

When Pull Comes To Shove... Do Both!

Ullas ☺

*Department of Computer Science and Automation,
Indian Institute of Science, Bangalore
email: ullas@iisc.ac.in*

Abstract

Graph algorithms are ubiquitous. Wait, graphs are ubiquitous. Period. It is an axiom of life [1] that the more something is ubiquitous, the more people want more of it. Graphs are no exception. And Computer Scientists eat graphs for breakfast — and weighted graphs, let's not forget (and if you ask, "What do they eat for lunch and dinner, then?", well, they don't. Processing the breakfast takes them enough time to last until next day, because... BIG DATA!!). The apocalyptic scenario where graphs have become so pervasive that one needs to breathe graphs rather than Oxygen for survival is not too far away. Sooooooo, it helps to be prepared, you know...

DIALOGUE

Reader: Wait, how is that an abstract?

Author: Because it is anything but concrete.

Reader: What is this paper even trying to tell? What research problem does it address?

Author: All in its own good time.

Reader: And that is definitely NOT a valid reference that you have put there.

Author: It doesn't violate any definition of reference that is known in the literature.

Reader: Self-references don't count!

Author: It doesn't invalidate the veracity of the statement nevertheless.

Reader: This is not even the way to write a research paper, you know? Research has to be pathbreaking. It has to do something that nobody has embarked upon before, something preferably so mindblowing that— Ohhhh!

Author: Exactly! I am so glad you see what I have done.
Reader: Wait, why you are writing down my words?
Author: To make this dialogue a part of the paper itself.
Reader: GASP! I see myself here! Dialogues in a research paper! WHAT HAVE YOU DONE?

PREREQUISITES

You are a Computer Scientist, in any sense of the phrase, since the field has grown so vast that it has encroached upon adjacent fields, and the neighbours are threatening litigation.

POSTREQUISITES

You remain a Computer Scientist, haha.

1 INTRODUCTION

Imagine you are the ringleader of an organization that kills terrorists. So wait, doesn't that make you a terrorist as well, because you inspire terror into the hearts of terrorists, and isn't inspiring terror the very dictionary definition of a terrorist? Ok, I see that the first word of this section was a colossal mistake. Do not imagine too much.

Devil: So, by that definition, if you are killing terrorists, shouldn't you eventually commit suicide?

Author: So much for the phrase "An idle mind is a devil's workshop".
Devil: Ok ok, now I get it — why you have asked the reader to imagine. So as to keep me at bay.

Reader: Um..., a Devil in a Research paper?
Author: Well, we do need someone to play the Devil's Advocate.
Reader: And a dialogue in the middle of the paper too?
Author: Hey, you are featuring in it as well. No complaining.

Let us return, refreshed, from that delightful and much-needed digression, in light of the heavyweight machinery that awaits. As a ringleader, what is your primary goal?

- (A) To buy a ring for yourself (to lend credibility to your title)
- (B) To try to remain in one piece by the end of the operation
- (C) To publish research papers on graphs
- (D) To, um, kill terrorists?

[The above MCQ is left an exercise to the alert reader]

Reader: I thought you said we had returned from the digression.

Author: We digressed again. As we will throughout. For that matter, any non-digression is a digression.

Your skills are so much in demand that you spend three-fourths of the year in fixing and replacing your door... This is owing to the fact that people come breaking down your door, demanding for your skills, probably because they have dispensed with all the necessary politeness of knocking on the door first because you are so much in demand, for the few seconds it takes them to knock might mean that you are already taken by their competitors.

Anyway, your task is to identify the terrorist networks so that you can be sure that once you locate and capture one terrorist in that network, that terrorist will lead to others in the network, who can in turn lead to others they know, and so on (it is inherently assumed that your interrogatory prowess involves the most advanced methods at your disposal, owing to which, the captured terrorists WILL certainly reveal all they know, out of sheer, er, terror at the thought of what you are capable of doing to them if they chose not to sing — which really *really* REALLY begs the question of who is the worse terrorist over here, really). Your repute as a ringleader of your organization is well known, to the point that every time you go out on your missions, dogs move out of your way. And any stray dog lingering on your path is pulled back by the other dogs who realize that you mean business. And in case you are wondering where dogs come in to the scene, aren't the terrorists themselves dogs of a kind, and the worst kind at that (the word *kind* may be a little too kind to be used on them)? Ok, such is your repute that nobody would date you out of sheer fear. In fact, you probably come from an era before the concept of dating was invented. I guess that makes you a "Predator", haha!

Anyway, this is the idea of a Connected Component in a Graph (er... no, not the dogs), where the nodes are the terrorists, an undirected edge connects two nodes if the two terrorists belong to the same organization and know each other (as with most terrorist organizations, not everyone will know everyone in it, but it is guaranteed that from any one of them in the organization, you can reach anyone else (apologies if that sounds too ominous)), and all the terrorists within a network form a single "connected component", because they are all connected (blood is thicker than water, and all that, you know), while those of other terrorist networks form their own components, disparate from each other (it is well known that terrorists, unlike in academia, do NOT collaborate — their Erdos numbers are laughable).

RIDDLE

Why are the nodes of a graph so stressed?
Because they are living on the edge.

The problem then, is simple. You need to identify and label each node in the graph with the component it belongs to.

Reader: Couldn't this one line alone appear in the abstract instead of all this rigmarole? It's precise. It would have saved so much time.

Author: Yes. And you can go visit Agra and come back home without seeing the Taj Mahal. It's precise. It would save you so much time.

DISCLAIMER

This paper discusses the Connected Components Graph Algorithm, and readers familiar with it can skip the rest of the paper, or read it for fun.

META-DISCLAIMER

The above disclaimer should really really have appeared in, like, page 1, possibly even before the abstract, and not this late in the paper.

META-META-DISCLAIMER

In the event that the disclaimer really did appear as suggested, the necessity for the above meta-disclaimer would have been entirely obviated, and hence it would have been absent.

META-DISCLAIMER

Oh yeah? Then in that case, the Meta-Meta-Disclaimer would have been absent as well. Take that, you nosy brat of a meta-meta-disclaimer!

META-DISCLAIMER:

HEY! You are encroaching upon my space!

Now I guess you merging into my box should make us a single connected component.

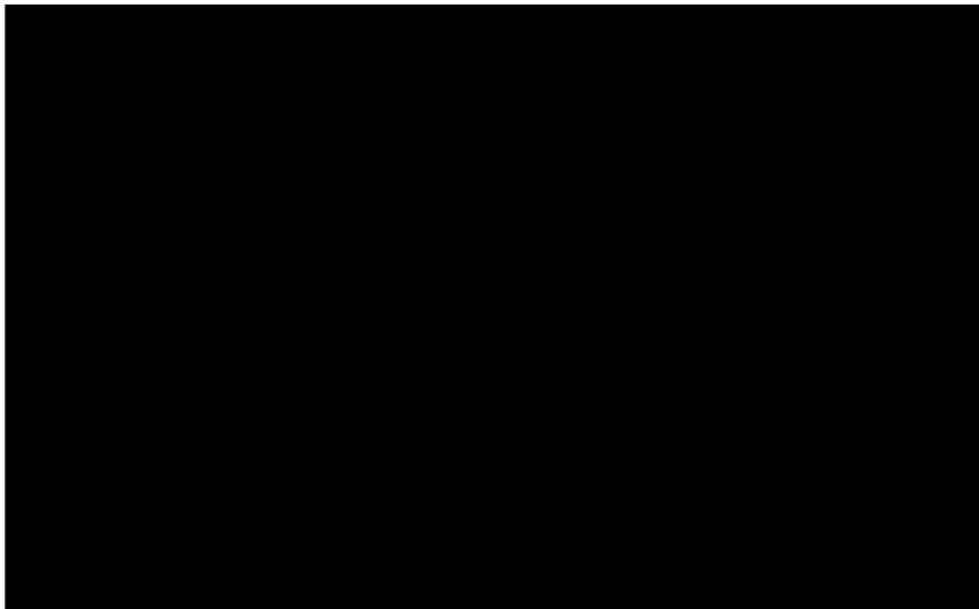
META-META-DISCLAIMER:

You don't deserve a box of your own. And I will teach you to call me a brat!

Hey, and who on earth are you down there?

AUTHOR:

Now now guys, this has gone too far.



The reader is strongly urged to desist from asking what happened within the above box. Let the black hole in the page above be a void of no return to

the imagination.

Anyway, as it so happens, the definition given above is an operational definition, and so the algorithm is evident to the most myopic bat—

Devil: Wait, aren't bats blind by default? Isn't a myopic bat a genetic improvement upon the species?

Having lost the beautiful metaphor to the Devil's Domain, we will proceed with a succinct description of the obvious algorithm to find Connected Components in a Graph.

Algorithm 1

Input: Terrorist Graph

Output: Terrorist Graph with labelled Components

```
function begin()
{
1   Give_Up();
}
function begin_again()
{
1   while nodes remain in the Terrorist graph:
2     Let t be a terrorist; visit t, add t to VisitedList, giving t a ComponentId.
3     while VisitedList is not empty:
4       Capture a terrorist from VisitedList and add him to CapturedList,
          giving him same ComponentId.
5       ****
6       For each contact c given above, pay a "friendly" visit, adding c to
          VisitedList.
7     Increment ComponentId.
8   return home
}
```

The above is the standard algorithm in the literature — ok ok, I think the word “standard” might have raised some eyebrows, and might have tested the elasticity of the word, but that can be easily remedied by massaging the eyebrows in order to lower them. Questions are awaited.

Reader: What is the point of function begin?

Author: To make a first attempt at solving the problem, but giving up owing to fear of the repercussions. Therefore, begin_again is entered into after much convincing. This is typical of all hard problems. They go by the

common name of “Not Programming”, from the very words of dismissal that programmers use when asked to program such problems. Often abbreviated to NP or NP-Hard.

Reader: Are you serious?

Author: Haha, no no of course not! But this makes for a more interesting definition of NP and NP-Hard Problems than what is defined in the literature.

Reader: What are those asterisks in Line 5?

Author: The unprintable means that you use to extract the information from the captured terrorist. The asterisks hide the gory details, leaving the means used to the reader’s imagination. It is a blocking statement, because it completes only after the list of the terrorist’s known contacts are given, leading to Line 6.

Reader: Why the VisitedList and CapturedList? Why two lists? Why not just one?

Author: You can revisit a terrorist but capture him only once. It is assumed that captured terrorists do not have the means to make an escape.

Reader: Why would you want to revisit a terrorist?

Author: To get more information.

Reader: Ok. All clear.

In summary, the above algorithm is a straightforward way to start from a given terrorist and follow through all of his known contacts, tarring them all with the same brush, and once this terrorist’s network is fully explored, moving on to a disparate and disconnected terrorist and repeating, with a different brush, the process all over again (because your organization’s name is: “*Kabhi AlQaeda Na Kehna*”).

Problem solved. End of pap— WAIT!! The order in which the terrorists are visited follows either Depth-First Search or Breadth-First Search as popularized in the literature. However, both of these are dimensional travesties upon considerations of efficiency. Therefore, we shall lift ourselves from the three-dimensional shackles that confine us and move into the fourth dimension by aiming for Time-First Search instead... You are supposed to be the ringleader of an ORGANIZATION. And unless, by the word “organization”, you mean the arrangement of your own body organs, it is likely that there are going to be others working with you. So if you alone are out hunting, what do your minions do? You don’t want them idle, if only because they need to be wary of the Devil.

Devil: Hey, did someone invite me?

Author: Speak of the Devil!

Devil: And the Devil comes! So, if you don’t use your minions, then you will be killing all the terrorists yourself? That would make you a serial killer!

And thus, the only way to avoid being labelled for life as a serial killer is to use your minions for you, and thereby be labelled a parallel killer instead.

Enter Parallelism!

SCRRRREEEEECCCCCHHHHHHHH!!!
CRAAAAAAAASSHHHH!!!

Life Lesson #1: Do not read research papers while driving. Let the gory scene of upturned wheels and flaming vehicle-parts and spattered blood and body parts strewn all across the road serve as a dire warning of the fatal consequences thereof.

Having witnessed at first hand the untimely demise of a precious reader, lost to surprise twists of a research paper, a stark reminder of how brutal academia can be, and having observed a minute's silence to commemorate the departed soul, let us now proceed to Section 2 with a cautious step, an acknowledgement of the dangers that embellish our path ahead.

2 MILPSRLAEAL NE SMSAE

I apologize for the title of this section. It's supposed to be "PARALLELISM EN MASSE", but well, this only just goes to show the inherent dangers of unconstrained parallelism. You attempt parallelism in displaying characters, and this is the result. You probably didn't use any locks. But then, if you use too many locks, it would hurt performance severely. Which I guess means that you need to use just the right number of locks. This, dear Readers, is called the GoldiLOCKS zone, haha.

Given the fact that you have many agents in your organization, how should you tackle the task of identifying the connected components in the terrorist graphs? And to make it more simple, let us assume that we already have a list of the names of the terrorists, we just don't know who belongs to which organization (We shall leave the aftermath of identifying this as an exercise in imagination, and concern ourselves with just identifying the connected components). And you have such a large number of employees that you don't even need to go admire the night sky for the stars — you only need to look at your organization from far away, and you can make out shapes of constellations owing to so many of the employees.

Devil: Speaking of constellations, aren't constellations really just another glorified name for connected components of a graph?

What you'd like, what you'd really like is, well, world peace, but since that's a lofty goal, let's aim for something more achievable. So what you'd like is to give one single algorithm for all your employees to follow. What you would NOT want is to give each employee a different algorithm, since that would be a nightmare for you to keep track of who's given what task.

Very crudely, this is the idea of a GPU (Graphics Processing Unit) — a processor that has massive parallelism in the form of “threads” that all execute the same instruction at a time.

Devil: You know you are lying, right? By omission.

Author: Sigh ok ok. To come entirely clean, not all threads are executing the same instruction. Threads are grouped into “warps” and all threads in a warp are executing the same instruction. But those of a different warp need not. But all are MOST DEFINITELY executing the same program — they just need not be at the same line in the program. And furthermore, warps are grouped into thread blocks. It is not ok to launch thread blocks with insufficient work to do, because they would then die off quickly, and you see... Block Lives Matter.

Devil: Groan! So much for pedantry. And puns.

What might, at first glance, seem like an evident and trivial algorithm, is, upon deep reflection (and refraction and diffraction and interference and other optical phenomena as well) not really, and the reader is therefore urged to desist from prematurely presenting an obviously incorrect algorithm.

Reader: Well, it *is* trivial. Just assign each minion to one terrorist in the graph. And each minion visits that node and labels it with a unique component id, and follows that terrorist's contacts, and labels them with the same component id.

Author: You have already been warned that this is not a trivial algorithm. So if we use your approach, when a minion follows a terrorist's contacts, he will reach other terrorists, right?

Reader: Yeah.

Author: Who have already been assigned to some other minions, who all followed the same algorithm, and have already labelled those terrorists with their own component ids.

Reader:

Author:

Reader:

Author:

Reader:

Author:

The above speechlessness proves the statement made earlier that this algorithm is not trivial. After a long enough interval of open-mouthed shock on

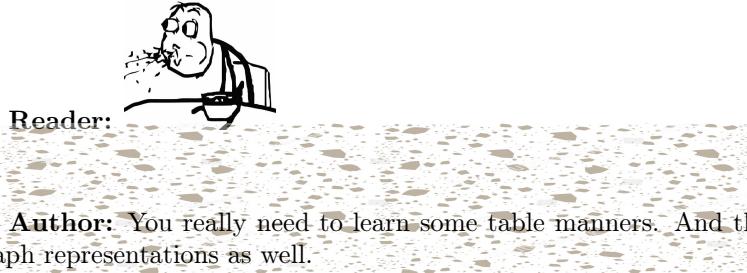
the part of the reader, an interval long enough to accumulate flies in the reader's mouth, necessitating drastic measures to purge them thereof, let us proceed to elicit further suggestions.

Reader: Ok, how about the following? Same as before — each minion gets one terrorist. But now, when that minion reaches the terrorist, maybe he paints the terrorist with some unique colour to act as component id. Then follows the contacts of this terrorist. If he sees an uncoloured contact, he goes ahead and colours that contact because clearly this contact belongs to the same component. And keeps following that contact's contacts and so on, until all nodes are coloured.

Author: And what if this minion finds a contact that has already been coloured by some other minion?

Reader: That means he has already been visited by some other minion, and the two minions really are exploring the same component. So one of the minions has to go back and recolour his terrorist with the other minion's colour, to show that there is only one component.

Author: And suppose the other minion also stumbles upon the first minion's terrorist, and sees him coloured a different colour, indicating that this terrorist was reached by someone else, so he goes back and recolours his original terrorist with the other minion's colour. And we are back to square one, or circle one, if you will, because two contacts are coloured differently, when they should be coloured the same.



Author: You really need to learn some table manners. And that includes graph representations as well.

Reader:

Author:

Reader:

Author:

Despite our repeated attempts at solving the problem, it appears as if some malevolent cosmic entity is operating behind the scenes to ensure failure with high probability, where high probability can be defined as—

Devil: Hey, did you call me?

Author: Oh damn! I was hoping that not using your name would help.

Devil: I go by many names. I am aliased. Just because nobody ever chants my *Sahasranama*, doesn't mean it doesn't exist.

Author: Exorcise yourself!

In the interests of preserving the reader's sanity and time, and preventing the reader from expending an inordinate amount of energy, losing excessive mass in the process (well, owing to Einstein's famous equation) in an attempt to solve the problem, the solution is presented herewith.

Without loss of generality, but with also no real gain of specificity, but with sufficient loss of interesting and vivid descriptions of a colourful nature, let us assume that the terrorists (or nodes in the graph) are labelled from 0 to n-1 (it also helps from a standard fantasy trope point of view, because according to lore, if you give a thing a name, you give it power, and hence, let us not vest too much power into the terrorists, and divest them of their names). Start off by assuming that each terrorist has their component id to be the same as their label. By starting off assuming that each terrorist is operating solo, the goal is to refine this hypothesis to identify who else they are collaborating with, and eventually identify the component id for each terrorist. Each minion is requested to execute the following algorithm:

Algorithm 2: Let's call it... um... PUSH?

Input: Terrorist graph with each node labelled uniquely with its node-id

Output: Terrorist graph with each node labelled with its component id, where component id is the minimum node-id among all the nodes in that component.

```
function CC()
{
    1   If terrorist is painted black, go back home. Forget the rest of this function.
    2   Paint the terrorist black.
    3   Let L be the component id of this terrorist.
    4   **** (You know what to do)
    5   Visit each contact c obtained in the previous step, and do:
    6       If component id of c is more than L, set it to L *** (and
           paint terrorist c white).
}
```

In case you are wondering what the CC stands for (apart from unity, integrity and all that), it is Cafe Coffee Day who has realized that its Days are over and it might as well indulge its remaining life in Graph algorithms and in particular, Connected Components, because you see, a lot can happen over Graphs. The beauty of the above function lies in the fact that it is the same function that can be asked to be carried out by every minion. After all, let us not forget that you are the RINGleader, and hence you would like... ONE RING TO RULE THEM ALL!

ASSUMPTIONS

1. The terrorists are docile enough to not put up a fight and get themselves painted as per the whims of the minion.
2. What are the terrorists doing after you paint them? Yeah, you got that right — they are sitting right there, waiting for you to come back again and check on them to see if they have changed their colours. Of course, they are not just twiddling their thumbs — they are praying for an asteroid to come and deliver them from their miseries (despite many of the terrorists not really believing in, or for that matter, even knowing about, asteroids), or failing that, they are willing to settle for an Amazon agent, given that they seem to be so good at making deliveries.

RIDDLE

Where do terrorists hang out?
The gallows.

With the pride of accomplishment akin to parental pride evident in the public display of their newborn child, let us climb the nearest tallest rooftop in the vicinity and shout out for the whole world to see and bestow their admiration and envy upon our worthy newborn baby algori— oopsie...

AAAAAAAAAAAAAAAHHHHHHHHHHHHHH!
CRRRAAAAAAASSSSSSSSHHHHH!

Life Lesson #2: Do not climb rooftops with babies in the hope of public display. Unless the baby in question is (a) Spiderman, or (b) aerodynamically prolific, or (c) having high coefficient of elasticity. Babies are inclusive of newfound algorithms as well.

Let us once again mourn the passing away of yet another reader (infant readers count). It is evident that the number of readers at the beginning of the paper is most certainly not going to equal that which remains at the end, but let us plod on, hoping that no more fatal casualties ensue. Academia can be brutal. And it is best that the reader is made aware of this fact upfront. Academia is brutal because it stabs you in the back, just like Brutus, after whom

the word “brutal” was coined, since that was what Brutus was known for, and everything else about him is forgotten. Brutus would seriously have blossomed and flourished in academia without breaking a sweat. And then one fine day when nobody was expecting it, he would have gone and joined the industry, having stabbed academia in the back.

Algorithm 2 has the advantage that it can be executed by all minions independently, reducing the overall runtime, so that each minion can go home for dinner (if not be dinner themselves at the hands of the terrorist they capture, but Assumption 1 provides them a protective shield of defense).

The reader’s teetering on the precipice of enquiry is manifestly discernible, and the reader is therefore encouraged to trip and give in to the temptation to satisfy curiosity.

Reader: But, but.... I don’t understand this whole thing. I am not able to pinpoint what, but something seems kind of off.

Author: That’s because the algorithm presented above is incomplete, and I haven’t told you the whole story, haha! I wanted to see the reaction on your face.



Reader:

Life Lesson #3: Do not read research papers while eating or drinking.

Let us shed copious tears at the devastating and unprecedeted loss of yet another callous reader to the brutal and unforgiving jaws of academia and research. Having attended the last rites and disposed of the remains of the eviden— er, I mean, the departed reader, let us plod staunchly ahead with a word of caution and warning to the remaining readers (if any) that by proceeding beyond the following point, they consent to do so at their own risk.



Algorithm 2 (The Rest Of It)

```
function RingLeader()
{
    1   Assign one minion each to each terrorist in the graph.
    2   As long as there is even one terrorist who is not painted black:
    3       Call CC()
}
```

Out of essential courtesy, let us pause to allow the reader to get dazzled at the sheer effulgence and brilliance of this simple algorithm. Sunglasses are offered to protect the reader's eyes from the radiant glow emitted by this algorithm's awesomeness. It is a shameful pity that the first seven wonders of the world are monuments, and to remedy this egregious oversight of leaving out algorithms, it will be agreed that the above algorithm deserves an honoured place in the list. Dear Reader, say Hello to the Eighth Wonder of the World.

Reader: Um, Hello? But I am not sure I understand it still.

Author: Ah, that's because the effulgence of the program is making you close your eyes to the beauty of it.

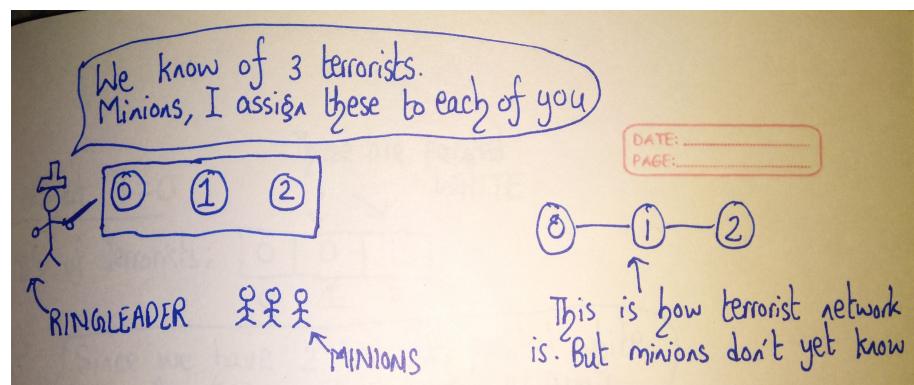
Reader: Could you explain it?

Author: Let me show you instead.

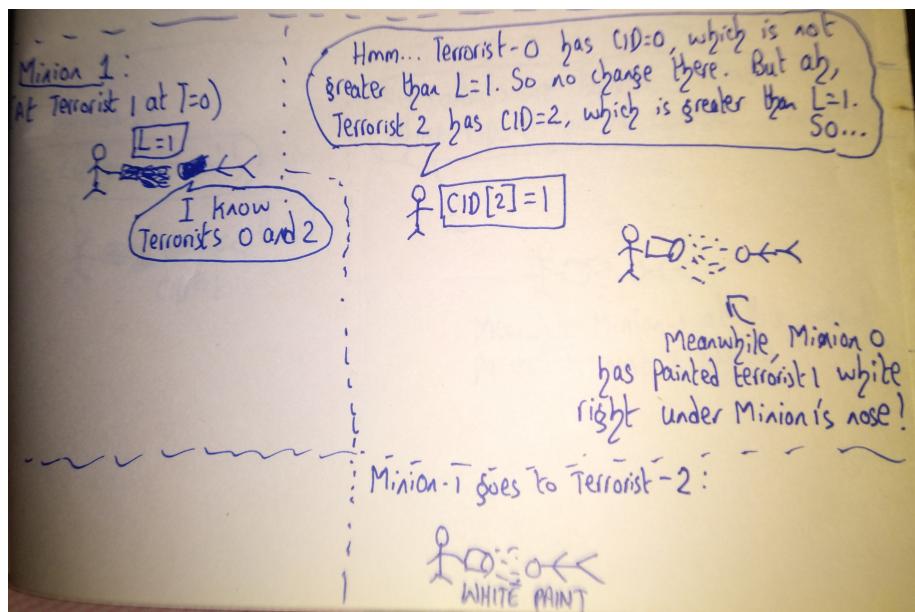
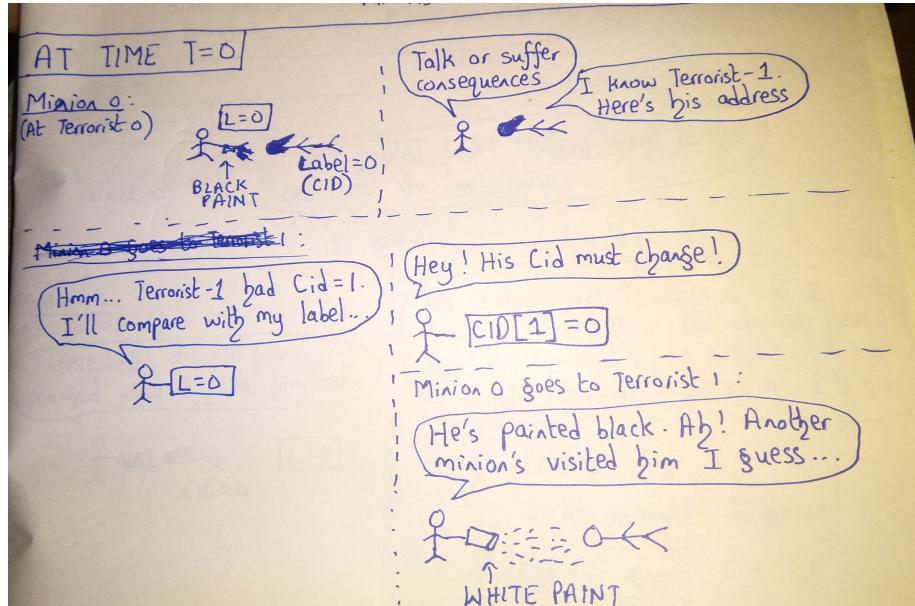
Reader: Huh? Show?

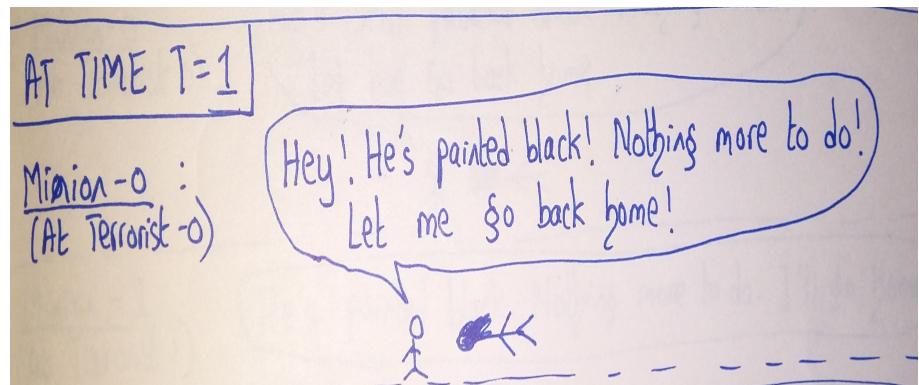
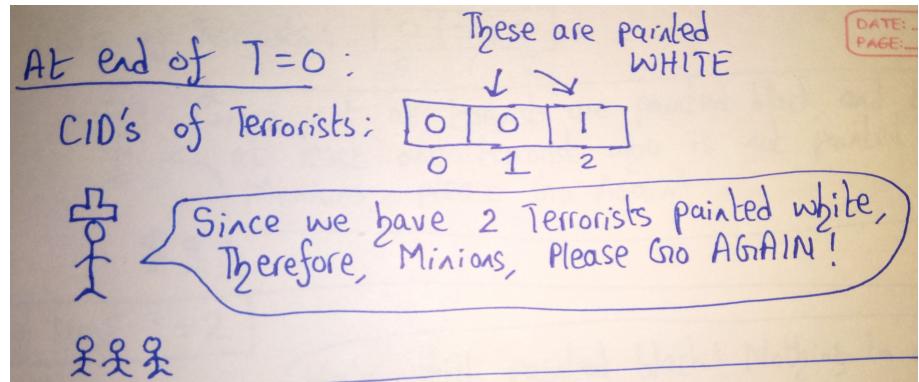
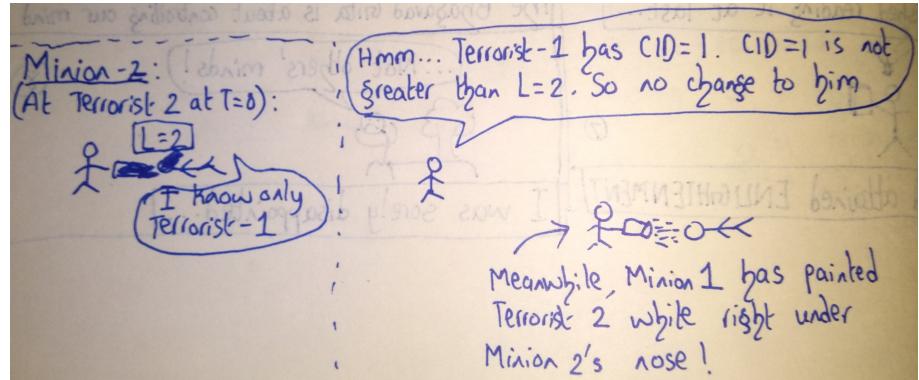
Author: Come, Harry. Into the Pensieve.

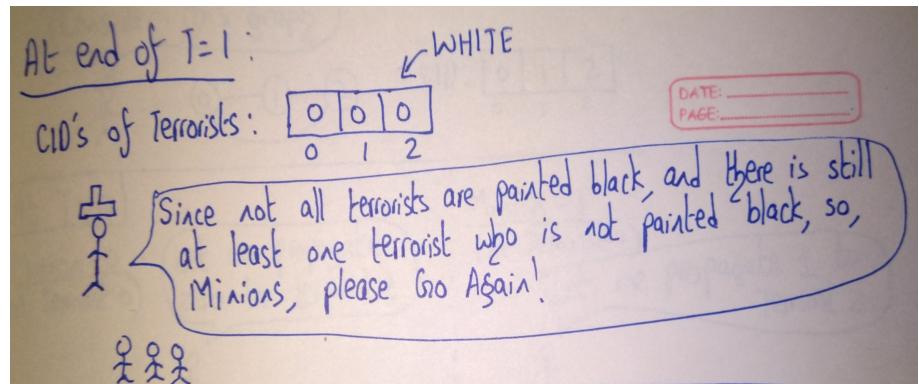
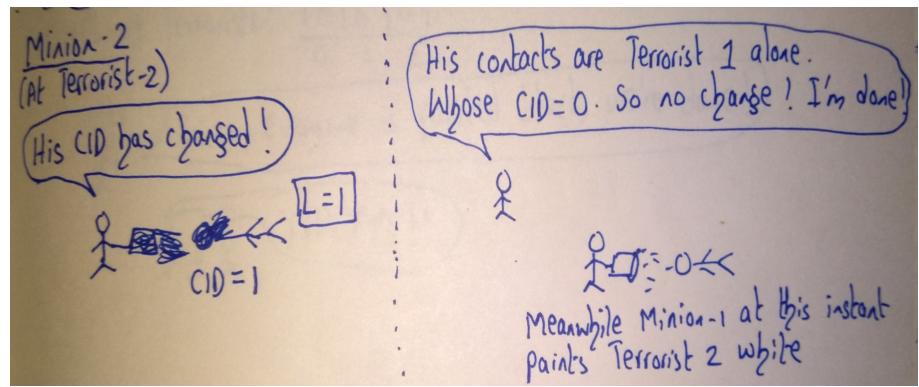
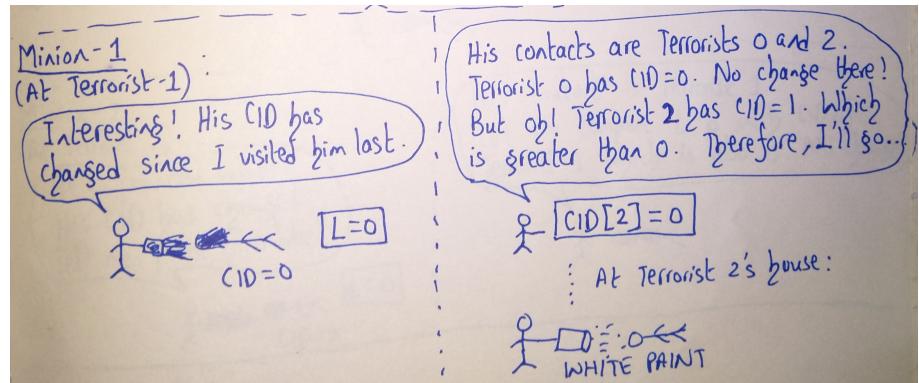
Reader: Harry? Pensieve?¹ Wait, what is happe—
AAAAAAAAAHHHHHHHH!

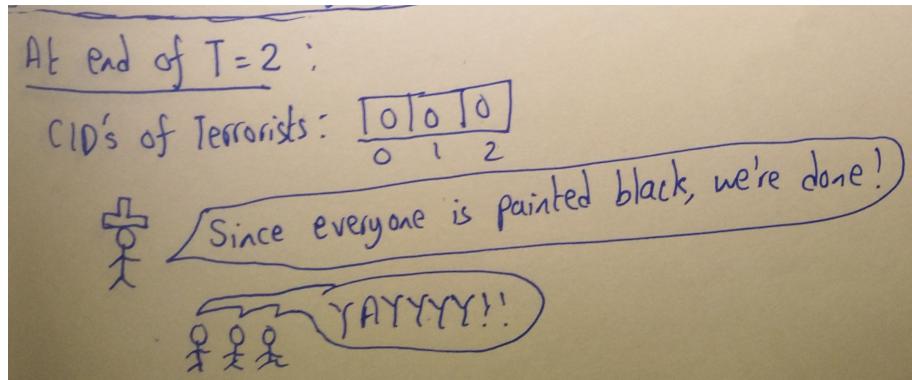
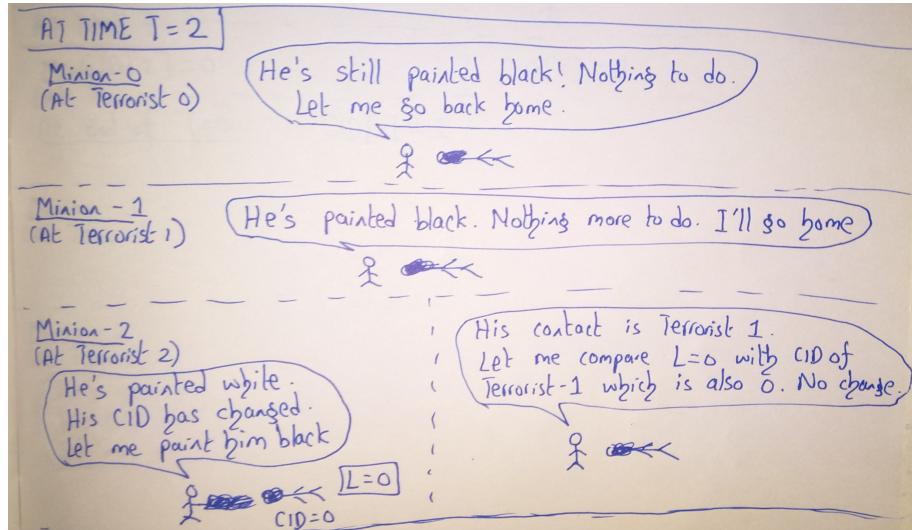


¹A Pensieve is a magical device to store memories in the Harry Potter series [6]. Let us observe 2 minutes of mournful silence for those readers who have wilfully handicapped their own childhood by not reading the Harry Potter series.









In summary, the Ringleader (the CPU) launches the operation (kernel) assigning minions (GPU threads) to each terrorist (node) in the graph. Each terrorist propagates his component id to his neighbours, so that any neighbours who have higher component ids update their component id so that within each connected component, each node will have the id of the lowest numbered node in that component. And when a terrorist's component id gets updated, that terrorist in turn must propagate this update to his neighbours, and thus, this is indicated by painting the terrorist white. A white terrorist indicates that in a subsequent iteration, his value needs to be propagated, whereas a black terrorist (apologies if this sounds overtly racist) indicates that his component id need not be propagated. Thus the algorithm terminates only when all terrorists have been painted black.

Alert Reader: I observed that when one of the minions is updating the

component ids, another minion enters the terrorist's house right under the former's nose and paints the terrorist white.

Author: That's a good observation. This is because the CC function is executed independently by each minion. So while one minion is executing one part of the function, another can be executing another part of it.

Alert Reader: Also, I feel there could be an optimization. You see, for instance at time $T=0$, the CID of Terrorist 1 is updated to 0 by Minion 0, but at the same time, Minion 1 notes the CID value into L as 1 and not zero. In other words, the change made by Minion 0 is not reflected to Minion 1.

Author: Good observation. This is because we have seen a serialized version where we saw Minion 0's operation before Minion 1's operation. But remember that each of them are running in parallel. So it is quite possible that Minion 1 executes before Minion 0. Hence we store the CID value into a temporary variable L, and any changes made to CID are only stored locally by each minion, who then goes back home at the end of that time step (iteration) and writes the changed value to a global place.

AMBUSH!

AAAAAAAHHHHH!!!

Life lesson #4: The above is to teach the reader that it is important to always be cautious while reading papers, because one never knows what is waiting around the corner.

Of course, since the goal of any research paper is to prepare the reader for life, and given that life is full of surprises, the above has been a step in the right direction, and must be applauded. Having given the readers sufficient time to catch their breath, let us proceed with the dialogue...

Alert Reader: We can optimize this, can't we? Maybe let each minion write the CID update onto the terrorist itself! So that when any subsequent minion comes there, the changed value will be propagated within that iteration, thereby reducing the number of iterations.

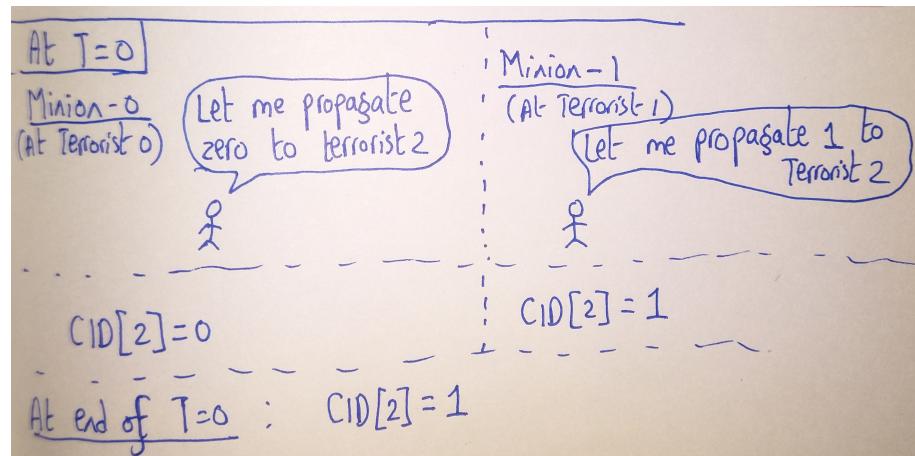
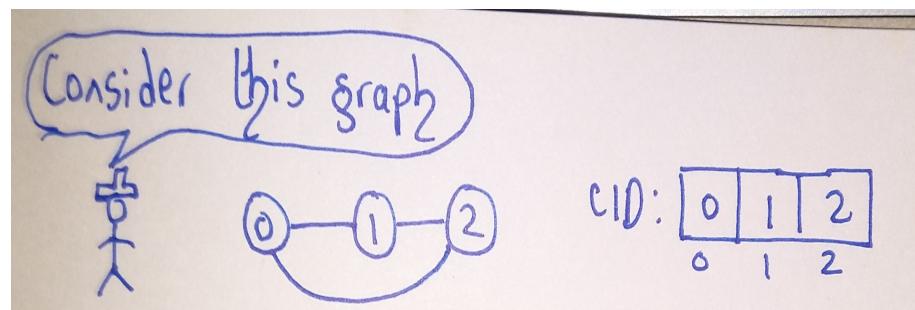
Author: Good. If we were to do that, such an algorithm would be called Asynchronous in the literature. What we have looked at is only the Synchronous case.

Alert Reader: I wondered about those asterisks in Algorithm 2 in the CC function. The last line of that function. Is that yet another place of third degree treatment?

Author: Excellent question. Those asterisks indicate a footnote to the alert reader. And since I am talking to one right now, let me expand on it. It indicates that that line is to be done atomically.

Alert Reader: Atomically? You mean, you go down to the level of atoms and change the positions and momentums of its electrons?

Author: Oh no no. Once again, let me show, rather than tell. Let us enter the subjunctive world where the operation was not atomic.



Author: And now, at this point, Terrorist 2 is stuck with the value 1, and the update by minion 0 is lost. This would have been solved if we insisted that each minion gets exclusive access to the terrorist while comparing the cid value as well as updating it. Then, even if Minion 1 managed to write 1 to $CID[2]$, Minion 0 would have seen that 1 at the time of its comparison and would have managed to overwrite that 1 with a zero eventually. Not like above where each of the minions could read the value of $CID[2]$ at the same time and update it at the same time.

Alert Reader: Ok, but what happens if we try to break up the atomic operation?

Author: Don't do it.

Alert Reader: Eh, how bad can it get? A minion can just interrupt—



Life Lesson #5: Atomic energy is dangerous. Splitting the Atomic can be as catastrophic as splitting the Atom. Respect Atomic Energy.

Alert Reader: Great. So we have solved the problem at last. How elegant! We can all go back home now.

Author: Alas. What if I tell you we have only just begun?

HONK! HONK! HOOOONNNNNNNNNNNKKKKKKKK!!!!

SCREEEEEEECCCCCHHHHHHHH!!!

CRRRRRAAAAAAAASSSSSSSHHHH!

++reader_toll

Life-Lesson #6: Do not read research papers while crossing the road. Look up from your research papers, look to the left, then to the right, cross the road after verifying that it is safe to do so, and then resume reading the research paper.

Aspiring researchers, please take note of the above, lest we have expiring researchers.

3 EXPANDING THE PROBLEM

It is a truth universally acknowledged, that a single problem, in possession of a good solution, must be in want of a modification of the problem statement. Apologies to Jane Austen who is rolling in her grave at seeing her renowned sentence being abused in the above manner. Alas, she probably had too much Pride in her sentences, and too much Prejudice against Connected Components Problems.

But leaving aside Jane Austen's Senses and Sensibilities, could you think of how you could modify the algorithm yet maintain its correctness?

Devil: If you can't Push a terrorist too much, do you just Pull along?

Let us extend the most heartfelt gratitude to the Devil for edging us closer to our goal.

Reader: Ok. I have an idea. When we are comparing a terrorist's cid with that of his contacts, we propagate the source terrorist's cid to his contacts if any of his contacts c have a higher cid than the source. But what if we do the opposite? What if we update the source terrorist's cid if any of his contacts has a lower cid than the source?

No, dear Reader, this is not the end of the paper already. This blank page is a refreshment break, a much needed feature of any research paper that has the reader glued to its pages. Readers are requested to look up from the paper, stare at the horizons for a while, let out their breaths which they had held for such a long time, and after verifying that the pulse rate and breathing rate are normal, turn over to the next page and resume reading.

Reader: Did we reach the end of the paper already?

Author: No, that blankness was the glare of the sheer brightness of your idea which affected the ocular power of the unsuspecting reader. This, dear Reader, is the Pull variant of the algorithm as opposed to the Push variant we saw before.

Algorithm 3: PULL VARIANT

```
function CC()
{
1 Let L be the component id of this terrorist.
2 **** (You know what to do)
3 Visit each contact c obtained in the previous step, and find the minimum
cid among all c's. Call this min.
4 If min < L, update Cid of this terrorist to min and paint him white.
}

function RingLeader()
{
1 Assign one minion each to each terrorist in the graph.
2 As long as there is even one terrorist who is not painted black:
3 Paint all terrorists black.
4 Call CC()
}
```

It is immediately apparent that there is some asymmetry between the push and the pull variants, especially when it comes to painting the terrorists. It is not a completely black and white matter, haha. In an attempt to prevent dizziness on the part of the reader owing to multiple forays into the pensieve, we shall not have another excursion therein, but leave the details to be worked out by the astute reader.

DISEMBODIED VOICE

Why aren't minions going back home in CC function if terrorist is already painted black? Why aren't minions painting terrorist black? What would happen if we follow very similar pattern of the Push variant and let the minions paint the terrorist black within CC, and not let Ringleader paint the terrorists black?

It is a basic premise, implicitly acknowledged in every fantasy, that one is obligated to listen to disembodied voices (if for no other reason that failure to do so might result in their voice becoming the next disembodied voice for others to hear), for they hold the key to many mysterious doors that remain otherwise stubbornly closed. Since this research paper is most unanimously agreed to be categorized under the fantasy genre, it behoves the reader to diligently attempt to follow the mysterious hints posed by the disembodied voice above, for this holds the answer to true understanding.

A detailed answer to all the above questions were given in the appendix, but the paper had to undergo an emergency appendicitis operation, and hence, its appendix is no longer available. The inconvenience is deeply regretted.

Reader: So which of the two versions is the better one?

Devil: Of course, the Push variant.

Reader: No, I think the Pull version is better.

Devil: Push.

Reader: Pull.

Author: How about we don the hat of a researcher and experimentally verify for ourselves? Isn't that much better than pushing around answers down our throats? Or pulling answers from a magic hat?

When you apply the tried and tested principles of inferring from data, you are abiding by the scientific spirit that provides everlasting salvation to its adherents. You might have to cross uncharted mountains, you might have to swim through shark-infested seas, you might have to face fire-breathing dragons, you might have to battle light-saber wielding aliens dropping from the sky through UFOs, but you can rest assured that the conclusions you reach are backed by hard solid data, and are not diaphanous wispy hypotheses drawn from a hat. You will attain the forbidden fruit of knowledge that can only be attained by the scientific spirit. You will then thank those mountains, seas, sharks, dragons, and aliens even if they are incapable of gracefully acknowledging your gratitude (though they might appreciate a bite from the forbidden fruit you are holding in your hand as you are thanking them).

Having gained sufficient experience with torturing confessions out of terrorists, let us now attempt to torture confessions from data, if only because it provides a welcome change from the monotony (although often enough dealing with data can cause far more terror than dealing with terrorists).

Type of Graph	V	E	Push-CC-Time	Pull-CC-Time	Push-Iter	Pull-Iter
gemsec-FB/artist	50515	819090	6.5194ms	9.4974ms	9	8
gemsec-FB/athletes	13866	86811	2.0312ms	2.7543ms	8	8
gemsec-FB/company	14113	52126	1.4808ms	1.8047ms	11	11
gemsec-FB/government	7057	89429	2.5921ms	3.6418ms	7	7
gemsec-FB/new_sites	27917	205964	3.4133ms	5.2294ms	11	10
gemsec-FB/politician	5908	41706	1.6289ms	2.3955ms	11	10
gemsec-FB/public_figure	11565	67038	1.8143ms	2.4578ms	11	10
gemsec-FB/tvshow	3892	17239	1.1815ms	1.3383ms	14	13
ca-CondMat	108300	93439	1.5367ms	2.2929ms	10	10
com-youtube	1157828	2987624	263.57ms	882.53ms	10	11
musae_git	37700	289003	143.44ms	159.07ms	9	7
roadNet-CA	1971281	2766607	92.329ms	79.105ms	554	542
Random	30K	10K	218.32us	196.08us	15	15
Random	100K	10K	109.33us	94.488us	7	7
Random	1M	1025	110.52us	123.25us	2	3
Random	1M	10K	198.13us	177.20us	3	4
Random	1M	100K	406.20us	359.71us	6	6
Clique	100	4950	309.57us	202.73us	2	2
Clique	1000	499500	10.896ms	2.2744ms	2	2
Clique	10K	49995000	179.02ms	92.475ms	2	2
Alternate-Edge	100	50	21.438us	16.638us	2	2
Alternate-Edge	1000	500	19.102us	16.703us	2	2
Alternate-Edge	10K	5000	31.965us	38.077us	2	2
Alternate-Edge	100K	50K	34.653us	24.542us	2	2
Alternate-Edge	1M	500K	165.71us	101.21us	2	2
Alternate-Edge	10M	5M	1.4220ms	824.05us	2	2
Alternate-Edge	50M	2.5M	24.180ms	4.7364ms	2	2
Alternate-Edge	100M	50M	27.113ms	29.344ms	2	2
Alternate-Edge	500M	250M	🐯	🐯	🐯	🐯
Alternate-Edge	1B	500M	🐯	🐯	🐯	🐯
Chain	100	99	941.57us	965.53us	100	100
Chain	1000	999	10.060ms	9.9702ms	1000	1000
Chain	10K	9999	111.12ms	102.14ms	10000	10000
Chain	100K	99999	1.48802s	1.47641s	100000	99986
RMAT	1M	1M	3.8789ms	5.7600ms	8	8

It is a well-known fact that any form of tabular data is completely glossed over by readers, and hence to avoid this indifference, two rows have been populated with dead cats, in the hope that they shall draw the reader's attention, and make them at least give a perfunctory glance at the table, to see if the other rows could hold any clue to the cause of their gruesome deaths. The dead cats are an omen. This provides a very chilling picture, and for this reason, it is very strongly recommended that this picture be placed inside multi-core processors so as to aid in the cooling process of the overheated system. Well, the dead cats in this case are an indication that flaming balls of hydrogen undergoing combustion reactions shine in the firmament, revolving in silent cosmological anticipation for the program to terminate and produce outputs for these inputs ².

As with all important questions in life that attempt to ascertain the relative superiority between multiple competing alternatives, the answer is inevitably,

²Program? Where did that come from, you wonder? Well, for the interested reader, CUDA programs have been written to run the above algorithms, and the times shown above are the results of this program running on an NVIDIA DGX1 Server. More details are skipped in the interests of retaining the readers' sanity and interest. If you are the kind of reader who is interested to know more about those details, you are probably reading the wrong paper

“It depends” (or if not that, then surely it is “None of the above”). So much for the scientific method. It is evident from the above that there really isn’t any clear winner (although it can be ascertained that the number of iterations of the Pull approach never exceeds that of the Push). Therefore, let us resort to the age-old, tried and well-tested principle, when faced with a dilemma of choosing among n alternatives, to introduce an $(n + 1)^{th}$ alternative, so that the original problem has been completely obviated (of course, to be supplanted by another problem, but let’s first bask in the rapture of annihilating the original problem, and pat ourselves on the back for such a commendable idea).

4 THE BEST OF BOTH WORLDS

Having basked in the ephemeral pleasure of having gotten rid of the original problem, long enough to realize that we have introduced another alternative to choose from, let us come to the sober realization that a problem (if not *the* problem) still exists, and we have a long way yet to go.

Reader: But what is the third alternative that you speak of?

Author: An amalgamation of the first two presented before!

Reader: Huh?

And before the reader even has time to draw his breath from the surprise of having received such an answer, or to realize what is just coming his way, we have—

BANG!!!!

Algorithm 4: HYBRID APPROACH

```
function CC()
{
    1   Wake up at 3 am, having tossed and turned on your bed for hours, unable
        to sleep.
    2   Question your career choices at having to go finding terrorists.
    3   Transition to questioning your life choices.
    4   Realize that you are duty-bound to identify the Connected Components
        of the terrorists' networks.
    5   Scream at the fates that have conspired to put you in this position.
    6   While there are no more neighbours attempting to batter down your door:
        7       Pacify the neighbours that your guttural screaming was nothing to
           worry about, and was an umm..., hehe... er... yeah, a car backfiring,
           right.
        8       Finally, get ready for your manhunting sojourn.
        9       Where the mind is without fear, and the head is held high...
       10      Realize that the previous statement wasn't really an algorithmic
           instruction in the truest sense of the phrase, but still... Your thoughts,
           your rules, or whatever...
       11      Get pushed out of your house to find the terrorists, and get pulled by
           your own moral and philosophical dilemmas.
       12      Realize that you have been both "pushed" and "pulled" in the previous
           step, and call this the "Hybrid Approach".
       13      THE END
}
```

Reader: WHAT ON EARTH WAS THAT?

Author: A much needed break from the intensity of the paper.

Reader: Um... So, there is no Hybrid Algorithm then?

Having given the reader enough breathing time to unwind, let us now present:

Algorithm 5: HYBRID APPROACH (the real one, because the previous one was just Pseudocode, living up to its name and being false code)

INPUT: Terrorist graph with each node labelled uniquely with its node-id.
OUTPUT: Terrorist graph with each node labelled with its component id, where component id is the minimum node-id among all the nodes in that component.

```
function CC()
{
1   Let L be the component id of this terrorist.
2   **** (You know what to do)
3   Visit each contact c obtained in the previous step, and find the minimum
      cid among all c's. Call this min.
4   If min < L, update Cid of this terrorist to min and paint him white.
5   Visit all contacts c of this terrorist and if their Cid's are more than min,
      update them to min, painting them white.
}

function RingLeader()
{
1   Assign one minion each to each terrorist in the graph.
2   As long as there is even one terrorist who is not painted black:
3     Paint all terrorists black.
4     Call CC().
}
```

Why would the above algorithm, which is seemingly more complicated than either the Push or the Pull, and moreover, does more work than each of them individually, seem to work better? The key insight lies in the fact that the Hybrid approach propagates the component id across a node and all its neighbours in one iteration, unlike the Push or the Pull approaches which propagate the minimum cid across multiple iterations. It is surmised that this will lead to a reduced number of iterations and hence a reduced execution time (sorry for the phrase, for this has nothing to do with actual terrorist execution, which is a separate matter altogether).

Devil: But couldn't it be possible that though the number of iterations have reduced, each iteration does more work, and hence the actual time increases?

Questions from the Devil typically require careful thought and any flippant answers must be avoided at all costs. Being fully cognizant of the fact that no amount of sacrificial offerings in the form of kittens, or poor innocent beasts or even human beings shall placate the Devil, let us therefore seek our refuge with Data, to save us from eternal perdition at the hands of the Devil.

Type of Graph	V	E	Push-CC-Time	Pull-CC-Time	Hybrid-CC-Time
gemsec-FB/artist	50515	819090	6.5194ms	9.4974ms	36.821ms
gemsec-FB/athletes	13866	86811	2.0312ms	2.7543ms	3.4300ms
gemsec-FB/company	14113	52126	1.4808ms	1.8047ms	1.9672ms
gemsec-FB/government	7057	89429	2.5921ms	3.6418ms	4.8743ms
gemsec-FB/new_sites	27917	205964	3.4133ms	5.2294ms	7.0720ms
gemsec-FB/politician	5908	41706	1.6289ms	2.3955ms	2.8489ms
gemsec-FB/public_figure	11565	67038	1.8143ms	2.4578ms	2.8717ms
gemsec-FB/tvshow	3892	17239	1.1815ms	1.3383ms	1.5726ms
ca-CondMat	108300	93439	1.5367ms	2.2929ms	2.1538ms
com-youtube	1157828	2987624	263.57ms	882.53ms	960.59ms
musae_git	37700	289003	143.44ms	159.07ms	247.81ms
roadNet-CA	1971281	2766607	92.329ms	79.105ms	92.766ms
Random	30K	10K	218.32us	196.08us	179.12us
Random	100K	10K	109.33us	94.488us	94.968us
Random	1M	1025	110.52us	123.25us	140.98us
Random	1M	10K	198.13us	177.20us	175.92us
Random	1M	100K	406.20us	359.71us	380.03us
Clique	100	4950	309.57us	202.73us	319.46us
Clique	1000	499500	10.896ms	2.2744ms	3.9708ms
Clique	10K	49995000	179.02ms	92.475ms	156.06ms
Alternate-Edge	100	50	21.438us	16.638us	18.335us
Alternate-Edge	1000	500	19.102us	16.703us	17.887us
Alternate-Edge	10K	5000	31.965us	38.077us	20.638us
Alternate-Edge	100K	50K	34.653us	24.542us	35.260us
Alternate-Edge	1M	500K	165.71us	101.21us	164.02us
Alternate-Edge	10M	5M	1.4220ms	824.05us	1.4191ms
Alternate-Edge	50M	2.5M	24.180ms	4.7364ms	20.782ms
Alternate-Edge	100M	50M	27.113ms	29.344ms	50.301ms
Alternate-Edge	500M	250M	🐯	🐯	🐯
Alternate-Edge	1B	500M	🐯	🐯	🐯
Chain	100	99	941.57us	965.53us	629.93us
Chain	1000	999	10.060ms	9.9702ms	6.6927ms
Chain	10K	9999	111.12ms	102.14ms	64.185ms
Chain	100K	99999	1.48802s	1.47641s	1.16323s
RMAT	1M	1M	3.8789ms	5.7600ms	5.5498ms



The Data has betrayed us. All readers have been dragged into the darkest crevices of Hell with us. The increment operation on the reader_toll counter has been applied so much that the reader_toll counter itself has overflowed. And apart from this, there are still dead cats in those two rows. For all intents and

purposes, the program has gone into a coma, refusing to listen to the tear-soaked pleas of its bedside relatives to wake up, or failing that, at least smile so that they'll know it's alright.

Moral of the story

Large inputs can cause unprepared programs to go comatose.

We are in a poor state, and readers who have been cursing the Author that this is what comes out of involving Devils in a self-respecting research paper must take solace in the fact that all is not yet lost. Because one must accept the fact that if we are being roasted alive in hell and are feeling the pain, then at least our nervous system is working fine, to be able to feel the pain, and hence all is not yet lost— QED! Kudos, Central Nervous System! Kudos!



Ok ok, that only seemed to have put oil to the fire.

RIDDLE

Q: How do you save one from being roasted in hell?

- (a) Apply burnol
- (b) Freeze hell over
- (c) Exorcise the Devil
- (d) Invoke the blessings of God

God: Say, did someone mention my name?

Author: Hallelujah! We have been saved! Save our Souls, God! SOS!

God: Um... you know what? Aren't souls supposed to be, like, you know, immortal or whatever? So they don't need to be saved really, do they?

Author: Then SOB God, SOB!

God: What! Are you using a swear word against me?



Author: No no, God! I meant, Save our Bodies!

God: Oh well alright. Then, tell me, why are you only looking at the time for CC? Shouldn't you be considering the overall execution time?

And that my dear Readers (or whatever charred remains of Readers there are), is the answer to the Riddle posed above. It is none of the given options (as should the answer to any self-respecting MCQ be). How do you save one from being roasted in hell? Well, by not letting them get into Hell in the first place, of course! Let us retract our steps and consider now the overall execution times, instead of merely the CC time.

The following table shows the overall runtimes of each of the inputs graphs for all the three variants, and like the previous table, for each input graph, the minimum execution time is highlighted in yellow. Oh, and by the way, if you're curious about the input graphs, the first few (until roadNet-CA) are standard real world benchmark graphs taken from the SNAP dataset [14], and the rest are synthetically generated graphs.

Type of Graph	V	E	Push-Time (s)	Pull-Time (s)	Hybrid-Time (s)	Hybrid-Iter
gemsec-FB/artist	50515	819090	1.1653	1.064	0.982	5
gemsec-FB/athletes	13866	86811	1.0826	0.8613	1.0323	5
gemsec-FB/company	14113	52126	0.979	1.0093	1.0853	6
gemsec-FB/government	7057	89429	1.0876	1.2233	0.8433	5
gemsec-FB/new_sites	27917	205964	1.0263	1.061	0.9446	7
gemsec-FB/politician	5908	41706	0.911	1.1803	1.0193	6
gemsec-FB/public_figure	11565	67038	1.26	0.8656	0.7556	6
gemsec-FB/tvshow	3892	17239	0.9536	0.9056	0.9846	8
ca-CondMat	108300	93439	0.9856	1.074	0.9023	5
com-youtube	1157828	2987624	1.8646	2.187	2.361	6
museae_git	37700	289003	1.0273	1.1236	1.075	5
roadNet-CA	1971281	2766607	3.132	2.8853	2.6363	263
Random	30K	10K	1.017	1.078	1.0293	8
Random	100K	10K	1.0533	1.0426	1.0126	4
Random	1M	1025	1.1997	1.277	1.1183	2
Random	1M	10K	1.1286	1.9726	1.496	2
Random	1M	100K	2.0693	1.9586	1.3246	3
Clique	100	4950	1.0526	0.9203	0.9983	2
Clique	1000	499500	0.947	0.8616	0.9166	2
Clique	10K	49995000	9.5026	9.4646	9.68	2
Alternate-Edge	100	50	1.115	0.976	1.098	2
Alternate-Edge	1000	500	0.9936	0.944	1.0303	2
Alternate-Edge	10K	5000	0.9913	0.868	0.9703	2
Alternate-Edge	100K	50K	1.0686	1.03	0.8613	2
Alternate-Edge	1M	500K	1.296	1.2453	1.3796	2
Alternate-Edge	10M	5M	5.4836	4.6967	4.8906	2
Alternate-Edge	50M	2.5M	21.4393	21.222	21.7033	2
Alternate-Edge	100M	50M	42.304	42.0686	41.9266	2
Alternate-Edge	500M	250M				
Alternate-Edge	1B	500M				
Chain	100	99	0.9973	1.0283	0.886	51
Chain	1000	999	1.0343	1.0726	1.1563	501
Chain	10K	9999	2.1616	2.0786	1.388	5001
Chain	100K	99999	33.447	31.3006	17.3906	49999
RMAT	1M	1M	1.443	1.6006	1.551	4



To preempt any critical readers from pointing out that we have the exact same figure repeated four times, and adding the comment that this is not the way to write research papers, let me clarify abundantly that the four figures are NOT the same, because each of them is happening at DIFFERENT instants of time, and this SHOULD be the way to write research papers, thank you very much.

Author: God, you betrayed us! Et tu, God!

God: Dude, don't expect me to spoonfeed you all the time. Just appeal to, um...

Author: Divine Intervention?

God: Yeah right!

Author: That's exactly what I did just now!

God (sheepishly): Ohhhh right! Ok well then, er... appeal to... um...

Author: Please don't tell Diabolical Intervention!

God: No no. Appeal to... you know, er..., yeah right, Curse of Higher Dimensions!

Author: What? That doesn't even make sense in the current context. Except maybe that we have been cursed by someone from Higher Dimensions. I hope you are getting the hint.

God: Ok ok fine. Look, the data is not all that bad. Some of them do support your hypothesis. Appeal to that. Appeal to Data Intervention!

Thus, thanks to Data Intervention, let us now analyze which inputs perform well with the Hybrid approach. We can make the following general observations from the data (any deviations in the data from these observations must be strictly considered to be noise in the data, which doesn't wish to bend itself to analysis or listen to reason):

1. For graphs where small number of nodes have high degree (like power-law graphs), the CC-time of the Push approach is the fastest although in terms of overall execution time, Hybrid sometimes does better.

2. For graphs with very small diameter, the Pull approach performs best. This is because the number of iterations is less, and hence walking through the neighbours (the phrase "walking through the neighbours" is not to be understood in a way that implies that we are ghosts — the neighbours are to be considered abstract entities, and "walking through" as an abstract operation of iteration) will not be too expensive. Hybrid approach also does fairly well, but the additional step of propagating minimum values to all neighbours does not gain much in terms of performance here because anyway, the values are going to converge in a very small number of iterations.

3. For graphs with large diameters, the Hybrid approach shines with the luminosity of a thousand suns! You are requested to wear protective sunglasses before looking at the results of its performance. Clearly, this performance boost is owing to the fact that the algorithm has to run through many iterations before reaching convergence, and hence the Hybrid approach would lead to savings in number of iterations in a large way, and per iteration, it helps in propagating the values further, which helps in the overall execution time as well.

4. The number of iterations with Hybrid approach is always less than (or equal to) the other two variants.

Having provided satisfactory explanations and observations, we can all finally go home happy. Ok, end of pap—

Reader: WAIT!

Author: Now what?

Reader: Um, those two inputs for which the program is crashing... Can't we do something about them?

Author: Well yes, hence the Grim Reaper has dragged away the dead cats from those rows. Because the corpses can't be allowed to remain there rotting throughout the paper.

Reader: No, I mean... The program is crashing or running out of time on those really large inputs.

Author: Look. The state that the program is in, it is basically a vegetable, and you can go ahead and chop it up and make gravy out of it and eat it whole.

DIGRESSION: THE BUTTERFLY EFFECT [11]

The flap of a butterfly's wings in one corner of the world might cause a hurricane in another corner of the world.

DIGRESSION-WITHIN-DIGRESSION:

Why can't you ask a hurricane to slow down?
Because if it did, it would no longer be a hurricane — it would be a relaxed-cane.

An innocuous pun made above has set the gears churning, leading to ideas for what needs to be done. The butterfly has flapped its wings and realizes the full enormity of what it has done, of being single-handedly (or rather, single-wingedly) responsible for creating a tornado on the opposite side of the globe, and frantically recalls Newton's Third Law and begs for Divine Intervention to invalidate the Law, or failing that, at least invent Newton's Fourth Law that provides an exemption clause to the Third Law.

DIGRESSION: Newton's Seventh Law of Motion

Newton's Fourth, Fifth and Sixth Laws of Motion are in constant Motion and cannot be pinned down or stated.

Because the word *chop* used above evokes a train of thought... What if we actually chop up the program? No, that probably doesn't make sense. But what if we chop up the graph?

5 RELATED WORK

[10]

END OF RELATED WORK

Author: Hey, you didn't really read that, did you?

Reader: No, I read it.

Author: Oh, come on!

Reader: Really! I swear, as sure as God is my witness, that I read it.

God: Ahem, I heard that.

Reader: G-G-God?

God: Yup, that's me. And you didn't read that paper.

Author: Aha!

Reader: G-G-God? I thought you didn't exist!

God: I heard that.

Devil: G-G-God? You exist?

God: I heard that too.

Reader: Are you vocabulary challenged?

God: I heard that.

Reader: Are you a machine preprogrammed to repeat just those words?



Reader: Whoa! What was that?

God: A display of my omnipotence.

Reader: But then, if you exist, how could you allow the Devil to exist as well?

God: Oh, um..., er..., *mumble* *mumble*... well, live and let live, you know?

Author: Ok, getting back on track, I urge you to read the reference.

[10]

Author: Now you have reached here too soon. You still didn't read it, did you? God, you can confirm this, can't you?

God: Look, I don't wish to be dragged into this.

Author: But God! You are the omnipresent and omniscient witness of everything! You know everything. You must have been a witness to whether the reader has read this or not.

God: Oh well, um..., er... you see, I wasn't.

Author: WHAT! BUT HOW? The Eye of God witnesses everything in the universe.

God: Er..., Um... Yeah that's true. But well, the Eye of God was a little preoccupied at the moment watching a really interesting sitcom on TV.

Reader: Ok, why don't you all trust me and believe me when I said that I did read the paper?

Author: Ok fine. Then let me ask you a test question to check if you really read the paper. Quick, tell me, what is Differential Privacy?

Reader: Um... Differential privacy is... er..., ok fine. I confess I didn't read it.

Author: Aha! Caught red-handed! Well, you really need to read that reference right now.

Reader: Ok ok fine. I will. But what is Differential Privacy anyway?

Author: I don't know. It certainly isn't mentioned in that reference, haha!



Reader:

6 UNRELATED WORK

While the reader is trying to come to grips, digesting the related work, here is a welcome digression to break the monotony. The reader is strongly encouraged to read the following works, because why should Jack have all work and no fun and become a dull boy, right?

[2], [3], [4], [5], [6], [7], [8], [9].

7 THE PARTING OF THE WAYS

Let us now return refreshed from that wondrous excursion, and come to the sudden unpleasant realization that the reader has probably forgotten the details of this paper. It is an uncomfortable realization on the part of the Author to know that the section on Unrelated Work was probably a mistake to be included, but by now, it is already too late and nothing can be done about it. Nevertheless, it has left the readers in a happy frame of mind, and since we know that the goal of life is happiness, we have nothing to complain about, I guess.

Meta-Moral of the Story

It is ok to make mistakes, maybe sometimes even in research papers.

However, to wrap things up, the Reader is strongly encouraged to ponder about partitioning the graph into multiple subgraphs. Because it is a well-established fact that Graphs are pervasive and growing in size by the day [1]. In other words, Why should all the terrorists be assigned to minions all at once? There probably aren't so many minions anyway, or else they would have been called *maxions*. So we can chop up the graph into different partitions. And let minions work on one such partition at a time. A natural question at this point is: How do we decide how to partition the graph? And how do each of the Push, Pull and Hybrid variants perform in the presence of partitioning?



8 CONCLUSION AND FUTURE WORK

Reader: Hey, what is that sound?

Author: What sound?

Reader: That ominous background music, that bodes something really bad is about to happen very soon. You are encouraged to scream to add to the special effects.

Author: Haha, you must be imagining things.

Reader: No, really. It's increasing in volume. And wait, do you hear that? That sounds like footsteps. A lot of footsteps. That's the sound of a stampede, footsteps of an unpacifiable mob, and they're approaching us!

Author: Yikes! You're right. But I haven't even had a chance to make my conclusion.

Reader: Let's run, because it's going to be a mob lynching.

Pant *Pant* So, to conclude, *pant* the connected components of a *pant* graph ***RUMBLE*** ***RUMBLE*** can be found *pant* ***RUMBLE*** ***RUMBLE*** no, NO NOOOO, I have more research directions to explore... I haven't even spoken about the Future Work, which is to explore Partitioning Strategies. Leave us alone! AAAAAAAAHHHHHHHHH!

THE END

AUTHOR'S AFTERWORD

The Nameless Horrors of Research Papers and What You Can Do About Them (Answer: Nothing)

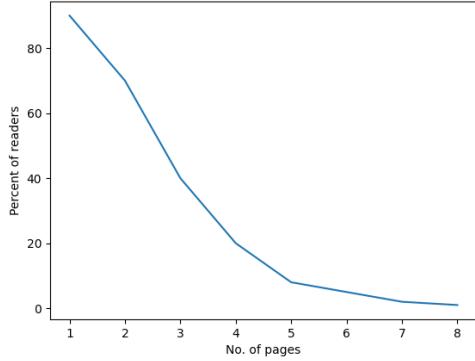


[12]

Yes yes, I know, I know. This is SO not the way to write a research paper, right? But apart from the fact that (a) it is informal, (b) there are way too many jokes, (c) digressions are the norm, (d) there are shocking twists and surprises often leading to catastrophic results, (e) there are life lessons, (f) there are dialogues, (g) there are comics strewn everywhere, (h) there are appearances of a Devil, and God as well, (i) it violates every possible known convention of any standard research paper..., there really is nothing wrong with this paper, and no reason why it should not be accepted as a research paper.

Research papers are known to be dry. The Sahara desert, if sentient, would feel deep jealousy towards most research papers in existence, and might even shed a few tears at its first position for dryness being usurped by many research papers, but then realize that in the process of shedding a few tears, it is no longer as dry as before, lowering its rank in this dry dryness list, and therefore, shed even more tears at this personal calamity, leading to an even further lowering of its rank, *ad infinitum*. THIS research paper, on the other hand, addresses this shortcoming by NOT being so dry. Well, I would rather have my paper be mosaic than prosaic, haha.

There are tons of research papers out there whose opacity is close to zero, inscrutable to all except for an enlightened few. The following graph illustrates:



The above graph presents an alarming picture. The coefficient of viscosity of research papers is so high that more than 50% of the readers have dropped off like flies by the end of page 3. And by the end of the paper, we only have 1 reader (which is probably why most papers have a page limit of 10-12). This is probably the Author who has written the paper himself/herself. This is alarming because it implies that not even the reviewers have been able to survive the vicious density of the paper to reach the end successfully. And often enough, it's either because (a) the paper is not self-contained, or (b) it oozes with terse jargon and deep math. Pure math is not everyone's cup of tea. I mean, it's like asking people to visualize an imaginary geometry of a three-headed dragon that curves in space-time and then proving that it cannot fly simply because breathing fire violates the law of commutativity. I can see you pure math enthusiasts moaning with pleasure at such abstractness, but this isn't for everyone. Because you see, here be the dragons.

Just getting through a page in any standard research paper of respectable quality requires immense concentration. Thus, we could say that the readers of such a paper are put in a concentration camp, haha! I hope to remedy this situation. The principle is simple: You do NOT want your readers to drop off dead like flies by the time they reach page 2 of your paper. Preferably you want them to die after they finish reading your paper. Whether as a result of the reading is of no consequence.

Reader: But this is a half-baked paper. It doesn't even go deep into the topic like any research paper should. Anybody who reads this will get half-baked knowledge.

Hey! This is supposed to be the Author's Afterword, and you aren't supposed to interrupt here. But oh well, seeing that the damage has been done anyway, let me reply to your point. Oh yes, this paper has half-baked knowledge, but that isn't the goal. This paper has half-baked knowledge because midway through the baking process, I suddenly realized that the knowledge wasn't meant to be baked, but rather fried in oil. So this resulted in an abrupt termination of the knowledge-baking process in order that it could be fried. So yes, this paper has

half-baked knowledge, but that's ok because it also gives deep-fried knowledge. And it's been marinated as well. The crispiness is testimony to this. You can't have just one, haha!

Ok ok, fine... for a more formal treatment of the work presented in this paper, the interested and hardened reader is urged to refer to [15].

My goal in writing this paper is straightforward: I wish research to be accessible to the motivated layman, even if it means that the actual quality of the research presented here is not dazzlingly novel. The goal is to etch the concepts presented here so deep into the reader that it becomes part of their DNA, so that the reader's offspring will be born with an innate understanding of these concepts. Ok, jokes aside, it takes one person to make a start to change long-established trends. Perhaps I have been such a person, and this paper has been perhaps an attempt in that direction, albeit maybe not the best attempt. This may not be a work of such monumental profundity and lucidity so as to catalyze a revolution in academia by dawn. But, at the very least... No animals have been harmed in writing this paper. However, a lot of rigour has been sacrificed. For the greater good. It is my hope that this will launch the "Make Research Papers Great Again" Movement. It is my hope that this will inspire many more readers to delve deeper into the field, contribute to it, and more importantly, dispense their learnings to others in an accessible way. May this not lose steam (because then, it would have to be called "Make Research Papers Great Again" Stagnant, haha). I have simply lit the torch. May the light of knowledge be passed on.

ACKNOWLEDGEMENTS

When one goes against the current to write a work of this sort, it is inherently obvious that nobody wishes to be seen associated dead or alive with the Author — imagine the ignominy! Anybody would conclude that a work of this sort must be the attempt of a demented and possessed researcher. But that would be wrong, because aren't all researchers possessed by default (by their institutions at the very least)? Perhaps the Author deserves to be cordoned off with a tape in the interests of public safety. So naturally the Author needs to be cautious while making acknowledgements to a work of this sort. Any names mentioned would automatically become the cast-out members of society, and those poor people, for no fault of theirs (except for making an appearance in an acknowledgement section) would be shunned by society, and probably die begging for foodgrains, clawing out their intestines in hunger because nobody is willing to have anything to do with them, much less provide them food. However, they are the brave few who have rendered help because they know what they are doing is right. So, for whatever it's worth, here goes:

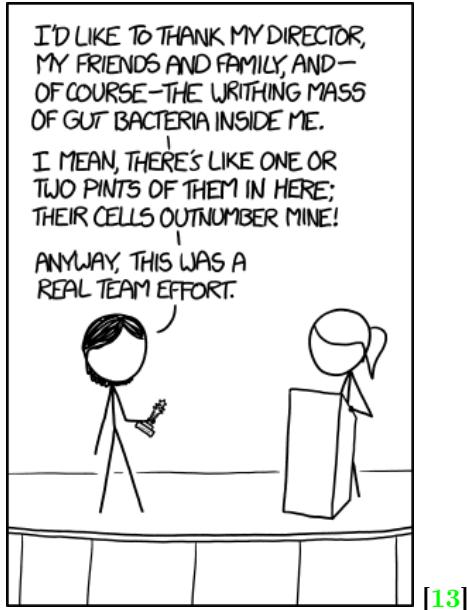
I owe the biggest debt of gratitude and acknowledgement to my mother, Geetha Chandrashekhar. Whatever I am in life today is entirely owing to her and her family (special mention goes to her parents G S Chandrashekhar and Shanthamma, her siblings, Sagar, Premalatha, and Raju, and their families (to take names, so as not to commit the sin of omission – Gayathri, Gouthami, Nikhil, Praveen, Jayashree, Vijay Chandra, Soujanya)). “The roots below the earth claim no rewards for making the branches fruitful” (Rabindranath Tagore) [9]. My mother is this kind of selfless person, who doesn't do a good turn for the sake of acknowledgement, but the branch in this case wishes to convey its gratitude to its root nevertheless.

A lot of teachers right from my childhood (from school to graduate college) have helped foster and encourage my creativity. To all of them, I prostrate with gratitude. I would especially like to acknowledge some of them: Govindarajan Sir, Rupesh Sir, Srikant Sir, Jayant Sir, Deepak Sir, Viraj Sir, NSK Sir, Arindam Sir, Rahul Sir, Dinesh Sir, Prema Ma'am, Jawahar Sir.

And more importantly, a lot of friends have been a staunch pillar of support for me through all my ups and downs. In particular, I would like to mention: Shubham Gupta, Chittaranjan, Varun, Sunil, Jeevan, Pradyumna, Hemanta, Gaurav, Srishty, Julian, AN, Nikitha, Shruti, Kavita, Pooja. There are others as well who wished not to be named. You know who you are, and I shall not embarrass you by naming you here. A huge shout-out to all of you, even if this does not reach your ears.

Now, I know I have ended up writing this Acknowledgement section in an unconventional way — certainly not in the way a research paper typically does. In other words, I have acknowledged not only those who were directly instrumental in bringing out this paper, like the way all research papers do. This is because I believe that any piece of monumental work is composed of many small parts, and it is unfair to only acknowledge the painters who put the

finishing touches of the monument, neglecting to mention all those other unsung contributors, without whom the monument would collapse. In other words...



As mentioned in the beginning of this section, I have also intentionally left out the names of so many people (not because they have not contributed, and not because I have neglected to acknowledge them, but because I don't know how they would react to being publicly named in a work of this sort). If any of you wish to be named (and face the consequences thereof), if you think you have made a contribution and deserve acknowledgement, kindly let me know and I shall do so.

References

- [1] Ullas 😊. "When Pull Comes To Shove... Do Both!" 2022
- [2] Rowling, Joanne K. "Harry Potter and the Philosopher's Stone." *Bloomsbury Publishing*, 1997.
- [3] Rowling, Joanne K. "Harry Potter and the Chamber of Secrets." *Bloomsbury Publishing*, 1998.
- [4] Rowling, Joanne K. "Harry Potter and the Prisoner of Azkaban." *Bloomsbury Publishing*, 1999.

- [5] Rowling, Joanne K. "Harry Potter and the Goblet of Fire." *Bloomsbury Publishing, 2000.*
- [6] Rowling, Joanne K. "Harry Potter and the Order of the Phoenix." *Bloomsbury Publishing, 2003.*
- [7] Rowling, Joanne K. "Harry Potter and the Half-Blood Prince." *Bloomsbury Publishing, 2005.*
- [8] Rowling, Joanne K. "Harry Potter and the Deathly Hallows." *Bloomsbury Publishing, 2007.*
- [9] Tagore, Rabindranath. "Stray Birds." *1916.*
- [10] Sabet, Amir Hossein Nodehi, Zhijia Zhao, and Rajiv Gupta. "Subway: Minimizing data transfer during out-of-GPU-memory graph processing." *Proceedings of the Fifteenth European Conference on Computer Systems, 2020.*
- [11] https://en.wikipedia.org/wiki/Butterfly_effect
- [12] <https://www.smbc-comics.com/?id=3225>
- [13] <https://xkcd.com/1543/>
- [14] SNAP Datasets: Stanford Large Network Dataset. <https://snap.stanford.edu/>
- [15] At the time of this writing, this paper does not yet exist. The author apologizes for leading the reader through a dangling reference, but offers consolation by the promise that this reference will exist in the future (hopefully, subject to paper acceptance constraints).