A Preliminary Model of the Piece Population in BitTorrent

Cameron Dale
School of Computing Science
Simon Fraser University
Burnaby, BC, Canada
Email: camerond@sfu.ca

***Abstract*— This paper introduces a new analytical model for describing BitTorrent systems. This model builds on previous work on the population of peers, but attempts to go further and describe the population of pieces in the system. The model is a simple fluid model based on a Markov Chain of states, each of which represents the number of copies a piece has in the system. Preliminary results show that for a large enough system, the population of pieces in steady-state is unaffected by further increases in the system size. Results for a couple of sample scenarios are shown.**

I. Introduction

OF all the peer-to-peer Internet applications available, BitTorrent [1] has become the most popular for the downloading of large files. Some reports indicate that half of all the current Internet traffic is due to BitTorrent [2]. One of the reasons it has become so popular, is that the sharing is very efficient [3], allowing downloads to scale well with the size of the downloading population. This efficiency is obtained by breaking up each large file into hundreds or thousands of segments (called *pieces*) which, once downloaded by a peer, can be shared with others while the rest of the download continues.

The BitTorrent system coordinates file sharing through the use of a centralized *tracker*. Upon receiving a request from a downloading peer's client, the tracker will provide a random list of peers. The client will then contact each of the peers to gather information about which pieces the peers have available for download. The client then chooses pieces to download based on a *rarest-first* policy, which ensures that pieces are distributed uniformly throughout the system. In a BitTorrent system, downloaders are typically referred to as *leechers*, while uploaders who have completed their download are referred to as *seeders*.

A. Relationship to prior work

Though there have been a few papers dealing specifically with the sharing present in BitTorrent ([4]–[6]), all have explored the population of peers in the system. In [7], a Markov Chain model was used to numerically study the service capacity of a BitTorrent-like P2P system. This model was expanded on and used in [3] to create a simple deterministic fluid model that was then analyzed to find simple expressions for the peer population of the system. This paper will build on this fluid model, in order to begin to explore the population of pieces in the system.

This paper will start from the model previously proposed [3] for the peer population, and extend it to explore the population of pieces in the system. This model represents several important features of a BitTorrent system:

- Peers arrive according to a Poisson process
- Leechers can abort the download at any time
- Seeders can leave the system

This model also involves some assumptions about the system that are used to simplify the calculations:

- All peers have the same download and upload bandwidths
- Peers have global knowledge, and so can always find a peer to download from

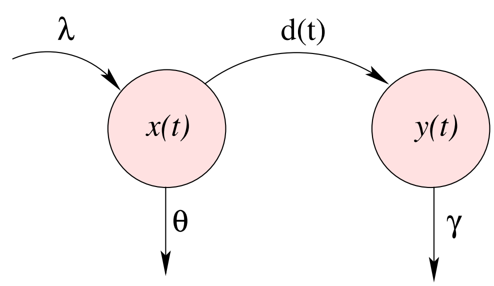

Fig. 1. The original model's two-state Markov Chain representing the peers present in the system.

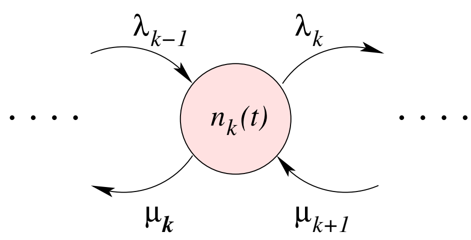

Fig. 2. A sample state from the new Markov Chain. This state would be occupied by all pieces having k copies in the system.

The original model made use of a two-state Markov Chain for the two states of leeching and seeding (see Fig. 1). This model uses the following variables (the file size is assumed to be 1)

- $x(t)$ number of downloaders in the system at time t.
- $y(t)$ number of seeds in the system at time t.
- λ the arrival rate of new requests.
- μ the uploading bandwidth of a given peer.
- c the downloading bandwidth of a given peer.
- θ the rate at which downloaders abort the download.
- γ the rate at which seeds leave the system.
- η indicates the effectiveness of the file sharing.

II. The Proposed Model

The new model will consist of a state k for every peer in the system, where being in a state k indicates that the piece has k copies of itself present in the system. The model will be a standard *birth/death process* [8], where each state k only has transitions to and from neighboring states $k+1$ and $k-1$ (see Fig. 2). This model will use the same variables as the original model, and also introduce the following

- N the number of pieces the download is broken up into.
- $n_k(t)$ the number of pieces with k copies at time t.

Due to the nature of a BitTorrent system, the number of interesting states is always limited to the range $y(t)$ to $y(t)+x(t)$. Any states below this range must have a population of zero, as the minimum number of copies of a piece is the number of seeders. Any states above this range are also empty, as it is not possible to have more copies of a piece than there are peers in the system.

Some assumptions are needed for an analytical analysis of the new model, as the number of variables is as large as the system itself. The most important simplifying assumption is that the transition rates are state independent, and therefore constant across all states.

$$\begin{aligned} \lambda_k &= \lambda_{k+1} = \lambda_{k+2} = \cdots = \lambda' \\ \mu_k &= \mu_{k+1} = \mu_{k+2} = \cdots = \mu' \end{aligned} \tag{1}$$

Also, as downloaders can leave the system at any point in their download, it is assumed that the number of pieces a departing downloader has completed will be random. Therefore, on average, a departing downloader will have half the pieces completed.

These assumptions, and the analysis from the original paper, then lead to the following definition of the transition rates. The transition probability for pieces from k to $k+1$ will be the same as the transition probability of a peer from a downloader to a seeder.

$$\lambda' = d(t) = \min\{cx(t), \mu(\eta x(t) + y(t))\} \tag{2}$$

Though the probability is the same, the population of pieces is much larger than (N times) the population of peers, and so the transition rate (the probability multiplied by the population of the state) will be much larger. Similarly, the transition probability for pieces from k to $k-1$ will be the sum of half the transition probability of a downloader departing and the transition probability of a seeder departing.

$$\mu' = \frac{1}{2}\theta x(t) + \gamma y(t) \tag{3}$$

Using these assumptions, the fluid model for the evolution of the population of pieces in the system is given by

$$\begin{aligned} \frac{\mathrm{d}n_k(t)}{\mathrm{d}t} &= \lambda'(k-1)n_{k-1}(t) - \lambda'(k)n_k(t) \\ &\quad + \mu'(k+1)n_{k+1}(t) - \mu'(k)n_k(t) \end{aligned} \tag{4}$$

Note that the state population used to calculate transition rates from the transition probabilities is actually the population of the state $n_k(t)$, times the state number k. This is a result of the state number representing k copies of the pieces in that state being present in the system. A special case of (4) is used for the end state of n_y, due to the limits on the occupiable states in the system.

$$\frac{\mathrm{d}n_y(t)}{\mathrm{d}t} = -\lambda'(y)n_y(t) + \mu'(y+1)n_{y+1}(t) \tag{5}$$

This large set of equations describes all the piece state transitions within the system.

III. Steady-State Analysis

This model can be analyzed in steady-state by setting (4) to zero and determining the steady-state population of each state ($\bar{n}_k$). Under steady-state conditions, the analysis in [3] has shown that the peer populations are stable and constant as

$$\bar{x} = \frac{\lambda}{\beta\left(1 + \frac{\theta}{\beta}\right)} \tag{6}$$

$$\bar{y} = \frac{\lambda}{\gamma\left(1+\frac{\theta}{\beta}\right)} \tag{7}$$

where

$$\frac{1}{\beta} = \max\left\{\frac{1}{c}, \frac{1}{\eta}\left(\frac{1}{\mu} - \frac{1}{\gamma}\right)\right\} \tag{8}$$

Therefore, the following analysis has been simplified by only referring to the states from $\bar{y}$ to $\bar{y}+\bar{x}$. The new variable k will now be used in the 0 to $\bar{x}$ range, to indicate the position of the state amongst the downloader states.

Using (6) and (7), a simple equation can be derived for the ratio of the transition probabilities (see Appendix A). In general, this ratio is given by

$$\frac{\lambda'}{\mu'} = \frac{1}{1+\frac{\theta}{2\beta}} \tag{9}$$

Since θ and β are always positive, this ration will be in the range [0,1].

Solving (4) set to zero (see Appendix B) gives a steady-state state population of

$$\bar{n}_{\bar{y}+k} = \left(\frac{\lambda'}{\mu'}\right)^k \frac{\bar{y}}{\bar{y}+k}\bar{n}_{\bar{y}} \tag{10}$$

The only variable left unknown in this set of equations is the initial state population ($\bar{n}_{\bar{y}}$). This can be determined by the normalization condition that the sum of all the state populations must be the number of pieces of the file

$$\sum_{k=0}^{\bar{x}} \bar{n}_{\bar{y}+k} = N \tag{11}$$

Unfortunately, (11) leads to a sum of the form

$$\sum_{x=0}^{n} \frac{r^x}{x} \tag{12}$$

which is not easily solvable. However, for large enough $\bar{y}$ and $\frac{\lambda'}{\mu'} < 1$, the solution to (11) can be approximated by (see Appendix C)

$$\bar{n}_{\bar{y}} \approx N\frac{1-\frac{\lambda'}{\mu'}}{1-\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}} \tag{13}$$

In cases where these conditions do not hold, the system is small enough that a value for $\bar{n}_{\bar{y}}$ can easily be determined numerically.

IV. Observations

The analysis in Section III has completely determined the piece population of the system in steady-state. This determination has several interesting results.

The most interesting result is that for a large enough system, the piece population will no longer depend on the size of the system. Appendix D shows that for systems with steady-state downloader size of at least

$$\bar{x} > \frac{\log\left(2N\left(\frac{1}{1+\frac{2\beta}{\theta}}\right)+1\right)}{\log\left(1+\frac{\theta}{2\beta}\right)} - 1 = x_{min} \tag{14}$$

then the initial state population of (13) will be given by

$$\bar{n}_{\bar{y}} \approx N\left(1-\frac{\lambda'}{\mu'}\right) \tag{15}$$

This equation no longer depends on the number of downloaders in the system, and so does not depend on λ, the arrival rate of downloaders to the system. Appendix D also shows that for a seeder population of at least

$$\bar{y} > 2N\frac{\lambda'}{\mu'}\left(1-\frac{\lambda'}{\mu'}\right) - 1 = y_{min} \tag{16}$$

then the piece population of each state from (10) will be given by

$$\bar{n}_{\bar{y}+k} = \left(\frac{\lambda'}{\mu'}\right)^k \bar{n}_{\bar{y}} \tag{17}$$

The simplicity of these equations which can be used to describe the population of all the pieces in the system is surprising. The equations depend on the variables introduced by the original model in Section I-A, but there is no dependence on the size of the system (i.e. the number of peers in the system), and therefore no dependence on the arrival rate λ.

V. Experiments

The combination of (9), (10) and (13) can now be used in experimental situations to calculate the population of all the states. This will be done for two cases from the original paper [3]: a download-limited situation called "Experiment 1" in the original paper, and an upload-limited situation called "Experiment 2." In both experiments, the number of pieces the downloading file is segmented into is assumed to be a conservative $N = 500$.

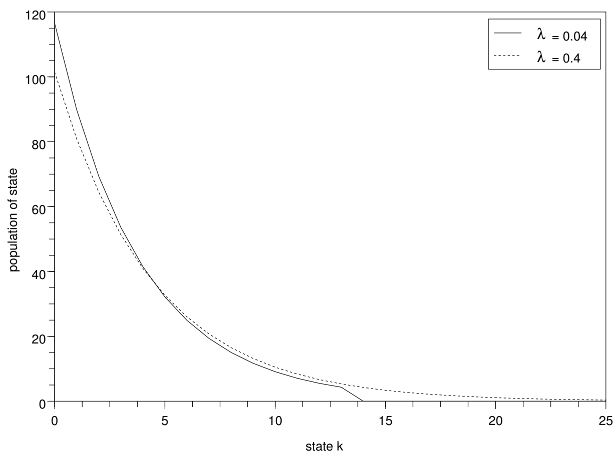

Fig. 3. Two of the populations from Experiment 1, showing the piece population for the two smallest arrival rates (larger rates result in curves identical to $\lambda = 0.4$).

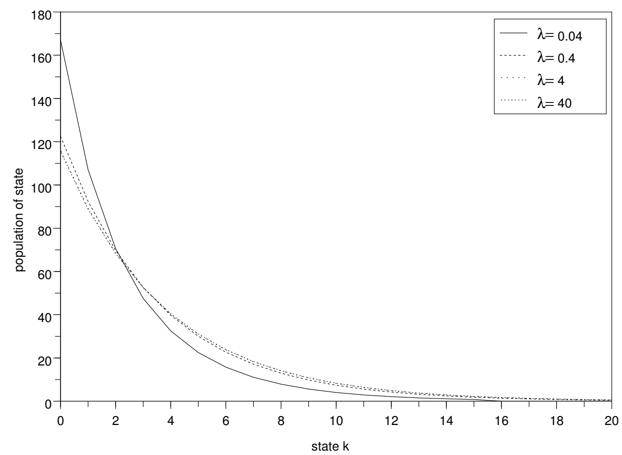

Fig. 4. All four piece populations from Experiment 2.

A. Experiment 1

In this download-limited system, the following parameters are used: $\mu = 0.00125$, $c = 0.002$, $\theta = \gamma = 0.001$ and $\eta = 1$ is assumed. The arrival rate λ will vary from 0.04 to 40 by factors of 10. Using (6) and (7), the steady-state peer populations are calculated to range from $\bar{x} = 13$ and $\bar{y} = 27$, to $\bar{x} = 13333$ and $\bar{y} = 26667$. The ratio of the transition rates is given by (9) to be $\frac{\lambda'}{\mu'} = 0.8$. These values can then be used to calculate the minimum peer populations from (14) and (16) for this to be considered a large system, which are $x_{min} = 23$ and $y_{min} = 159$. The initial state population for a large system is given by (13) to be $\bar{n}_{\bar{y}} = 100$. Only the smallest system of $\lambda = 0.04$ does not meet the minimum requirements for a large system, and in this case the initial state population was numerically calculated to be $\bar{n}_{\bar{y}} = 116$.

In Fig. 3, the piece population is shown for the two smallest arrival rates, 0.04 and 0.4. The other two arrival rates resulted in curves that exactly overlapped the curve for 0.4, which is expected as all three meet the condition for a large system, and so are expected to be independent of the system size and the arrival rate. This independence is interesting, as it suggests that most of the pieces have very few copies beyond the required seed copies. In fact, the initial state of no additional copies is occupied by a full 20% of the pieces in the system.

B. Experiment 2

In this upload-limited system, the parameters used are the same, except that the seed leaving rate is increased to $\gamma = 0.005$. The arrival rate λ will again vary from 0.04 to 40 by factors of 10. Using (6) and (7), the steady-state peer populations are calculated to range from $\bar{x} = 15$ and $\bar{y} = 5$, to $\bar{x} = 15000$ and $\bar{y} = 5000$. The ratio of the transition rates is given by (9) to be $\frac{\lambda'}{\mu'} = 0.77$. These values can then be used to calculate the minimum peer populations from (14) and (16) for this to be considered a large system, which are $x_{min} = 20$ and $y_{min} = 176$. The initial state population for a large system is given by (13) to be $\bar{n}_{\bar{y}} = 115$. Now, the smallest two systems with arrival rates of 0.04 and 0.4 do not meet the minimum requirements for a large system, and in these cases the initial state populations were numerically calculated to be $\bar{n}_{\bar{y}} = 167$ and $\bar{n}_{\bar{y}} = 123$, respectively.

In Fig. 4, the piece population is shown for all four arrival rates. The two smallest arrival rates, 0.04 and 0.4, are distinct, but the other two largest arrival rates resulted in curves that exactly overlapped each other. This is as expected again, as those two meet the condition for a large system, and so are expected to be independent of the system size and the arrival rate. In the $\lambda = 0.04$ case, the initial state of no additional copies is occupied by a full 33% of the pieces in the system.

VI. Future Work

The analysis has been completed, but the results have not yet been verified to be correct. The most important work required to further this model, is to verify the results obtained here by comparing them to actual measurements of a BitTorrent system.

Once verified, the model can then be used for may interesting explorations of the interactions in a BitTorrent system. One possible avenue of exploration is the relaxation of the assumption that departing downloaders have, on average, half completed the download. A more realistic assumption might be that the number of pieces a departing downloader has completed would be exponentially distributed, as a downloader is probably more likely to leave after recently joining the system. Another area of exploration is the different policies that BitTorrent implements. For example, downloaders choose pieces to download based on a rarest-first policy that encourages copying of pieces in the system that are under-represented. This policy would definitely have an impact on the general population of pieces, and should be considered.

The model presented here could also be extended to allow for the modelling of pieces in a dynamic environment, in which pieces are created and leave the system regularly. This would then allow for the analysis of many more interesting situations, including the streaming BitTorrent case which is becoming more popular for streaming video [9].

VII. Conclusions

In this paper, a previously proposed simple fluid model for the population of peers in the system has been extended to obtain expressions for the population of pieces under steady-state conditions. The resulting system-sized Markov Chain has been analyzed, and using some simplifying assumptions, has been solved analytically for large systems. Surprisingly, these systems are shown to have a constant piece population, unaffected by further increases in the system size. This result was unexpected, and needs further investigation to explain properly. Also, the analysis is currently preliminary in nature, and has not yet been confirmed by measurement.

Appendix A
The Ratio $\frac{\lambda'}{\mu'}$

The ratio can be determined in steady-state from (2) and (3) to be

$$\frac{\lambda'}{\mu'} = \frac{\min\{c\bar{x}, \mu(\eta\bar{x} + \bar{y})\}}{\frac{1}{2}\theta\bar{x} + \gamma\bar{y}}$$

Dividing through by $\bar{x}$ gives

$$\frac{\lambda'}{\mu'} = \frac{\min\{c, \mu(\eta + \frac{\bar{y}}{\bar{x}})\}}{\frac{1}{2}\theta + \gamma\frac{\bar{y}}{\bar{x}}} \tag{18}$$

Using the steady-state values for $\bar{x}$ and $\bar{y}$ in (6) and (7) gives

$$\frac{\bar{y}}{\bar{x}} = \frac{\lambda}{\gamma\left(1 + \frac{\theta}{\beta}\right)} \frac{\beta\left(1 + \frac{\theta}{\beta}\right)}{\lambda} = \frac{\beta}{\gamma}$$

Substituting into (18) gives

$$\frac{\lambda'}{\mu'} = \frac{\min\{c, \mu(\eta + \frac{\beta}{\gamma})\}}{\frac{1}{2}\theta + \beta} \tag{19}$$

In the upload-limited case,

$$\frac{1}{c} \leq \frac{1}{\eta}\left(\frac{1}{\mu} - \frac{1}{\gamma}\right) \quad \Rightarrow \quad \beta = \frac{\eta\mu\gamma}{\gamma - \mu}$$

solving (19) for this case gives

$$\frac{\lambda'}{\mu'} = \frac{1}{\frac{\theta}{2}\frac{1}{\eta}\left(\frac{1}{\mu} - \frac{1}{\gamma}\right) + 1} = \frac{1}{\frac{\theta}{2\beta} + 1}$$

In the download-limited case,

$$\frac{1}{c} \geq \frac{1}{\eta}\left(\frac{1}{\mu} - \frac{1}{\gamma}\right) \quad \Rightarrow \quad \beta = c$$

solving (19) for this case gives

$$\frac{\lambda'}{\mu'} = \frac{1}{\frac{\theta}{2c} + 1} = \frac{1}{\frac{\theta}{2\beta} + 1}$$

In general, the ratio is then given by

$$\frac{\lambda'}{\mu'} = \frac{1}{1 + \frac{\theta}{2\beta}} \tag{20}$$

which, since θ and β are always positive, will be in the range [0,1].

Appendix B
Steady-State Birth Death Solution

The changing state population described by (4) can be solved in steady-state by setting all the population changes to zero.

$$\begin{aligned} 0 = &\lambda'(k-1)\bar{n}_{k-1} - \lambda'(k)\bar{n}_k \\ &+\mu'(k+1)\bar{n}_{k+1} - \mu'(k)\bar{n}_k \end{aligned} \tag{21}$$

In the end state case of $k = y$ described by (5), the steady-state equation is of the form

$$0 = -\lambda'(\bar{y})\bar{n}_{\bar{y}} + \mu'(\bar{y}+1)\bar{n}_{\bar{y}+1}$$

Solving this for the state $\bar{y}+1$ gives

$$\bar{n}_{\bar{y}+1} = \frac{\lambda'}{\mu'}\left(\frac{\bar{y}}{\bar{y}+1}\right)\bar{n}_{\bar{y}} \tag{22}$$

Solving (21) for the state $k+1$ gives

$$\bar{n}_{k+1} = \frac{\lambda' + \mu'}{\mu'}\frac{k}{k+1}\bar{n}_k - \frac{\lambda'}{\mu'}\frac{k-1}{k+1}\bar{n}_{k-1}$$

Replacing k with $k-1$ gives

$$\bar{n}_k = \frac{\lambda' + \mu'}{\mu'}\frac{k-1}{k}\bar{n}_{k-1} - \frac{\lambda'}{\mu'}\frac{k-2}{k}\bar{n}_{k-2} \tag{23}$$

Now, the state $\bar{y}+2$ can be described using (23) and (22) to be

$$\bar{n}_{\bar{y}+2} = \left(\frac{\lambda'}{\mu'}\right)^2 \left(\frac{\bar{y}}{\bar{y}+2}\right) \bar{n}_{\bar{y}}$$

Using this equation, the next state, $y+3$, can be shown to be

$$\bar{n}_{\bar{y}+3} = \left(\frac{\lambda'}{\mu'}\right)^3 \left(\frac{\bar{y}}{\bar{y}+3}\right) \bar{n}_{\bar{y}}$$

The general solution is now obvious, and is given by

$$\bar{n}_{\bar{y}+k} = \left(\frac{\lambda'}{\mu'}\right)^k \left(\frac{\bar{y}}{\bar{y}+k}\right) \bar{n}_{\bar{y}} \quad (24)$$

Appendix C
Normalization of the State Populations

The normalization condition for the steady-state population is given by

$$N = \sum_{k=0}^{\bar{x}} \bar{n}_{\bar{y}+k}$$

Using (24), the sum is

$$N = \bar{y}\bar{n}_{\bar{y}} \sum_{k=0}^{\bar{x}} \left(\frac{\lambda'}{\mu'}\right)^k \left(\frac{1}{\bar{y}+k}\right)$$

Unfortunately, this sum is not easily solved. Under the assumption that $\bar{y}$ is large and $\frac{\lambda'}{\mu'} < 1$, it can be approximated by assuming that $\bar{y}+k \approx \bar{y}$ (see Appendix D for when this is valid).

$$N \approx \bar{y}\bar{n}_{\bar{y}} \left(\frac{1}{\bar{y}}\right) \sum_{k=0}^{\bar{x}} \left(\frac{\lambda'}{\mu'}\right)^k \quad (25)$$

This sum is well known,

$$\sum_{x=0}^{n} r^x = \frac{1-r^{n+1}}{1-r}$$

Substitution into (25) gives

$$N \approx \bar{n}_{\bar{y}} \frac{1-\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}}{1-\frac{\lambda'}{\mu'}}$$

Solving for $\bar{n}_{\bar{y}}$ then gives

$$\bar{n}_{\bar{y}} \approx N \frac{1-\frac{\lambda'}{\mu'}}{1-\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}} \quad (26)$$

Appendix D
Large System Conditions

A. Large Number of Downloaders

For large systems, the normalization constraint on the system in (26) can be rewritten as

$$\bar{n}_{\bar{y}} \approx N\left(1-\frac{\lambda'}{\mu'}\right) \quad (27)$$

since $\frac{\lambda'}{\mu'} < 1$ the denominator term $\left(1-\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}$ will be very close to 1 for large $\bar{x}$. This approximation will be considered valid if

$$N\frac{1-\frac{\lambda'}{\mu'}}{1-\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}} - N\left(1-\frac{\lambda'}{\mu'}\right) < \frac{1}{2}$$

Solving this equation for $\bar{x}$ gives

$$\frac{1}{1-\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}} - 1 < \frac{1}{2N\left(1-\frac{\lambda'}{\mu'}\right)}$$

$$\frac{\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}}{1-\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}} < \frac{1}{2N\left(1-\frac{\lambda'}{\mu'}\right)}$$

$$\frac{1}{\left(\frac{\lambda'}{\mu'}\right)^{\bar{x}+1}} > 2N\left(1-\frac{\lambda'}{\mu'}\right)+1$$

$$-(\bar{x}+1)\log\left(\frac{\lambda'}{\mu'}\right) > \log\left(2N\left(1-\frac{\lambda'}{\mu'}\right)+1\right)$$

$$\bar{x} > \frac{\log\left(2N\left(1-\frac{\lambda'}{\mu'}\right)+1\right)}{-\log\left(\frac{\lambda'}{\mu'}\right)} - 1 = x_{min}$$

This can be rewritten using the equation for the ratio of the transition rates, (20), to be

$$\bar{x} > \frac{\log\left(2N\left(\frac{1}{1+\frac{2\beta}{\theta}}\right)+1\right)}{\log\left(1+\frac{\theta}{2\beta}\right)} - 1 = x_{min}$$

B. Large Number of Seeders

Appendix C derived an equation for the initial population state by assuming that for large enough $\bar{y}$ and $\frac{\lambda'}{\mu'} < 1$,

$$\bar{n}_{\bar{y}+k} = \left(\frac{\lambda'}{\mu'}\right)^k \frac{\bar{y}}{\bar{y}+k}\bar{n}_{\bar{y}}$$

can be approximated by assuming that

$$\bar{y}+k \approx \bar{y}$$

which then gives a simpler equation

$$\bar{n}_{\bar{y}+k} = \left(\frac{\lambda'}{\mu'}\right)^k \bar{n}_{\bar{y}}$$

To determine when this approximation is valid, it is sufficient to determine that

$$\left(\frac{\lambda'}{\mu'}\right)^k \frac{\bar{y}}{\bar{y}+k}\bar{n}_{\bar{y}} - \left(\frac{\lambda'}{\mu'}\right)^k \bar{n}_{\bar{y}} < \frac{1}{2}$$

for a worst-case value of k. This can be simplified

$$\frac{\bar{y}+k}{k} > 2\left(\frac{\lambda'}{\mu'}\right)^k \bar{n}_{\bar{y}}$$

$$\bar{y} > k\left[2\left(\frac{\lambda'}{\mu'}\right)^k \bar{n}_{\bar{y}} - 1\right]$$

Since $\frac{\lambda'}{\mu'} < 1$, the largest the right hand side can be (worst case) is when $k = 1$, at which point

$$\bar{y} > 2\frac{\lambda'}{\mu'}\bar{n}_{\bar{y}} - 1 = y_{min}$$

Using the result from Appendix D-A, this can be further simplified by inserting the approximate value for $\bar{n}_{\bar{y}}$ in (27)

$$\bar{y} > 2N\frac{\lambda'}{\mu'}\left(1 - \frac{\lambda'}{\mu'}\right) - 1 = y_{min}$$

References

[1] B. Cohen. (2003, May) Incentives build robustness in BitTorrent. [Online]. Available: http://bitconjurer.org/BitTorrent/bittorrentecon.pdf

[2] (2004) CacheLogic. [Online]. Available: http://www.cachelogic.com

[3] D. Qiu and R. Srikant, "Modeling and performance analysis of BitTorrent-like peer-to-peer networks," in *Proc. SIGCOMM '04*, Portland, Oregon, USA, Aug. 30–Sept. 3, 2004.

[4] M. Izal, G. Urvoy-Keller, E. Biersack, P. Felber, A. A. Hamra, and L. Garces-Erice, "Dissecting BitTorrent: Five months in a torrents lifetime," in *Passive and Active Measurements*, Antibes Juan-les-Pins, France, Apr. 2004.

[5] J. A. Pouwelse, P. Garbacki, D. H. J. Epema, and H. J. Sips, "The BitTorrent P2P file-sharing system: Measurements and analysis," in *Proc. 4th International Workshop on Peer-to-Peer Systems*, Ithaca, NY, USA, Feb. 2005.

[6] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang, "Measurements, analysis, and modeling of BitTorrent-like systems," in *Proc. Internet Measurement Conference*, Berkeley, CA, USA, Oct. 2005.

[7] G. de Veciana and X. Yang, "Fairness, incentives and performance in peer-to-peer networks," in *Forty-first Annual Allerton Conference on Communication, Control and Computing*, Monticello, Illinois, USA, Oct. 2003.

[8] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing Networks and Markov Chains*, 2nd ed. Hoboken, New Jersey: John Wiley & Sons, Inc., 2006.

[9] X. Zhang, J. Liu, B. Li, and T.-S. P. Yum, "CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming," in *Proc. IEEE INFOCOM*, Miami, FL, USA, Mar. 2005.