock from 1985-2100
    material_stock.sim<- matrix(nrow=48, ncol=5000) #define a matrix to store the material
stock
    #historical material stock from 1985-2011; 'stock_u_his' is a vector that contains the urban
housing stock from 1985-2011
    for(i in 1:27)
    {
        material_stock.sim[i,]<-stock_u_his[i]*intensity/1000
    }
    #material stock from 2012-2100
    for(j in 28:48)
    {
        material_stock.sim[j,]<-stock_u[j-27,]*intensity/1000
    }
```

(2) Material inflow

```
    #material inflow from 1982-2100
    material_inflow.sim<- matrix(nrow=88, ncol=5000) #define a matrix to store the material
inflow
    for(i in 1:119)
    {
        material_inflow.sim[i,]<-complete.sim[i,]*intensity/1000
    }
```

(3) Material outflow

```
#material outflow from 2013-2100
material_outflow.sim<- matrix(nrow=88, ncol=5000)
for(i in 1:88)
{
  material_outflow.sim[i,]<-output.sim[i,]*intensity/1000
}
```

Section S10. Material flows of various material types

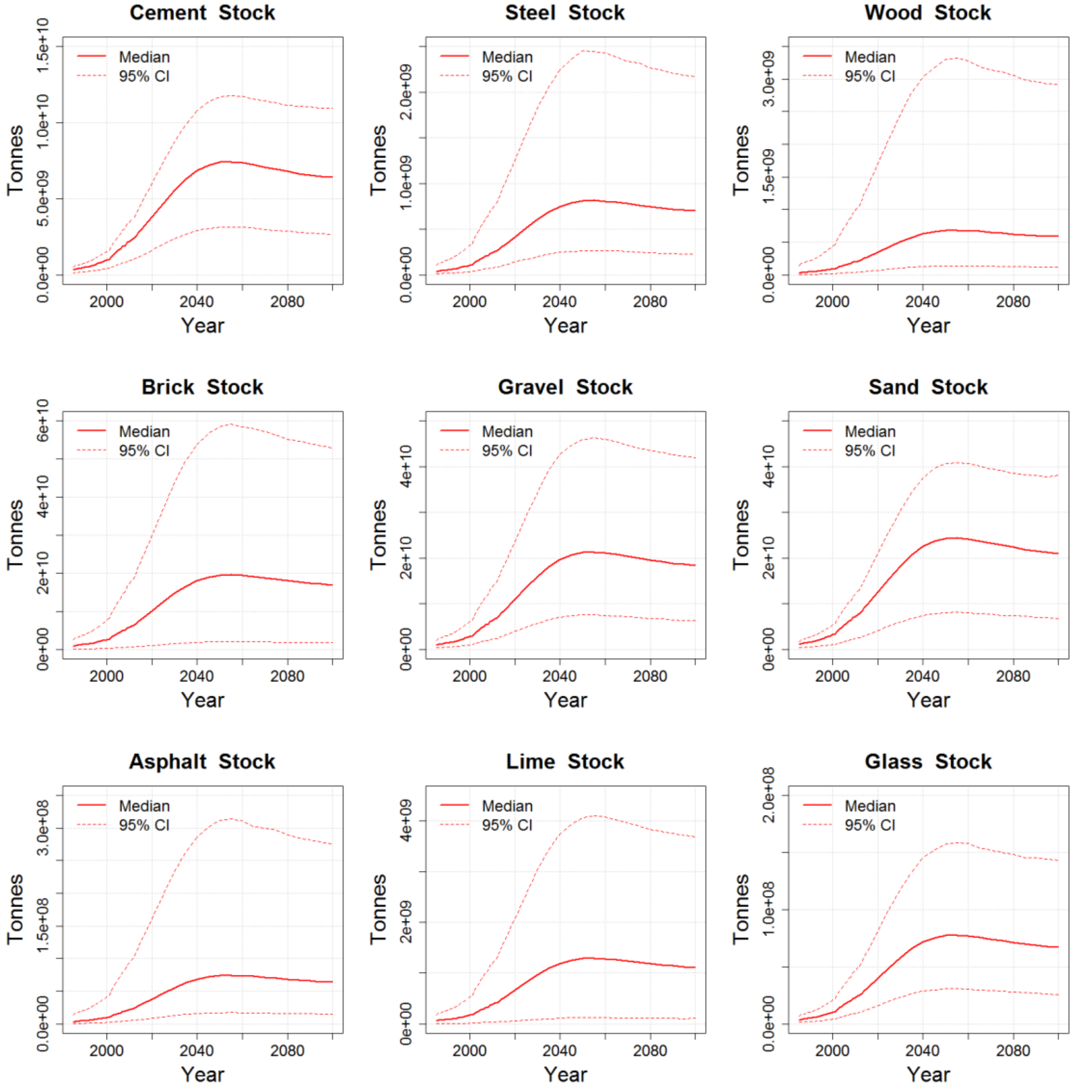


Figure S3 The material stock of the urban housing stock during 1985-2100 (cement, steel, wood, brick, gravel, sand, asphalt, lime and glass).

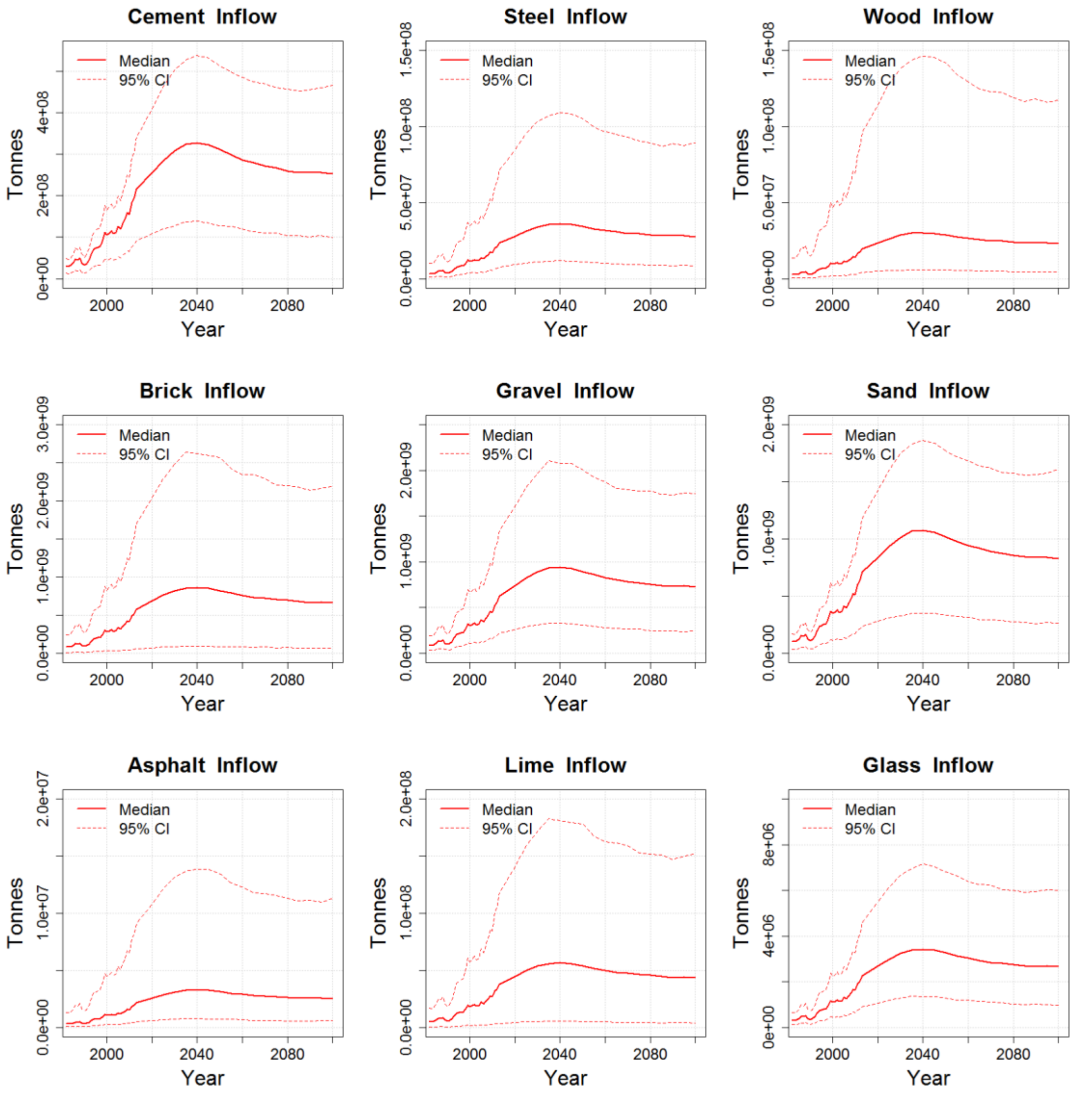
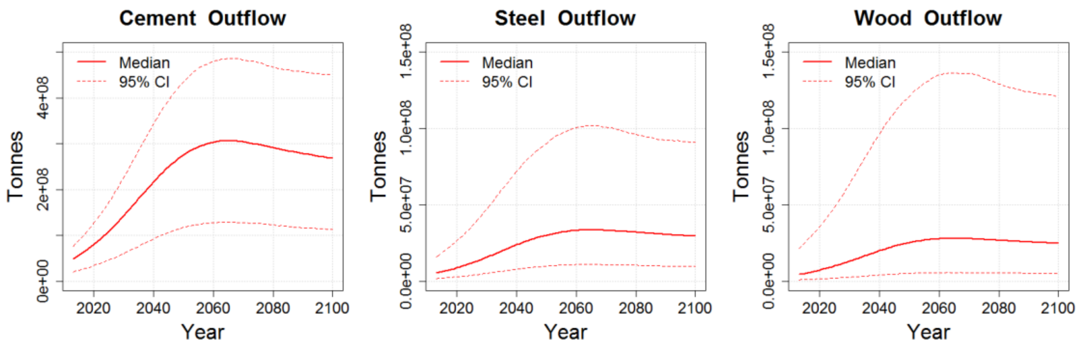


Figure S4 The material inflow of the urban housing stock during 1985-2100 (cement, steel, wood, brick, gravel, sand, asphalt, lime and glass).



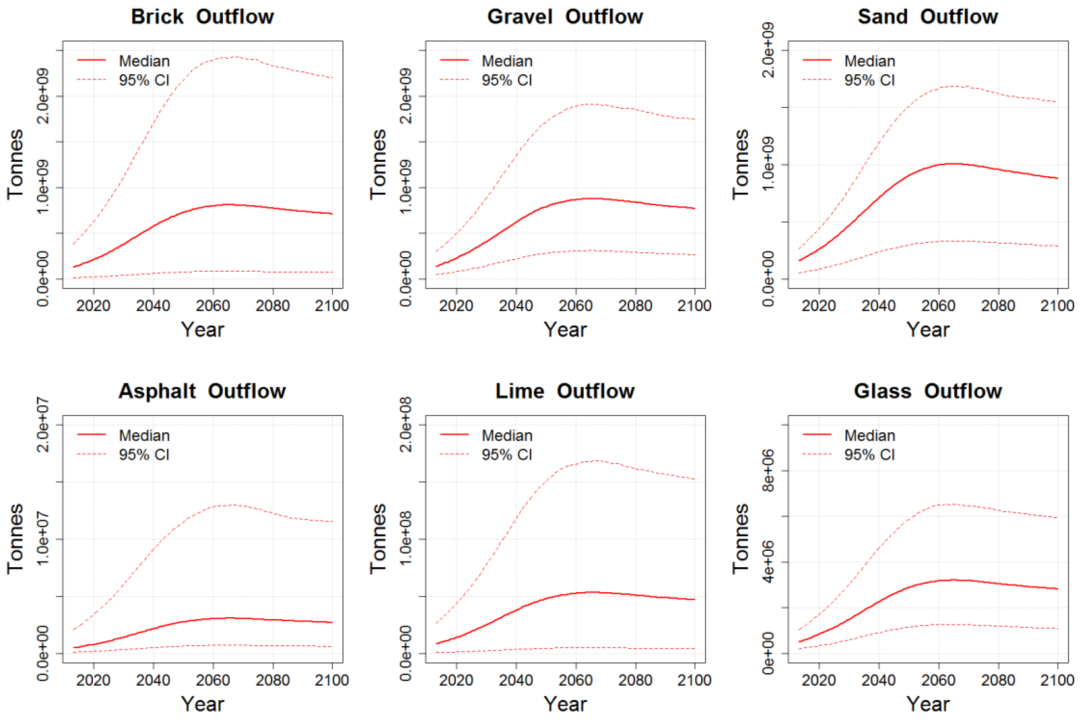


Figure S5 The material outflow of the urban housing stock during 2013-2100 (cement, steel, wood, brick, gravel, sand, asphalt, lime and glass).

Section S11. Time-delay effect of model inputs on model outputs

To explain the time-delay effect of model inputs on model outputs, a simple formula derivation is presented here.

From the following equations

$$CA_t = SA_t - SA_{t-1} + DA_t \ (1)$$

$$SA_t = P_t \times U_t \times a_t \ (2)$$

$$DA_t = \sum_{t'=t_0}^{t'=t-1} CA_{t'} \times (1 - S_{t-t'}) \ (3)$$

$$S(t) = exp(-(\frac{t}{\lambda})^k) \ (8)$$

we can obtain housing stock, demolished floor area and newly completed floor area for year 2013:

$$SA_{2013} = P_{2013} \times U_{2013} \times a_{2013}$$

$$DA_{2013} = \sum_{t'=t_0}^{t'=2012} CA_{t'} \times (1 - exp(-(\frac{2013 - t'}{\lambda})^k))$$

$$CA_{2013} = P_{2013} \times U_{2013} \times a_{2013} - P_{2012} \times U_{2012} \times a_{2012} + \sum_{t'=t_0}^{t'=2012} CA_{t'} \times (1 - exp(-(\frac{2013-t'}{\lambda})^k))$$

Then we could obtain demolished floor area and newly completed floor area for year 2014:

$$DA_{2014} = \sum_{t'=t_0}^{t'=2013} CA_{t'} \times (1 - S_{2014-t'})$$

$$= \sum_{t'=t_0}^{t'=2012} CA_{t'} \times (1 - S_{2014-t'}) + CA_{2013} \times \left(1 - S_{2014-2013}\right)$$

$$= \sum_{t'=t_0}^{t'=2012} CA_{t'} \times (1 - S_{2014-t'}) + (P_{2013} \times U_{2013} \times a_{2013} - P_{2012} \times U_{2012} \times a_{2012} +$$

$$\sum_{t'=t_0}^{t'=2012} CA_{t'} \times (1 - exp(-(\frac{2013-t'}{\lambda})^k)))) \times \left(1 - exp(-(\frac{1}{\lambda})^k)\right)$$

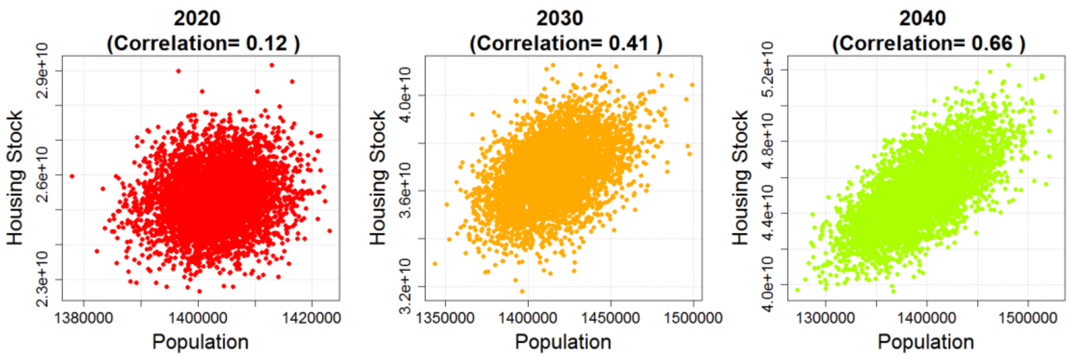
$$CA_{2014} = P_{2014} \times U_{2014} \times a_{2014} - P_{2013} \times U_{2013} \times a_{2013} +$$

$$\sum_{t'=t_0}^{t'=2012} CA_{t'} \times (1 - S_{2014-t'}) + (P_{2013} \times U_{2013} \times a_{2013} - P_{2012} \times U_{2012} \times a_{2012} +$$

$$\sum_{t'=t_0}^{t'=2012} CA_{t'} \times (1 - exp(-(\frac{2013-t'}{\lambda})^k)))) \times \left(1 - exp(-(\frac{1}{\lambda})^k)\right)$$

Since P_{2012}, U_{2012}, a_{2012} and $\sum_{t'=t_0}^{t'=2012} CA_{t'}$ are already known, uncertainties in the newly completed floor area and demolished floor area at year 2014 might be affected by the evolution of population, urbanization rate and per capita floor area in 2013. The whole time-delay effect of model inputs in the preceding years on 2100's newly completed floor area or demolished floor area would be a tedious work. Nonetheless, the deduction demonstrated above has revealed the time-delay effect amongst the dynamic housing stock model.

Section S12. Spearman's rank correlation coefficients between the total population and the urban housing stock



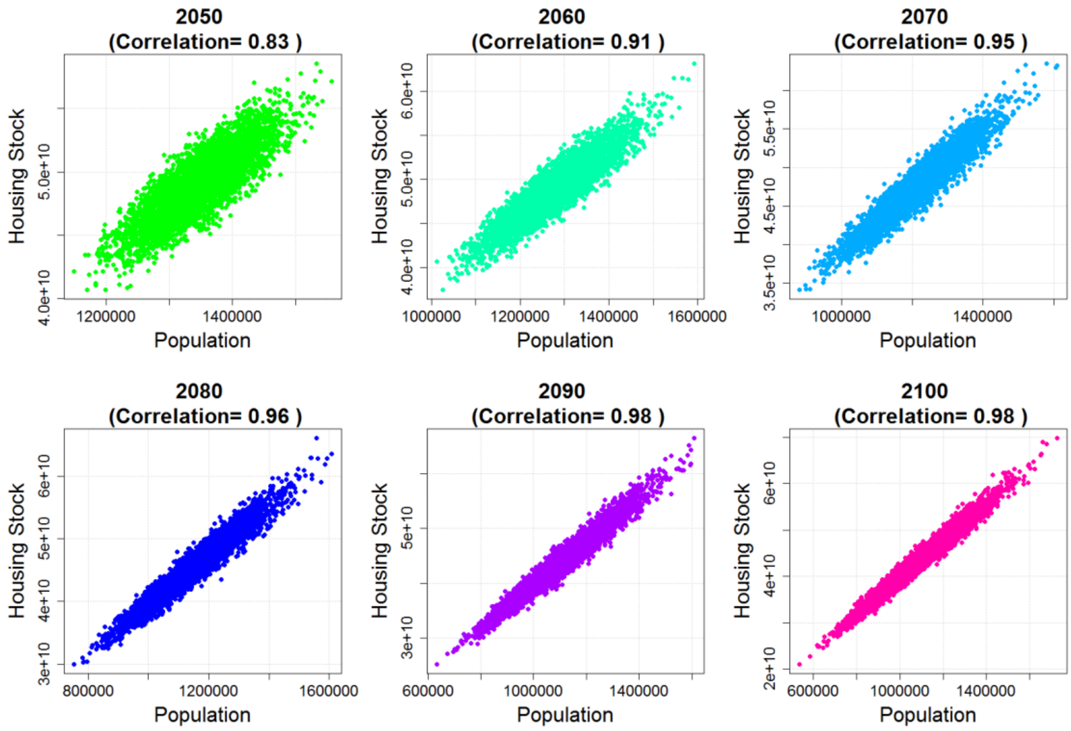


Figure S6 Spearman's rank correlation coefficients between the total population and the urban housing stock.

Section S13. Data Sources

(1) Urban housing floor area per capita. Historical data of floor area in the urban region is quoted from the China Statistical Yearbook (NBSC, 1983-2015).

(2) Building lifetime. The floor area completed annually can be obtained from the annual official statistical publication (NBSC, 1983-2015). The floor area demolished annually is calculated using the estimation approach proposed by Cai et al. (2015).

(3) Population. Data of historical total fertility rate and life expectancy is obtained from the United Nations Population Division (UNDP, 2015).

(4) Urbanization rate. Data of population urbanization rate is obtained from the China Population & Employment Statistics Yearbook (NBSPSD, 1988-2014).

(5) Material intensity. 146 urban residential building project samples including various kinds of building structure are collected from the Construction Project Investment Estimation Handbook (Yu and Li, 1999).

Reference

Alkema, L., Raftery, A. E., Gerland, P., Clark, S. J., Pelletier, F., Buettner, T., & Heilig, G. K. (2011). Probabilistic projections of the total fertility rate for all countries. Demography, 48(3), 815-839.

Cai, W., Wan, L., Jiang, Y., Wang, C., & Lin, L. (2015). Short-Lived Buildings in China: Impacts on Water, Energy, and Carbon Emissions. Environmental science & technology, 49(24), 13921-13928.

Delignette-Muller, M. L. and Dutang, C. (2014). fitdistrplus: An R Package for Fitting Distributions. Journal of Statistical. Software, 64, 1-34.

Hyndman, R. (2016). forecast: Forecasting Functions for Time Series and Linear Models version 7.3. (Available at http://CRAN.R-project.org/package=forecast)

NBSC, National Bureau of Statistics of China. (1983-2015). China Statistical Yearbook. China Statistics Press, Beijing.

NBSPSD, National Bureau of Statistics of Population and Statistics Division. (1988-2014). China Population & Employment StatisticsYearbook. China Statistics Press, Beijing.

Raftery, A. E., Li, N., Ševčíková, H., Gerland, P., & Heilig, G. K. (2012). Bayesian probabilistic population projections for all countries. Proceedings of the National Academy of Sciences, 109(35), 13915-13921.

Raftery, A. E., Chunn, J. L., Gerland, P., & Ševčíková, H. (2013). Bayesian probabilistic projections of life expectancy for all countries. Demography, 50(3), 777-801.

Ševčíková H, Alkema L, Raftery AE (2011). bayesTFR: An R package for probabilistic projections of the total fertility rate. Journal of Statistical Software, 43(1), 1-29.

Ševčíková H, Raftery AE (2011). bayesLife: Bayesian Projection of Life Expectancy R package version 0.4-0. (Available at http://CRAN.R-project.org/package=bayesLife)

Ševčíková H, Raftery AE (2012). bayesPop: Probabilistic Population Projection R package version 1.0-3. (Available at http://CRAN.R-project.org/package=bayesPop)

UNDP, United Nations, Department of Economic and Social Affairs, Population Division (2015). World Population Prospects: The 2015 Revision, DVD Edition.

Yu, H. and Li, G. J. (1999). Construction Project Investment Estimation Handbook. (In Chinese). Beijing, China: China Architecture &Building Press.