Algorithm 12 Approximately counting the number of instances of K_r

procedure CountClique$(n, r, \lambda, \epsilon, m, L_r)$
  $\gamma \leftarrow \epsilon/(8r \cdot r!)$, $\beta \leftarrow 1/(6r)$
  for each $t \in [2, r-1]$, set $\tau_t \leftarrow \frac{r^{4r}}{\beta^r \cdot \gamma^2} \cdot \lambda^{r-t}$;
    $\tau_r \leftarrow 1$; $\tau_1 \leftarrow \frac{r^{4r}}{\gamma^2} \cdot \min\{\lambda^{r-1}, L_r^{(r-1)/r}\}$,
    $\vec{\tau} \leftarrow \{\tau_1, \ldots, \tau_r\}$
  for all $j = 1, \ldots, q = \Theta(\log(n))$ **do**
    Invoke ApproxCliques$(n, r, \lambda, \epsilon, L_r, m, \vec{\tau})$.
    Let χ_j be the returned value.
  Let $\hat{n}_r$ be the the median value of $\chi_1, \cdots, \chi_q$
  return $\hat{n}_r$.

Algorithm 13 Approximately counting the number of instances of K_r

procedure ApproxClique$(n, r, \lambda, \epsilon, L_r, m, \vec{\tau})$
  set $\mathcal{R}_0 \leftarrow V$, $\mathrm{dg}(\mathcal{R}_0) \leftarrow n$, $\tilde{\omega}_0 = (1-\epsilon/2)L_r$, $\beta \leftarrow 1/(18r)$
    and $\gamma \leftarrow \epsilon/(2r)$
  sample $s_1 = \lceil \frac{n\tau_1}{\tilde{\omega}_0} \cdot \frac{3\ln(2/\beta)}{\gamma^2} \rceil$ vertices u.a.r
    and let $\mathcal{R}_1$ be the chosen multiset
  for all $t = 1, \ldots, r-1$ **do**
    Compute $\mathrm{dg}(\mathcal{R}_t)$ and set $\tilde{\omega}_t = (1-\gamma)\frac{\tilde{\omega}_{t-1}}{\mathrm{dg}(\mathcal{R}_{t-1})} \cdot s_t$
      and $s_{t+1} \leftarrow \lceil \frac{\mathrm{dg}(\mathcal{R}_t)\tau_{t+1}}{\tilde{\omega}_t} \cdot \frac{3\ln(2/\beta)}{\gamma^2} \rceil$
    If $s_{t+1} > \frac{4m\lambda^{t-1} \cdot \tau_{t+1}}{L_r} \cdot \frac{(r!)^2 \cdot 3\ln(2/\beta)}{\beta^t \cdot \gamma^2}$ then **abort**
    Invoke SampleASet$(t, \mathcal{R}_t, s_{t+1})$
      and let $\mathcal{R}_{t+1}$ be the returned multiset.
  $\hat{n}_r = \frac{n \cdot \mathrm{dg}(\mathcal{R}_1) \cdot \cdots \cdot \mathrm{dg}(\mathcal{R}_{r-1})}{s_1 \cdot \cdots \cdot s_r} \sum_{\vec{C} \in \mathcal{R}_r} \text{IsAssigned}(\vec{C}, r, \lambda, \epsilon, L_r, m, \vec{\tau})$
  return $\hat{n}_r$.

Algorithm 14 Sampling a set of ordered $(t+1)$-cliques

procedure SampleASet$(t, \mathcal{R}_t, s_{t+1})$
  Compute $\mathrm{dg}(\mathcal{R}_t)$ and set up a data structure
    to sample each $\vec{T} \in \mathcal{R}_t$ with probability $\mathrm{dg}(\vec{T})/\mathrm{dg}(\mathcal{R}_t)$
  initialize $\mathcal{R}_t = \emptyset$
  for all $\ell = 1, \ldots, s_{t+1}$ **do**
    invoke the above to generate $\vec{T}_\ell$
    find the minimum degree vertex u of $\vec{T}_\ell$
    sample a random neighbor w of u
    If the $(t+1)$-tuple $(\vec{T}_\ell, w)$ is an ordered $(t+1)$-clique, add it to $\mathcal{R}_{t+1}$
  return $\mathcal{R}_{t+1}$.

Algorithm 15 Check if an ordered r-clique $\vec{C}$ is assigned to the un-ordered clique

procedure IsAssigned$(\vec{C}, r, \lambda, \epsilon, L_r, m, \vec{\tau})$
  Let C be the un-ordered clique corresponding to $\vec{C}$
  for all ordered r-clique $\vec{C}'$
    whose un-ordered clique equals C **do**
    for all prefix $\vec{C}'_{\leq t}$, $t \in [r-1]$ **do**
      invoke IsActive$(t, \vec{C}'_{\leq t}, r, \lambda, \epsilon, L_r, m, \vec{\tau})$
        and if it returns **Non-Active**
        then **abort** and **return** 0
  if $\vec{C}$ is the lexicographically first ordered r-clique
    in the above set of active ordered cliques **then**
    return 1
  else
    return 0.

Algorithm 16 Check if an ordered i-clique $\vec{I}$ is active

procedure IsActive$(i, \vec{I}, r, \lambda, \epsilon, L_r, m, \vec{\tau})$
  for all $\ell = 1, \cdots, q = 12\ln(n^{r+10})$ **do**
    set $\mathcal{R}_i = \{\vec{I}\}$, $\tilde{\omega}_i = (1-\epsilon/2)\tau_i$, $\beta = 1/(6r)$,
      and $\gamma = \epsilon/(8r \cdot r!)$.
    for all $t = i, \ldots, r-1$ **do**
      Compute $\mathrm{dg}(\mathcal{R}_t)$
      for $t > i$, set $\tilde{\omega}_t = (1-\gamma)\frac{\tilde{\omega}_{t-1} \cdot s_t}{\mathrm{dg}(\mathcal{R}_{t-1})}$
        and $s_{t+1} = \frac{\mathrm{dg}(\mathcal{R}_t) \cdot \tau_{t+1}}{\tilde{\omega}_t} \cdot \frac{3\ln(2/\beta)}{\gamma^2}$.
      if $s_{t+1} > \frac{2m\lambda^{t-1} \cdot \tau_{t+1}}{L_r} \cdot \frac{12\ln(1/\beta)}{\beta^r \cdot \gamma^3}$,
        then set $\chi_\ell = 0$ and continue to next ℓ.
      invoke SampleASet$(t, \mathcal{R}_t, s_{t+1})$
        and let $\mathcal{R}_{t+1}$ be the returned multiset.
    set $\hat{c}_r(\vec{I}) = \frac{\mathrm{dg}(\mathcal{R}_i) \cdot \cdots \cdot \mathrm{dg}(\mathcal{R}_{r-1})}{s_{i+1} \cdot \cdots \cdot s_r} \cdot |\mathcal{R}_r|$.
    if $\hat{c}_r(\vec{I}) \leq \frac{\tau_i}{4}$, then $\chi_\ell = 1$, otherwise $\chi_\ell = 0$.
  if $\sum_{\ell=1}^{q} \chi_\ell \geq q/2$ **then**
    return Active
  else
    return Non-Active.

D MISSING ALGORITHMS FROM SECTION 5.2

Algorithm 17 Check if an ordered r-clique $\vec{C}$ is assigned to its unordered clique

procedure StrIsAssigned($\vec{C}, r, \lambda, \epsilon, m, \vec{\tau}$)
  $C \leftarrow$ the unordered clique corresponding to $\vec{C}$
  initialize dictionary A
  parallel for all ordered r-cliques $\vec{C}'$ isomorphic to C
    parallel for all $t \in \{2, \ldots, r\}$
      passes 1 to 2t-1; input: $t, \vec{C}'_{\leq t}, r, \lambda, \epsilon, m, \vec{\tau}$
        $A[\vec{C}'_{\leq t}] \leftarrow$ StrAct($t, \vec{C}'_{\leq t}, r, \lambda, \epsilon, m, \vec{\tau}$)
  for ordered r-clique $\vec{C}'$ of C **do**
    if $\vec{C}'$ "lex. <" $\vec{C}$
      $\wedge \forall t \in \{2, \ldots, r\} : A[\vec{C}'_{\leq t}] =$ **active then**
      return 0
  return 1

Algorithm 18 Check if an ordered i-clique $\vec{I}$ is active

procedure StrAct($i, \vec{I}, r, \lambda, \epsilon, m, \vec{\tau}$)
  parallel for all $\ell \leftarrow 1, \ldots, q = 12 \ln(n^{r+10}/\delta)$
    $\mathcal{R}_i \leftarrow \{\vec{I}\}$, $\tilde{\omega}_i \leftarrow (1-\epsilon/2)\tau_i$, $\beta \leftarrow 1/(6r)$,
      and $\gamma \leftarrow \epsilon/(8r \cdot r!)$
    pass 1; input: $\mathcal{R}_i$
      construct $d[\mathcal{R}_i]$ ▷ f_2
    for all $t \leftarrow i, \ldots, r-1$ **do**
      $\mathrm{dg}(\mathcal{R}_t) \leftarrow \sum_{\vec{T} \in \mathcal{R}_t} \mathrm{dg}(\vec{T}) = \sum_{\vec{T} \in \mathcal{R}_t} \min_{v \in \vec{T}} d[\mathcal{R}_t]_v$
      for $t > i$, set $\tilde{\omega}_t = (1-\gamma)\frac{\tilde{\omega}_{t-1} \cdot s_t}{\mathrm{dg}(\mathcal{R}_{t-1})}$
        and $s_{t+1} = \frac{\mathrm{dg}(\mathcal{R}_t) \cdot \tau_{t+1}}{\tilde{\omega}_t} \cdot \frac{3 \ln(2/\beta)}{\gamma^2}$
      if $s_{t+1} > \frac{2m\lambda^{t-1} \cdot \tau_{t+1}}{\#K_r} \cdot \frac{12 \ln(1/\beta)}{\beta^r \cdot \gamma^3}$ **then**
        $\chi_\ell \leftarrow 0$ and break loop iteration for ℓ
      passes 2t to 2t+1; input: $t, \mathcal{R}_t, d[\mathcal{R}_t], s_{t+1}$
        $\mathcal{R}_{t+1}, d[\mathcal{R}_{t+1}]$
          $\leftarrow$ StreamSet($t, \mathcal{R}_t, d[\mathcal{R}_t], s_{t+1}$)
    $\hat{c}_r(\vec{I}) \leftarrow \frac{\mathrm{dg}(\mathcal{R}_i) \cdot \cdots \cdot \mathrm{dg}(\mathcal{R}_{r-1})}{s_{i+1} \cdot \cdots \cdot s_r} \cdot |\mathcal{R}_r|$
    if $\hat{c}_r(\vec{I}) \leq \frac{\tau_i}{4}$, **then** $\chi_\ell \leftarrow 1$, **else** $\chi_\ell \leftarrow 0$
  if $\sum_{\ell=1}^{q} \chi_\ell \geq q/2$ **then**
    return active
  else
    return non-active