Online Independent Set with Stochastic Adversaries

Oliver Göbel Martin Hoefer Thomas Kesselheim

Thomas Schleiden Berthold Vöcking

mhoefer@mpi-inf.mpg.de

CLUSTER OF EXCELLENCE

max planck institut informatik

Online Capacity Maximization in Wireless Networks

- Wireless devices located in a metric space
- Set of n communication requests
- Transmissions with Interference (and Noise)

- Wireless devices located in a metric space
- Set of n communication requests
- Transmissions with Interference (and Noise)

- Foundational problem for more complicated tasks is Capacity Maximization:

  Maximize number of successful transmissions.

Online Capacity Maximization in Wireless Networks

- Wireless devices located in a metric space
- Set of n communication requests
- Transmissions with Interference (and Noise)

- Foundational problem for more complicated tasks is Capacity Maximization:

  Maximize number of successful transmissions.

- Design online algorithms when requests arrive one-by-one over time.

Online Capacity Maximization in Wireless Networks

- Wireless devices located in a metric space
- Set of n communication requests
- Transmissions with Interference (and Noise)

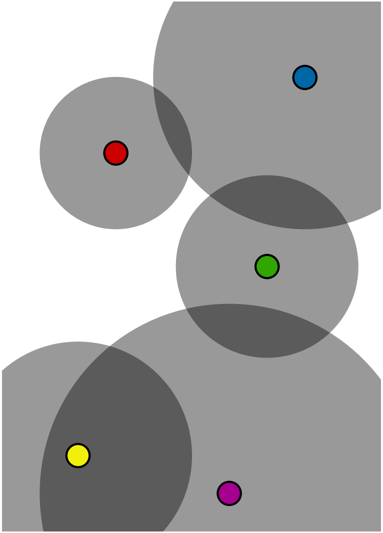
- Foundational problem for more complicated tasks is Capacity Maximization:

  Maximize number of successful transmissions.

- Design online algorithms when requests arrive one-by-one over time.

- Approximation algorithms with provable performance guarantees.

Interference?

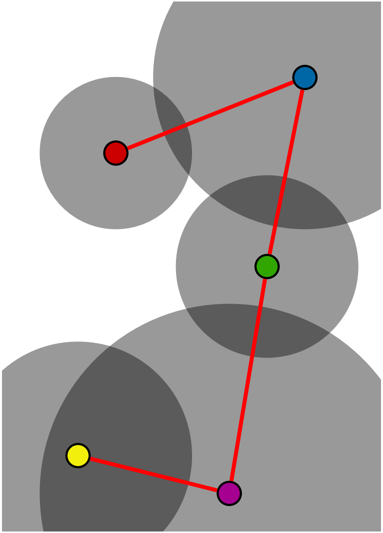


Disk Graph Model

- Users are transmitters in the plane
- User i has a transmission range
- Two transmitters can get assigned the same channel if their ranges do not intersect.

Set I of users is successful if there is no intersection among ranges of users in I, i.e., I is an independent set in the intersection graph.

Interference?

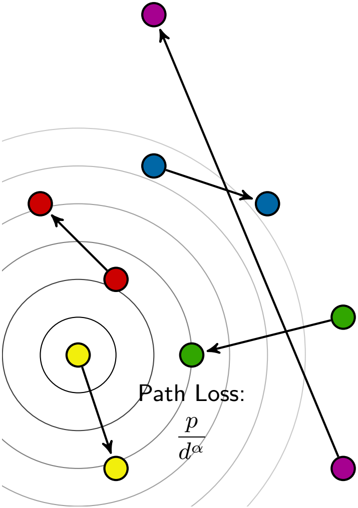


Disk Graph Model

- Users are transmitters in the plane
- User i has a transmission range
- Two transmitters can get assigned the same channel if their ranges do not intersect.

Set I of users is successful if there is no intersection among ranges of users in I, i.e., I is an independent set in the intersection graph.

Physical Model of Interference



Physical Model

Underlying Metric Space (V, d)
Requests between points in V

Parameter:

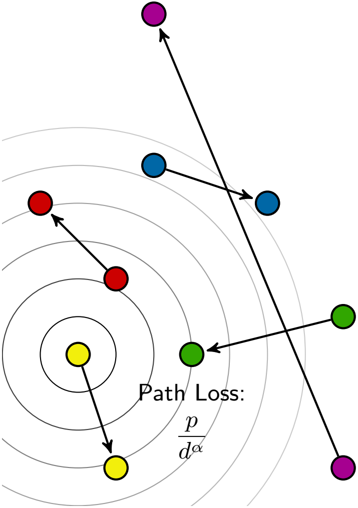
- Path loss exponent α
- Decay: $g_{ij} = 1/d_{ij}^\alpha$
- Threshold $\beta > 0$
- Noise $\nu \geq 0$

SINR Condition:

$$g_{ii} \cdot p_i \quad \geq \quad \beta \cdot \left(\nu + \sum_{j \neq i} g_{ji} \cdot p_j \right)$$

Successful requests are simultaneously feasible w.r.t. their SINR condition.

Physical Model of Interference



Weighted Conflict Graph

- Fixed distances d_{ij} and powers p_i
- Complete directed graph
- $w(i,j)$ for ordered pair of requests i,j
- Measures impact of interference of i on j, relative to j's signal strength
- Affectance:

$$w(i,j) \quad = \quad \frac{\beta \cdot g_{ij} \cdot p_i}{g_{jj} p_j - \beta \nu}$$

SINR Condition:

$$\sum_{j \neq i} w(j,i) \quad \leq \quad 1$$

Inductive Independence

In general, independent set is $O(n^{1-\varepsilon})$-hard to approximate, but affectances are based on distances in a metric space.

Inductive Independence

In general, independent set is $O(n^{1-\varepsilon})$-hard to approximate, but affectances are based on distances in a metric space.

Define "undirected weights"

$$\bar{w}(i,j) = w(i,j) + w(j,i) \ .$$

For request j, ordering π of requests, the forward set of j is

$$\Gamma_\pi(j) = \{i \mid \pi(i) > \pi(j)\} \ .$$

Inductive Independence

In general, independent set is $O(n^{1-\varepsilon})$-hard to approximate, but affectances are based on distances in a metric space.

Define "undirected weights"

$$\bar{w}(i,j) = w(i,j) + w(j,i) \ .$$

For request j, ordering π of requests, the forward set of j is

$$\Gamma_\pi(j) = \{i \mid \pi(i) > \pi(j)\} \ .$$

G has inductive independence number $\rho \Leftrightarrow$ The best ordering bounds the incoming weight from every independent set in every forward set to at most ρ.

Definition

The inductive independence number of G is the minimum number ρ s.t. there is ordering π which has for all j and independent sets I:

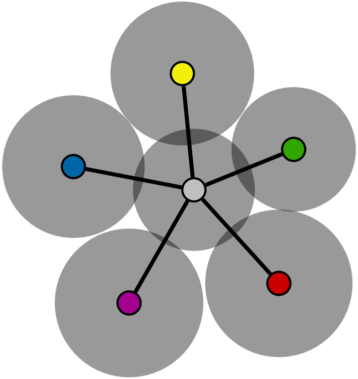
$$\sum_{i \in I \cap \Gamma_\pi(j)} \bar{w}(i,j) \leq \rho \ .$$

Proposition

For disk graphs, the inductive independence number ρ is at most 5.

Idea:

- Non-decreasing order of radius
- Geometric Argument:
  At most $\rho = 5$ intersecting disks with larger radius and without mutual intersection. □



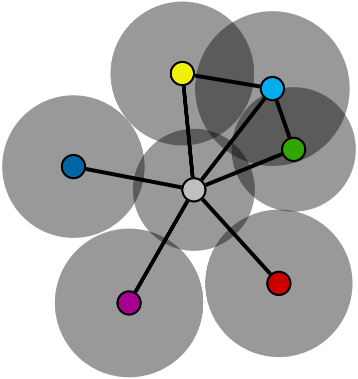
Inductive Independence Number

Proposition

For disk graphs, the inductive independence number ρ is at most 5.

Idea:

- Non-decreasing order of radius
- Geometric Argument:
  At most $\rho = 5$ intersecting disks with larger radius and without mutual intersection. $\square$



Inductive Independence in Interference Models

All prominent interference models have small upper bounds on ρ. These bounds hold even for trivial orderings.

Model	Order	Bound	Ref.
Disk Graphs	Radius	5	[Folklore]
Protocol Model	Length	$\left\lceil \frac{\pi}{\arcsin \frac{\Delta}{2(\Delta+1)}} \right\rceil - 1$	[Wan, MobiCom'09]
IEEE 802.11 model	Length	23	[Wan, MobiCom'09]
Distance-2-Match	Radius	$O(1)$	[Barrett et al, PERCOMW'06]
Distance-2-Color	Radius	$O(1)$	[Hoefer et al, SPAA'11]
SINR, Monotone	Length	$O(\log n)$	[Kesselheim, Vöcking, DISC'10]
SINR, Mean	Length	$O(1)$, $O(\log \log \Delta)$	[Halldórsson et al, SODA'13]
SINR, Power Ctrl.	Length	$O(1)$	[Kesselheim SODA'11, ESA'12]

Algorithms for Unweighted Conflict Graphs [Akcoglu et al, 2002]

Greedy Algorithm for MaxIS

1. Set $S = \emptyset$.
2. For each node i in order of π do:
3. If i is not discarded do:
4. Add i to S and discard every forward neighbor j of i.
5. Output S

- A local ratio argument shows that greedy computes a ρ-approximation.
- There is no $\rho/\omega(\log^4 \rho)$-approximation algorithm for independent set. Follows from a lower bound in regular graphs. [Chan, STOC'13]

Algorithms for Unweighted Conflict Graphs [Akcoglu et al, 2002]

Greedy Algorithm for MaxIS

1. Set $S = \emptyset$.
2. For each node i in order of π do:
3. If i is not discarded do:
4. Add i to S and discard every forward neighbor j of i.
5. Output S

- A local ratio argument shows that greedy computes a ρ-approximation.
- There is no $\rho/\omega(\log^4 \rho)$-approximation algorithm for independent set. Follows from a lower bound in regular graphs. [Chan, STOC'13]
- Similar algorithm gives ρ-approximation for Max-Weight-IS.

Algorithms for Unweighted Conflict Graphs [Akcoglu et al, 2002]

Greedy Algorithm for MaxIS

1. Set $S = \emptyset$.
2. For each node i in order of π do:
3. If i is not discarded do:
4. Add i to S and discard every forward neighbor j of i.
5. Output S

- A local ratio argument shows that greedy computes a ρ-approximation.
- There is no $\rho/\omega(\log^4 \rho)$-approximation algorithm for independent set. Follows from a lower bound in regular graphs. [Chan, STOC'13]
- Similar algorithm gives ρ-approximation for Max-Weight-IS.
- There are algorithms with factor $O(\rho)$ $(O(\rho \cdot \log n))$ for MaxIS (Max-Weight-IS) in weighted conflict graphs. [Kesselheim SODA'11] [H., Kesselheim, Vöcking SPAA'11]

Online Optimization

Online Scenario:

- Nodes from a conflict graph arrive iteratively one-by-one
- Each node i reveals upon arrival all edges to previous nodes.
- Decision to include or reject i before seeing the next node(s).
- Impossible to revoke decisions made in earlier rounds.
- Keep all accepted nodes feasible $\rightarrow$ Build an IS

Online Optimization

Online Scenario:

- Nodes from a conflict graph arrive iteratively one-by-one
- Each node i reveals upon arrival all edges to previous nodes.
- Decision to include or reject i before seeing the next node(s).
- Impossible to revoke decisions made in earlier rounds.
- Keep all accepted nodes feasible $\rightarrow$ Build an IS

Worst-Case Analysis:

- Adversary determines (unweighted) conflict graph $G = (V, E)$ adaptively
- Decides in each round which node to reveal next
- Strives to make algorithm perform as bad as possible

Online Optimization

Online Scenario:

- Nodes from a conflict graph arrive iteratively one-by-one
- Each node i reveals upon arrival all edges to previous nodes.
- Decision to include or reject i before seeing the next node(s).
- Impossible to revoke decisions made in earlier rounds.
- Keep all accepted nodes feasible $\rightarrow$ Build an IS

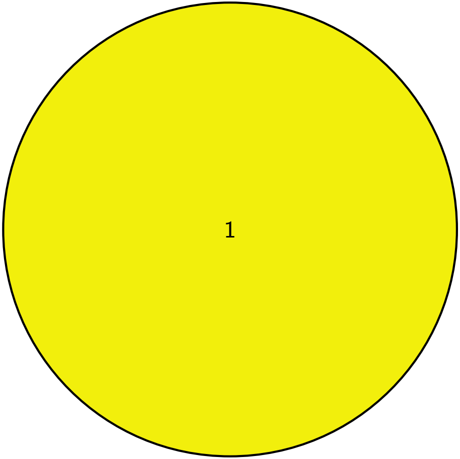
Worst-Case Analysis:

- Adversary determines (unweighted) conflict graph $G = (V, E)$ adaptively
- Decides in each round which node to reveal next
- Strives to make algorithm perform as bad as possible

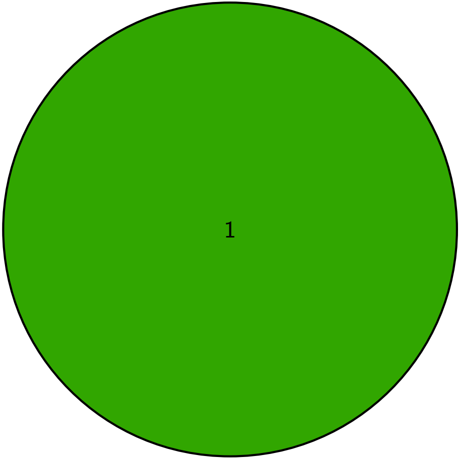
Competitive Ratio:

- S^* is optimum IS for G, S is IS constructed by online algorithm
- Competitive ratio given by $|S^*|/|S| \geq 1$.

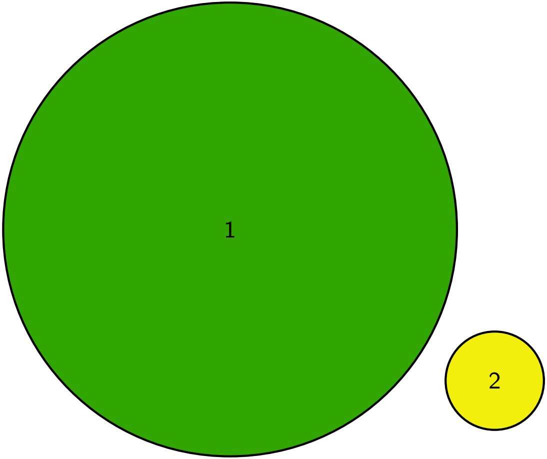
Online Arrivals – Example



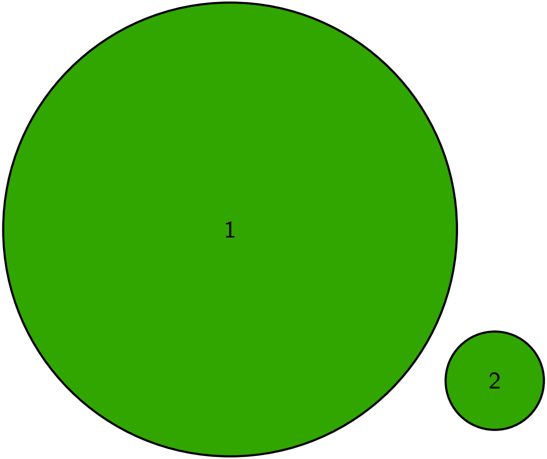
Online Arrivals – Example



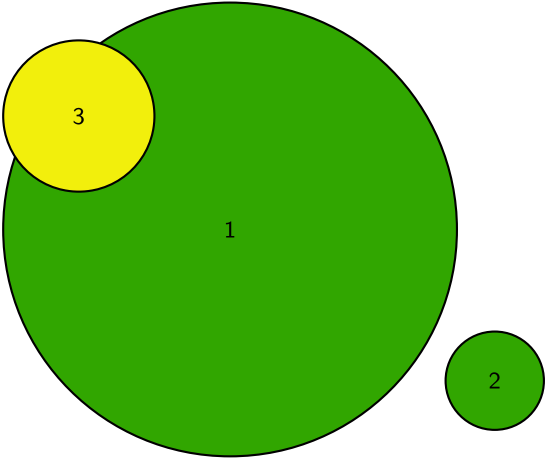
Online Arrivals – Example



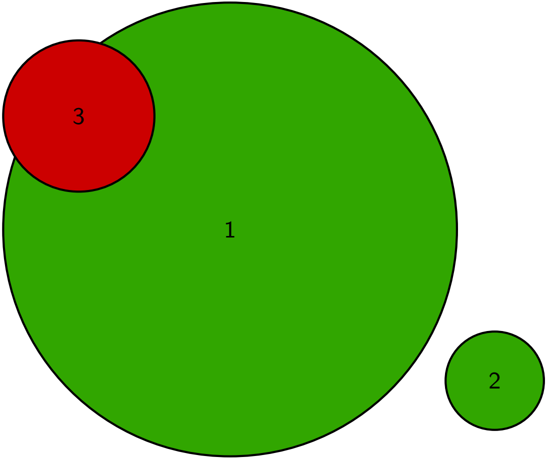
Online Arrivals – Example



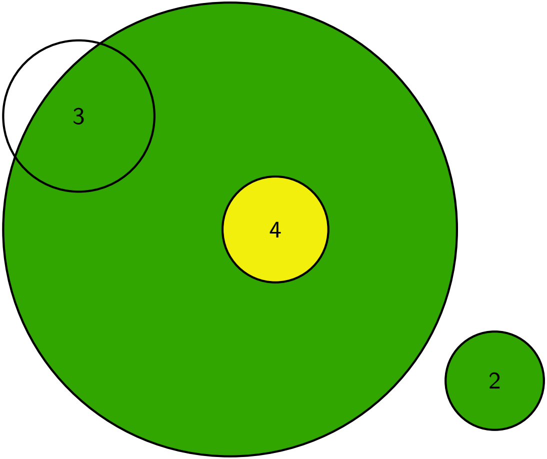
Online Arrivals – Example



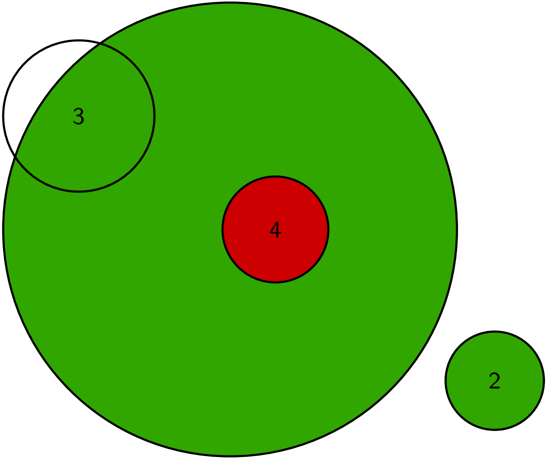
Online Arrivals – Example



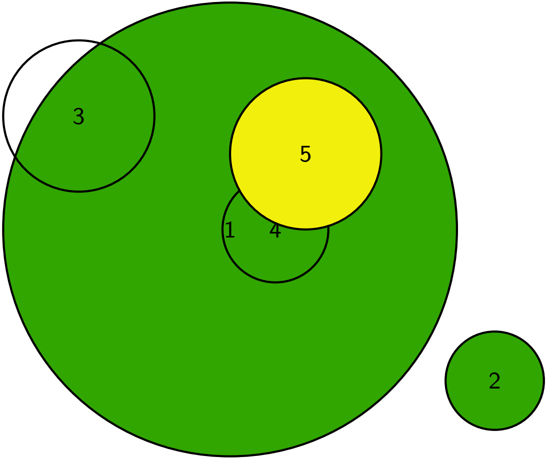
Online Arrivals – Example



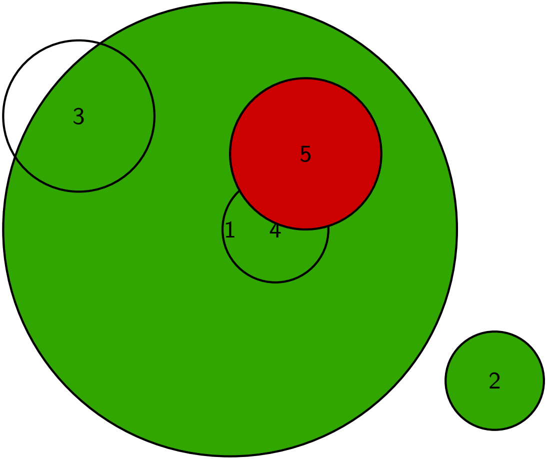
Online Arrivals – Example



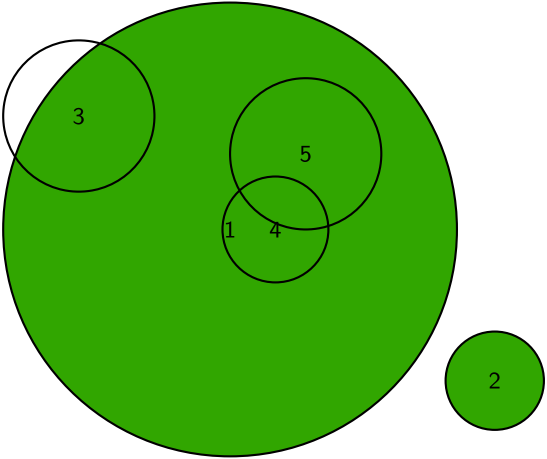
Online Arrivals – Example



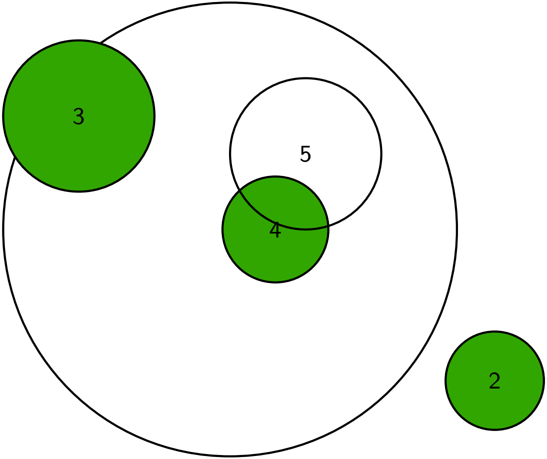
Online Arrivals – Example



Online Arrivals – Example



Online Arrivals – Example



Online IS in the Worst-Case

The worst-case competitive ratio is $\Omega(n)$ for every deterministic or randomized online algorithm, even when both the following hold:

- Adversary restricted to interval graphs G with $\rho = 1$
- Interval representation induces π with $\rho = 1$ and is shown to the algorithm for the revealed subgraph in every round
- n and size of the optimum $|S^*|$ revealed to the algorithm in advance

Online IS in the Worst-Case

The worst-case competitive ratio is $\Omega(n)$ for every deterministic or randomized online algorithm, even when both the following hold:

- Adversary restricted to interval graphs G with $\rho = 1$
- Interval representation induces π with $\rho = 1$ and is shown to the algorithm for the revealed subgraph in every round
- n and size of the optimum $|S^*|$ revealed to the algorithm in advance

For SINR models, there are algorithms maintaining a "safety distance" around accepted requests. They give competitive ratios based on distances of requests, dimension of metric space, and chosen powers.

[Fanghänel, Geulen, H., Vöcking J. Sched 2013]

Online IS in the Worst-Case

The worst-case competitive ratio is $\Omega(n)$ for every deterministic or randomized online algorithm, even when both the following hold:

- Adversary restricted to interval graphs G with $\rho = 1$
- Interval representation induces π with $\rho = 1$ and is shown to the algorithm for the revealed subgraph in every round
- n and size of the optimum $|S^*|$ revealed to the algorithm in advance

For SINR models, there are algorithms maintaining a "safety distance" around accepted requests. They give competitive ratios based on distances of requests, dimension of metric space, and chosen powers.

[Fanghänel, Geulen, H., Vöcking J. Sched 2013]

Worst-case online analysis in this scenario pointless, all algorithms equally bad. In practice, request structure often is not entirely adversarial.

Secretary Model

Online Scenario:

- Requests (i.e., nodes from a conflict graph) arrive iteratively one-by-one
- Each node i reveals upon arrival all edges to previous nodes.
- Decision to include or reject i before seeing the next node(s).
- Impossible to revoke decisions made in earlier rounds.
- Keep all accepted nodes feasible $\rightarrow$ Build an IS

Stochastic Analysis:

- Adversary determines $G = (V, E)$ in advance, nodes arrive in random order
- Ordering π with low ρ for revealed subgraph in each step is known
- n revealed to the algorithm in advance
- Strives to make algorithm perform as bad as possible

Secretary Model

Online Scenario:

- Requests (i.e., nodes from a conflict graph) arrive iteratively one-by-one
- Each node i reveals upon arrival all edges to previous nodes.
- Decision to include or reject i before seeing the next node(s).
- Impossible to revoke decisions made in earlier rounds.
- Keep all accepted nodes feasible $\rightarrow$ Build an IS

Stochastic Analysis:

- Adversary determines $G = (V, E)$ in advance, nodes arrive in random order
- Ordering π with low ρ for revealed subgraph in each step is known
- n revealed to the algorithm in advance
- Strives to make algorithm perform as bad as possible

Competitive Ratio:

- S^* is optimum IS for G, S is IS constructed by online algorithm
- S is a random variable, as arrival order is random
- Competitive ratio given by $|S^*|/\mathbb{E}[|S|] \geq 1$.

Secretary Algorithm

[Göbel, H., Kesselheim, Schleiden, Vöcking CoRR'13]

Sample-and-Inject for Unweighted Conflict Graphs

1. Reject the first $k = Binom(n, 0.5)$ requests, denote this set by V_s
2. Set output $S = \emptyset$
3. For each subsequent request i do
4. Would Greedy on $V_s \cup i$ take i? No: Reject i.
5. Reject i with probability $1 - {}^{1}/_{2\rho}$.
6. If i survived and $S \cup i$ is IS, accept i and set $S \leftarrow S \cup i$.
7. Otherwise reject i.

Theorem

Sample-and-Inject is $O(\rho^2)$-competitive for unweighted conflict graphs.

Proof Idea:

- Greedy algorithm on $V_s \cup i$ gives a ρ-approximation
- Due to random arrival, V_s is a "representative" subset of V
- Surviving requests are feasible w.r.t. V_s but not mutually conflict-free
- Second filtering step destroys mutual conflicts among surviving requests
- Implies a factor of $O(\rho^2)$ in expectation

Values and Lower Bounds

Algorithm for Online Max-Weight-IS (Sketch):

- At the end of the sampling phase create $O(\log n)$ classes of values based on $\max_{i \in V_s} v_i$ and choose one class uniformly at random.
- Reject and discard all nodes (also in V_s) with values below this class. Run the previous algorithm on the remaining nodes.

Values and Lower Bounds

Algorithm for Online Max-Weight-IS (Sketch):

- At the end of the sampling phase create $O(\log n)$ classes of values based on $\max_{i \in V_s} v_i$ and choose one class uniformly at random.
- Reject and discard all nodes (also in V_s) with values below this class. Run the previous algorithm on the remaining nodes.

Theorem

Weighted-Sample-and-Inject is $O(\rho^2 \cdot \log n)$-competitive for unweighted conflict graphs and node values $v_i \geq 0$.

Values and Lower Bounds

Algorithm for Online Max-Weight-IS (Sketch):

- At the end of the sampling phase create $O(\log n)$ classes of values based on $\max_{i \in V_s} v_i$ and choose one class uniformly at random.
- Reject and discard all nodes (also in V_s) with values below this class. Run the previous algorithm on the remaining nodes.

Theorem

Weighted-Sample-and-Inject is $O(\rho^2 \cdot \log n)$-competitive for unweighted conflict graphs and node values $v_i \geq 0$.

In general, an increase of (almost) $\log n$ is unavoidable:

Theorem

There is a set of instances with $\rho = 1$ such that every secretary online algorithm has competitive ratio at least $\Omega(\log n / (\log \log n)^2)$.

SINR Model and Weighted Conflict Graphs

For edge-weighted conflict graphs, we obtain the following bounds:

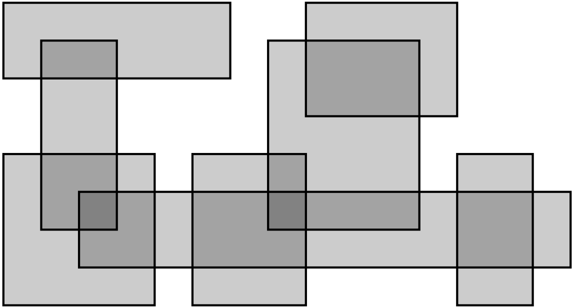
	Unweighted CG	Weighted CG
$v_i = 1$	$O(\rho^2)$	$O(\rho^2 \log^2 n)$
arbitrary	$O(\rho^2 \log n)$	$O(\rho^2 \log^3 n)$

Adjustment:

- On $V_s \cup i$ apply the $O(\rho)$ approximation algorithm
- Resolving conflicts is more demanding because of aggregation effects
- More aggressive filtering resolves mutual conflicts among surviving nodes
- Yields an additional $O(\log^2 n)$ factor in both cases

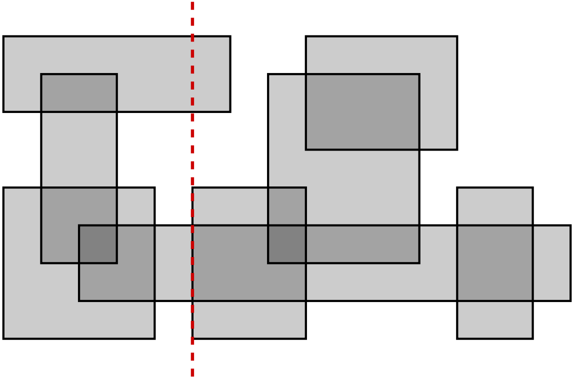
Arrival and Departure

- Requests revealed one-by-one uniformly at random on one day.
- Request demands channel for some period on the next day.
- At any time during the next day, the accepted set of requests must be conflict-free.



Arrival and Departure

- Requests revealed one-by-one uniformly at random on one day.
- Request demands channel for some period on the next day.
- At any time during the next day, the accepted set of requests must be conflict-free.



Arrival and Departure

We recursively partition our sample to identify a number of critical time points. We decompose the instance, consider only IS problems at these time points, to which we apply previous algorithms. This yields another $O(\log n)$ factor:

	Unweighted CG	Weighted CG
$v_i = 1$	$O(\rho^2 \log n)$	$O(\rho^2 \log^3 n)$
arbitrary	$O(\rho^2 \log^2 n)$	$O(\rho^2 \log^4 n)$

This also implies:

Corollary

There is an $O(\log n)$-competitive secretary algorithm for online MaxIS in rectangle graphs.

Stochastic Input

- **Prophet-Inequality Model**
  We know for each node the probability distribution of its value

1. We are presented probability distributions for the node values
2. Values are realized
3. In each round, adversary decides which node is revealed next
4. We must decide immediately without seeing the next node value(s).

Stochastic Input

- **Prophet-Inequality Model**
  We know for each node the probability distribution of its value
- **Secretary Model**
  Adversary fixes values but arrival in random order

Stochastic Input

- **Prophet-Inequality Model**
  We know for each node the probability distribution of its value
- **Secretary Model**
  Adversary fixes values but arrival in random order
- **Period Model**
  We have reference data: Each node shows up with a similar probability as "last week" at the same time

Graph Sampling Model

Input graph:

- Node values $v(i)$ are drawn from unknown distributions
- Adversary determines node arrival order after values have been determined

Graph Sampling Model

Input graph:

- Node values $v(i)$ are drawn from unknown distributions
- Adversary determines node arrival order after values have been determined

Sample graph:

- Node values $v'(i)$ are drawn from unknown distributions
- All nodes with positive value are presented to the algorithm in advance

Graph Sampling Model

Input graph:

- Node values $v(i)$ are drawn from unknown distributions
- Adversary determines node arrival order after values have been determined

Sample graph:

- Node values $v'(i)$ are drawn from unknown distributions
- All nodes with positive value are presented to the algorithm in advance

Independence between different nodes,
Arbitrary correlation between v and v'

Graph Sampling Model

Input graph:

- Node values $v(i)$ are drawn from unknown distributions
- Adversary determines node arrival order after values have been determined

Sample graph:

- Node values $v'(i)$ are drawn from unknown distributions
- All nodes with positive value are presented to the algorithm in advance

Independence between different nodes,
Arbitrary correlation between v and v'

Stochastic similarity: For every node $i \in V$ and every $b > 0$,

$$1/c \cdot \mathbf{Pr}\left[v'(i) = b\right] \quad \leq \quad \mathbf{Pr}\left[v(i) = b\right] \quad \leq \quad c \cdot \mathbf{Pr}\left[v'(i) = b\right]$$

Open Problems

- Can we turn all ρ^2-factors into ρ-factors?
- Can we turn various $O(\log n)$-factors into $O(1)$:
  MaxIS in weighted conflict graphs?
  For (classes of) the SINR model?
  For arrivals and departures?
- Correlation between values of different nodes?
- etc.