



#ASLI ENGINEERING

The Choke Algorithm

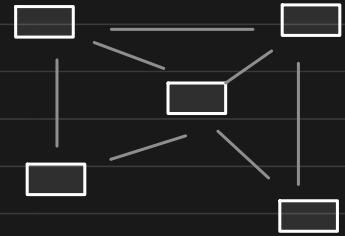


BY

ARPIT BHAYANI

The Choke Algorithm

BitTorrent works on a P2P network, hence there is no central resource allocation unit. How would the peers ensure



1. maximum download speed
2. prevent anyone from abusing the network

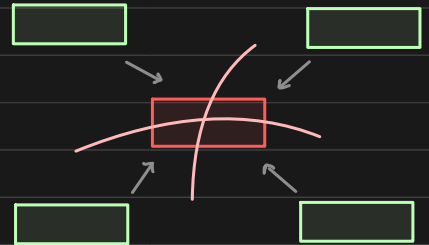
* peer would naturally try to download from whoever it can

The choke algorithm was introduced to **guarantee** a reasonable upload and download reciprocation.

The choke algorithm is a variant of "tit-for-tat" algorithm

Free rider problem

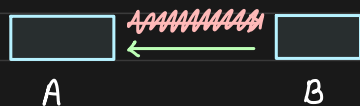
Free-riders are peers that never uploads, but always downloads, should be penalized



Hence our criteria to choose a peer to send our content to cannot be "simple" it should be based on **reciprocation**.

Choking and Interested

Choking is a temporary refusal to upload



We say that peer A has choked peer B because peer B is unable to download file from A, but peer A can download from peer B

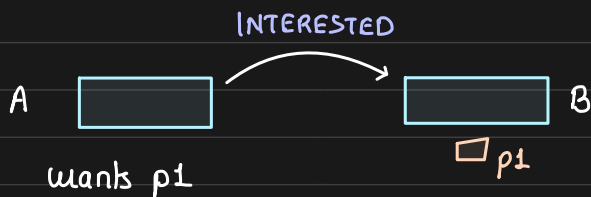
Choking is necessary because

1. TCP congestion when we send across many connections at once.
2. prevent network abuse and starvation

Choking and Unchoking aren't perpetual rather it is periodic. The upload happens only until a peer is unchoked.

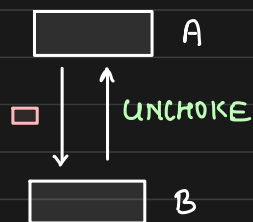
Interested peer is someone who wants a piece that you have

A wants a piece p_1 that is there with B, hence we say that A is INTERESTED in peer B



How to find peers to "unchoke"?

we are prioritizing to **unchoke** peers who let us download the pieces recently



* **Reciprocity** - returning a favour

Example:

A is choked by B i.e. it cannot get any piece from B.

B wanted a piece that A had and since A gave that piece B **unchokes** A allowing it to now download from B.

* There may be thousands of nodes in the swarm but a few of them are randomly picked by tracker as a peer for each.

But, how would we find peers to choke and unchoke?

↳ **Download rate**

↳ **Reciprocity**

"Any peer will upload to peers, who give the best download rate"

↳ This encourages peers to let others download

↳ prohibits free riders that never uploads

Choke algorithm for leecher

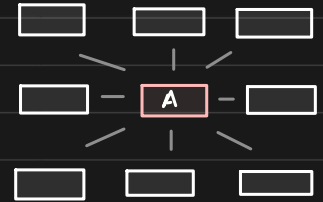
When in leecher state, the choke algorithm is called

↳ every 10 seconds

↳ everytime a peer leaves a peerset

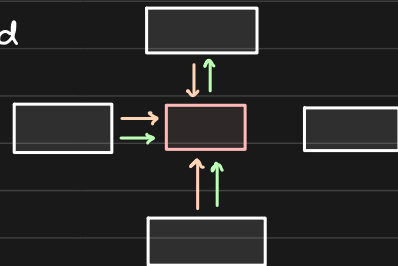
↳ everytime an unchoked peer becomes interested or not.

1. Every 10 seconds, peer A orders the interested remote peers by their download rate to A and the fastest 3 are **unchoked**.



↖
Regular Unchoke

2. For regular unchoke, the peers are ordered by their download rate to local peer and who have sent at least one block in last 30 sec. This guarantees only active peers are unchoked.

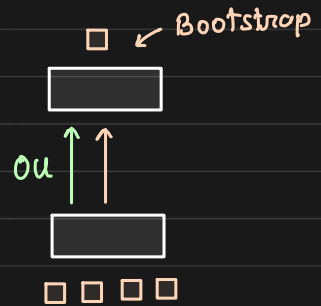


3. Every 30 seconds, one additional interested peer is unchoked at random \leftarrow To promote fairness to new peers
[no need of reciprocation]

Optimistic Unchoke

Advantages of optimistic unchoke

- ↳ evaluates download capacity of new peers
- ↳ bootstrap new peers who do not have any piece to share, by giving them the first piece.



4. if the optimistic unchoked peer is from the 3 fastest peer, another peer is chosen for an unchoke

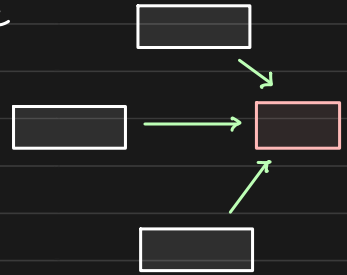
↳ if peer was also interested, the round completes

↳ if peer is not interested, still it is unchoked and we continue to unchoke other peers optimistically

* we would have more than 4 unchokes
but at max 4 interested unchokes

Choking algorithm boosts a "good" behaviour

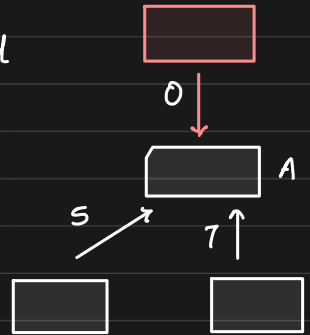
- ↳ if upload rate of A is high, more peers will unchoke it, giving A a faster download



Free Riders are penalized

Free riders never upload, they only download. Hence when other peer is choosing whom to unchoke, the free rider will be last in the list because of sorting criteria.

- ↳ ordered by download rate to A



The only hope for a free rider is Optimistic Unchoke

Anti-snubbing

There is a possibility that a peer is CHOKED by all others

How would it proceed?

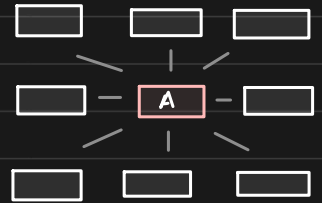
peer who has not sent anything in last 60 sec

- ↳ Optimistic unchoke → infrequent
- ↳ it retaliates and refuses to upload to the peer
- ↳ which increases optimistic unchokes in the network

Choke algorithm for Seeder

- ↳ every 10 seconds or everytime a peer leaves a peerset
- ↳ everytime an unchoked peer becomes interested / not interested

1. The algorithm orders the peers according to the time they were last **unchoked** or that have pending request for blocks.



Higher upload rate is given a priority Penalizes free riders with high bandwidth

2. The other peers (never unchoked) are ordered by upload rates
3. For first 20 seconds

↳ unchoke first 3 peers ← Seed kept unchoked

↳ and unchokes one peer at **random** ← Seed random Unchoked

For the next 10 seconds

↳ unchokes the first 4 peers ← Seed kept unchoked

Seeder is not unchoking based on upload rate

instead it is using time they were last unchoked

peers in active peer set are changed regularly

random peer taking a slot from too ordered peer