Locality List Update Seminar

Loric ANDRE

05.07.2021

Outline

Introduction

- Add dependencies to classic list update problem
- Goal : Get ratio in terms of locality

MRF

- Move accessed item to first dependency in front
- Then move said dependency the same way
- Repeat
- **We already have a competitive ratio of 4 in the case with dependencies**

Base bound

- Not in terms of locality yet
- We want another way to prove the bound of 4

Notations

- $ALG^*(\sigma)$: partial cost of ALG over the sequence σ; accessing node i costs only $i-1$.
- σ_{xy} : sequence without accesses to nodes different from x and y
- $ALG(x, j)$: 1 if x in front of $\sigma(j)$ at time j
- $ALG_{xy}(\sigma)$: sum of all $ALG(x, j)$ and $ALG(y, j)$ where j is x or y : projection of the cost for x and y

Pairwise property

Base version : ALG satisfies the pairwise property if

$$ALG_{xy}(\sigma) = ALG(\sigma_{xy})$$

ALG satisfies the **weak** pairwise property if

$$ALG_{xy}(\sigma) \leq ALG(\sigma_{xy})$$

Move Recursively Forward

MRF satisies the weak pairwise property :

- Cost of MRF : access and reconfiguration
- Access : $pos(x)$
- Reconfiguration : at least $pos(x) - \#\{\text{parents of x}\}$
- $MRF(\sigma_{xy}) =$
  $2pos(x) + 2pos(y) - a(x)occ(x, \sigma) - a(y)occ(y, \sigma)$
- $a(x)occ(x, \sigma) + a(y)occ(y, \sigma)$ less than optimal cost so less than $MRF_{xy}(\sigma) = pos(x) + pos(y)$

Ratio on pairs

Independent nodes :

- i and j : 2 consecutive indexes of accesses to the back node by MRF. $j - i \geq 1$
- MRF : cost $3 + 3 + (j - i - 1) = 5 + j - i$
- OPT pays at least 2 (access to back node) 1 time : cost at least $2 + j - i$
- $\frac{MRF(\sigma_{xy})}{OPT(\sigma_{xy})} \leq \frac{5+j-i}{2+j-i} \leq \frac{6}{3} = 2$

Full ratio

- If satisfies pairwise property, can sum ratios
- In the case of MRF, $MRF(\sigma) \leq 2 \sum_{x \neq y} MRF_{xy}(\sigma)$
- Related nodes : MRF = OPT
- Conclusion : **Ratio of 4**
- Ratio decreases with number of related nodes

Model

- Locality : grows with accesses to same node
- Short and long runs
- Long run changes
- $\lambda = \frac{l_c(\sigma)}{r(\sigma)}$

Phases

2 forms starting with x :

- split after every long run or at the end
- (a). $(xy)^k x^l$: xyxyxyxyxxxxx
- (b). $(xy)^k y^l$: xyxyxyxyyyyyy

Optimal Algorithm

We deal with pairs of unrelated nodes for now.
An optimal algorithm is to move a node to the front of the list at the beginning of every long run.

Cost of OPT

- Analyze cost over phases ($(xy)^k x^l$ or $(xy)^k y^l$)
- Beginning of phase starting with x : y in front
- Phase other than first or last :
  - (b) costs $\frac{r(\pi(i))}{2}$ (number of accesses to x)
  - (a) costs $\frac{r(\pi(i))-1}{2} + 2$ (number of accesses to x then 2 for the long run)
  - Conclusion : cost of $\frac{r(\pi(i))+3l_c(\pi(i))}{2}$ ($l_c(\pi(i))$ number of long run changes)
- Last run : no move if no long run change, same cost minus $f_e(\sigma_{xy})$ (whether it ends in a long run change or not)
- First run : x can be in front : cost at least $\frac{r(\pi(1))+3l_c(\pi(1))-f_b(\sigma_{xy})}{2}$

Cost of MRF

All runs cost 2, first one can cost 0 if first accessed node in front :

$$MRF_U^*(\sigma_{xy}) = 2r(\sigma_{xy}) - 2f_b(\sigma_{xy})$$

Related nodes

- Related nodes : list fixed, same cost
- $\eta(\sigma_{xy})$: Number of accesses to back node
- Cost $\eta(\sigma_{xy})|\sigma_{xy}|$

Full ratio - strict competitiveness

- $\alpha(\sigma) = \frac{(1+\eta(\sigma))|\sigma| - f_{b,U}}{r_U(\sigma)}$
- $\beta(\sigma) = \frac{3l_{c,U}(\sigma) - 2f_{e,U}(\sigma)}{r_U(\sigma)} \geq \frac{l_{c,U}(\sigma)}{r_U(\sigma)} \geq \lambda_U$
- Ratio (strict competitiveness) : $R(\sigma) \leq \frac{4}{1+\frac{\alpha+\beta}{\alpha+1}}$
- Issue : α could be negative or not, so transition to $R(\sigma) \leq \frac{4}{1+\beta}$ not yet proven
- Once done, $R(\sigma) \leq \frac{4}{1+\lambda_U}$

Full ratio - down to one

- $\alpha'(\sigma) = \frac{(1+\eta(\sigma))|\sigma|-f_{b,U}--2f_{e,U}(\sigma)}{r_U(\sigma)}$
- $\beta'(\sigma) = \frac{l_{c,U}(\sigma)}{r_U(\sigma)} \geq \frac{l_{c,U}(\sigma)}{r_U(\sigma)} \geq \lambda_U$
- Ratio (strict competitiveness) :
  $MRF(\sigma) \leq \frac{4OPT(\sigma)}{1+\frac{\alpha'+3\beta'}{\alpha'+1}} + 4f_{e,U}(\sigma)\,(1+2f_{b,U})$
- Same issue
- Once done, $MRF(\sigma) \leq \frac{4OPT(\sigma)}{1+3\lambda_U} + O(l^4)$, l number of nodes
- This goes down to one !

Conclusions

- Need one step cleared to get a ratio of 4, going down to 2 (strict competitiveness) or 1
- Always removed the cost over related nodes when we could : ratio of one means that the full ratio decreases with the number of related pairs !

Questions

Thank you for listening, any questions ?