Advanced Algorithms
COMP4121

Aleks Ignjatovic

School of Computer Science and Engineering
University of New South Wales

Database Access Problem

Database Access Problem

(More details can be found on pages 707–714 in *Algorithm Design* by Kleinberg and Tardos.)

- Assume that n processes want to access a database, and that the time is discrete, $t = 1, 2, 3, \ldots$
- If at any instant t two or more processes (simultaneously) request access, there is a conflict and all processes are locked out of access.
- We assume that the processes cannot communicate with each other to agree on a joint strategy; each process has to decide if it will request access to database on its own, in isolation of other processes.
- One possible approach in such a situation is that each process at each instant t tosses a coin which produces outcome "request access" with a probability p and "do not request access" with probability $1 - p$.
- How should we choose p to maximise the probability of a successful access to the database for a process at any instant t?

Database Access Problem

(More details can be found on pages 707-714 in *Algorithm Design* by Kleinberg and Tardos.)

- Assume that n pro[illegible]nt to access a database, and that the time is discrete, $t =$ [illegible]..
- If at any instant t [illegible]sses (simultaneously) request access, there is a conflict and all processes are locked out of access.
- We assume that the processes cannot communicate with each other to agree on a joint strategy; each process has to decide if it will request access to database on its own, in isolation of other processes.
- One possible approach in such a situation is that each process at each instant t tosses a coin which produces outcome "request access" with a probability p and "do not request access" with probability $1 - p$.
- How should we choose p to maximise the probability of a successful access to the database for a process at any instant t?

Database Access Problem

(More details can be found on pages 707-7[illegible]4 in *Algorithm Design* by Kleinberg and Tardos.)

- Assume that n pro[illegible]nt to access a database, and that the time is discrete, $t =$ [illegible]..
- If at any instant t [illegible]sses (simultaneously) request access, there is a conflict and all processes are locked out of access.
- We assume that the processes cannot communicate with each other to agree on a joint strategy; each process has to decide if it will request access to database on its own, in isolation of other processes.
- One possible approach in such a situation is that each process at each instant t tosses a coin which produces outcome "request access" with a probability p and "do not request access" with probability $1 - p$.
- How should we choose p to maximise the probability of a successful access to the database for a process at any instant t?

Database Access Problem

(More details can be found on pages 707-714 in *Algorithm Design* by Kleinberg and Tardos.)

- Assume that n processes want to access a database, and that the time is discrete, $t = 1, 2, 3, \ldots$
- If at any instant t two or more processes (simultaneously) request access, there is a conflict and all processes are locked out of access.

- We assume that the processes cannot communicate with each other to agree on a joint strategy; each process has to decide if it will request access to database on its own, in isolation of other processes.

- One possible approach in such a situation is that each process at each instant t tosses a coin which produces outcome "request access" with a probability p and "do not request access" with probability $1 - p$.
- How should we choose p to maximise the probability of a successful access to the database for a process at any instant t?

Database Access Problem

(More details can be found on pages 707–714 in *Algorithm Design* by Kleinberg and Tardos.)

- Assume that n pro[illegible]nt to access a database, and that the time is discrete, $t =$ [illegible] ..
- If at any instant t [illegible]sses (simultaneously) request access, there is a conflict and all processes are locked out of access.
- We assume that the processes cannot communicate with each other to agree on a joint strategy; each process has to decide if it will request access to database on its own, in isolation of other processes.
- One possible approach in such a situation is that each process at each instant t tosses a coin which produces outcome "request access" with a probability p and "do not request access" with probability $1 - p$.
- How should we choose p to maximise the probability of a successful access to the database for a process at any instant t?

Database Access Problem

(More details can be found on pages 707-714 in *Algorithm Design* by Kleinberg and Tardos.)

- Assume that n processes want to access a database, and that the time is discrete, $t = 1, 2, 3, \ldots$.
- If at any instant t two or more processes (simultaneously) request access, there is a conflict and all processes are locked out of access.
- We assume that the processes cannot communicate with each other to agree on a joint strategy; each process has to decide if it will request access to database on its own, in isolation of other processes.
- One possible approach in such a situation is that each process at each instant t tosses a coin which produces outcome "request access" with a probability p and "do not request access" with probability $1 - p$.
- How should we choose p to maximise the probability of a successful access to the database for a process at any instant t?

Database Access Problem

- For a given probability p, the probability of the success of a process i at any instant t is $\mathcal{P}(\mathcal{S}(i,t)) = p(1-p)^{n-1}$, because a process i requests access with probability p and the probability that no other process has requested access is $(1-p)^{n-1}$.
- For what value of p does $\mathcal{P}(\mathcal{S}(i,t))$ have the largest possible value?

- Extremal points are found by solving

$$\frac{d}{dp}\mathcal{P}(\mathcal{S}(i,t)) = (1-p)^{n-1} - p(n-1)(1-p)^{n-2} = 0$$

- Dividing both sides by $(1-p)^{n-2}$ we get that the above equality holds just in case $1-p-p(n-1)=0$ which is equivalent to $p = 1/n$, as one might expect.

Database Access Problem

- For a given probability p, the probability of the success of a process i at any instant t is $\mathcal{P}(\mathcal{S}(i,t)) = p(1-p)^{n-1}$, because a process i requests access with probability p and the probability that no other process has requested access is $(1-p)^{n-1}$.
- For what value of p does $\mathcal{P}(\mathcal{S}(i,t))$ have the largest possible value?

- Extremal points are found by solving

$$\frac{d}{dp}\mathcal{P}(\mathcal{S}(i,t)) = (1-p)^{n-1} - p(n-1)(1-p)^{n-2} = 0$$

- Dividing both sides by $(1-p)^{n-2}$ we get that the above equality holds just in case $1 - p - p(n-1) = 0$ which is equivalent to $p = 1/n$, as one might expect.

- For a given probability p, the probability of the success of a process i at any instant t is $\mathcal{P}(\mathcal{S}(i,t)) = p(1-p)^{n-1}$, because a process i requests access with probability p and the probability that no other process has requested access is $(1-p)^{n-1}$.
- For what value of p does $\mathcal{P}(\mathcal{S}(i,t))$ have the largest possible value?
- Extremal points are found by solving

$$\frac{d}{dp}\mathcal{P}(\mathcal{S}(i,t)) = (1-p)^{n-1} - p(n-1)(1-p)^{n-2} = 0$$

- Dividing both sides by $(1-p)^{n-2}$ we get that the above equality holds just in case $1 - p - p(n-1) = 0$ which is equivalent to $p = 1/n$, as one might expect.

Database Access Problem

- For a given probability p, the probability of the success of a process i at any instant t is $\mathcal{P}(\mathcal{S}(i,t)) = p(1-p)^{n-1}$, because a process i requests access with probability p and the probability that no other process requested access is $(1-p)^{n-1}$.
- For what value of p does $\mathcal{P}(\mathcal{S}(i,t))$ have the largest possible value?
- Extremal points are found by solving

$$\frac{d}{dp}\mathcal{P}(\mathcal{S}(i,t)) = (1-p)^{n-1} - p(n-1)(1-p)^{n-2} = 0$$

- Dividing both sides by $(1-p)^{n-2}$ we get that the above equality holds just in case $1-p-p(n-1)=0$ which is equivalent to $p = 1/n$, as one might expect.

Database Access Problem

- So for the optimal [illegible] $1/n$ the probability of success for a process i at an inst[illegible]

$$\mathcal{P}(\mathcal{S}(i,t)) = p(1-p)^{n-1} = \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1}$$

- To establish the asymptotic behaviour of $\mathcal{P}(\mathcal{S}(i,t))$ the following two facts are useful; both can be proved by establishing the signs of the relevant first derivatives:

1. $\left(1-\frac{1}{n}\right)^{n}$ increases monotonically from $1/4$ up to $1/e$ as n increases from 2 to ∞ ($e = 2.718\ldots$).
2. $\left(1-\frac{1}{n}\right)^{n-1}$ decreases monotonically from $1/2$ down to $1/e$ as n increases from 2 to ∞.

Database Access Problem

- So for the optimal value $p = 1/n$ the probability of success for a process i at an instant t is

$$\mathcal{P}(\mathcal{S}(i,t)) = p(1-p)^{n-1} = \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1}$$

- To establish the asymptotic behaviour of $\mathcal{P}(\mathcal{S}(i,t))$ the following two facts are useful; both can be proved by establishing the signs of the relevant first derivatives.

1. $\left(1-\frac{1}{n}\right)^{n}$ increases monotonically from $1/4$ up to $1/e$ as n increases from 2 to ∞ ($e = 2.718\ldots$).
2. $\left(1-\frac{1}{n}\right)^{n-1}$ decreases monotonically from $1/2$ down to $1/e$ as n increases from 2 to ∞.

Database Access Problem

- Thus, since we had

$$\cdots = \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1}$$

we obtain

$$\frac{1}{n}\cdot\frac{1}{e} \leq \mathcal{P}(\mathcal{S}(i,t)) \leq \frac{1}{n}\cdot\frac{1}{2}$$

- Thus $\mathcal{P}(\mathcal{S}(i,t)) = \Theta\left(\frac{1}{n}\right)$.
- So after t many instants, what is the probability that a particular process has succeeded?
- It is actually easier to estimate the probability of failure after t instants:

Database Access Problem

- Thus, since we had

$$) = \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}$$

  we obtain

$$\frac{1}{n} \cdot \frac{1}{e} \leq \mathcal{P}(\mathcal{S}(i,t)) \leq \frac{1}{n} \cdot \frac{1}{2}$$

- Thus $\mathcal{P}(\mathcal{S}(i,t)) = \Theta\left(\frac{1}{n}\right)$.

- So after t many instants, what is the probability that a particular process has succeeded?

- It is actually easier to estimate the probability of failure after t instants:

Database Access Problem

- Thus, since we had

$$\ldots) = \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}$$

we obtain

$$\frac{1}{n} \cdot \frac{1}{e} \leq \mathcal{P}(\mathcal{S}(i,t)) \leq \frac{1}{n} \cdot \frac{1}{2}$$

- Thus $\mathcal{P}(\mathcal{S}(i,t)) = \Theta\left(\frac{1}{n}\right)$.

- So after t many instants, what is the probability that a particular process has succeeded?

- It is actually easier to estimate the probability of failure after t instants:

- Thus, since we had

$$) = \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1}$$

we obtain

$$\frac{1}{n}\cdot\frac{1}{e} \leq \mathcal{P}(\mathcal{S}(i,t)) \leq \frac{1}{n}\cdot\frac{1}{2}$$

- Thus $\mathcal{P}(\mathcal{S}(i,t)) = \Theta\left(\frac{1}{n}\right)$.

- So after t many instants, what is the probability that a particular process has succeeded?

- It is actually easier to estimate the probability of failure after t instants:

Database Access Problem

$$\mathcal{P}\left(\text{failure after } t \text{ instants}\right) = \left(1 - \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}\right)^{t}$$

- Using the second of the inequalities we get

$$\mathcal{P}\left(\text{failure after } t \text{ instants}\right) \approx \left(1 - \frac{1}{e\,n}\right)^{t}$$

- We can now observe a somewhat strange phenomenon:

  - If we chose $t = e\,n$ many consecutive instants (we ignore floors or ceilings necessary to make quantities integers) then the probability of a failure is quite large, because

$$\mathcal{P}\left(\text{failure after } t = e\,n \text{ instants}\right) \approx \left(1 - \frac{1}{e\,n}\right)^{e\,n} \approx \frac{1}{e}$$

which is quite large, larger than $1/3$.

Database Access Problem

$$\mathcal{P}\,(\text{failure after } t \text{ instants}) = \left(1 - \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}\right)^t$$

- Using the second o[illegible] inequalities we get

$$\mathcal{P}\,(\text{failure after } t \text{ instants}) \approx \left(1 - \frac{1}{e\,n}\right)^t$$

- We can now observe a somewhat strange phenomenon:
  - If we chose $t = e\,n$ many consecutive instants (we ignore floors or ceilings necessary to make quantities integers) then the probability of a failure is quite large, because

$$\mathcal{P}\,(\text{failure after } t = e\,n \text{ instants}) \approx \left(1 - \frac{1}{e\,n}\right)^{e\,n} \approx \frac{1}{e}$$

which is quite large, larger than $1/3$.

Database Access Problem

$$\mathcal{P}\,(\text{failure after } t \text{ instants}) = \left(1 - \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}\right)^{t}$$

- Using the second of the inequalities we get

$$\mathcal{P}\,(\text{failure after } t \text{ instants}) \approx \left(1 - \frac{1}{e\,n}\right)^{t}$$

- We can now observe a somewhat strange phenomenon:

  - If we chose $t = e\,n$ many consecutive instants (we ignore floors or ceilings necessary to make quantities integers) then the probability of a failure is quite large, because

$$\mathcal{P}\,(\text{failure after } t = e\,n \text{ instants}) \approx \left(1 - \frac{1}{e\,n}\right)^{e\,n} \approx \frac{1}{e}$$

which is quite large, larger than $1/3$.

$$\mathcal{P}\,(\text{failure after } t \text{ instants}) = \left(1 - \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}\right)^{t}$$

- Using the second o[illegible] inequalities we get

$$\mathcal{P}\,(\text{failure after } t \text{ instants}) \approx \left(1 - \frac{1}{e\,n}\right)^{t}$$

- We can now observe a somewhat strange phenomenon:

  - If we chose $t = e\,n$ many consecutive instants (we ignore floors or ceilings necessary to make quantities integers) then the probability of a failure is quite large, because

$$\mathcal{P}\,(\text{failure after } t = e\,n \text{ instants}) \approx \left(1 - \frac{1}{e\,n}\right)^{e\,n} \approx \frac{1}{e}$$

which is quite large, larger than $1/3$.

- However, if we increase the number of instants only slightly, by taking $t = e\,n \cdot 2\ln n$, then

$$
\begin{aligned}
\mathcal{P}\,(\text{failure after } e\,n \cdot 2\ln n \text{ instants}) &\approx \left(1 - \frac{1}{e\,n}\right)^{e\,n\,2\ln n} \\
&= \left(\left(1 - \frac{1}{e\,n}\right)^{e\,n}\right)^{\ln n^2} \\
&\approx \left(\frac{1}{e}\right)^{\ln n^2} \\
&= \frac{1}{n^2}
\end{aligned}
$$

- Thus, a slight increase in the number of time instants, from $e\,n$ to $2e\,n\,\ln n$ caused a dramatic reduction in the probability of failure of a process to access the database.

- However, if we increase the number of instants only slightly, by taking $t = e\,n \cdot 2\ln n$, then

$$\mathcal{P}\,(\text{failure after } e\,n \cdot 2\ln n \text{ instants}) \approx \left(1 - \frac{1}{e\,n}\right)^{e\,n\,2\ln n}$$

$$= \left(\left(1 - \frac{1}{e\,n}\right)^{e\,n}\right)^{\ln n^2}$$

$$\approx \left(\frac{1}{e}\right)^{\ln n^2}$$

$$= \frac{1}{n^2}$$

- Thus, a slight increase in the number of time instants, from $e\,n$ to $2e\,n\,\ln n$ caused a dramatic reduction in the probability of failure of a process to access the database.

Database Access Problem

- What is the probability that at least one process has failed to access data base af[illegible] n instants?
- If probability of fai[illegible]ch process is less than $1/n^2$ and there are n processes, th[illegible]bability that at least one process failed cannot be larger than $n \cdot \frac{1}{n^2} = \frac{1}{n}$.
- Thus after $2e\,n \ln n$ instants all processes succeeded to access the data base with probability at least $1 - 1/n$.
- So if the processes could communicate and agree upon the order in which they would access the database, it would take n instants for all of them to access the database.
- If they use randomised process which does not require any communication, then they will be able to access the database with probability $1 - 1/n$ in time which is larger only by a relatively small factor of $2e \ln n$.

Database Access Problem

- What is the probability that at least one process has failed to access data base after n instants?
- If probability of failure of each process is less than $1/n^2$ and there are n processes, then the probability that at least one process failed cannot be larger than $n \cdot \frac{1}{n^2} = \frac{1}{n}$.

- Thus after $2e\, n \ln n$ instants all processes succeeded to access the data base with probability at least $1 - 1/n$.

- So if the processes could communicate and agree upon the order in which they would access the database, it would take n instants for all of them to access the database.

- If they use randomised process which does not require any communication, then they will be able to access the database with probability $1 - 1/n$ in time which is larger only by a relatively small factor of $2e \ln n$.

- What is the probability that at least one process has failed to access data base af[illegible] n instants?
- If probability of fai[illegible]ch process is less than $1/n^2$ and there are n processes, th[illegible]bability that at least one process failed cannot be larger than $n \cdot \frac{1}{n^2} = \frac{1}{n}$.
- Thus after $2e\, n \ln n$ instants all processes succeeded to access the data base with probability at least $1 - 1/n$.
- So if the processes could communicate and agree upon the order in which they would access the database, it would take n instants for all of them to access the database.
- If they use randomised process which does not require any communication, then they will be able to access the database with probability $1 - 1/n$ in time which is larger only by a relatively small factor of $2e \ln n$.

- What is the probability that at least one process has failed to access data base after $2e\,n \ln n$ instants?
- If probability of failure for each process is less than $1/n^2$ and there are n processes, then the probability that at least one process failed cannot be larger than $n \cdot \frac{1}{n^2} = \frac{1}{n}$.
- Thus after $2e\,n\ \ln n$ instants all processes succeeded to access the data base with probability at least $1 - 1/n$.
- So if the processes could communicate and agree upon the order in which they would access the database, it would take n instants for all of them to access the database.
- If they use randomised process which does not require any communication, then they will be able to access the database with probability $1 - 1/n$ in time which is larger only by a relatively small factor of $2e\ \ln n$.

Database Access Problem

- What is the probability that at least one process has failed to access data base af[illegible] n instants?
- If probability of fai[illegible]ch process is less than $1/n^2$ and there are n processes, th[illegible]bability that at least one process failed cannot be larger than $n \cdot \frac{1}{n^2} = \frac{1}{n}$.
- Thus after $2e\,n\,\ln n$ instants all processes succeeded to access the data base with probability at least $1 - 1/n$.
- So if the processes could communicate and agree upon the order in which they would access the database, it would take n instants for all of them to access the database.
- If they use randomised process which does not require any communication, then they will be able to access the database with probability $1 - 1/n$ in time which is larger only by a relatively small factor of $2e\,\ln n$.