In [1]:
```
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import colorsys
import seaborn as sns

def scale_lightness(rgb, scale_l):
    h, l, s = colorsys.rgb_to_hls(*rgb)
    return colorsys.hls_to_rgb(h, min(1, l * scale_l), s = s)
from sympy import *
init_printing(use_latex='mathjax')
```

Fig.3

b)

In [2]:
```
%matplotlib inline

Corr=np.loadtxt('fig3b.dat')

Corr=Corr-np.mean(Corr)
np.fill_diagonal(Corr, np.nan)
im=plt.imshow(Corr)
```

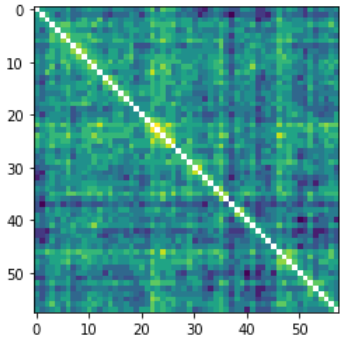


Fig.4

In [3]:
```
%matplotlib inline

Corr=np.loadtxt('fig4.dat')

np.fill_diagonal(Corr, np.nan)
im=plt.imshow(Corr)
```

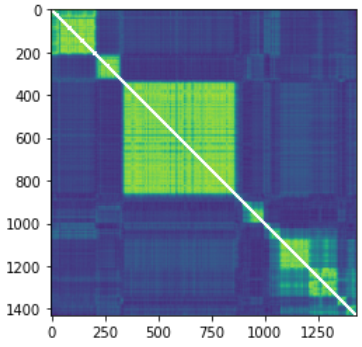


Fig.5

In [4]:
```
q,b,h = symbols(r"q \beta h", positive=True)
f = Function('f')

f2=lambda q: q**3/2+h**2*q

def RSpot(F):
    RSS = b**2*(f(1)-f(q))/2 + q/(1-q)/2+log(1 - q)/2 + 1 + log(2*np.pi);
```

```
    return (-RSS/b).replace(f,F).doit()
RSpot(f)
```

Out[4]:

$$\frac{-\frac{\beta^2(f(1)-f(q))}{2}-\frac{q}{2\cdot(1-q)}-\frac{\log(1-q)}{2}-2.83787706640935}{\beta}$$

In [5]:
```
%matplotlib inline
CLR=plt.rcParams['axes.prop_cycle'].by_key()['color']

rsHB=lambda B,H,Q: -RSpot(f2).subs(h,H).subs(q,Q).subs(b,B)

RSH={};RSHb={};RSHT={};

Temp=0.5
QH=np.arange(0.999,1-Temp,-0.01)
QHb=np.arange(0.5,0.001,-0.01)

for h in [1,2,4,8]:
    RSH[h]=[rsHB(1/Temp,h,q) for q in QH]
    RSHb[h]=[rsHB(1/Temp,h,q) for q in QHb]

Temp=0.55
QHT=np.arange(0.999,1-Temp,-0.01)

for h in [1,2,4,8]:
    RSHT[h]=[rsHB(1/Temp,h,q) for q in QHT]

plt.yscale('log')
plt.xlim([0., 1])

i=0;
for h in [1,2,4,8]:
    plt.plot(QH,RSH[h],color=CLR[i])
    plt.plot(QHb,RSHb[h],linestyle='dotted',color=CLR[i])
    plt.plot(QHT,RSHT[h],color=CLR[i],linestyle='dashed',alpha=0.3)
    i+=1;

plt.ylabel(r"$V_{RS}$")
plt.xlabel(r"$q$");
```

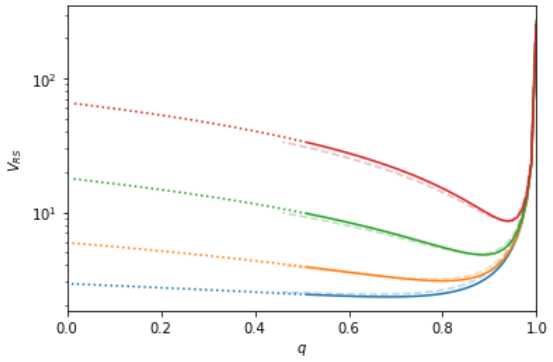


Fig.6

In [6]:
```
q,b,h = symbols(r"q \beta h", positive=True)
q01 = symbols(r"q_01", positive=True)
f = Function('f')

f3=lambda q: q**3/2

def q01sp(F):
    return sqrt(q - b**2*(1 - q)**2*diff(f(q),q)).replace(f,F).doit()

q01sp(f)
```

Out[6]:

$$\sqrt{-\beta^2(1-q)^2\frac{d}{dq}f(q)+q}$$

In [7]:
```
def FPpot(F):
    FP = b**2*(f(1)-f(q)+diff(f(q),q)*(1-q))/2 + b**2*f(q01)+ log(1 - q)/2 + 1 + log(2*np.pi);
    return (-FP/b).replace(f,F).doit().replace(q01,q01sp(F))

FPpot(f)
```

Out[7]:

$$\frac{-\frac{\beta^2\left((1-q)\frac{d}{dq}f(q)+f(1)-f(q)\right)}{2}-\beta^2 f\left(\sqrt{-\beta^2(1-q)^2\frac{d}{dq}f(q)+q}\right)-\frac{\log(1-q)}{2}-2.83787706640935}{\beta}$$

In [8]:

```
def Mpot(F):
    FP = b**2*(f(1)+f(q))/2 + (log(1 - q) + q)/2 + 1 + log(2*np.pi);
    return (-FP/b).replace(f,F).doit()
Mpot(f)
```

Out[8]:

$$\frac{-\frac{\beta^2(f(1)+f(q))}{2}-\frac{q}{2}-\frac{\log(1-q)}{2}-2.83787706640935}{\beta}$$

In [9]:

```
def LREq(F):
    LR = 2/(1-q)**2-2*b**2*diff(f(q),q,2)
    return LR.replace(f,F).doit()
LREq(f)
```

Out[9]:

$$-2\beta^2\frac{d^2}{dq^2}f(q)+\frac{2}{(1-q)^2}$$

In [10]:

```
mm=lambda Q: Mpot(f3).subs(b,1/0.59).subs(q,Q)
qq01=lambda Q: q01sp(f3).subs(b,1/0.59).subs(q,Q)
ffp=lambda Q: FPpot(f3).subs(b,1/0.59).subs(q,Q)
llr=lambda Q: LREq(f3).subs(b,1/0.59).subs(q,Q)
```

In [11]:

```
%matplotlib inline
CLR=plt.rcParams['axes.prop_cycle'].by_key()['color']

Q1=np.arange(0.,0.4,0.002)
Q2=np.arange(0.4,0.9,0.01)

QQ1=[q for q in Q1 if llr(q)>0]
MM1=[mm(q) for q in Q1 if llr(q)>0]
QQ011=[qq01(q) for q in Q1 if llr(q)>0]
FFP1=[ffp(q) for q in Q1 if llr(q)>0]

QQ2=[q for q in Q2 if llr(q)>0]
MM2=[mm(q) for q in Q2 if llr(q)>0]
QQ012=[qq01(q) for q in Q2 if llr(q)>0]
FFP2=[ffp(q) for q in Q2 if llr(q)>0]

Q3=np.arange(0.1,0.7,0.01)

QQ3=[q for q in Q3 if llr(q)<0]
MM3=[mm(q) for q in Q3 if llr(q)<0]
QQ013=[qq01(q) for q in Q3 if llr(q)<0]
FFP3=[ffp(q) for q in Q3 if llr(q)<0]


plt.xlim([0., 0.8])
plt.ylim([-2.1, -2.085])

plt.plot(QQ1,MM1,label=r"$V_{M}(q)$",color=CLR[0])
plt.plot(QQ011,FFP1,label=r"$V_{FP}(p)$",color=CLR[1])

plt.plot(QQ2,MM2,color=CLR[0])
plt.plot(QQ012,FFP2,color=CLR[1])

plt.plot(QQ3,MM3,linestyle='dotted',color=CLR[0])
plt.plot(QQ013,FFP3,linestyle='dotted',color=CLR[1])

plt.ylabel(r"$V_{M \;/\; FP}$")
plt.xlabel(r"$q$ or $p$");
```

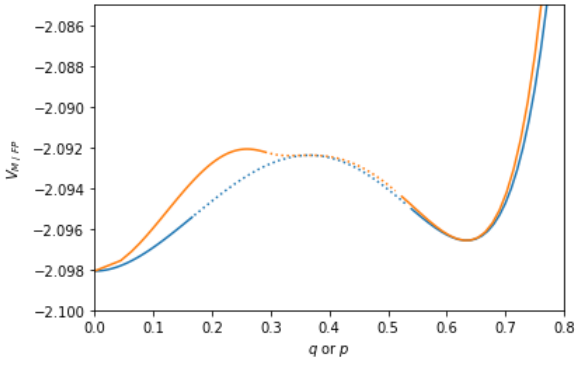


Fig.7

In [12]:
```
%matplotlib inline

plt.yscale('log')

CL=[(0.12151575, 0.07117741, 0.12698763), (0.19093944, 0.1283047, 0.25077369), (0.23670785, 0.190297, 0.38504973), (0.25404842, 0.
CL2=[(0.1125315, 0.06742194, 0.17414848), (0.22741085, 0.10140876, 0.25340813), (0.34940101, 0.11863987, 0.3138355), (0.47114798,

def chi_M(q):
    return ((-1 + q)**2 *(-1 + 5*q))/(1 + q*(-5 + 6*q))

def chi_FP(q):
    return ((-1 + q)**2 *(-1 + 4*q))/(1 + q*(-5 + 6*q))

def chi_MB(q,T):
    beta=1./T;
    return ((-1+q)**2*(-1-3*q-(-1+q)**3 * beta**2))/(-1-q+ 2*(-1+q)**2 * beta**2+(-1+q)**5 * beta**4)

def chi_FPB(q,T):
    beta=1./T;
    return ((-1+q)**2*(-1-2*q-(-1+q)**3 * beta**2))/(-1-q+ 2*(-1+q)**2 * beta**2+(-1+q)**5 * beta**4)

beta=2

tQ=np.arange(0.501,1,0.001)
tchiM=[chi_M(q) for q in tQ]
tchiMB=[chi_MB(q,1/beta) for q in tQ]
tchiFP=[chi_FP(q) for q in tQ]
tchiFPB=[chi_FPB(q,1/beta) for q in tQ]

qq={}
cchiFP={}
cchiM={}
for H in [1,2,4,8]:
        qq[H]={}
        cchiFP[H]={}
        cchiM[H]={}
        c=1
        for N in [50,100,200]:
            qq[H][N]=np.loadtxt("fig7_q_h"+str(H)+"_N"+str(N)+".dat")
            cchiFP[H][N]=np.loadtxt("fig7_chiFP_h"+str(H)+"_N"+str(N)+".dat")
            cchiM[H][N]=np.loadtxt("fig7_chiM_h"+str(H)+"_N"+str(N)+".dat")

            plt.scatter(qq[H][N],cchiFP[H][N],color=CL[c],s=12)
            c+=2
            plt.scatter(qq[H][N],cchiM[H][N],color=CL2[c],s=12)

plt.plot(tQ,tchiFPB,'-',color='red',linewidth=0.8)
plt.plot(tQ,tchiMB,'-',color='blue',linewidth=0.8)

axes = plt.gca()
axes.set_ylim([0.001,2])
axes.set_xlim([0.55,1])

plt.xlabel(r"$q$")
plt.ylabel(r"$\hat{\chi}_{FP}$, $\hat{\chi}_{M}$");
```

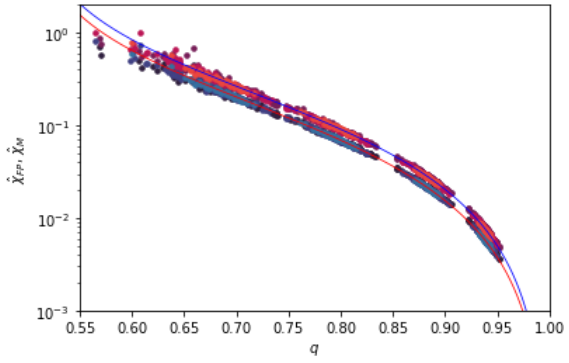


Fig.8

In [13]:
```
%matplotlib inline

import pandas as pd

plt.yscale('log')
```

```
N=200;

HH=[1,2,4,8];

CHIFP=[0.25425306646769386, 0.07383360842347114, 0.02075512401662805, 0.0056169385574390525]
CHIM=[0.25425306646769386, 0.07383360842347114, 0.02075512401662805, 0.0056169385574390525]
CHIFPi=[0.1646145606038888, 0.05117062102901271, 0.013243564750903023, 0.003413632838711431]
CHIMi=[0.0865767464777667, 0.02807282908622482, 0.006696427173706508, 0.0016299037430920968]
CHIFPi2=[0.1528174177919001,0.03269201596608891,0.007331507931929829 4,0.00178539310682821]
CHIMi2=[0.07574824474583446,0.010650702376405571,0.0010096595827595472,7.884549417486774e-05]


errCHIFP=[0.009964508699285467, 0.001528500029881874, 0.0003117655854351441, 7.537507804533772e-05]
errCHIM=[0.012876849329055749, 0.0019925871282696385, 0.0004077596879623603 6, 9.878752892955916e-05]
errCHIFPi=[0.04833354592491031, 0.005161460628566164, 0.0012036236267906082, 0.00029980350383556 99]
errCHIMi=[0.026578729727508445, 0.006378375070315 93, 0.0011930248190997745, 0.0002930212853481007]
errCHIFPi2=[0.020564640448557873,0.0027414719766916664,0.0002273603755353009,2.1096864424710125e-05]
errCHIMi2=[0.013560212675732234,0.001993725546832866,0.0002181841319161539 5,1.8324375536903565e-05]


plt.errorbar(HH,CHIFP,yerr=errCHIFP,c='blue',fmt='o')
plt.errorbar(HH,CHIM,yerr=errCHIM,c='red',fmt='o')
plt.errorbar(HH,CHIFPi,yerr=errCHIFPi,c='green',fmt='o')
plt.errorbar(HH,CHIMi,yerr=errCHIMi,c='orange',fmt='o')
plt.errorbar(HH,CHIFPi2,yerr=errCHIFPi2,c='green',fmt='x')
plt.errorbar(HH,CHIMi2,yerr=errCHIMi2,c='orange',fmt='x')

chiFP=pd.read_csv("fig8_FP.csv",header=None)[0]
chiM=pd.read_csv("fig8_M.csv",header=None)[0]
chiFPi=pd.read_csv("fig8_FPi.csv",header=None)[0]
chiMi=pd.read_csv("fig8_Mi.csv",header=None)[0]
chiFPi2=pd.read_csv("fig8_FPi2.csv",header=None)[0]
chiMi2=pd.read_csv("fig8_Mi2.csv",header=None)[0]
OV=pd.read_csv("fig8_Mi.csv",header=None)[1]
Hfield=pd.read_csv("fig8_FP.csv",header=None)[2]

plt.plot(Hfield,chiFP,'--',color='blue',label=r'$N\to\infty$'+', Gaussian h')
plt.plot(Hfield,chiM,'--',color='red',label=r'$N\to\infty$'+', Gaussian h')

plt.plot(Hfield,chiFPi2,'--',color='green',label=r'$N\to\infty$'+', Gaussian h')
plt.plot(Hfield,chiMi2,'--',color='orange',label=r'$N\to\infty$'+', Gaussian h')
plt.plot(Hfield,chiFPi,':',color='green',label=r'$N\to\infty$'+', homogenous h')#/Hfield
plt.plot(Hfield,chiMi,':',color='orange',label=r'$N\to\infty$'+', homogenous h')#/Hfield

plt.xlabel(r"$h$")
plt.ylabel(r"$\hat{\chi}$");
```

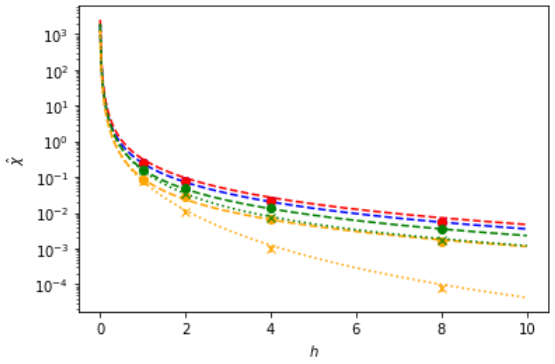
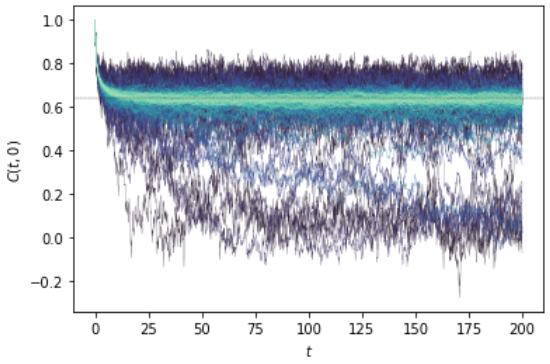


Fig.9

a)

In [14]:

```
CL=sns.color_palette("mako", 8)

c=0;
for N in [100,200,400,800,1600,3200,6400,12800]:
    for i in range(20,40):
        C=np.loadtxt("fig9a_N"+str(N)+"_S"+str(i)+".dat")
        plt.plot(C[0],C[1],lw=0.2,color=CL[c])
    c+=1;

plt.axhline(y=0.6339, color='black', linestyle='--',linewidth=0.3)
plt.ylabel(r"$C(t,0)$")
plt.xlabel(r"$t$");
```



b)

In [15]:
```
CL=sns.color_palette("mako", 8)

def Eeq(q):
    return -np.sqrt(1/6.)/np.sqrt(q - q**2)

EMCT=-1/np.sqrt(3/8)/2
EK=-1/0.5861/2

qMCT=0.5
qK=0.644883

qq=np.arange(0.5,0.644883,0.001)
Eq=[Eeq(q) for q in qq]

T=0.59
c=0
for N in [100,200,400,800,1600,3200,6400,12800]:
    Q=np.loadtxt("fig9b_Q_N"+str(N)+".dat")
    E=np.loadtxt("fig9b_E_N"+str(N)+".dat")
    plt.scatter(E,Q,color=CL[c],label="N="+str(N))
    c+=1

plt.axvline(EK, color='black', linestyle='--',linewidth=0.3)
plt.axvline(EMCT, color='black', linestyle='--',linewidth=0.3)

plt.axhline(y=qK, color='black', linestyle='--',linewidth=0.3)
plt.axhline(y=qMCT, color='black', linestyle='--',linewidth=0.3)

plt.plot(-1/T/2,0.6339,'x',color='red',markersize=10,label='expected at T=0.59')

plt.plot(Eq,qq,'-.k',label='mean-field')

plt.xlabel(r"$\langle H \rangle/N$")
plt.ylabel(r"$q$");
```

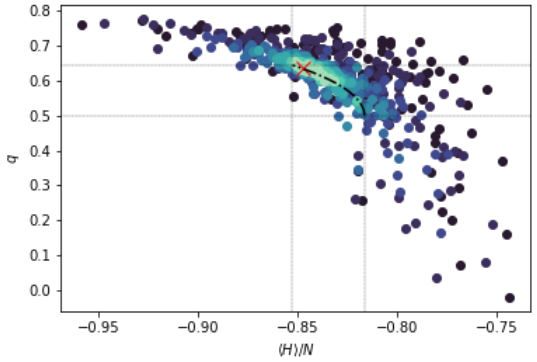
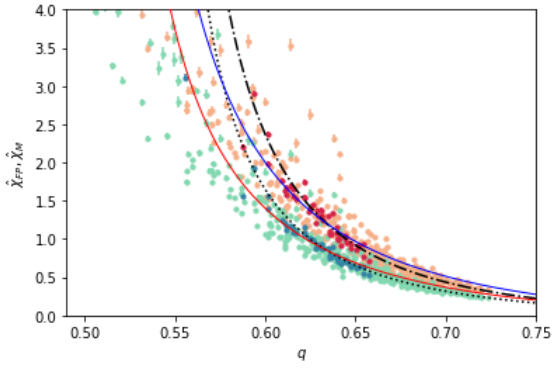


Fig. 10

a)

In [16]:
```
%matplotlib inline

CL=[(0.12151575, 0.07117741, 0.12698763), (0.19093944, 0.1283047, 0.25077369), (0.23670785, 0.190297, 0.38504973), (0.25404842, 0.
CL2=[(0.1125315, 0.06742194, 0.17414848), (0.22741085, 0.10140876, 0.25340813), (0.34940101, 0.11863987, 0.3138355), (0.47114798,
```

```
def chi_M(q):
    return ((-1 + q)**2 *(-1 + 5*q))/(1 + q*(-5 + 6*q))

def chi_FP(q):
    return ((-1 + q)**2 *(-1 + 4*q))/(1 + q*(-5 + 6*q))

def chi_MB(q,T):
    beta=1./T;
    return (2 *(1 + 3*q - 3*(-1 + q)**2 *q*beta**2))/((2/(-1 + q)**2 - 6*q*beta**2)*(1 + q - 3*(-1 + q)**2 *q*beta**2))

def chi_FPB(q,T):
    beta=1./T;
    return (2 *(1 + 2*q - 3*(-1 + q)**2 *q*beta**2))/((2/(-1 + q)**2 - 6*q*beta**2)*(1 + q - 3*(-1 + q)**2 *q*beta**2))

T=0.59
tQ=np.arange(0.535,0.75,0.0001)
tchiM=[chi_M(q) for q in tQ]
tchiMB=[chi_MB(q,T) for q in tQ]
tchiFP=[chi_FP(q) for q in tQ]
tchiFPB=[chi_FPB(q,T) for q in tQ]

c=10
for N in [400,6400]:
        C=np.loadtxt("fig10_all_N"+str(N)+".dat")

        plt.errorbar(C[0],C[2],yerr=C[4],xerr=C[5],ls = "None",color=CL2[c],zorder=1)#,s=12)
        plt.errorbar(C[0],C[1],yerr=C[3],xerr=C[5],ls = "None",color=CL[c],zorder=1)#,s=12)
        plt.scatter(C[0],C[2],color=CL2[c],s=10,zorder=2)
        plt.scatter(C[0],C[1],color=CL[c],s=10,zorder=2)

        c-=4


plt.plot(tQ,tchiMB,'-.k',label=r'$\hat{\chi}_{M}$, $N\to\infty$',zorder=3)
plt.plot(tQ,tchiFPB,':k',label=r'$\hat{\chi}_{FP}$, $N\to\infty$',zorder=3)

plt.plot(tQ,tchiM,'-',color='blue',zorder=3,linewidth=1)
plt.plot(tQ,tchiFP,'-',color='red',zorder=3,linewidth=1)

axes = plt.gca()
axes.set_ylim([0,4])
axes.set_xlim([0.49,0.75])

plt.xlabel(r"$q$")
plt.ylabel(r"$\hat{\chi}_{FP},\hat{\chi}_{M}$");
```



b)

In [17]:

```
%matplotlib inline

CL=[(0.921460863132993, 0.9512120106378944, 0.9302557793948234), (0.8546193719792059, 0.9145821230194208, 0.8771231358554057), (0.
qMCT=0.5
qK=0.644883

def M1B(q,T):
    beta=1./T;
    return 2./(1-q)**2 - 6*q*beta**2

def M1(q):
    beta=1./T;
    return 2./(1-q)**2 - 4/(1-q)

tQ=np.arange(0.501,0.75,0.001)
m1b=[M1B(q,T) for q in tQ]
m1=[M1(q) for q in tQ]
```

```
T=0.59


c=3
for N in [400,6400]:
        C=np.loadtxt("fig10_all_N"+str(N)+".dat")

        plt.errorbar(C[0],-2/(C[2]-2*C[1]),yerr=-2*(C[3]+2*C[4])/(C[1]-2*C[2])**2,xerr=C[5],ls = "None",color=CL[c],zorder=1)#,s=1
        plt.scatter(C[0],-2/(C[2]-2*C[1]),color=CL[c],label="N="+str(N),zorder=2)

        c+=5

plt.plot(tQ,np.array(m1b),':k',label=r'$\lambda_R$ $N\to\infty$',zorder=3)
clr='blue'
plt.plot(tQ,np.array(m1),'-',color=clr,label=r'$\lambda_R(q_{EA})$ $N\to\infty$',zorder=3,linewidth=1)

plt.axvline(x=qK, color='black', linestyle='--',linewidth=0.3)
plt.axvline(x=qMCT, color='black', linestyle='--',linewidth=0.3)

axes = plt.gca()
axes.set_ylim([0,14])

plt.xlabel(r"$q$")
plt.ylabel(r"$\lambda_{R}$");
```

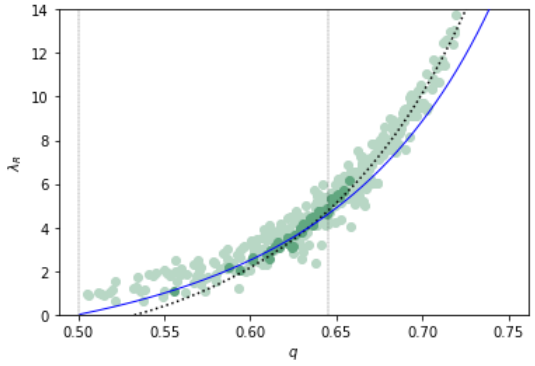


Fig.11

In [18]:
```
chiFP = 0.8521
chiM = 1.2039
chiFPi = 2.1805
chiMi = 1.8287

NN=[400,800,1600,3200,6400]

Q=[0.6395533134228835, 0.6243785781986531, 0.6193596717492259, 0.629870756227491, 0.6265782798426055]
CHIFP=[0.9946161355768357, 1.2820100697502377, 1.0953357791105385, 0.8906609561792096, 0.9769845158358322]
CHIFPi=[1.7955575799281445, 2.026707033911463, 2.8306169004077684, 3.265426595418967, 3.292505512589724]
CHIM=[1.4629906584493966, 1.8688984584193895, 1.6082031441841602, 1.2603883858783331, 1.3966626578593218]
CHIMi=[0.8676204376961049, 1.608644857812294, 2.3257430450768783, 2.4556019630438732, 2.3915406620696533]
errCHIFP=[0.043699390621309044, 0.1460282820990271, 0.07291586594909162, 0.07743321827487956, 0.06085608583901214]
errCHIFPi=[0.28224835802379117, 0.3827865152647233, 0.6533299813500267, 0.6912580679659036, 1.4322139869215613]
errCHIM=[0.06927414104364965, 0.23103196890979805, 0.10984414336237652, 0.10322890455626804, 0.08770362651944394]
errCHIMi=[0.05679204838332164, 0.24702665384749478, 0.3368807551371768, 0.6027780034477775, 0.7256104045382744]

plt.errorbar(NN,CHIFP,yerr=errCHIFP,c='blue',fmt='-o')
plt.errorbar(NN,CHIM,yerr=errCHIM,c='red',fmt='-o')
plt.errorbar(NN,CHIFPi,yerr=errCHIFPi,c='green',fmt='-o')
plt.errorbar(NN,CHIMi,yerr=errCHIMi,c='orange',fmt='-o')


plt.axhline(y=chiFP, color='blue', linestyle='--',label=r'$N\to\infty$')
plt.axhline(y=chiM, color='red', linestyle='--',label=r'$N\to\infty$')
plt.axhline(y=chiFPi, color='green', linestyle='--',label=r'$N\to\infty$')
plt.axhline(y=chiMi, color='orange', linestyle='--',label=r'$N\to\infty$')

plt.xscale('log')

plt.xlabel(r"$N$")
plt.ylabel(r"$\hat{\chi}$");
```

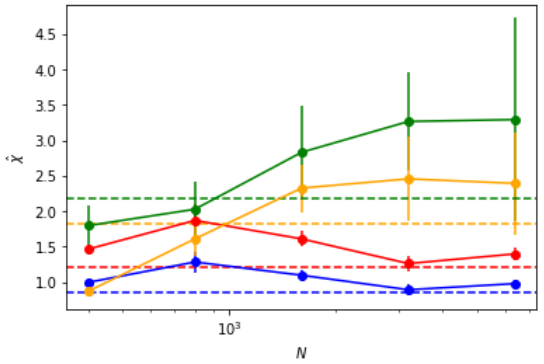
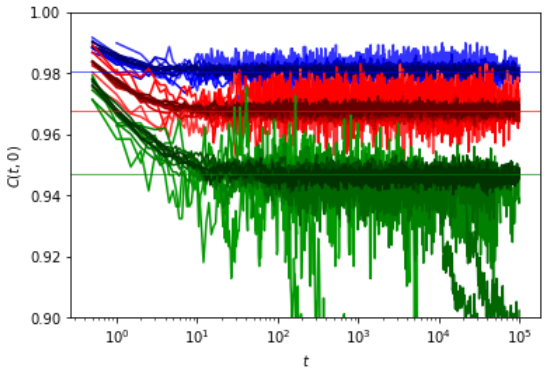


Fig.12

a)

In [19]:
```
%matplotlib inline

plt.xscale('log')

NN=[256,512,1024,2048,4096]

T=0.19
color = matplotlib.colors.ColorConverter.to_rgb("blue")
rgbs = [scale_lightness(color, scale) for scale in [.4, 0.6, 0.8, 1., 1.2]]

i=4
for N in NN:
    C=np.loadtxt("fig12_q_T"+str(T)+"_N"+str(N)+".dat")
    plt.plot(C[0],C[1],c=rgbs[i])
    plt.plot(C[0],C[2],c=rgbs[i])
    plt.plot(C[3],C[4],c=rgbs[i])
    i-=1

T=0.21
color = matplotlib.colors.ColorConverter.to_rgb("red")
rgbs = [scale_lightness(color, scale) for scale in [.4, 0.6, 0.8, 1., 1.2]]

i=4
for N in NN:
    C=np.loadtxt("fig12_q_T"+str(T)+"_N"+str(N)+".dat")
    plt.plot(C[0],C[1],c=rgbs[i])
    plt.plot(C[0],C[2],c=rgbs[i])
    plt.plot(C[3],C[4],c=rgbs[i])
    i-=1

T=0.23
color = matplotlib.colors.ColorConverter.to_rgb("green")
rgbs = [scale_lightness(color, scale) for scale in [.4, 0.6, 0.8, 1., 1.2]]

i=4
for N in NN:
    C=np.loadtxt("fig12_q_T"+str(T)+"_N"+str(N)+".dat")
    plt.plot(C[0],C[1],c=rgbs[i])
    plt.plot(C[0],C[2],c=rgbs[i])
    plt.plot(C[3],C[4],c=rgbs[i])
    i-=1

plt.axhline(y=0.980511, color='blue', linestyle='-',linewidth=0.6, label="T=0.19")
plt.axhline(y=0.967679, color='red', linestyle='-',linewidth=0.6, label="T=0.21")
plt.axhline(y=0.946886, color='green', linestyle='-',linewidth=0.6, label="T=0.23")
plt.ylim((0.9,1))

plt.xlabel(r"$t$")
plt.ylabel(r"$C(t,0)$");
```



b)

In [20]:

```
%matplotlib inline

def mean_Energy(P,T):
    b=1/T;
    return -((-1+np.sqrt(1+4*b*(-1+2*P+b)))/(4*b))

NN=[256,512,1024,2048,4096]
T=0.19
color = matplotlib.colors.ColorConverter.to_rgb("blue")
rgbs = [scale_lightness(color, scale) for scale in [.4, 0.6, 0.8, 1., 1.2]]

i=4
for N in NN:
    C=np.loadtxt("fig12_pE_T"+str(T)+"_N"+str(N)+".dat")
    plt.scatter(C[1],C[0],color=rgbs[i])
    i-=1

T=0.21
color = matplotlib.colors.ColorConverter.to_rgb("red")
rgbs = [scale_lightness(color, scale) for scale in [.4, 0.6, 0.8, 1., 1.2]]

i=4
for N in NN:
    C=np.loadtxt("fig12_pE_T"+str(T)+"_N"+str(N)+".dat")
    plt.scatter(C[1],C[0],color=rgbs[i])
    i-=1

T=0.23
color = matplotlib.colors.ColorConverter.to_rgb("green")
rgbs = [scale_lightness(color, scale) for scale in [.4, 0.6, 0.8, 1., 1.2]]

i=4
for N in NN:
    C=np.loadtxt("fig12_pE_T"+str(T)+"_N"+str(N)+".dat")
    plt.scatter(C[1],C[0],color=rgbs[i])
    i-=1

EMCT=mean_Energy(0.3,0.2465)
EK=mean_Energy(0.3,0.1803)

qMCT=0.896857
qK=0.985042

EN=pd.read_csv("fig12b_Eq.csv",header=None)[0]
OV=pd.read_csv("fig12b_Eq.csv",header=None)[1]
plt.plot(EN,OV,'-.k',label='mean-field')

#plt.axvline(-1/T/2, color='black', linestyle='-')
plt.axvline(EK, color='black', linestyle='--',linewidth=0.3)
plt.axvline(EMCT, color='black', linestyle='--',linewidth=0.3)

#plt.axhline(y=0.6339, color='black', linestyle='-')
plt.axhline(y=qK, color='black', linestyle='--',linewidth=0.3)
plt.axhline(y=qMCT, color='black', linestyle='--',linewidth=0.3)

plt.plot(-0.435466,0.980511,'x',color='w',markersize=10)
plt.plot(-0.428911,0.967679,'x',color='w',markersize=10)
plt.plot(-0.422402,0.946886,'x',color='w',markersize=10)

plt.ylim((0.88,1))
```

```
plt.xlabel(r"$\langle H \rangle/N$")
plt.ylabel("q");
```

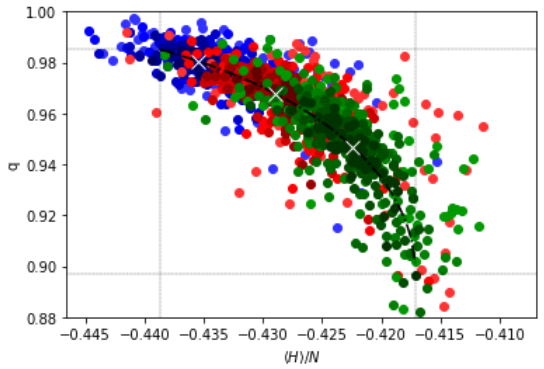


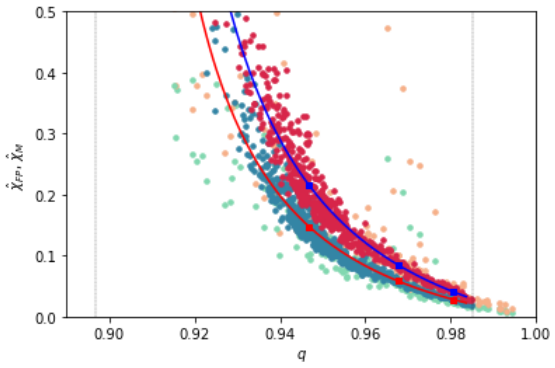
Fig.13

a)

In [21]:

```
%matplotlib inline

CL=[(0.12151575, 0.07117741, 0.12698763), (0.19093944, 0.1283047, 0.25077369), (0.23670785, 0.190297, 0.38504973), (0.25404842, 0.
CL2=[(0.1125315, 0.06742194, 0.17414848), (0.22741085, 0.10140876, 0.25340813), (0.34940101, 0.11863987, 0.3138355), (0.47114798,

qMCT=0.896857
qK=0.985042

ov19,chiFP19=0.980511, 0.0564897
ov21,chiFP21=0.967679, 0.116633
ov23,chiFP23=0.946886, 0.294383

ov19,chiM19=0.980511, 0.0820467
ov21,chiM21=0.967679, 0.169492
ov23,chiM23=0.946886, 0.429628

tQ=np.arange(0.501,1,0.001)
tchiM=[chi_M(q) for q in tQ]
tchiMB=[chi_MB(q,1/beta) for q in tQ]
tchiFP=[chi_FP(q) for q in tQ]
tchiFPB=[chi_FPB(q,1/beta) for q in tQ]

qq={}
cchiFP={}
cchiM={}

c=10
for N in [256,4096]:

    for T in [0.19,0.21,0.23]:

        qq=np.loadtxt("fig13_q_T"+str(T)+"_N"+str(N)+".dat")
        cchiFP=np.loadtxt("fig13_chiFP_T"+str(T)+"_N"+str(N)+".dat")
        cchiM=np.loadtxt("fig13_chiM_T"+str(T)+"_N"+str(N)+".dat")

        plt.scatter(qq,cchiFP,color=CL[c],s=12)
        plt.scatter(qq,cchiM,color=CL2[c],s=12)
    c-=4
CHIFP=pd.read_csv("fig13_FP.csv",header=None)[0]
CHIM=pd.read_csv("fig13_M.csv",header=None)[0]
OV=pd.read_csv("fig13_FP.csv",header=None)[1]
plt.plot(OV,CHIFP,color='red')
plt.plot(OV,CHIM,color='blue')

plt.plot(ov19,chiFP19/2,'s',color='red',markersize=4)
plt.plot(ov21,chiFP21/2,'s',color='red',markersize=4)
plt.plot(ov23,chiFP23/2,'s',color='red',markersize=4)

plt.plot(ov19,chiM19/2,'s',color='blue',markersize=4)
plt.plot(ov21,chiM21/2,'s',color='blue',markersize=4)
plt.plot(ov23,chiM23/2,'s',color='blue',markersize=4)


plt.axvline(x=qK, color='black', linestyle='--',linewidth=0.3)
plt.axvline(x=qMCT, color='black', linestyle='--',linewidth=0.3)

axes = plt.gca()
```

```
axes.set_xlim([0.89,1])
axes.set_ylim([0,0.5])

plt.xlabel(r"$q$")
plt.ylabel(r"$\hat{\chi}_{FP}$, $\hat{\chi}_{M}$");
```



b)

In [22]:

```
%matplotlib inline
plt.rcParams["figure.figsize"] = (8,3)

CL=sns.color_palette("mako", 10)
CL2=sns.color_palette("rocket", 10)

every=1
i=0

for Temp in [0.19,0.21,0.23]:

    if Temp==0.19:
        st = '19'
        ov = ov19
        chiFP = chiFP19/2
        chiM = chiM19/2
        labb = "(b)"

    if Temp==0.21:
        st = '21'
        ov = ov21
        chiFP = chiFP21/2
        chiM = chiM21/2
        labb = "(c)"

    if Temp==0.23:
        st = '23'
        ov = ov23
        chiFP = chiFP23/2
        chiM = chiM23/2
        labb = "(d)"

    ax = plt.subplot(1,3,i+1)
    ax.set_title("T="+str(Temp))

    ax.axvline(x=qK, color='black', linestyle='--',linewidth=0.3)

    CHIFP=pd.read_csv("fig13_FP.csv",header=None)[0]
    CHIM=pd.read_csv("fig13_M.csv",header=None)[0]
    OV=pd.read_csv("fig13_FP.csv",header=None)[1]
    ax.plot(OV,CHIFP,color='red')
    ax.plot(OV,CHIM,color='blue')

    CHIFP=pd.read_csv("fig13_FP"+st+".csv",header=None)[0]
    CHIM=pd.read_csv("fig13_M"+st+".csv",header=None)[0]
    OV=pd.read_csv("fig13_FP"+st+".csv",header=None)[1]
    plt.plot(OV,CHIFP/2,':k')
    plt.plot(OV,CHIM/2,'-.k')

    plt.plot(ov,chiFP,'s',color='red',markersize=4)
    plt.plot(ov,chiM,'s',color='blue',markersize=4)

    N=4096

    qq=np.loadtxt("fig13_q_T"+str(Temp)+"_N"+str(N)+".dat")
    cchiFP=np.loadtxt("fig13_chiFP_T"+str(Temp)+"_N"+str(N)+".dat")
    cchiM=np.loadtxt("fig13_chiM_T"+str(Temp)+"_N"+str(N)+".dat")

    ax.scatter(qq,cchiFP,color=CL[6],s=12)
```

```
    ax.scatter(qq,cchiM,color=CL2[6],s=12)

    axes = plt.gca()

    axes.set_xlim([0.93,0.99])
    axes.set_ylim([0,0.5])

    i+=1
```

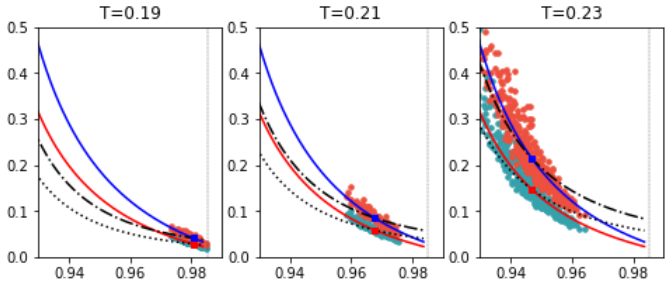


Fig.14

In [23]:
```
%matplotlib inline

TT=[0.19,0.21,0.23]
Q=[0.9792766213743865, 0.9673931583781937, 0.9453126371709056]
CHIFP=[0.03083305901698348, 0.06019265767444122, 0.1665210830610323]
CHIFPi=[0.045892281329222434, 0.09353916222892071, 0.30584941970839874]
CHIM=[0.0448158829737383, 0.08816749796625782, 0.23244863967627064]
CHIMi=[0.02850722363719875, 0.06682186815898843, 0.2422412271667983]
errCHIFP=[0.00071266322793561 17, 0.001470212975432196, 0.006016559120962704]
errCHIFPi=[0.0069434216809834415, 0.014462123940409069, 0.04812461477342985]
errCHIM=[0.0009832137102557903, 0.0021339430125366815, 0.008651620850258262]
errCHIMi=[0.0043184456415275956, 0.008165969313500355, 0.04020913516711039]


plt.errorbar(TT,CHIFP,yerr=errCHIFP,c='blue',fmt='-o')
plt.errorbar(TT,CHIM,yerr=errCHIM,c='red',fmt='-o')
plt.errorbar(TT,CHIFPi,yerr=errCHIFPi,c='green',fmt='-o')
plt.errorbar(TT,CHIMi,yerr=errCHIMi,c='orange',fmt='-o')

chiT=pd.read_csv("fig14_FP.csv",header=None)[0]
chiD=pd.read_csv("fig14_FPi.csv",header=None)[0]
chiH=pd.read_csv("fig14_M.csv",header=None)[0]
chiV=pd.read_csv("fig14_Mi.csv",header=None)[0]
Temp=pd.read_csv("fig14_FP.csv",header=None)[2]
plt.plot(Temp,chiT,'--',color='blue',label=r'$N\to\infty$')
plt.plot(Temp,chiD,'--',color='red',label=r'$N\to\infty$')
plt.plot(Temp,chiH,'--',color='green',label=r'$N\to\infty$')
plt.plot(Temp,chiV,'--',color='orange',label=r'$N\to\infty$')

plt.ylim((0.,0.4))
plt.xlim((0.18,0.24))

plt.xlabel(r"$T$")
plt.ylabel(r"$\hat{\chi}$");
```

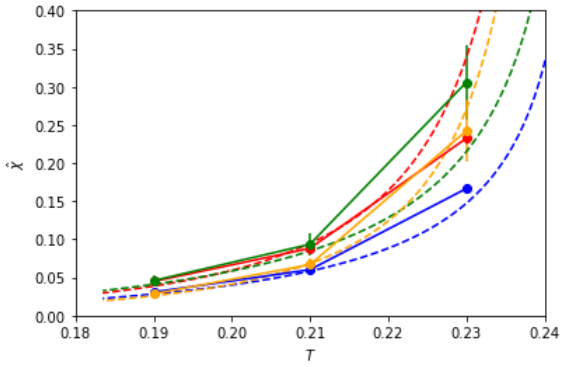


Fig.15

In [24]:
```
%matplotlib inline
```

```
plt.ylim([1,20])

plt.xscale('log')
plt.yscale('log')

CLR=plt.rcParams['axes.prop_cycle'].by_key()['color']
i=0

C=np.loadtxt("fig15_D"+str(4)+".txt")
plt.plot(C[1],C[2],color=CLR[0],marker='x',zorder=4)
C=np.loadtxt("fig15_D"+str(8)+".txt")
plt.plot(C[1],C[2],color=CLR[1],marker='o',zorder=4)
C=np.loadtxt("fig15_D"+str(16)+".txt")
plt.plot(C[1],C[2],color=CLR[2],marker='s',zorder=4)

C=np.loadtxt("fig15_Dinf.txt")
plt.scatter(C[1],C[2],color='black',s=45,marker='s',zorder=4)

C=np.loadtxt("fig15_Phi.txt")
plt.plot(C[0],C[1],label=r"$d\to\infty$",color='black')

plt.axvline(x=2.40339, color='black', linestyle='dashed')
plt.axhline(y=2*1.15335*2.40339**2, color='black', linestyle='dashed')


plt.ylabel(r"$\hat{\Delta}\hat{\varphi}^2$")
plt.xlabel(r"$\hat{\varphi}$");
```

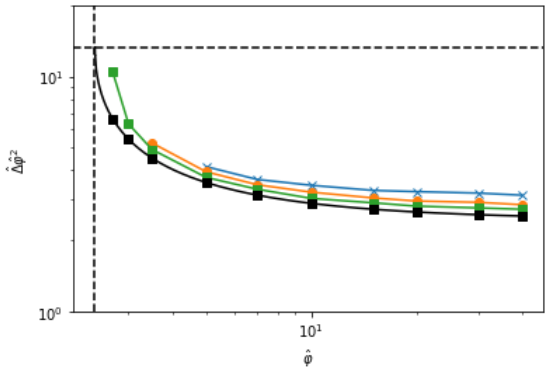


Fig.16

In [25]:
```
%matplotlib inline

CLR=plt.rcParams['axes.prop_cycle'].by_key()['color']

C=np.loadtxt("fig16_all.dat")

plt.rcParams["figure.figsize"] = (7,3)


ax = plt.subplot(1,2,1)
ax.set_xscale('log')
ax.set_xlim([0.0002, 2*1.15335+0.5])
ax.plot(C[0],C[5]*C[0]**2,label=r"$m_1=\lambda_R$",color=CLR[1])
ax.plot(C[0],C[6]*C[0]**2,label=r"$m_2$",color=CLR[2])
ax.plot(C[0],C[7]*C[0]**2,label=r"$m_3$",color=CLR[3])

ax.set_ylabel(r"$m \hat{\Delta}^2$")
ax.set_xlabel(r"$\hat{\Delta}$")

ax.axvline(x=2*1.15335, color='black', linestyle='dashed')
ax.axhline(y=0, color='black', linestyle='dashed')

ax = plt.subplot(1,2,2)
ax.set_xscale('log')
ax.set_yscale('log')

ax.set_xlim([0.0002, 2*1.15335+0.5])
ax.set_ylim([0.8,200])

ax.plot(C[0],C[2],label=r"$\hat{\chi}^{FP}_{intra}$",color=CLR[0])
ax.plot(C[0],C[1],label=r"$\hat{\chi}^{M}_{intra}$",color=CLR[3])
ax.plot(C[0],C[4],label=r"$\hat{\chi}^{FP}_{sample}$",color=CLR[2])
ax.plot(C[0],C[3],label=r"$\hat{\chi}^{M}_{sample}$",color=CLR[1])

ax.set_ylabel(r"$\hat{\chi}/\hat{\Delta}^2$")
```

```
ax.set_xlabel(r"$\hat{\Delta}$")
ax.yaxis.set_label_position("right")
```

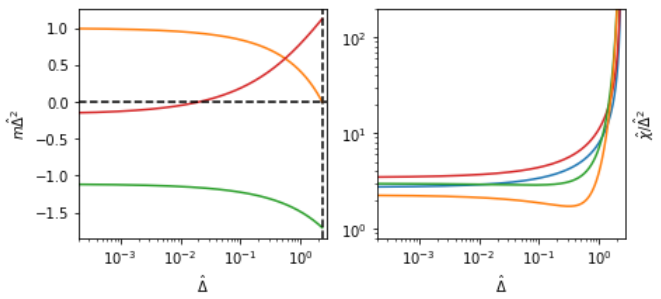
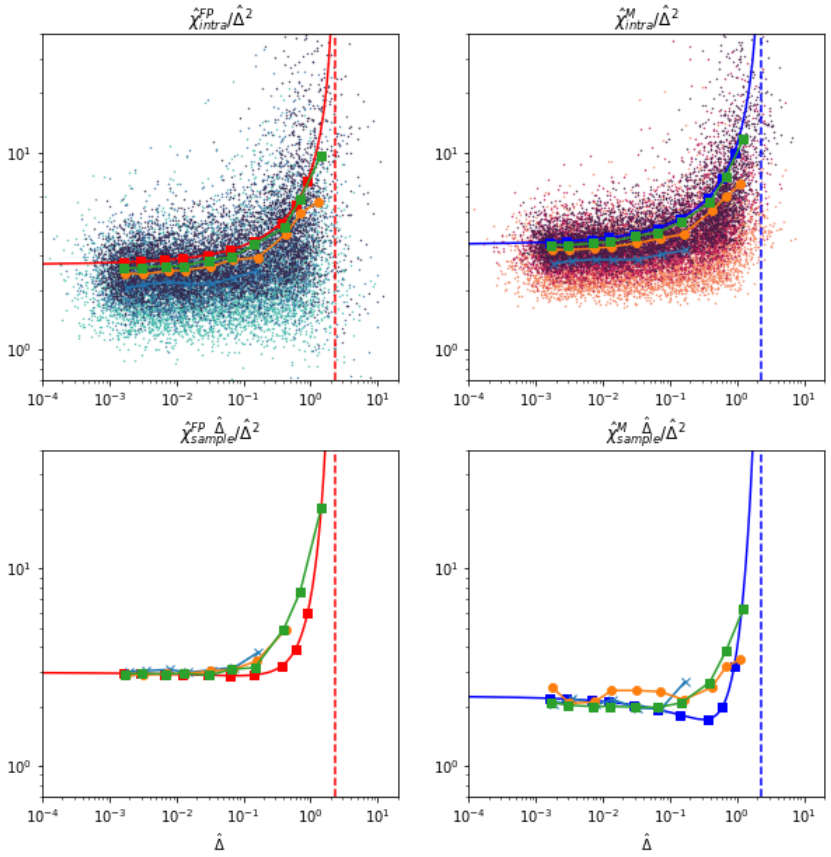


Fig.17

a,b,c,d)

In [26]:
```
%matplotlib inline
plt.rcParams["figure.figsize"] = (10,10)

CL=sns.color_palette("mako", 10)
CL2=sns.color_palette("rocket", 10)

ax = plt.subplot(2,2,1)
ax.set_title(r"$\hat{\chi}^{FP}_{intra}/\hat{\Delta}^2$")

ax.set_xscale('log')
ax.set_yscale('log')
ax.set_xlim([0.0001, 20])
ax.set_ylim([0.7,40])

i=0

c=7
for D in [4,8,16]:
    for Ph in [2.7,3,3.5,5,7,10,15,20,30,40]:

        C=np.loadtxt("fig17a_D"+str(D)+"_Ph"+str(Ph)+".txt")
        ax.scatter(C[0],C[1],color=CL[c],s=0.2,zorder=2)
    c-=3


C=np.loadtxt("fig17a_av_Dinf.txt")
ax.scatter(C[0],C[1],color='red',s=45,marker='s',zorder=4)

C=np.loadtxt("fig17a_av_D4.txt")
ax.plot(C[0],C[1],color=CLR[0],marker='x',label="d=4",zorder=4)

C=np.loadtxt("fig17a_av_D8.txt")
ax.plot(C[0],C[1],color=CLR[1],marker='o',label="d=8",zorder=4)

C=np.loadtxt("fig17a_av_D16.txt")
ax.plot(C[0],C[1],color=CLR[2],marker='s',label="d=16",zorder=4)

C=np.loadtxt("fig17a_all_Dinf.txt")
ax.plot(C[0],C[1],color='red',label=r"$d\to\infty$")

ax.axvline(x=2*1.15335, color='r', linestyle='dashed')

ax.set_xlabel(r"$\hat{\Delta}$")


ax = plt.subplot(2,2,2)
ax.set_title(r"$\hat{\chi}^{M}_{intra}/\hat{\Delta}^2$")
ax.set_xscale('log')
ax.set_yscale('log')
ax.set_xlim([0.0001, 20])
ax.set_ylim([0.7,40])

i=0

c=7
for D in [4,8,16]:
    for Ph in [2.7,3,3.5,5,7,10,15,20,30,40]:

        C=np.loadtxt("fig17b_D"+str(D)+"_Ph"+str(Ph)+".txt")
        ax.scatter(C[0],C[1],color=CL2[c],s=0.2,zorder=2)
```

```
    c-=3


C=np.loadtxt("fig17b_av_Dinf.txt")
ax.scatter(C[0],C[1],color='blue',s=45,marker='s',zorder=4)

C=np.loadtxt("fig17b_av_D4.txt")
ax.plot(C[0],C[1],color=CLR[0],marker='x',label="d=4",zorder=4)

C=np.loadtxt("fig17b_av_D8.txt")
ax.plot(C[0],C[1],color=CLR[1],marker='o',label="d=8",zorder=4)

C=np.loadtxt("fig17b_av_D16.txt")
ax.plot(C[0],C[1],color=CLR[2],marker='s',label="d=16",zorder=4)

C=np.loadtxt("fig17b_all_Dinf.txt")
ax.plot(C[0],C[1],color='blue',label=r"$d\to\infty$")

ax.axvline(x=2*1.15335, color='b', linestyle='dashed')

ax.set_xlabel(r"$\hat{\Delta}$")


ax = plt.subplot(2,2,3)
ax.set_title(r"$\hat{\chi}^{FP}_{sample}/\hat{\Delta}^2$")
ax.set_xscale('log')
ax.set_yscale('log')
ax.set_xlim([0.0001, 20])
ax.set_ylim([0.7,40])

C=np.loadtxt("fig17c_av_Dinf.txt")
ax.scatter(C[0],C[1],color='red',s=45,marker='s',zorder=4)

C=np.loadtxt("fig17c_av_D4.txt")
ax.plot(C[0],C[1],color=CLR[0],marker='x',label="d=4",zorder=4)

C=np.loadtxt("fig17c_av_D8.txt")
ax.plot(C[0],C[1],color=CLR[1],marker='o',label="d=8",zorder=4)

C=np.loadtxt("fig17c_av_D16.txt")
ax.plot(C[0],C[1],color=CLR[2],marker='s',label="d=16",zorder=4)

C=np.loadtxt("fig17c_all_Dinf.txt")
ax.plot(C[0],C[1],color='red',label=r"$d\to\infty$")

ax.axvline(x=2*1.15335, color='r', linestyle='dashed')

ax.set_xlabel(r"$\hat{\Delta}$")

ax = plt.subplot(2,2,4)
ax.set_title(r"$\hat{\chi}^{M}_{sample}/\hat{\Delta}^2$")
ax.set_xscale('log')
ax.set_yscale('log')
ax.set_xlim([0.0001, 20])
ax.set_ylim([0.7,40])

C=np.loadtxt("fig17d_av_Dinf.txt")
ax.scatter(C[0],C[1],color='blue',s=45,marker='s',zorder=4)

C=np.loadtxt("fig17d_av_D4.txt")
ax.plot(C[0],C[1],color=CLR[0],marker='x',label="d=4",zorder=4)

C=np.loadtxt("fig17d_av_D8.txt")
ax.plot(C[0],C[1],color=CLR[1],marker='o',label="d=8",zorder=4)

C=np.loadtxt("fig17d_av_D16.txt")
ax.plot(C[0],C[1],color=CLR[2],marker='s',label="d=16",zorder=4)

C=np.loadtxt("fig17d_all_Dinf.txt")
ax.plot(C[0],C[1],color='blue',label=r"$d\to\infty$")

ax.axvline(x=2*1.15335, color='b', linestyle='dashed');

ax.set_xlabel(r"$\hat{\Delta}$");
```



e)

In [27]:

```
%matplotlib inline
plt.rcParams["figure.figsize"] = (5,4)

CLR=plt.rcParams['axes.prop_cycle'].by_key()['color']

CL=sns.light_palette("seagreen", 10)

i=0

c=9

for D in [4,8,16]:


    #plt.set_title(r"$\hat{\varphi}$="+str(Ph))

    plt.xscale('log')
    #plt.yscale('log')
    plt.ylim([-1,3])
    plt.xlim([0.001,5])

    plt.xlabel(r"$\hat{\Delta}$")
    plt.ylabel(r"$\lambda_{R}\hat{\Delta}^2$")


    for Ph in [40,30,20,15,10,7,5,3.5,3,2.7]:

        C=np.loadtxt("fig17e_D"+str(D)+"_Ph"+str(Ph)+".txt")

        plt.scatter(C[1],C[0],color=CL[c],s=1,zorder=2)

    plt.grid()

    c-=3


C=np.loadtxt("fig17e_all_Dinf.txt")
plt.plot(C[0],C[1],'r-')

C=np.loadtxt("fig17e_av_D4.txt")
plt.plot(C[0],C[1],color=CLR[0],marker='x',label="d=4",zorder=4)

C=np.loadtxt("fig17e_av_D8.txt")
```

```
plt.plot(C[0],C[1],color=CLR[1],marker='o',label="d=8",zorder=4)

C=np.loadtxt("fig17e_av_D16.txt")
plt.plot(C[0],C[1],color=CLR[2],marker='s',label="d=16",zorder=4)

C=np.loadtxt("fig17e_av_Dinf.txt")
plt.plot(C[0],C[1],'--ro',label=r"$d \to \infty$")

plt.axhline(y=0, color='r', linestyle='dashed')

plt.axvline(x=2*1.15335, color='red', linestyle='dashed');
```

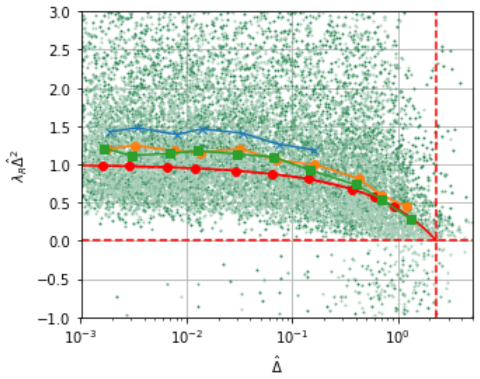


Fig.18

In [28]:
```
%matplotlib inline
plt.rcParams["figure.figsize"] = (10,10)

CL=sns.color_palette("mako", 10)
CL2=sns.color_palette("rocket", 10)

i=0

[aa,am,bb,bm]=np.loadtxt("fig18_av.txt")

for Ph in [30,20,15,10,7,5,3.5,3,2.7]:
    ax = plt.subplot(3,3,i+1)
    ax.set_title(r"$\hat{\varphi}$="+str(Ph))

    ax.set_xscale('log')
    ax.set_yscale('log')
    ax.set_xlim([0.00005, 20])
    ax.set_ylim([0.1,100])


    if i%3==0:
        ax.set_ylabel(r"$\hat{\chi}_{th}/\hat{\Delta}^2$")
    if i>5:
        ax.set_xlabel(r"$\hat{\Delta}$")


    c=7
    for D in [4,8,12,16,20]:

            C=np.loadtxt("fig18_D"+str(D)+"_Ph"+str(Ph)+".txt")

            plt.scatter(C[1],C[0],color=CL[c],s=0.5,zorder=2)

            c-=4

    plt.grid()

    plt.scatter(bb[i],aa[i],color='red',s=45,marker='s',zorder=4)

    plt.scatter(bm[i],am[i],color='orange',s=45,marker='x',label="d=16",zorder=4)

    i+=1

    plt.axvline(x=2*1.15335, color='r', linestyle='dashed')
```

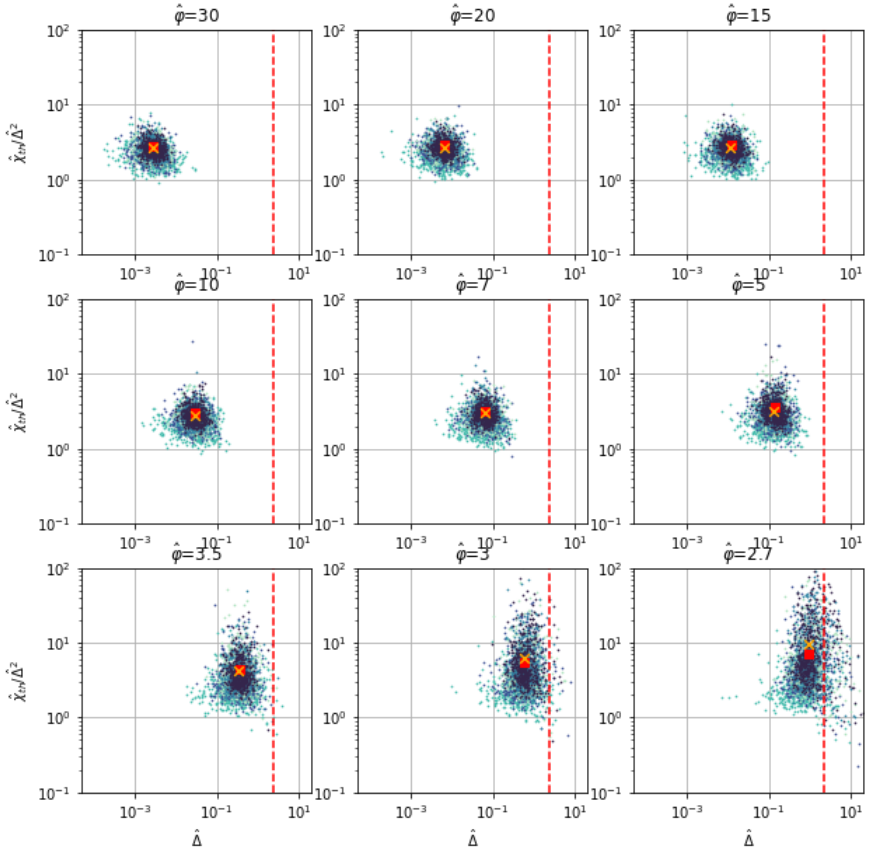
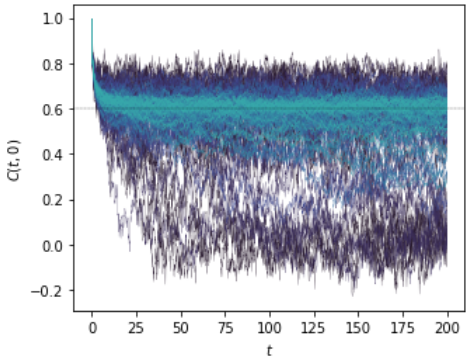


Fig.19

a)

In [29]:
```
plt.rcParams["figure.figsize"] = (5,4)

CL=sns.color_palette("mako", 8)

c=0
for N in [100,200,400,800,1600,3200]:
    for i in range(20,40):
        C=np.loadtxt("fig19a_N"+str(N)+"_S"+str(i)+".dat")
        plt.plot(C[0],C[1],lw=0.2,color=CL[c])
    c+=1;

plt.axhline(y=0.6, color='black', linestyle='--',linewidth=0.3)
plt.ylabel(r"$C(t,0)$")
plt.xlabel(r"$t$");
```



b)

In [30]:
```
plt.rcParams["figure.figsize"] = (5,4)

CL=sns.color_palette("mako", 8)
```

```
def Eeq(q):
    return -np.sqrt(1/6.)/np.sqrt(q - q**2)

EMCT=-1/np.sqrt(3/8)/2
EK=-1/0.5861/2

qMCT=0.5
qK=0.644883

qq=np.arange(0.5,0.644883,0.001)
Eq=[Eeq(q) for q in qq]

T=0.6
c=0
for N in [100,200,400,800,1600,3200]:
    Q=np.loadtxt("fig19b_Q_N"+str(N)+".dat")
    E=np.loadtxt("fig19b_E_N"+str(N)+".dat")
    plt.scatter(E,Q,color=CL[c],label="N="+str(N))
    c+=1


plt.plot(Eq,qq,'-.k',label='mean-field')

plt.axvline(EK, color='black', linestyle='--',linewidth=0.3)
plt.axvline(EMCT, color='black', linestyle='--',linewidth=0.3)

#plt.axhline(y=0.6339, color='black', linestyle='-')
plt.axhline(y=qK, color='black', linestyle='--',linewidth=0.3)
plt.axhline(y=qMCT, color='black', linestyle='--',linewidth=0.3)

plt.plot(-1/T/2,0.6,'x',color='red',markersize=10,label='expected at T=0.6')


plt.xlabel(r"$\langle H \rangle/N$")
plt.ylabel(r"$q$");
```

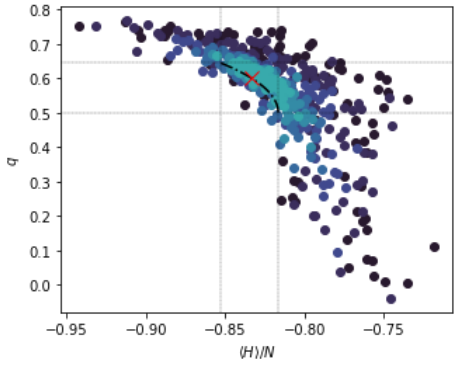


Fig.20

In [31]:
```
%matplotlib inline

CL=[(0.12151575, 0.07117741, 0.12698763), (0.19093944, 0.1283047, 0.25077369), (0.23670785, 0.190297, 0.38504973), (0.25404842, 0.
CL2=[(0.1125315, 0.06742194, 0.17414848), (0.22741085, 0.10140876, 0.25340813), (0.34940101, 0.11863987, 0.3138355), (0.47114798,


def chi_M(q):
    return ((-1 + q)**2 *(-1 + 5*q))/(1 + q*(-5 + 6*q))

def chi_FP(q):
    return ((-1 + q)**2 *(-1 + 4*q))/(1 + q*(-5 + 6*q))

def chi_MB(q,T):
    beta=1./T;
    return (2 *(1 + 3*q - 3*(-1 + q)**2 *q*beta**2))/((2/(-1 + q)**2 - 6*q*beta**2)*(1 + q - 3*(-1 + q)**2 *q*beta**2))

def chi_FPB(q,T):
    beta=1./T;
    return (2 *(1 + 2*q - 3*(-1 + q)**2 *q*beta**2))/((2/(-1 + q)**2 - 6*q*beta**2)*(1 + q - 3*(-1 + q)**2 *q*beta**2))

T=0.6
tQ=np.arange(0.535,0.75,0.0001)
tchiM=[chi_M(q) for q in tQ]
tchiMB=[chi_MB(q,T) for q in tQ]
tchiFP=[chi_FP(q) for q in tQ]
tchiFPB=[chi_FPB(q,T) for q in tQ]
```

```
c=10
for N in [400,6400]:
        C=np.loadtxt("fig20_all_N"+str(N)+".dat")

        plt.errorbar(C[0],C[2],yerr=C[4],xerr=C[5],ls = "None",color=CL2[c],zorder=1)#,s=12)
        plt.errorbar(C[0],C[1],yerr=C[3],xerr=C[5],ls = "None",color=CL[c],zorder=1)#,s=12)
        plt.scatter(C[0],C[2],color=CL2[c],s=10,zorder=2)
        plt.scatter(C[0],C[1],color=CL[c],s=10,zorder=2)

        c-=4


plt.plot(tQ,tchiMB,'-.k',label=r'$\hat{\chi}_{M}$, $N\to\infty$',zorder=3)
plt.plot(tQ,tchiFPB,':k',label=r'$\hat{\chi}_{FP}$, $N\to\infty$',zorder=3)

plt.plot(tQ,tchiM,'-',color='blue',zorder=3,linewidth=1)
plt.plot(tQ,tchiFP,'-',color='red',zorder=3,linewidth=1)

axes = plt.gca()
axes.set_ylim([0,4])
axes.set_xlim([0.49,0.75]);
```

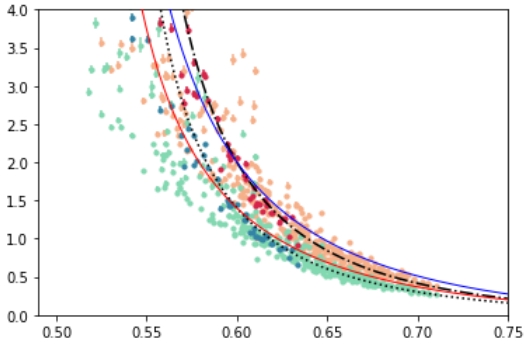
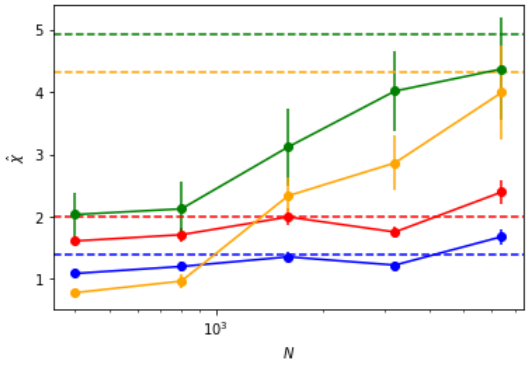
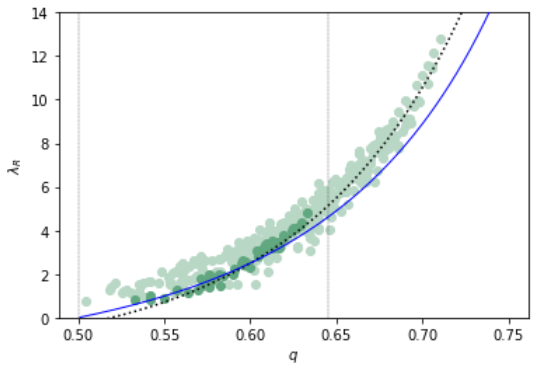


Fig.21

Left

In [32]:
```
chiFP = 1.4
chiM = 2.
chiFPi = 4.92
chiMi = 4.32

NN=[400,800,1600,3200,6400]

Q=[0.6249663743484504, 0.6107876762770563, 0.597544467555147, 0.6002979626160989, 0.5926857154602924]
CHIFP=[1.086286602798387, 1.1987910345843906, 1.3535068992484978, 1.2206758221400122, 1.677099137248427]
CHIFPi=[2.0320482814322474, 2.1234006440095166, 3.116579499890713, 4.014072187895014, 4.368703750243864]
CHIM=[1.607717404482736, 1.7084733523507605, 1.9939562817049232, 1.7518147941010906, 2.393901265321116]
CHIMi=[0.7775881309844566, 0.9637043408746779, 2.3331148473289813, 2.861228440882177, 3.991040460402569]
errCHIFP=[0.0394712719103694, 0.07826347759025222, 0.0859545651952067, 0.06581588202636864, 0.128887021638102]
errCHIFPi=[0.3467410843484087, 0.42792795971860287, 0.6224482690558387, 0.6395202089200717, 0.8168406985315849]
errCHIM=[0.06391759109821617, 0.11342394847001853, 0.1305495854202818, 0.09821605398943192, 0.18323749439671516]
errCHIMi=[0.0430565469466054, 0.11853841073244241, 0.30343837791184386, 0.43250218684929087, 0.7521561897440705]

plt.errorbar(NN,CHIFP,yerr=errCHIFP,c='blue',fmt='-o')
plt.errorbar(NN,CHIM,yerr=errCHIM,c='red',fmt='-o')
plt.errorbar(NN,CHIFPi,yerr=errCHIFPi,c='green',fmt='-o')
plt.errorbar(NN,CHIMi,yerr=errCHIMi,c='orange',fmt='-o')


plt.axhline(y=chiFP, color='blue', linestyle='--',label=r'$N\to\infty$')
plt.axhline(y=chiM, color='red', linestyle='--',label=r'$N\to\infty$')
plt.axhline(y=chiFPi, color='green', linestyle='--',label=r'$N\to\infty$')
plt.axhline(y=chiMi, color='orange', linestyle='--',label=r'$N\to\infty$');

plt.xscale('log')

plt.xlabel(r"$N$")
plt.ylabel(r"$\hat{\chi}$");
```



Right

In [33]:
```
%matplotlib inline

CL=[(0.921460863132993, 0.9512120106378944, 0.9302557793948234), (0.8546193719792059, 0.9145821230194208, 0.8771231358554057), (0.
qMCT=0.5
qK=0.644883

def M1B(q,T):
    beta=1./T;
    return 2./(1-q)**2 - 6*q*beta**2

def M1(q):
    beta=1./T;
    return 2./(1-q)**2 - 4/(1-q)

tQ=np.arange(0.501,0.75,0.001)
m1b=[M1B(q,T) for q in tQ]
m1=[M1(q) for q in tQ]

T=0.6


c=3
for N in [400,6400]:
        C=np.loadtxt("fig20_all_N"+str(N)+".dat")

        plt.errorbar(C[0],-2/(C[2]-2*C[1]),yerr=-2*(C[3]+2*C[4])/(C[1]-2*C[2])**2,xerr=C[5],ls = "None",color=CL[c],zorder=1)#,s=1
        plt.scatter(C[0],-2/(C[2]-2*C[1]),color=CL[c],label="N="+str(N),zorder=2)

        c+=5

plt.plot(tQ,np.array(m1b),':k',label=r'$\lambda_R$ $N\to\infty$',zorder=3)
clr='blue'
plt.plot(tQ,np.array(m1),'-',color=clr,label=r'$\lambda_R(q_{EA})$ $N\to\infty$',zorder=3,linewidth=1)

plt.axvline(x=qK, color='black', linestyle='--',linewidth=0.3)
plt.axvline(x=qMCT, color='black', linestyle='--',linewidth=0.3)

axes = plt.gca()
axes.set_ylim([0,14])

plt.xlabel(r"$q$")
plt.ylabel(r"$\lambda_{R}$");
```