random routing

Valiant 1981

one of the great classics

the historic idea of ‚random routing‘

what did the masters of freight stations do
when the station was congested by too many cars?

the historic idea of ‚random routing‘

what did the masters of freight stations do
when the station was congested by too many cars?

- for enough cars p
  - hang car p with destination $\pi(p)$...
  - on random train with destination $\rho(p)$
  - then from station $\rho(p)$
  - route car p to its original destination $\pi(p)$

the historic idea of ‚random routing‘

what did the masters of freight stations do
when the station was congested by too many cars?

- for enough cars p
  - hang car p with destination $\pi(p)$...
  - on random train with destination $\rho(p)$
  - then from station $\rho(p)$
  - route car p to its original destination $\pi(p)$

Valiant 1981:

- hypercube networks
  - 2^n nodes, diameter n
- route for each node p a packet to $\pi(p)$
  - π permutation of nodes
- with random routing for a constant c
  - run time $> nc$ extremely unlikely

n dimensional hypercubes

indexing of bits strings: $u = u[1:n]$

hypercubes: graphs $H_n = (V_n, E_n)$

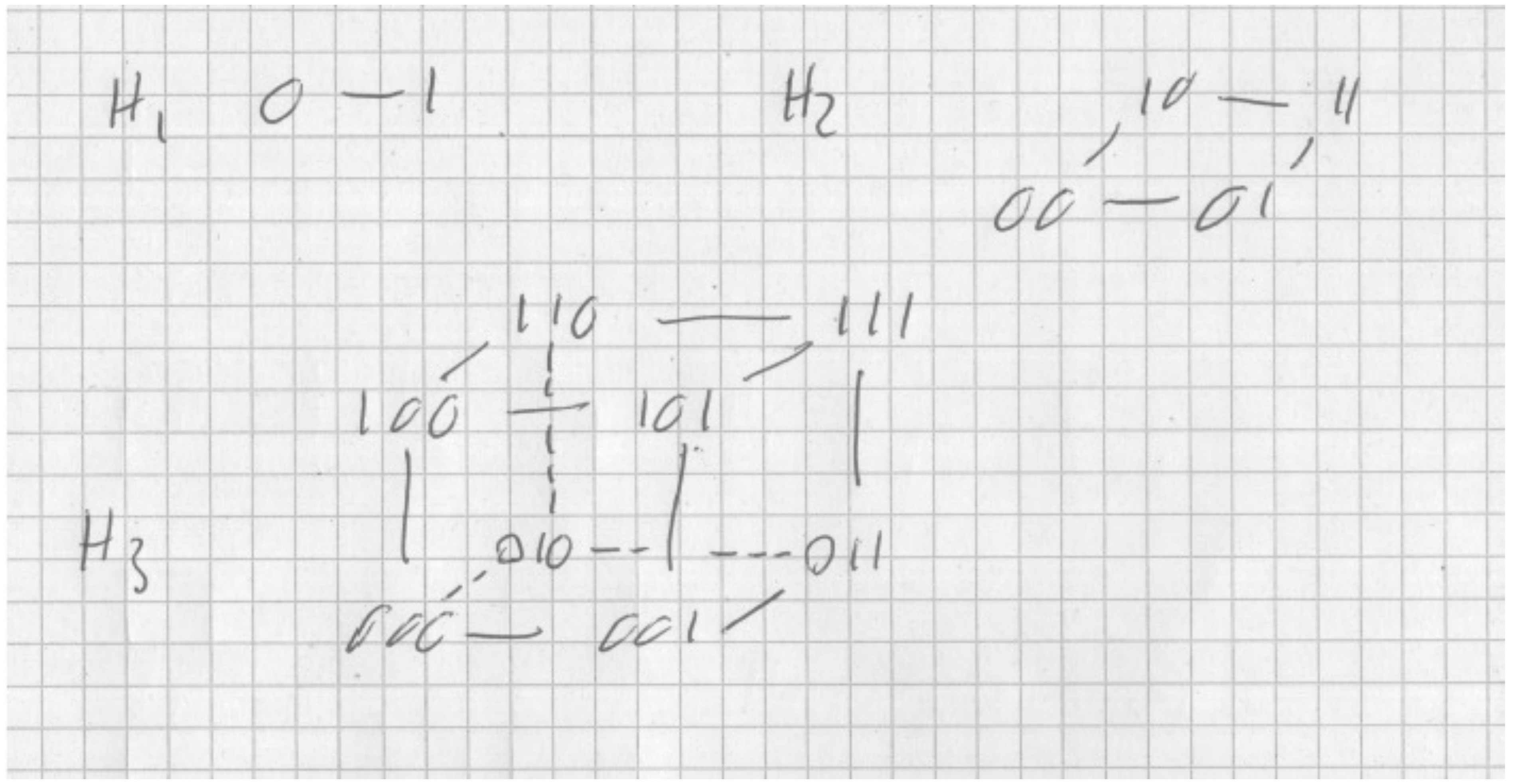
- nodes

$$V_n = \mathbb{B}^n$$

- edges

$$\{u,v\} \in E_n \leftrightarrow \#\{i \mid u_i \neq v_i\} = 1$$

nodes connected by an edge differ in exactly one position.



n dimensional hypercubes

indexing of bits strings: $u = u[1:n]$

hypercubes: graphs $H_n = (V_n, E_n)$

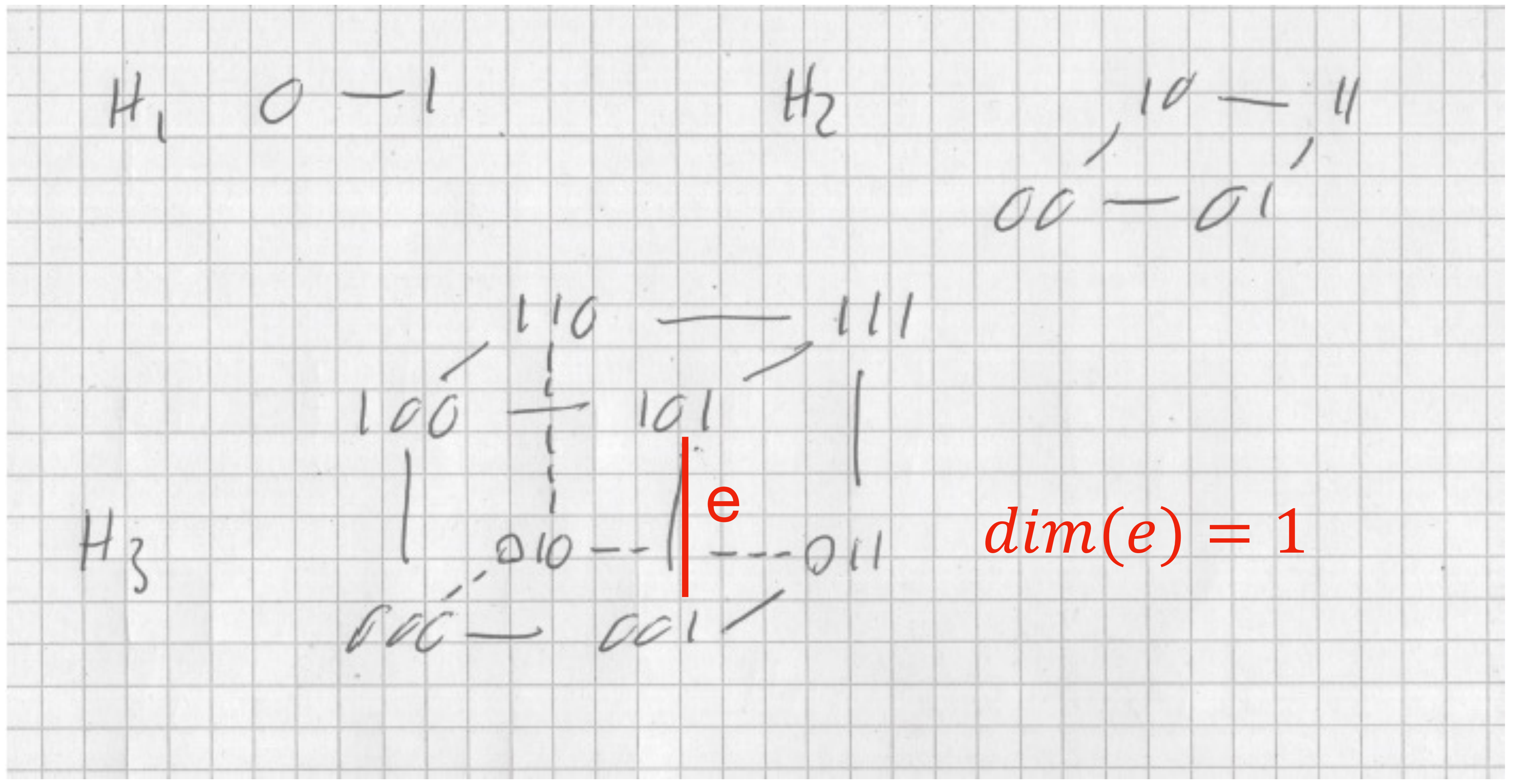
- nodes

$$V_n = \mathbb{B}^n$$

- edges

$$\{u,v\} \in E_n \leftrightarrow \#\{i \mid u_i \neq v_i\} = 1$$

nodes connected by an edge differ in exactly one position.

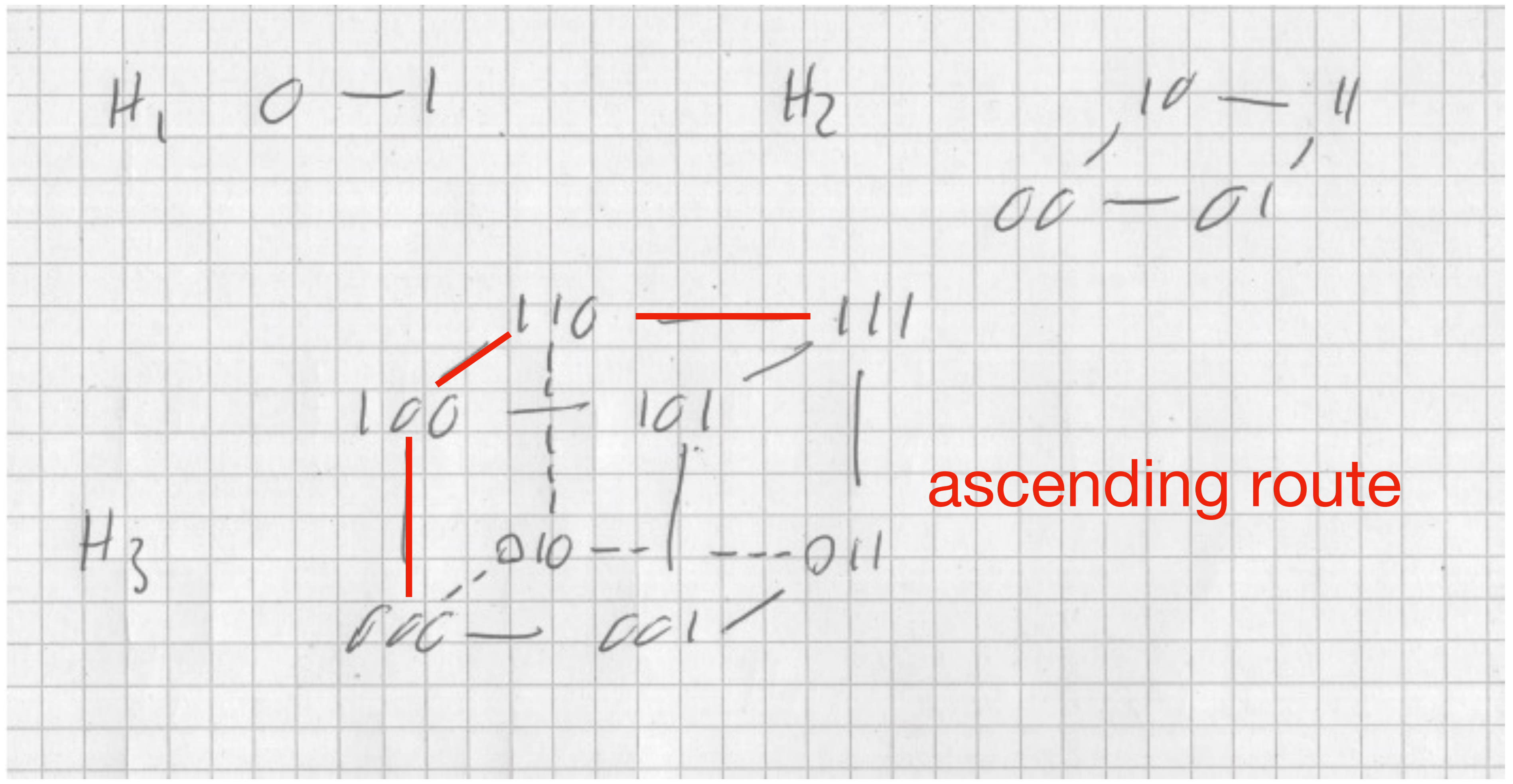


- dimension of edge: the position where end points differ

$$e = \{u,v\} \in E_n \wedge u_i \neq v_i \rightarrow dim(e) = i$$

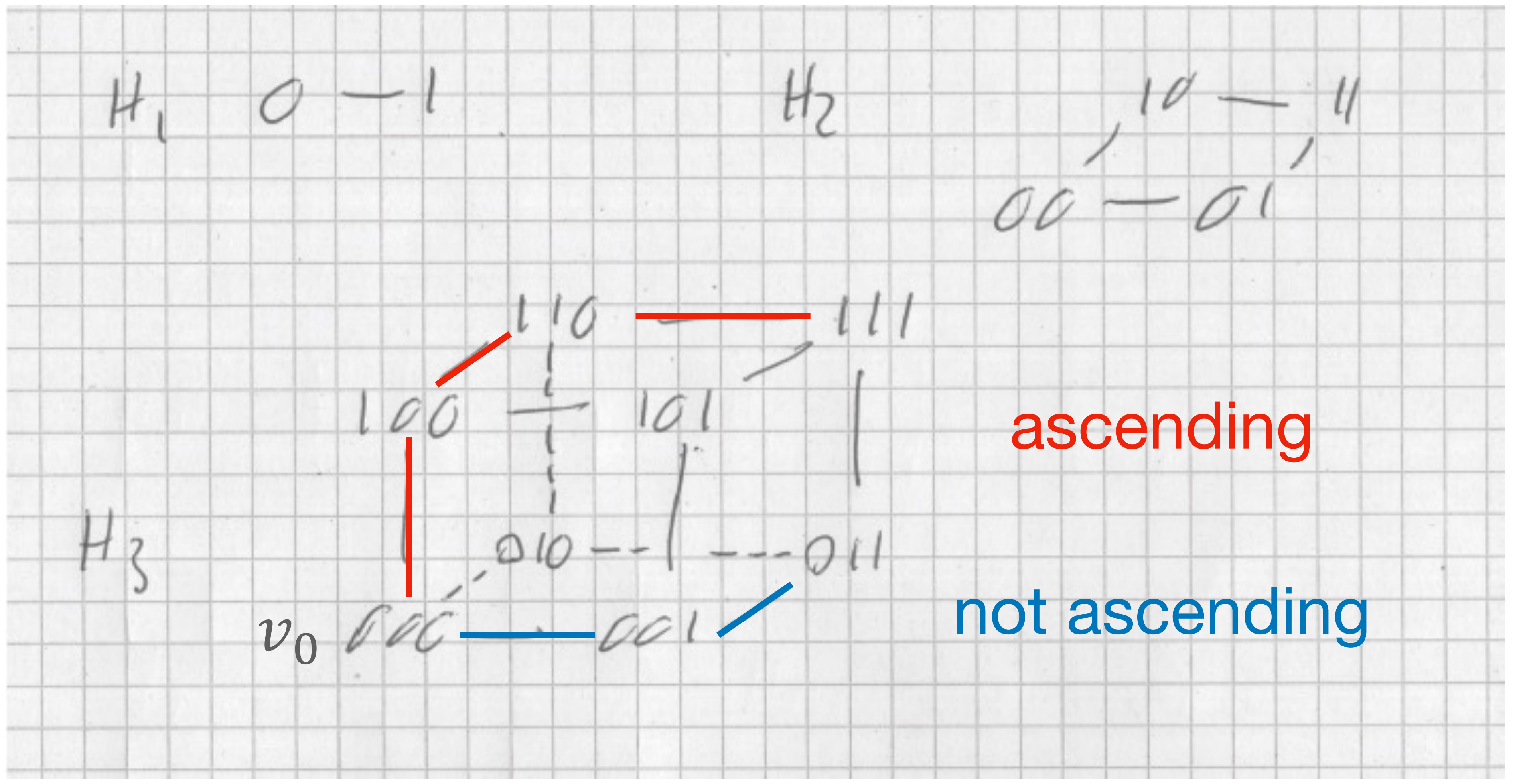
routes

- route
$$R = (v_0, v_1, \ldots v_k) \text{ path in } H_n$$
- route R is *ascending* if dimensions are traversed in increasing order
$$\forall i \in [1, k-1].\ dim(\{v_{i-1}, v_i\}) < dim(\{v_i, v_{i+1}\})$$



routes

- route
$$R = (v_0, v_1, \ldots v_k) \text{ path in } H_n$$
- route R is *ascending* if dimensions are traversed in increasing order
$$\forall i \in [1, k-1].\ dim(\{v_{i-1}, v_i\}) < dim(\{v_i, v_{i+1}\})$$



routes

- route

$$R = (v_0, v_1, \ldots v_k) \text{ path in } H_n$$

- route R is *ascending* if dimensions are traversed in increasing order

$$\forall i \in [1, k-1].\ dim(\{v_{i-1}, v_i\}) < dim(\{v_i, v_{i+1}\})$$

- routes $R = (v_0, v_1, \ldots v_k)$ and $Q = (u_0, u_1, \ldots u_m)$ *collide*, if they share at least one edge

$$col(R, Q) \leftrightarrow \exists i, j.\ \{v_i, v_{i+1}\} = \{u_j, u_{j+1}\}$$

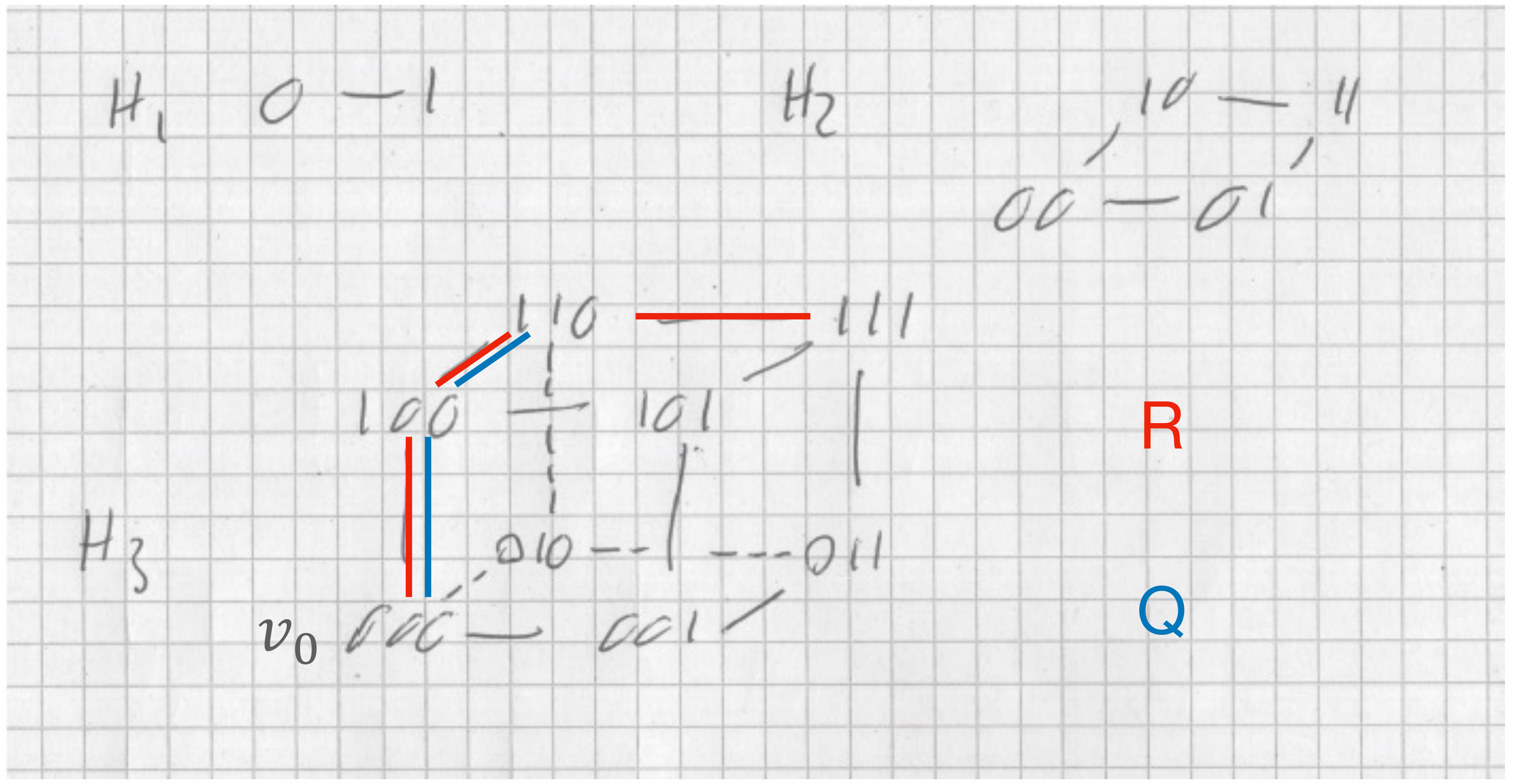
Their *collision dimension* is the dimension of the first edge on R which they share (if both are ascending, it is also the first edge on Q.

such

$$cd(R, Q) = \min i \,|\, \exists j.\ \{v_i, v_{i+1}\} = \{u_j, u_{j+1}\}$$

Their *collision edge* is the edge on R in the collision dimension

$$ce(R, Q) = \{v_{cd(R,Q)}, v_{cd(r,Q)+1}\}$$



- R, Q colliding
- collision dimension 1
- collision edge $\{000{,}100\}$

- $des(R) = 111$
- $des(Q) = 110$

routes

- route

$$R = (v_0, v_1, \ldots v_k) \text{ path in } H_n$$

- route R is *ascending* if dimensions are traversed in increasing order

$$\forall i \in [1, k-1].\ dim(\{v_{i-1}, v_i\}) < dim(\{v_i, v_{i+1}\})$$

- routes $R = (v_0, v_1, \ldots v_k)$ and $Q = (u_0, u_1, \ldots u_m)$ *collide*, if they share at least one edge

$$col(R,Q) \leftrightarrow \exists i, j.\ \{v_i, v_{i+1}\} = \{u_j, u_{j+1}\}$$

Their *collision dimension* is the dimension of the first edge on R which they share (if both are ascending, it is also the first edge on Q.

such

$$cd(R,Q) = \min i \mid \exists j.\ \{v_i, v_{i+1}\} = \{u_j, u_{j+1}\}$$

Their *collision edge* is the edge on R in the collision dimension

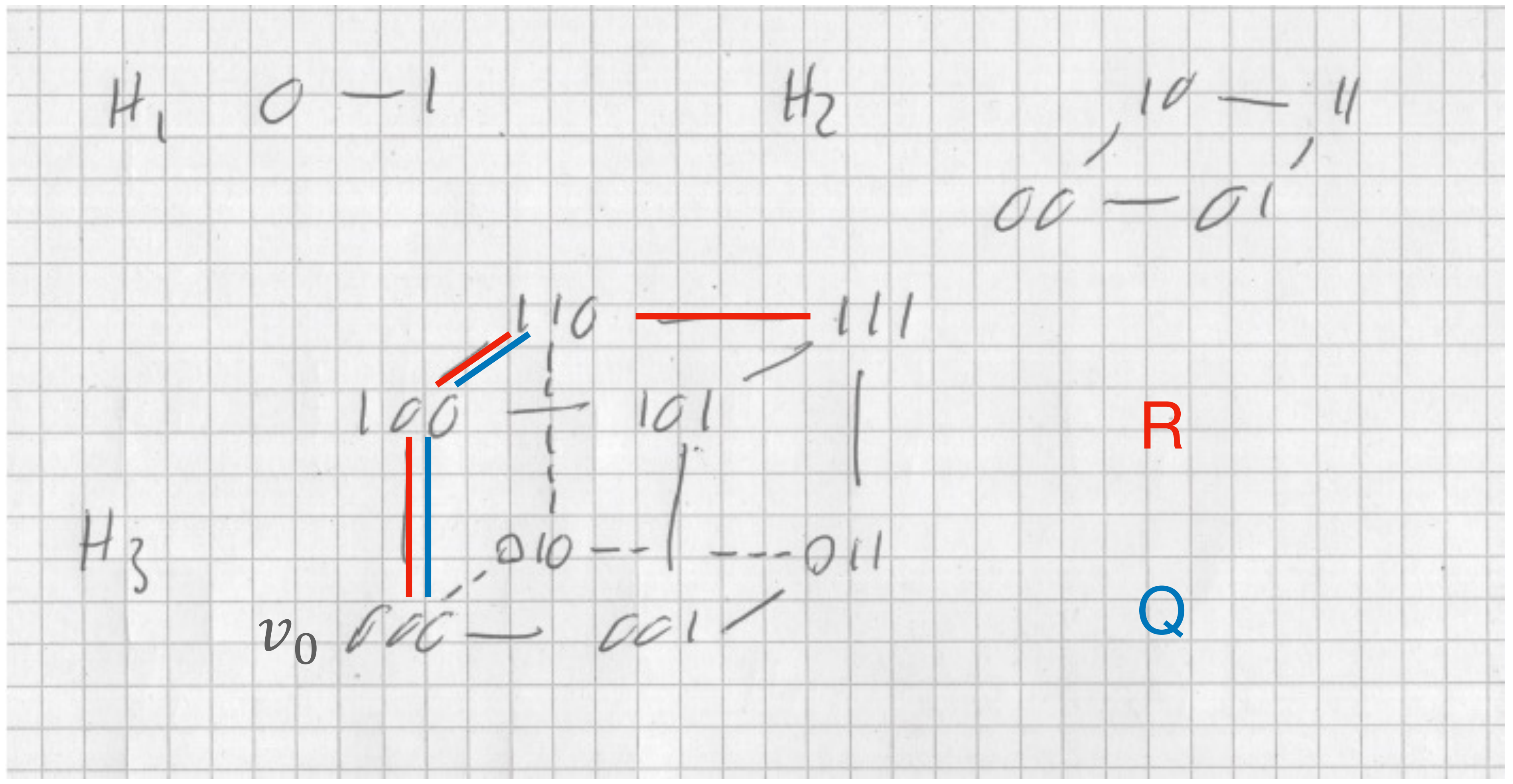
$$ce(R,Q) = \{v_{cd(R,Q)}, v_{cd(R,Q)+1}\}$$

exercise

- standard (partial) description of route R: Bit j of $des(R) \in \mathbb{B}^n$ is 1, if an edge of R has dimension j.

$$des(R) \in \mathbb{B}^n\ ,\ des(R)_j = 1 \leftrightarrow \exists i.\ dim\{v_{i-1}, v_i\} = j$$

An ascending route R is determined by start point v_0 and $des(R)$.



- R, Q colliding
- collision dimension 1
- collision edge $\{000, 100\}$

- $des(R) = 111$
- $des(Q) = 110$

packet routing

- one packet in each node u
- $\pi : \mathbb{B}^n \to \mathbb{B}^n$ permutation.
- goal: for all u move packet from u to $\pi(u)$.
- packets can be stored in nodes
- in each step each edge can only be passed by 1 packet.
- if several packets in a node try to traverse the same edge, the order cannot be predicted.

random routing

packet routing

- one packet in each node u
- $\pi : \mathbb{B}^n \to \mathbb{B}^n$ permutation.
- goal: for all u move packet from u to $\pi(u)$.
- packets can be stored in nodes
- in each step each edge can only be passed by 1 packet.
- if several packets in a node try to traverse the same edge, the order cannot be predicted.

random routing

two rounds:

- round 1: each node u determines random $des(R(u)) \in \mathbb{B}^n$. Packet from u is sent to random destination

$$\rho(u) = u \oplus des(R(u))$$

exercise: This is intuitive „thing"

via ascending route.

- round 2: packets from u are sent from $\rho(u)$ to $\pi(u)$ via ascending routes.

probability space

packet routing

- one packet in each node u
- $\pi : \mathbb{B}^n \to \mathbb{B}^n$ permutation.
- goal: for all u move packet from u to $\pi(u)$.
- packets can be stored in nodes
- in each step each edge can only be passed by 1 packet.
- if several packets in a node try to traverse the same edge, the order cannot be predicted.

random routing

two rounds:

- round 1: each node u determines random $des(R(u)) \in \mathbb{B}^n$. Packet from u is sent to random destination

$$\rho(u) = u \oplus des(R(u))$$

  via ascending route.

- round 2: packets from u are sent from $\rho(u)$ to $\pi(u)$ via ascending routes.

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n}, \; p(d) = \frac{1}{2^{n \cdot 2^n}} \quad \text{for } d \in S$$

probability space

packet routing

- one packet in each node u
- $\pi : \mathbb{B}^n \to \mathbb{B}^n$ permutation.
- goal: for all u move packet from u to $\pi(u)$.
- packets can be stored in nodes
- in each step each edge can only be passed by 1 packet.
- if several packets in a node try to traverse the same edge, the order cannot be predicted.

random routing

two rounds:

- round 1: each node u determines random $des(R(u)) \in \mathbb{B}^n$. Packet from u is sent to random destination

$$\rho(u) = u \oplus des(R(u))$$

  via ascending route.
- round 2: packets from u are sent from $\rho(u)$ to $\pi(u)$ via ascending routes.

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n} \;,\; p(d) = \frac{1}{2^{n \cdot 2^n}} \quad \text{for } d \in S$$

Lemma 1. *For events $E \subseteq S$*

$$p(E) = \frac{\#E}{2^{n \cdot 2^n}}$$

probability space

packet routing

- one packet in each node u
- $\pi : \mathbb{B}^n \to \mathbb{B}^n$ permutation.
- goal: for all u move packet from u to $\pi(u)$.
- packets can be stored in nodes
- in each step each edge can only be passed by 1 packet.
- if several packets in a node try to traverse the same edge, the order cannot be predicted.

random routing

two rounds:

- round 1: each node u determines random $des(R(u)) \in \mathbb{B}^n$. Packet from u is sent to random destination

$$\rho(u) = u \oplus des(R(u))$$

  via ascending route.
- round 2: packets from u are sent from $\rho(u)$ to $\pi(u)$ via ascending routes.

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n} \ , \ p(d) = \frac{1}{2^{n \cdot 2^n}} \quad \text{for } d \in S$$

Lemma 1. *For events $E \subseteq S$*

$$p(E) = \frac{\#E}{2^{n \cdot 2^n}}$$

run time of a round:

- in 1 step 1 packet can traverse 1 edge.
- traversal of different edges performed in parallel.
- fix any strategy to schedule conflicting packets on edges.
- $T(d)$= number of steps until all packets from nodes u have reached their destination $\rho(u)$

probability of long runs

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n} \;,\; p(d) = \frac{1}{n \cdot 2^n} \quad \text{for } d \in S$$

Lemma 1. *For events $E \subseteq S$*

$$p(E) = \frac{\#E}{n \cdot 2^n}$$

run time of a round:

- in 1 step 1 packet can traverse 1 edge.
- traversal of different edges performed in parallel.
- fix any strategy to schedule conflicting packets on edges.
- $T(d)$= number of steps until all packets from nodes u have reached their destination $\rho(u)$

long run times (in round 1) are unlikely:

Lemma 2. *For any constant $c > 0$:*

$$p(\{d \mid T(d) \geq n + c\}) \leq 2^{-(c-5n)/2}$$

routes with many collision routes

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n} \; , \; p(d) = \frac{1}{n \cdot 2^n} \quad \text{for } d \in S$$

Lemma 1. *For events $E \subseteq S$*

$$p(E) = \frac{\#E}{n \cdot 2^n}$$

run time of a round:

- in 1 step 1 packet can traverse 1 edge.
- traversal of different edges performed in parallel.
- fix any strategy to schedule conflicting packets on edges.
- $T(d)$= number of steps until all packets from nodes u have reached their destination $\rho(u)$

long run times (in round 1) are unlikely:

Lemma 2. *For any constant $c > 0$:*

$$p(\{d \mid T(d) \geq n + c\}) \leq 2^{-(c-5n)/2}$$

- If $T(d) \geq n + c$, then the packet from some node u is delayed by at least c other packets from nodess v_i.
- The routes $R_d(v_i)$ for these packets collide with route $R_d(u)$
- the event that this situation occurs:

$$E = \{d \mid \exists u, v_1, \ldots v_c. \, \forall j. \; col(R_d(u), R_d(v_j))\}$$

routes with many collision routes

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n} \;,\; p(d) = \frac{1}{n \cdot 2^n} \quad \text{for } d \in S$$

Lemma 1. *For events $E \subseteq S$*

$$p(E) = \frac{\#E}{n \cdot 2^n}$$

run time of a round:

- in 1 step 1 packet can traverse 1 edge.
- traversal of different edges performed in parallel.
- fix any strategy to schedule conflicting packets on edges.
- $T(d)$= number of steps until all packets from nodes u have reached their destination $\rho(u)$

long run times (in round 1) are unlikely:

Lemma 2. *For any constant $c > 0$:*

$$p(\{d \mid T(d) \geq n + c\}) \leq 2^{-(c-5n)/2}$$

- If $T(d) \geq n + c$, then the packet from some node u is delayed by at least c other packets from nodess v_i.
- The routes $R_d(v_i)$ for these packets collide with route $R_d(u)$
- the event that this situation occurs:

$$E = \{d \mid \exists u, v_1, \ldots v_c.\ \forall j.\ col(R_d(u), R_d(v_j))\}$$

- $$\{d \mid T(d) \geq n + c\} \subseteq E$$

  lemma 1 $\rightarrow$

  $$p(\{d \mid T(d) \geq n + c\} \leq \frac{\#E}{n \cdot 2^n}$$

- now estimate $\#E$ ...

routes with many collision routes

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n} \ , \ p(d) = \frac{1}{n \cdot 2^n} \quad \text{for } d \in S$$

Lemma 1. *For events $E \subseteq S$*

$$p(E) = \frac{\#E}{n \cdot 2^n}$$

run time of a round:

- in 1 step 1 packet can traverse 1 edge.
- traversal of different edges performed in parallel.
- fix any strategy to schedule conflicting packets on edges.
- $T(d)$= number of steps until all packets from nodes u have reached their destination $\rho(u)$

Lemma 2. *For any constant $c > 0$:*

$$p(\{d \mid T(d) \geq n + c\}) \leq 2^{-(c-5n)/2}$$

- If $T(d) \geq n + c$, then the packet from some node u is delayed by at least c other packets from nodess v_i.
- The routes $R_d(v_i)$ for these packets collide with route $R_d(u)$
- the event that this situation occurs:

$$E = \{d \mid \exists u, v_1, \ldots v_c. \ \forall j. \ col(R_d(u), R_d(v_j))\}$$

- $$\{d \mid T(d) \geq n + c\} \subseteq E$$

  lemma 1 $\rightarrow$

  $$p(\{d \mid T(d) \geq n + c\} \leq \frac{\#E}{n \cdot 2^n}$$

- now estimate $\#E$ ...

routes with many collision routes

Probability space:

all bits in

$$d = des(R(0^n)) \circ \ldots \circ des(R(1^n)) \in \mathbb{B}^{n \cdot 2^n}$$

chosen independently and with probability $1/2$. Probability space $W = (S, p)$

$$S = \mathbb{B}^{n \cdot 2^n}, \; p(d) = \frac{1}{n \cdot 2^n} \quad \text{for } d \in S$$

Lemma 1. *For events $E \subseteq S$*

$$p(E) = \frac{\#E}{n \cdot 2^n}$$

run time of a round:

- in 1 step 1 packet can traverse 1 edge.
- traversal of different edges performed in parallel.
- fix any strategy to schedule conflicting packets on edges.
- $T(d)$= number of steps until all packets from nodes u have reached their destination $\rho(u)$

Lemma 2. *For any constant $c > 0$:*

$$p(\{d \mid T(d) \geq n + c\}) \leq 2^{-(c-5n)/2}$$

- If $T(d) \geq n + c$, then the packet from some node u is delayed by at least c other packets from nodess v_i.
- The routes $R_d(v_i)$ for these packets collide with route $R_d(u)$
- the event that this situation occurs:

$$E = \{d \mid \exists u, v_1, \ldots v_c. \, \forall j. \, col(R_d(u), R_d(v_j))\}$$

- $$\{d \mid T(d) \geq n + c\} \subseteq E$$

  lemma 1 $\rightarrow$

  $$p(\{d \mid T(d) \geq n + c\} \leq \frac{\#E}{n \cdot 2^n}$$

- now estimate $\#E$ ...

idea:

- if a route $R_d(u)$ with c collision routes $R_d(v_j)$ exists, then this can be used to find a compressed encoding $d' \in \mathbb{B}^x$ of d with $x < n \cdot 2^n$.
- as each element in E is determined by such an encoding, the number of such encodings bouds $\#E$:

$$\#E \leq 2^x$$

compressing encodings along a route $R_d(u)$ with c collisions

compressed desciptions d' have 4 parts

$$d' = A \circ B \circ C \circ D$$

- the route from u

$$A = u \circ des(u)$$

We have

$$|A| = 2n$$

compressing encodings along a route $R_d(u)$ with c collisions

compressed descriptions d' have 4 parts

$$d' = A \circ B \circ C \circ D$$

- the route from u

$$A = u \circ des(u)$$

  We have

$$|A| = 2n$$

- For each dimension $i > 1$ let $c(i)$ be the number of collision routes colliding with $R_d(u)$ in dimension i. With

$$c^*(i) = \begin{cases} c(i) & c(i) \text{ even} \\ c(i) - 1 & c(i) \text{ odd} \end{cases}$$

  For each odd $c(i)$ we disregard route $R_d(v_j)$ colliding in dimension i. This removes at most n routes and all $c^*(i)$ are even.

  c(1) at most 1.

$$c - n \leq c^* = \sum_{i=2}^{n} c^*(i) \leq c$$

  We code the sequence of the $c^*(i)/2$ in *unary* in the collision vector

$$B = 1^{c^*(2)/2}01^{c^*(3)/2}0\ldots01^{c^*(n)}$$

  Then

$$|B| < n + c^*/2$$

- we order nodes v_j with collision routes in the order of the collision dimension with $R_d(u)$. In this order the collision dimension or route $R_d(v_j)$

$$cd(j) = cd(R_d(u), R_d(v_j)) = \overset{\max}{\cancel{\min}}\{i \,|c^*(i) \neq 0\ ,\ \sum_{k<i} c^*(k) < j\}$$

We code the collision routes in this order with descriptions

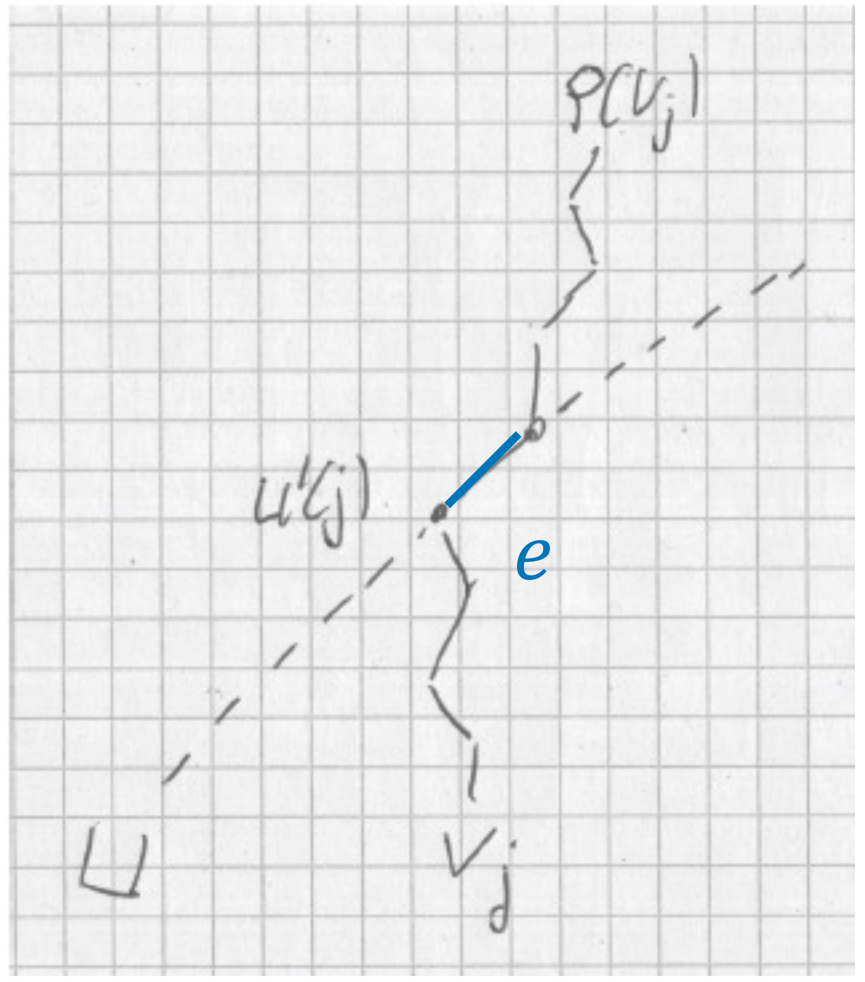
$$\underline{C = des'(v_1) \circ \ldots \circ des'(v_{c^*})}$$

where we omit t for each v_j the bit for the collision dimension $cd(j)$ (which we already know to be 1).

$$\underline{des'(v_j) = des(v_j)[1 : cd(j) - 1] \circ des(v_j)[cd(j) + 1 : n]}$$

Then for all j

$$\begin{aligned} |des'(v_j)| &= n - 1 \\ |C| &= c^* \cdot (n - 1) \end{aligned}$$



no need to specify presence
or absence of e in $des(v_j)$
if we know the collision dimension

- we order nodes v_j with collision routes in the order of the collision dimension with $R_d(u)$. In this order the collision dimension or route $R_d(v_j)$

$$cd(j) = cd(R_d(u), R_d(v_j)) = \overset{max}{\cancel{\min}}\{i \,|c^*(i) \neq 0 \;,\; \sum_{k<i} c^*(k) < j\}$$

  We code the collision routes in this order with descriptions

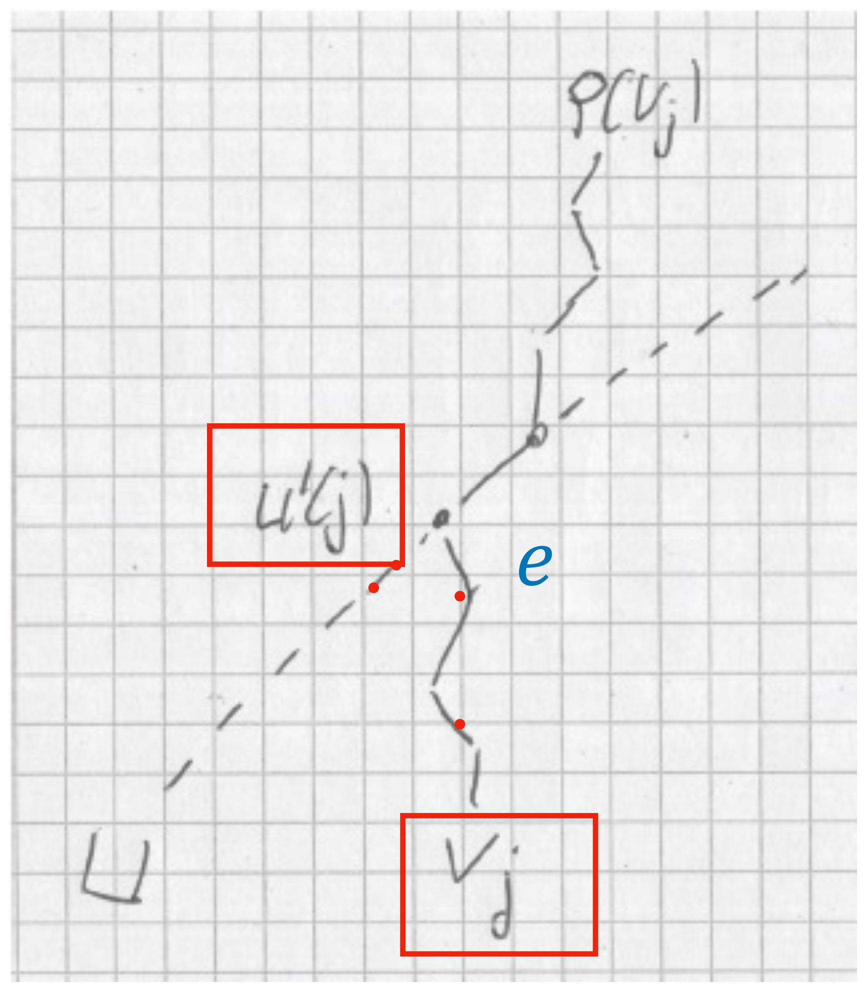
$$C = des'(v_1) \circ \ldots \circ des'(v_{c^*})$$

  where we omit t for each v_j the bit for the collision dimension $cd(j)$ (which we already know to be 1).

$$des'(v_j) = des(v_j)[1 : cd(j) - 1] \circ des(v_j)[cd(j) + 1 : n]$$

  Then for all j

$$\begin{aligned} |des'(v_j)| &= n - 1 \\ |C| &= c^* \cdot (n-1) \end{aligned}$$



no need to specify presence
or absence of e in $des(v_j)$
if we know the collision dimension

With known collision dimensions $cd(j)$ the start points v_j themselves can be reconstructed: from u follow path $R_d(u)$ until the start u' of the collision edge

$$u'(j) = u \oplus des(u)[1 : cd(j) - 1] \circ 0^{n-cd(j)+1}$$

then follow route $r_d(v_j)$ backward in the dimensions before the collision dimension

$$v_j = u'(j) \oplus des'(v_j)[1 : cd(j) - 1] \circ 0^{n-cd(j)+1}$$

- we order nodes v_j with collision routes in the order of the collision dimension with $R_d(u)$. In this order the collision dimension or route $R_d(v_j)$

$$cd(j) = cd(R_d(u), R_d(v_j)) = \overset{max}{\cancel{\min}}\{i \,|c^*(i) \neq 0\ ,\ \sum_{k<i} c^*(k) < j\}$$

We code the collision routes in this order with descriptions

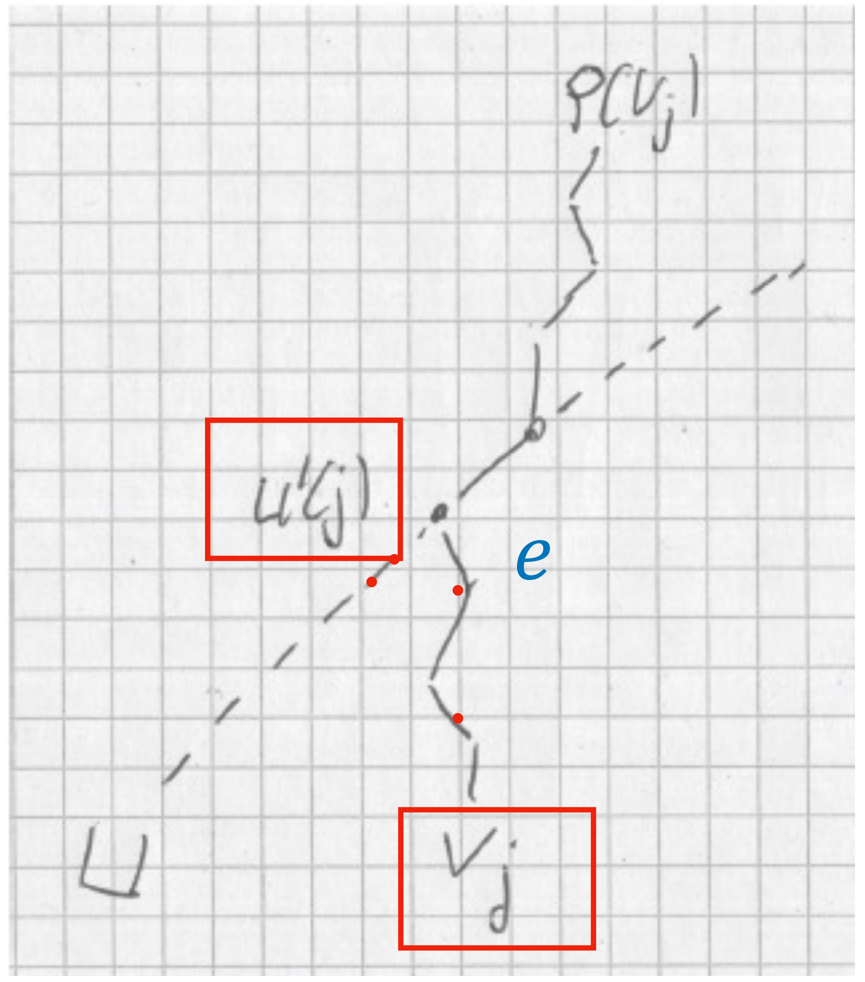
$$C = des'(v_1) \circ \ldots \circ des'(v_{c^*})$$

where we omit t for each v_j the bit for the collision dimension $cd(j)$ (which we already know to be 1).

$$des'(v_j) = des(v_j)[1 : cd(j)-1] \circ des(v_j)[cd(j)+1 : n]$$

Then for all j

$$\begin{aligned} |des'(v_j)| &= n-1 \\ |C| &= c^* \cdot (n-1) \end{aligned}$$



no need to specify presence or absence of e in $des(v_j)$ if we know the collision dimension

With known collision dimensions $cd(j)$ the start points v_j themselves can be reconstructed: from u follow path $R_d(u)$ until the start u' of the collision edge

$$u'(j) = u \oplus des(u)[1 : cd(j)-1] \circ 0^{n-cd(j)+1}$$

then follow route $r_d(v_j)$ backward in the dimensions before the collision dimension

$$v_j = u'(j) \oplus des'(v_j)[1 : cd(j)-1] \circ 0^{n-cd(j)+1}$$

- we order the remaining

$$m = 2^n - c^* - 1$$

nodes u_i in lexicographic order and store their descriptors

$$D = des(u_1) \circ \ldots \circ des(u_m)$$

We have

$$|D| = n \cdot (2^n - c^* - 1)$$

estimating the length $|d| = |ABCD|$ of the compressed description

- estimating $|d|$:

$$
\begin{aligned}
|d| &= |A| + |B| + |C| + |D| \\
&\leq 2n + c^*/2 + n + c^* \cdot (n-1) + n \cdot (2^n - c^* - 1) \\
&= n \cdot 2^n - \frac{c^*}{2} + 2n \\
&\leq n \cdot 2^n - \frac{c-n}{2} + 2n \quad (c^* \geq c - n) \\
&= n \cdot 2^n + \frac{5n}{2} - \frac{c}{2}
\end{aligned}
$$

estimating the probability of long runs

- estimating $|d|$:

$$
\begin{aligned}
|d| &= |A|+|B|+|C|+|D| \\
&\leq 2n+c^*/2+n+c^*\cdot(n-1)+n\cdot(2^n-c^*-1) \\
&= n\cdot 2^n-\frac{c^*}{2}+2n \\
&\leq n\cdot 2^n-\frac{c-n}{2}+2n \quad (c^*\geq c-n) \\
&= n\cdot 2^n+\frac{5n}{2}-\frac{c}{2}
\end{aligned}
$$

- estimating $\#E$ and $p(E)$

$$
\begin{aligned}
\#E &\leq 2^{|d|} \\
&\leq 2^{n\cdot 2^n-\frac{c-5n}{2}} \\
p(E) &= 2^{-n\cdot 2^n}\cdot \#E \quad (\text{lemma } 1\ ) \\
&\leq 2^{-\frac{c-5n}{2}}
\end{aligned}
$$

estimating the probability of long runs

- estimating $|d|$:

$$\begin{aligned} |d| &= |A|+|B|+|C|+|D| \\ &\leq 2n + c^*/2 + n + c^* \cdot (n-1) + n \cdot (2^n - c^* - 1) \\ &= n \cdot 2^n - \frac{c^*}{2} + 2n \\ &\leq n \cdot 2^n - \frac{c-n}{2} + 2n \quad (c^* \geq c-n) \\ &= n \cdot 2^n + \frac{5n}{2} - \frac{c}{2} \end{aligned}$$

- estimating $\#E$ and $p(E)$

$$\begin{aligned} \#E &\leq 2^{|d|} \\ &\leq 2^{n \cdot 2^n - \frac{c-5n}{2}} \\ p(E) &= 2^{-n \cdot 2^n} \cdot \#E \quad \text{(lemma 1)} \\ &\leq 2^{-\frac{c-5n}{2}} \end{aligned}$$

- instantiating $c = 9$: 9n

$$\boxed{\begin{aligned} p(\{d \mid T(d) \geq 10n\}) &\leq p(E) \\ &\leq 2^{-2n} \\ &= \frac{1}{N^2} \quad (\text{number of nodes } N = 2^n) \end{aligned}}$$

round 2

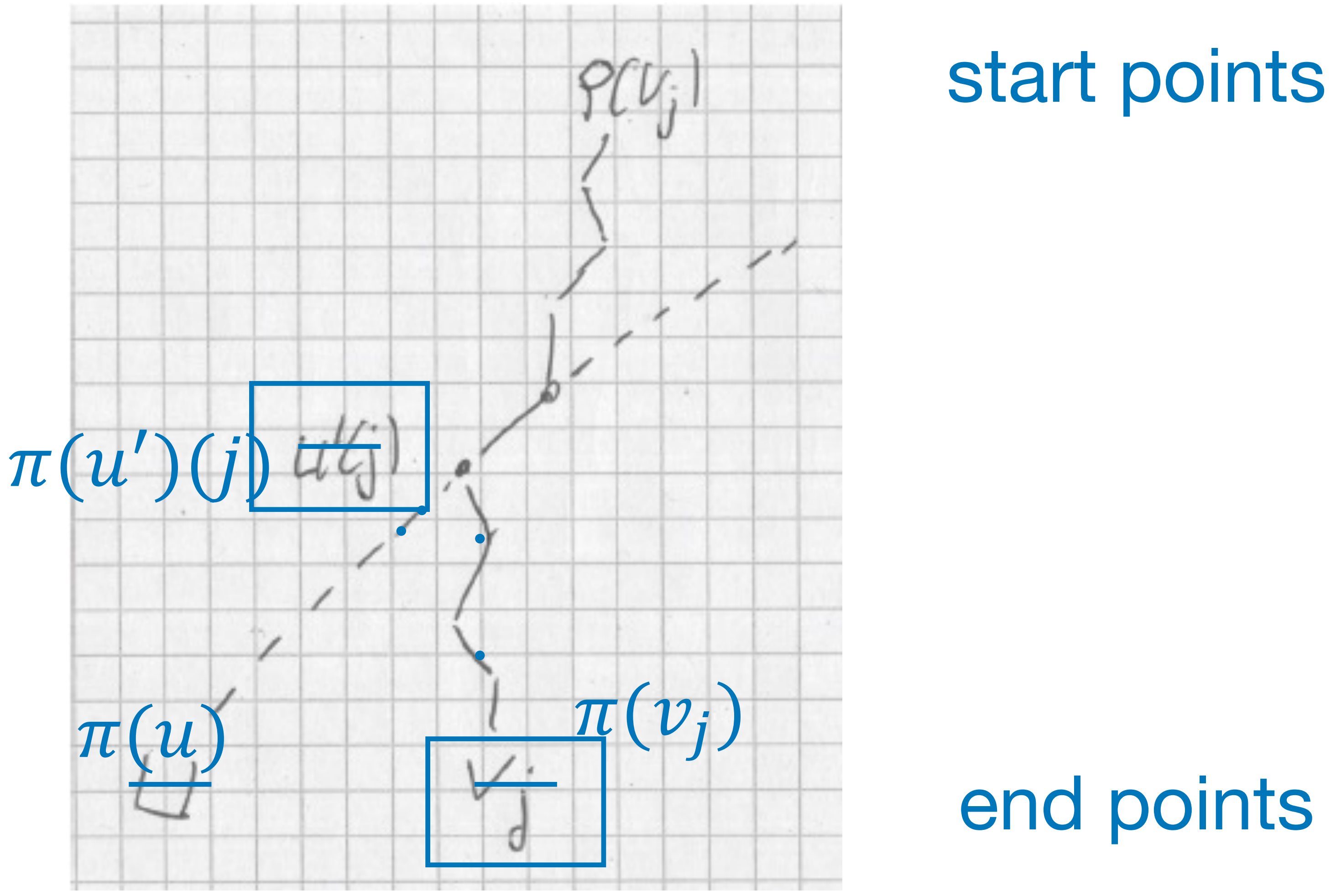
The probability of runs longer than $n+c$ can be estimated exactly as in phase 1. The previous argument is adapted as follows.

- start points of routes u and v_j are replaced by final destinations $\pi(u)$ and $\pi(v_j)$ of packets on ascending routes from intermediate destination $\rho(v_j)$ to $\pi(v_j)$.
- as collision edges one takes the *last* edge (i.e. with the largest dimension) shared by colliding pathes.
- descriptions $des[1:n]$ of paths specify as before the dimensions, in which edges are traversed, but to completely specify the path, the endpoints (of the form $\pi(v)$) are specified.

round 2

The probability of runs longer than $n+c$ can be estimated exactly as in phase 1. The previous argument is adapted as follows.

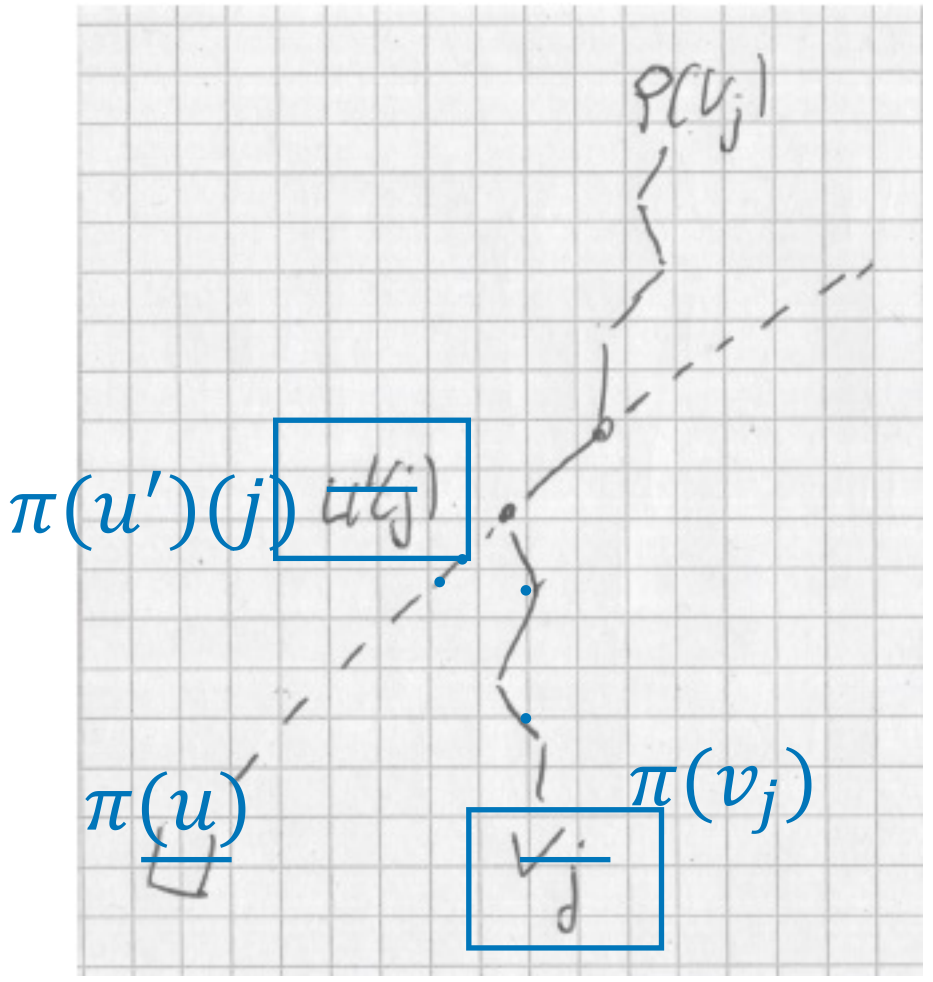
- start points of routes u and v_j are replaced by final destinations $\pi(u)$ and $\pi(v_j)$ of packets on ascending routes from intermediate destination $\rho(v_j)$ to $\pi(v_j)$.
- as collision edges one takes the *last* edge (i.e. with the largest dimension) shared by colliding pathes.
- descriptions $des[1:n]$ of paths specify as before the dimensions, in which edges are traversed, but to completely specify the path, the endpoints (of the form $\pi(v)$) are specified.



- reconstruction of an endpoint $\pi(v_j)$ from a compressed description $des'(\pi(v_j))$ and known collision dimendsion i is done with the *last* bits of the descriptions.

$$
\begin{aligned}
\pi(u)'(j) &= \pi(u) \oplus 0^i \circ des(\pi(u))[n-i+1:n] \\
\pi(v_j) &= \pi(u)'(j) \oplus 0^i \circ des'(\pi(v_j))[n-i+1:n]
\end{aligned}
$$

round 2



start points

end points

The probability of runs longer than $n+c$ can be estimated exactly as in phase 1. The previous argument is adapted as follows.

- start points of routes u and v_j are replaced by final destinations $\pi(u)$ and $\pi(v_j)$ of packets on ascending routes from intermediate destination $\rho(v_j)$ to $\pi(v_j)$.
- as collision edges one takes the *last* edge (i.e. with the largest dimension) shared by colliding pathes.
- descriptions $des[1:n]$ of paths specify as before the dimensions, in which edges are traversed, but to completely specify the path, the endpoints (of the form $\pi(v)$) are specified.

- reconstruction of an endpoint $\pi(v_j)$ from a compressed description $des'(\pi(v_j))$ and known collision dimendsion i is done with the *last* bits of the descriptions.

$$
\begin{aligned}
\pi(u)'(j) &= \pi(u) \oplus 0^i \circ des(\pi(u))[n-i+1:n] \\
\pi(v_j) &= \pi(u)'(j) \oplus 0^i \circ des'(\pi(v_j))[n-i+1:n]
\end{aligned}
$$

- this gives short descriptions d' of all paths in phase 2, in particular all intermediate destinations $\rho(v)$. From them one gets the routes and the sequence of random bits d generated in phase 1.