Dining Philosophers (Short Note)

David Ponarovsky

April 30, 2023

Abstract

This paper presents an alternative proof for the Dining Philosophers problem's impossibility of solving by deterministic protocol. The proof offered in this article is not novel; instead, it is an alternate way of providing the understanding that the protocols that intend to resolve the problem deterministically are bound to fail. By providing a new perspective, the paper aims to help computer science researchers and students understand the problem and the impossibility of deterministic protocols. Overall, the proof highlights the familiar idea that deterministic protocols for resolving the Dining Philosophers problem violate one or more of the necessary properties.

1 Todo.

1. Write the table (sage script). **[COMMENT]** all most done, it's left to handle the parameterized lines.
2. Add definitions. Problem description.
3. Complete the 'proof'.
4. Prove lower bound.
5. Add figures of covering the space by balls and cones.

2 Introduction.

The Dining Philosophers problem is a classic synchronization problem in computer science, where a group of philosophers sit at a round table and alternate between thinking and eating. The problem arises when they share common resources (forks) and a set of rules must be established to prevent deadlocks and starvation.

The impossibility result for the deterministic case states that it is impossible to design a solution to the Dining Philosophers problem if the philosophers behave deterministically and the resource allocation is symmetric. This is because each philosopher would require the same resources as their neighbor at the same time, which leads to deadlocks.Michael Rabin proposed a solution to the randomized case, where the philosophers behave randomly in choosing which fork to pick up first. This randomization breaks the symmetry and prevents deadlocks.

Attempts have also been made to solve the problem using quantumness, with **[COMMENT]** Adi Shamir and Avi Wigderson proposing a quantum analog to the Dining Philosophers problem. In **[COMMENT]** 2003, Dorit Aharonov proposed a quantum solution to the problem, which involves using entanglement to share the forks between philosophers.

An example of a real-world application of the Dining Philosophers problem is in resource allocation in computer networks, where multiple nodes may need access to a shared resource. The problem can also be used as a teaching tool in computer science to illustrate the importance of synchronization and avoiding deadlocks.

An incident related to the problem occurred in [COMMENT] 2008, when a bug in the synchronization code of the iPhone's email application caused it to hang, leading to a flurry of frustrated complaints from iPhone users dubbed the "Dining Philosopher's bug".

Theorem 1. *It is impossible to solve the Dining Philosophers problem for any number of philosophers larger than one.*

We prove the theorem by induction. First, we show that the base case of two philosophers is impossible to solve.

Claim 1. *It is impossible to solve the Dining Philosophers problem for a pair of philosophers.*

Proof. Suppose there are two philosophers sitting at a table, each with a fork. Each philosopher needs both forks to eat, but only one fork is available to each philosopher. Thus, they will forever wait for the other philosopher to release the fork, resulting in a deadlock. □

Claim 2. *If there exists an odd number $n > 1$ such that there is a valid protocol to solve the Dining Philosophers problem for n philosophers, then there exists an even number $n' = n + 1$ such that the problem can be solved by a protocol.*

Proof. Let $P_1, P_2, ..., P_n$ denote the philosophers sitting around the table, and let $F_1, F_2, ..., F_n$ denote the forks placed between them, with fork F_i between philosophers P_i and P_{i+1} (with $P_{n+1} = P_1$). We add an additional fork F_{n+1} between philosophers P_1 and $P_{\frac{n}{2}+1}$. That is, for $1 \leq i \leq \frac{n}{2}$, fork F_{n+i} is placed between philosophers P_i and $P_{i+\frac{n}{2}}$.

Define the rules of the protocol as follows:

1. Each philosopher can only pick up one fork at a time.
2. In order to eat, a philosopher must have both the fork to their left and the fork to their right in their possession.
3. Philosophers $P_1, P_2, ..., P_n$ take turns picking up one fork at a time, in a clockwise direction starting from P_1.
4. When a philosopher has F_1 and F_2 in their possession, they can also pick up F_{n+1} and eat.
5. When a philosopher has F_{n+1} in their possession, they must put it down and wait for their next turn to attempt to pick it up again.

It is easy to verify that these rules ensure that the philosophers can eat when their respective two forks are available, and that there is no deadlock. Since there is a protocol for an odd number of philosophers, there must exist a protocol for an even number of philosophers. □

Claim 3. *The existence of such even number follows a valid solution for the base case of only two philosophers.*

Proof. Suppose there is an even number $n = 2m$. We show that a protocol exists in which two Philosophers simulate the n Philosophers in the original protocol.

Let the original protocol have n Philosophers and n forks, labeled $F_0, F_1, ..., F_{n-1}$, where F_i is between Philosopher i and Philosopher $(i + 1) \mod n$. In the new protocol, the two simulating Philosophers, $P1$ and $P2$, will each be responsible for simulating a continuous half of the original Philosophers, i.e., $P1$ will simulate Philosophers 0 to $m - 1$ and $P2$ will simulate Philosophers m to $n - 1$.

To do this, $P1$ will define $m - 1$ imaginary forks, labeled $I_1, I_2, ..., I_{m-1}$, where I_i is between $P1$ and $P2$. Similarly, $P2$ will define $m - 1$ imaginary forks, labeled $I'_1, I'_2, ..., I'_{m-1}$, where I'_i is also between $P1$ and $P2$. Note that I_1 and I'_1 are shared forks, held by both $P1$ and $P2$.

1. $P1$ and $P2$ both pick up their left-hand fork and I_1.
2. $P1$ picks up their right-hand fork and I_2.

3. $P2$ picks up their right-hand fork and I_2'.
4. Both $P1$ and $P2$ now have two forks, and can eat.
5. After eating, $P1$ and $P2$ put down their forks and I_1.
6. $P1$ puts down their right-hand fork and I_2.
7. $P2$ puts down their right-hand fork and I_2'.
8. $P1$ and $P2$ are now ready to repeat the process.

It can be seen that this protocol never allows for a deadlock to occur, as both $P1$ and $P2$ are only ever holding two forks and either $P1$ or $P2$ will always be able to obtain another fork when they need it. Thus, the claim is proved.

$\square$