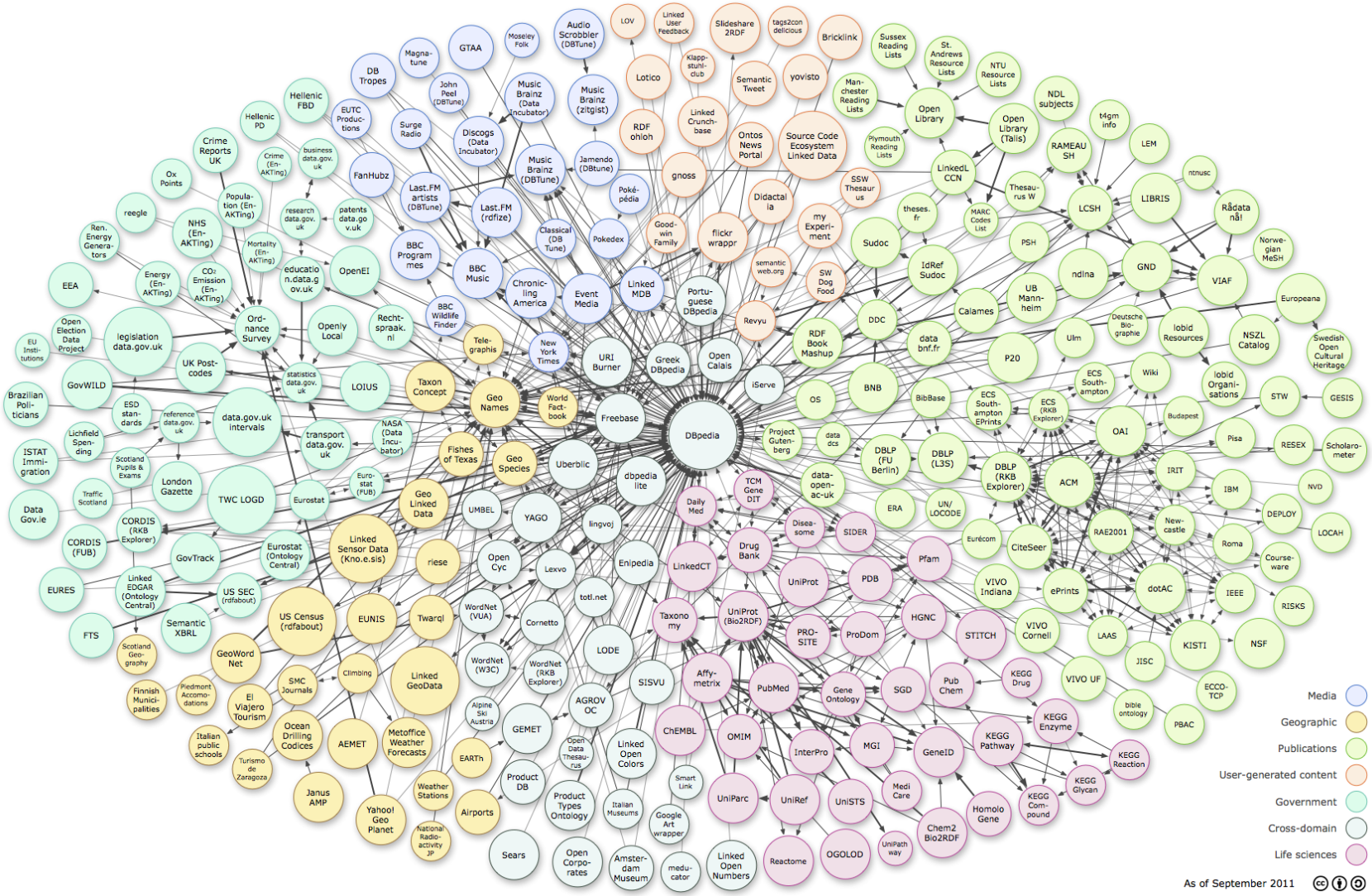
Link Discovery with Guaranteed Reduction Ratio in Affine Spaces with Minkowski Measures

Axel-Cyrille Ngonga Ngomo
ngonga@informatik.uni-leipzig.de

University of Leipzig
AKSW Research Group

November 13th, 2012
ISWC 2012, Boston MA, USA

Introduction



Link Discovery

Definition (Link Discovery)

- Given
  - Set S of source instances,
  - Set T of target instances,
  - Relation R,
- Find $M = \{(s, t) \in S \times T : R(s, t)\}$

Link Discovery

Definition (Link Discovery)

- Given
  - Set S of source instances,
  - Set T of target instances,
  - Relation R,
- Find $M = \{(s, t) \in S \times T : R(s, t)\}$
- Common approach: Find $M' = \{(s, t) \in S \times T : \delta(s, t) \leq \theta\}$

Link Discovery

Definition (Link Discovery)

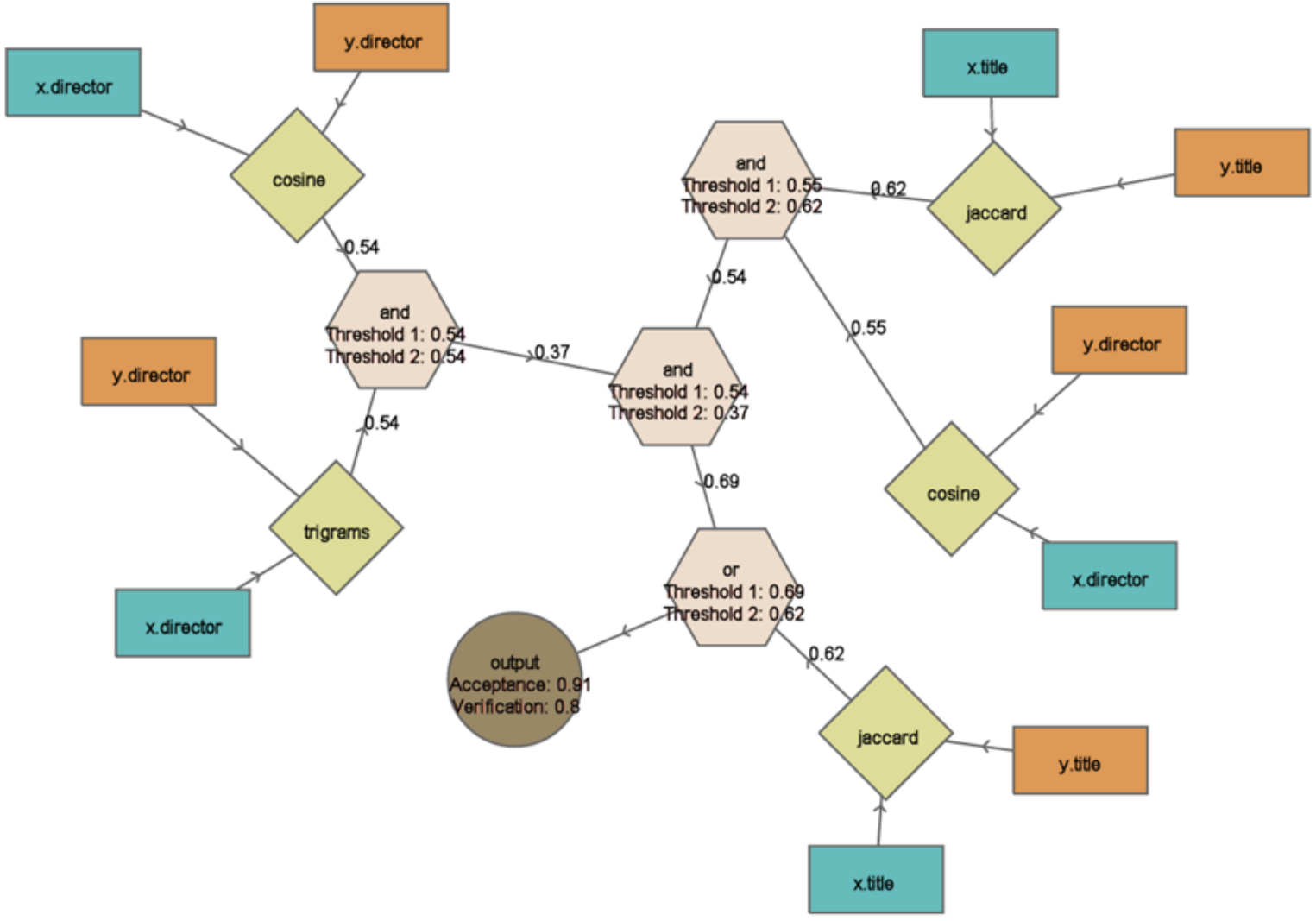
- Given
  - Set S of source instances,
  - Set T of target instances,
  - Relation R,
- Find $M = \{(s,t) \in S \times T : R(s,t)\}$
- Common approach: Find $M' = \{(s,t) \in S \times T : \delta(s,t) \leq \theta\}$

Problems

- Time complexity
- Resource management
- Complexity of specifications

Example

- Linking directors from DBpedia with LinkedMDB



http://saim.aksw.org

Runtime Optimization

- Reduce the number of comparisons $C(\mathcal{A}) \geq |M'|$
- Maximize reduction ratio:

$$RR(\mathcal{A}) = 1 - \frac{C(\mathcal{A})}{|S||T|}$$

Runtime Optimization

- Reduce the number of comparisons $C(\mathcal{A}) \geq |M'|$
- Maximize reduction ratio:

$$RR(\mathcal{A}) = 1 - \frac{C(\mathcal{A})}{|S||T|}$$

Question

- Can we devise lossless approaches with guaranteed RR?
- Advantages
  - Space management
  - Runtime prediction
  - Resource scheduling

RR Guarantee

- Best achievable reduction ratio: $RR_{\max} = 1 - \frac{|M'|}{|S||T|}$

RR Guarantee

- Best achievable reduction ratio: $RR_{\max} = 1 - \frac{|M'|}{|S||T|}$
- Approach $\mathcal{H}(\alpha)$ fulfills RR guarantee criterion, iff:

$$\forall r < RR_{\max}, \exists \alpha : RR(\mathcal{H}(\alpha)) \geq r$$

RR Guarantee

- Best achievable reduction ratio: $RR_{\max} = 1 - \frac{|M'|}{|S||T|}$
- Approach $\mathcal{H}(\alpha)$ fulfills RR guarantee criterion, iff:

$$\forall r < RR_{\max}, \exists \alpha : RR(\mathcal{H}(\alpha)) \geq r$$

- Here, we use relative reduction ratio (RRR):

$$RRR(\mathcal{A}) = \frac{RR_{\max}}{RR(\mathcal{A})}$$

Goal

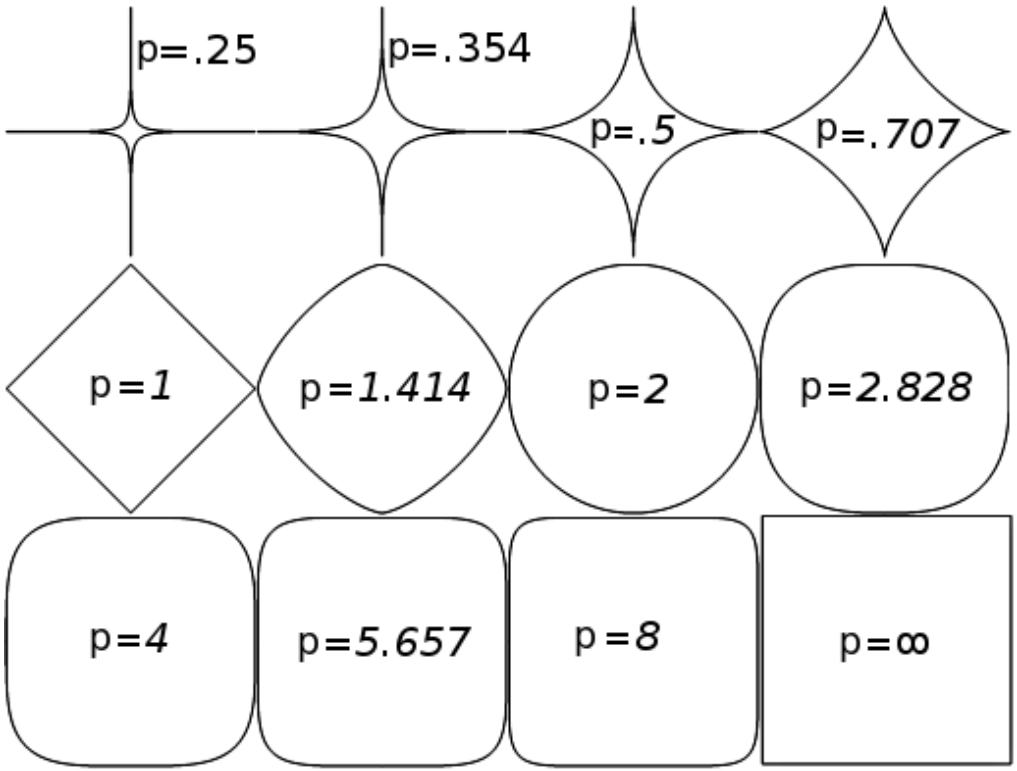
Formal Goal

Devise $\mathcal{H}(\alpha) : \forall r > 1, \exists \alpha : RRR(\mathcal{H}(\alpha)) \leq r$

Restrictions

Minkowski Distance

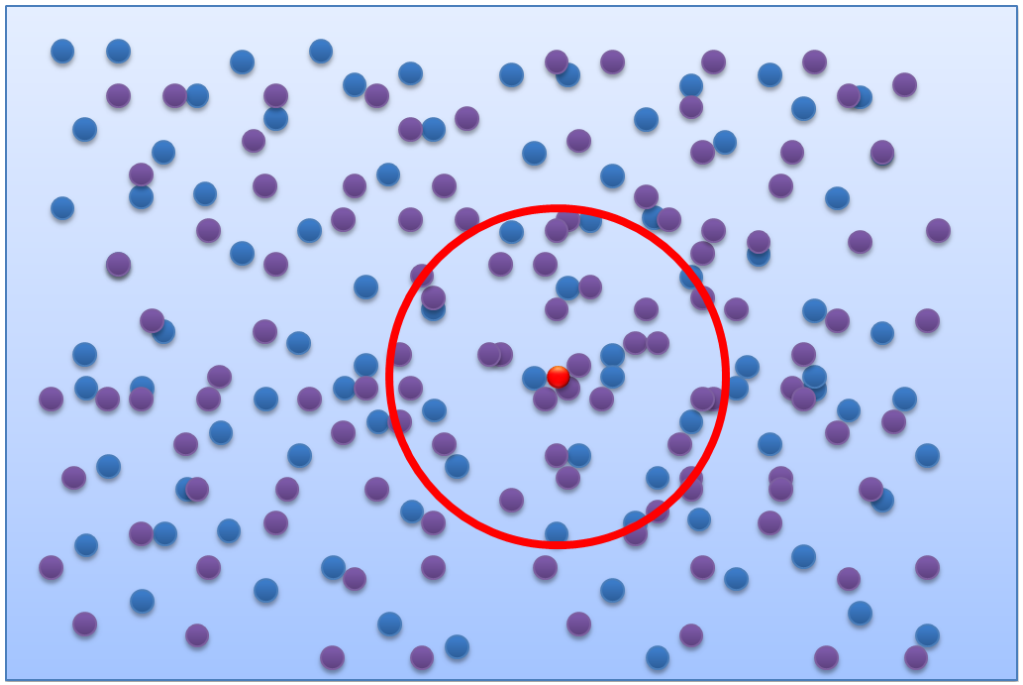
$$\delta(s,t) = \sqrt[p]{\sum_{i=1}^{n} |s_i - t_i|^p}, p \geq 2$$



Space Tiling

HYPPO

- $\delta(s, t) \leq \theta$ describes a hypersphere
- Approximate hypersphere by using a hypercube
  - Easy to compute
  - No loss of recall (blocking)



Space Tiling

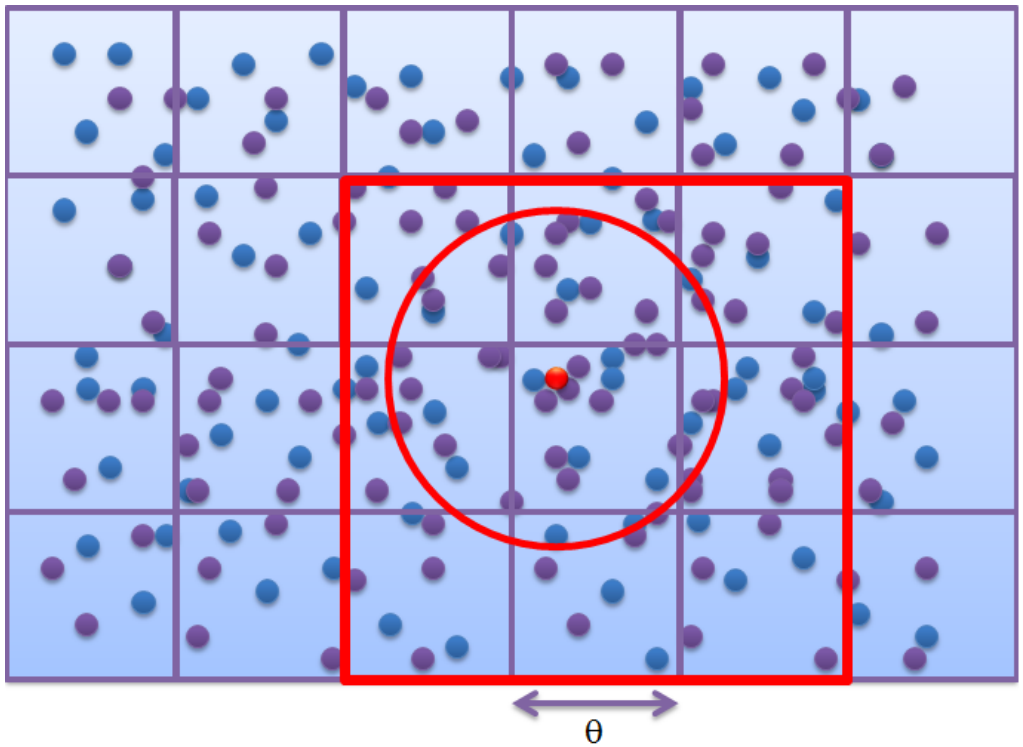
- Set width of single hypercube to $\Delta = \theta/\alpha$

Space Tiling

- Set width of single hypercube to $\Delta = \theta/\alpha$
- Tile $\Omega = S \cup T$ into the adjacent cubes C
  - Coordinates: $(c_1, \ldots, c_n) \in \mathbb{N}^n$
  - Contains points $\omega \in \Omega : \forall i \in \{1 \ldots n\}, c_i\Delta \leq \omega_i < (c_i + 1)\Delta$

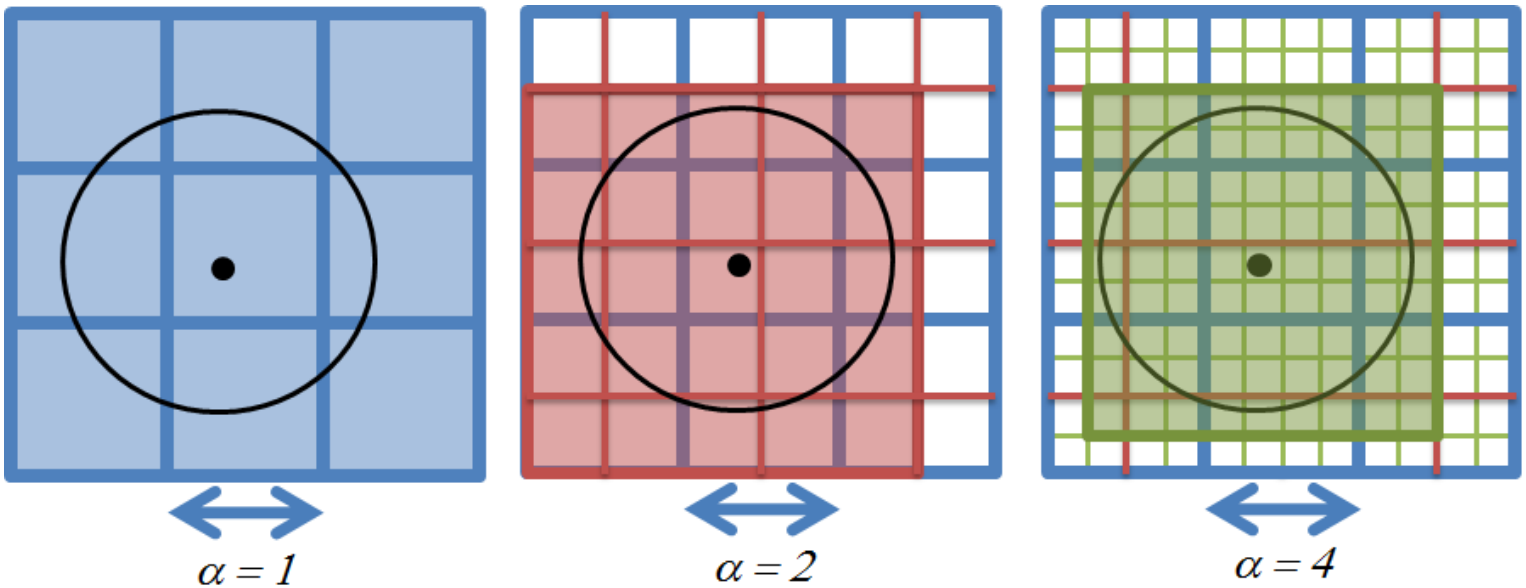
Space Tiling

- Set width of single hypercube to $\Delta = \theta/\alpha$
- Tile $\Omega = S \cup T$ into the adjacent cubes C
  - Coordinates: $(c_1, \ldots, c_n) \in \mathbb{N}^n$
  - Contains points $\omega \in \Omega : \forall i \in \{1 \ldots n\}, c_i\Delta \leq \omega_i < (c_i + 1)\Delta$



HYPPO

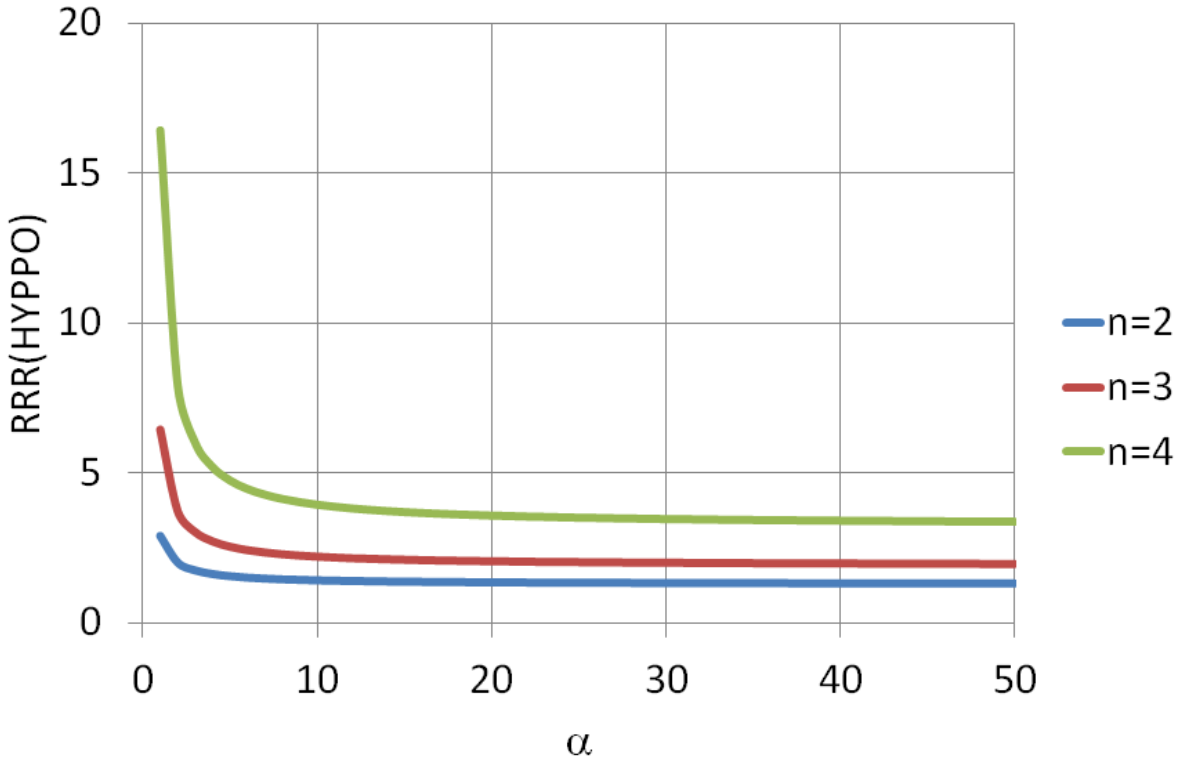
- Combine $(2\alpha + 1)^n$ hypercubes around $C(\omega)$ to approximate hypersphere



- $RRR(HYPPO(\alpha)) = \frac{(2\alpha+1)^n}{\alpha^n S(n)}$
- $\lim_{\alpha \to \infty} RRR(HYPPO(\alpha)) = \frac{2^n}{S(n)}$

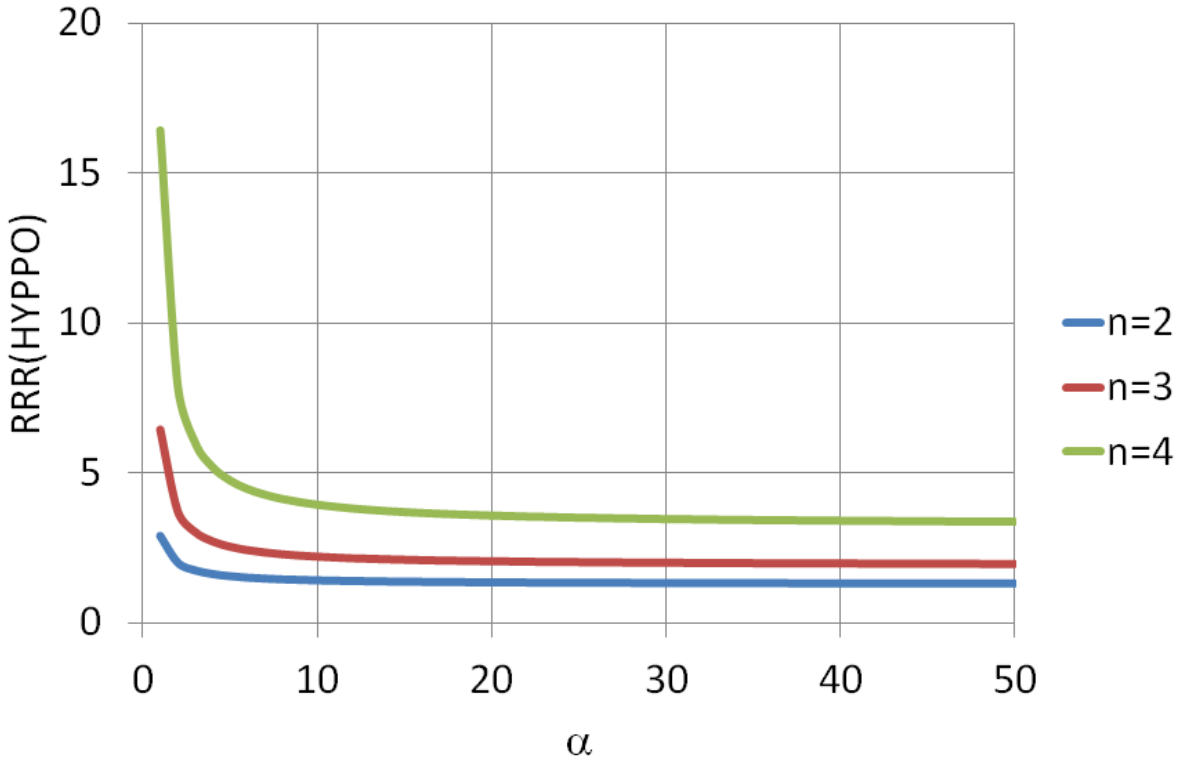
HYPPO

- RRR(HYPPO) for $p = 2$, $n = 2, 3, 4$ and $2 \leq \alpha \leq 50$



HYPPO

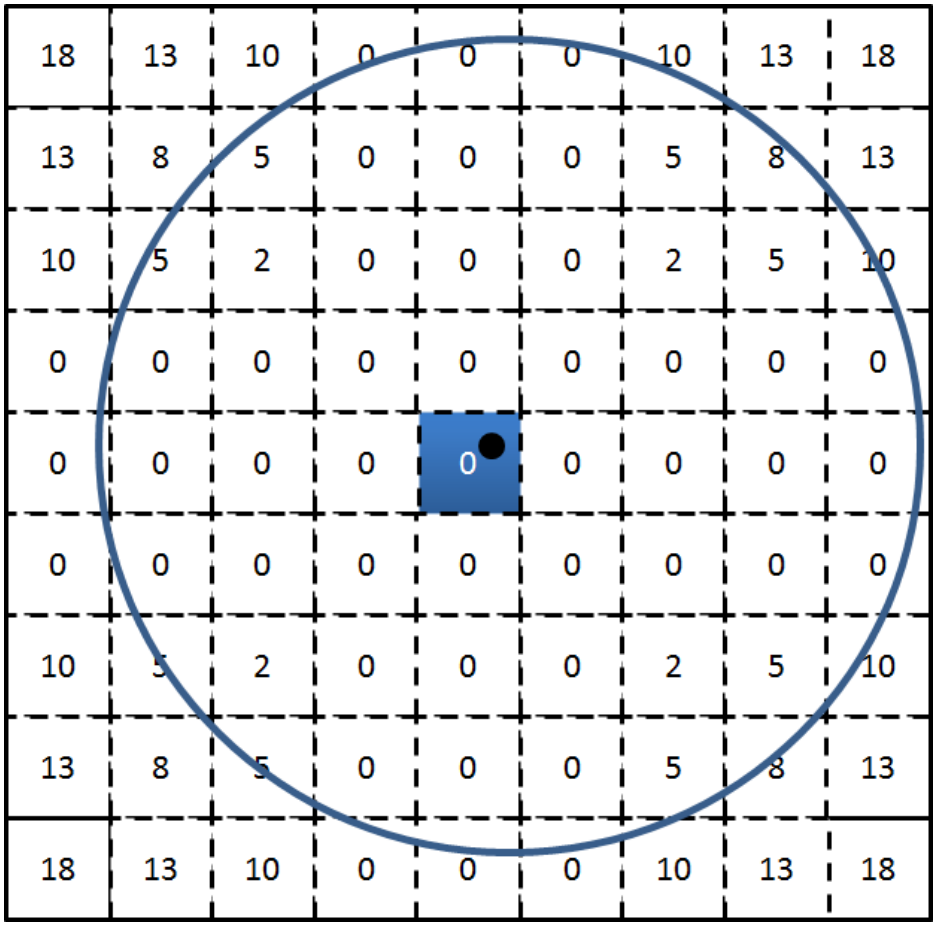
- RRR(HYPPO) for $p = 2$, $n = 2, 3, 4$ and $2 \leq \alpha \leq 50$



- $\lim_{\alpha \to \infty} RRR(HYPPO(\alpha)) = \frac{4}{\pi} \approx 1.27 \ (n = 2)$
- $\lim_{\alpha \to \infty} RRR(HYPPO(\alpha)) = \frac{6}{\pi} \approx 1.91 \ (n = 3)$
- $\lim_{\alpha \to \infty} RRR(HYPPO(\alpha)) = \frac{32}{\pi^2} \approx 3.24 \ (n = 4)$

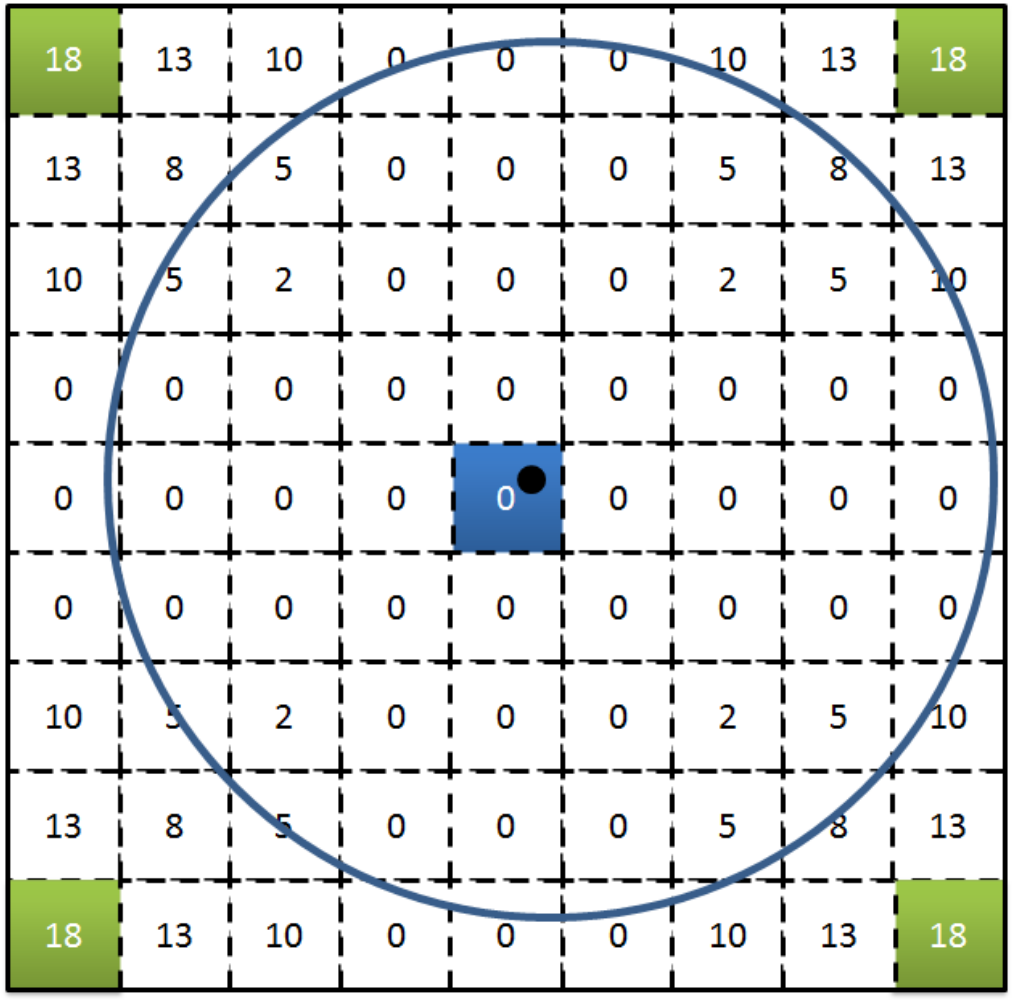
$\mathcal{HR}^3$: Idea

$$index(C,\omega) = \begin{cases} 0 \text{ if } \exists i : |c_i - c(\omega)_i| \leq 1, 1 \leq i \leq n, \\ \sum_{i=1}^{n} (|c_i - c(\omega)_i| - 1)^p \text{ else,} \end{cases}$$



$\mathcal{HR}^3$: Idea

- Compare $C(\omega)$ with C iff $index(C, \omega) \leq \alpha^p$
- $\alpha = 4, p = 2$



$\mathcal{HR}^3$: Idea

Claims

- No loss of recall
- $\lim_{\alpha \to \infty} RRR(\mathcal{HR}^3(\alpha)) = 1$

$\mathcal{HR}^3$: Lemma 1

Lemma

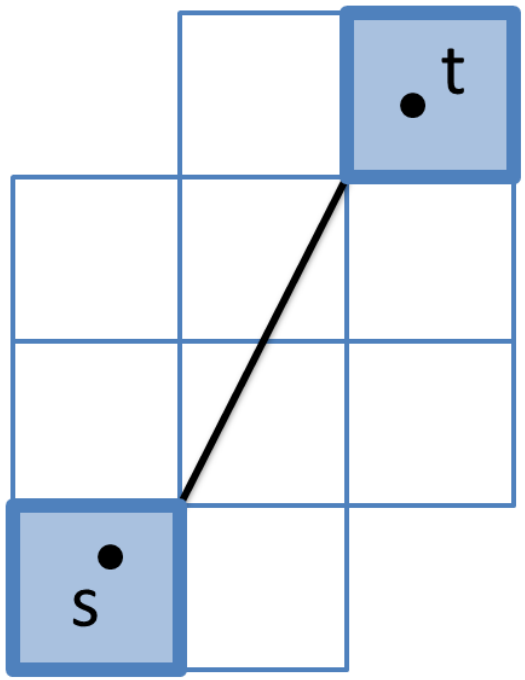
$index(C, s) = x \Rightarrow \forall t \in C \ \delta^p(s, t) > x\Delta^p$

$\mathcal{HR}^3$: Lemma 1

Lemma

$index(C, s) = x \Rightarrow \forall t \in C \ \delta^p(s, t) > x\Delta^p$

- $p = 2, n = 2, index(C, s) = 5$



$\mathcal{HR}^3$: Lemma 1

Lemma

$index(C, s) = x \Rightarrow \forall t \in C\ \delta^p(s, t) > x\Delta^p$

Proof.

- $index(C, s) = x \Rightarrow \sum_{i=1}^{n}(|c_i - c_i(s)| - 1)^p = x$
- Out of definition of cube index follows $|s_i - t_i| > (|c_i - c_i(s)| - 1)\Delta$
- Thus, $\sum_{i=1}^{n}|s_i - t_i|^p > \sum_{i=1}^{n}(|c_i - c_i(s)| - 1)^p\Delta^p$
- Therewith, $\delta^p(s, t) > x\Delta^p$

$\mathcal{HR}^3$: Lemma 2

Lemma

$\forall s \in S$: *index*$(C, s) > \alpha^p$ *implies that all* $t \in C$ *are non-matches*

Proof.

Follows directly from Lemma 1:
index$(C, s) > \alpha^p \Rightarrow \forall t \in C, \delta^p(s, t) > \Delta^p \alpha^p = \theta^p$ □

$\mathcal{HR}^3$: Idea

Claims

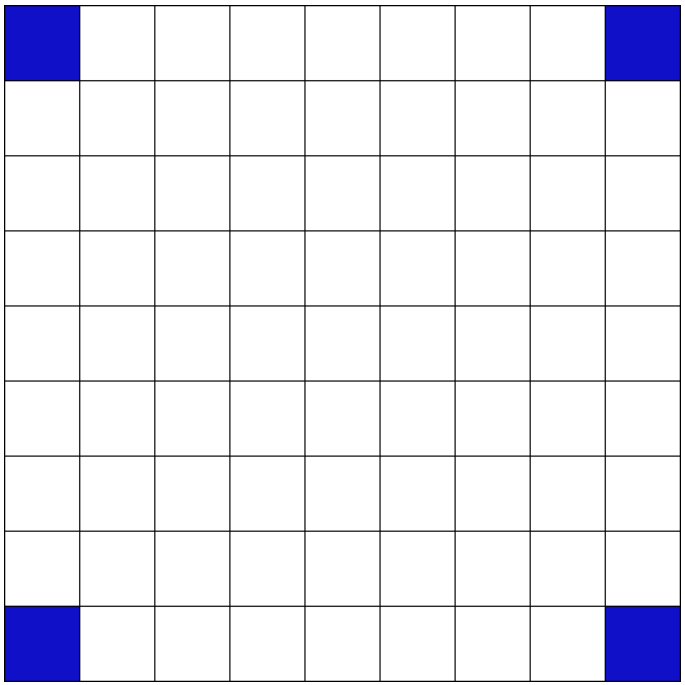
- No loss of recall ✓
- $\lim_{\alpha \to \infty} RRR(\mathcal{HR}^3(\alpha)) = 1$

$\mathcal{HR}^3$: Lemma 3

Lemma

$\forall \alpha > 1 \; RRR(\mathcal{HR}^3(2\alpha)) < RRR(\mathcal{HR}^3(\alpha))$

- $p = 2, \alpha = 4$

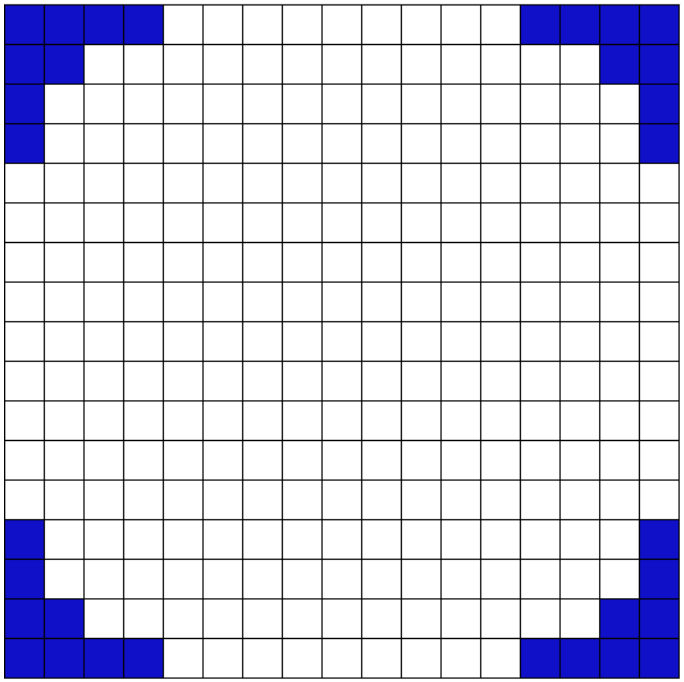


$\mathcal{HR}^3$: Proof

Lemma

$\forall \alpha > 1 \; RRR(HR^3(2\alpha)) < RRR(HR^3(\alpha))$

- $p = 2, \alpha = 8$

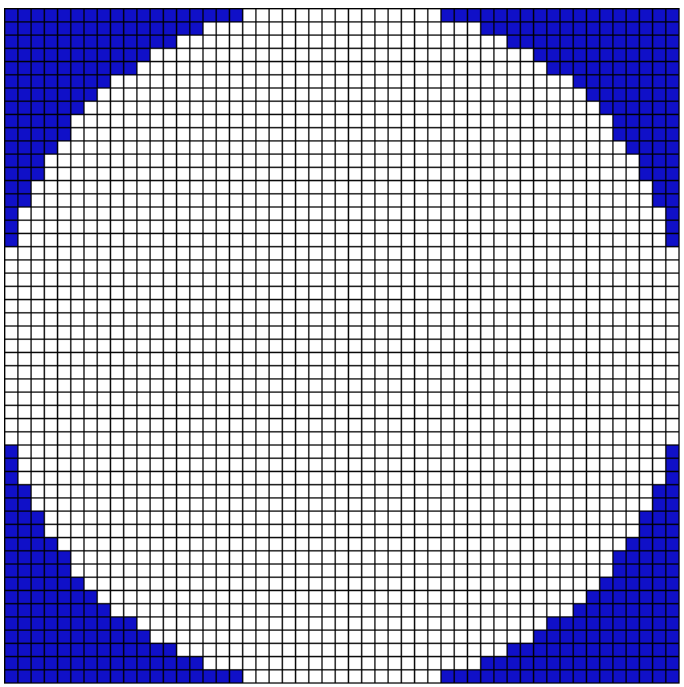


$\mathcal{HR}^3$: Proof

Lemma

$\forall \alpha > 1 \; RRR(HR^3(2\alpha)) < RRR(HR^3(\alpha))$

- $p = 2, \alpha = 25$

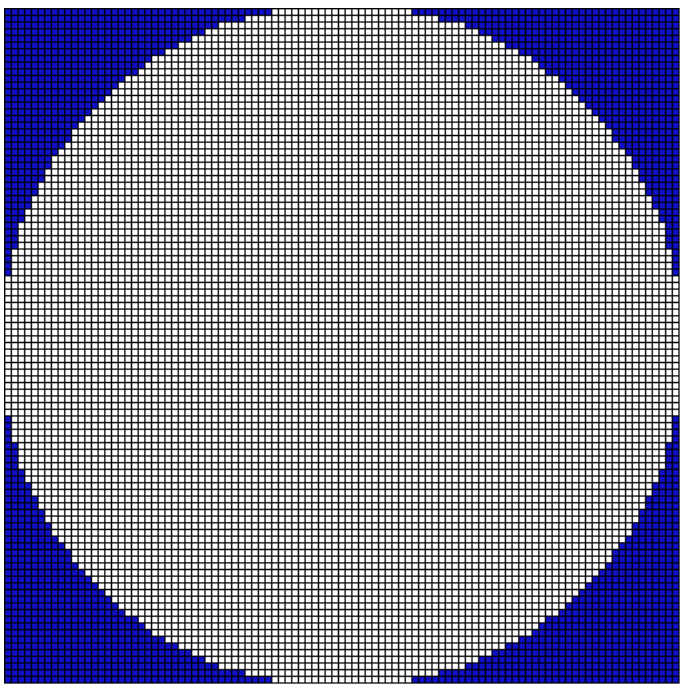


$\mathcal{HR}^3$: Proof

Lemma

$\forall \alpha > 1 \; RRR(HR^3(2\alpha)) < RRR(HR^3(\alpha))$

- $p = 2, \alpha = 50$



$\mathcal{HR}^3$: Theorem

Theorem

$$\lim_{\alpha \to \infty} RRR(\mathcal{HR}^3(\alpha)) = 1$$

Proof.

- $\alpha \to \infty \Rightarrow \Delta \to 0$
- Thus, $C(s) = \{s\}, C = \{t\}$
- Index function : $\sum_{i=1}^{n} \Delta^p(|c_i(s) - c_i| - 1)^p \leq \Delta^p \alpha^p$
- $\Delta \to 0 \Rightarrow \sum_{i=1}^{n} |s_i - t_i|^p \leq \theta^p$

$\mathcal{HR}^3$: Idea

Claims

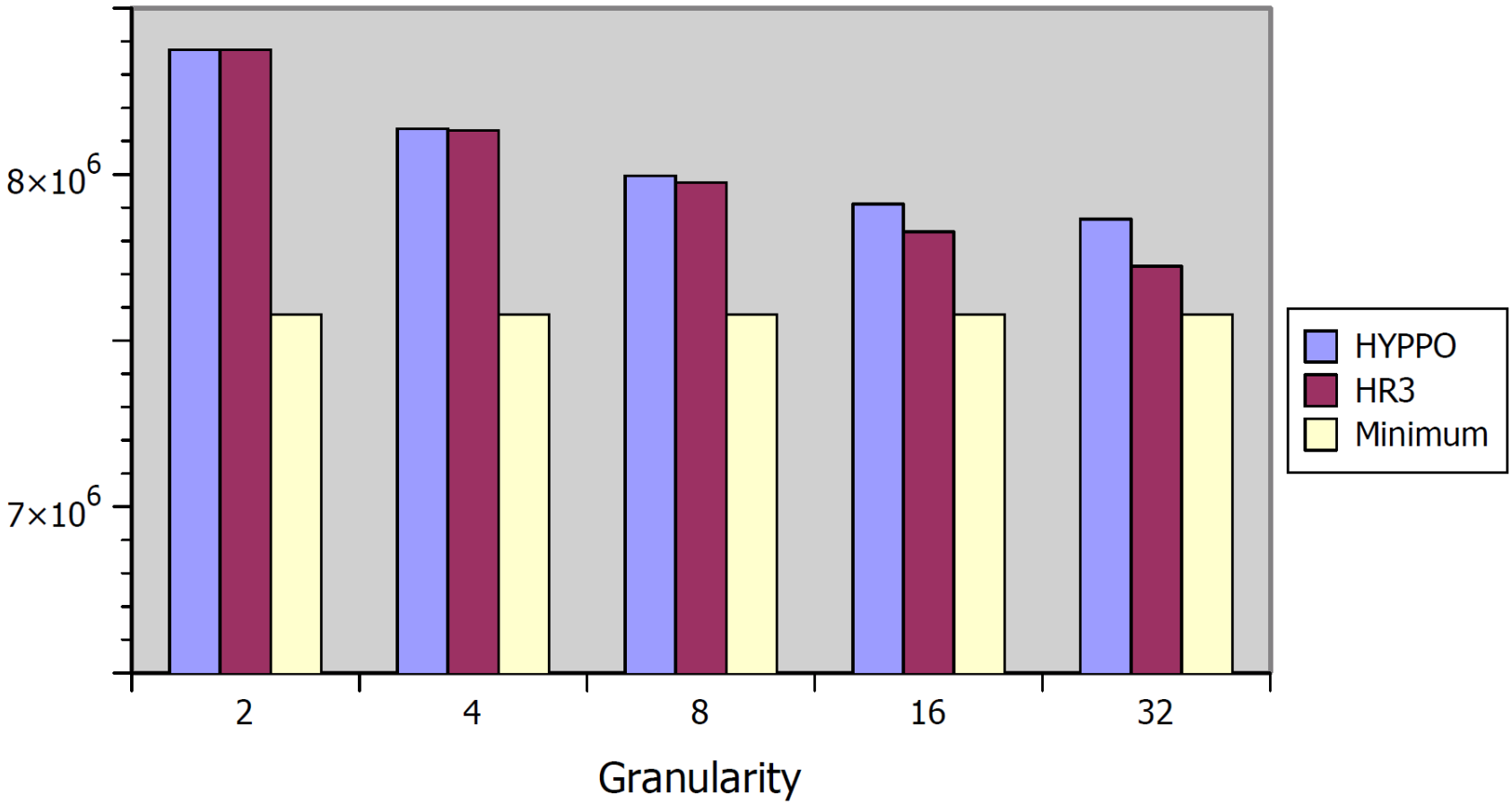
- No loss of recall ✓
- $\lim_{\alpha \to \infty} RRR(\mathcal{HR}^3(\alpha)) = 1$ ✓

$\mathcal{HR}^3$: Experiments

- Compare $\mathcal{HR}^3$ with LIMES 0.5's HYPPO and SILK 2.5.1
- Experimental Setup:
  - Deduplicating DBpedia places by minimum elevation, elevation and maximum elevation ($\theta = 49m, 99m$).
  - Geonames and LinkedGeoData by longitude and latitude ($\theta = 1°, 9°$)
- Windows 7 Enterprise machine 64-bit computer with a 2.8GHz i7 processor with 8GB RAM.

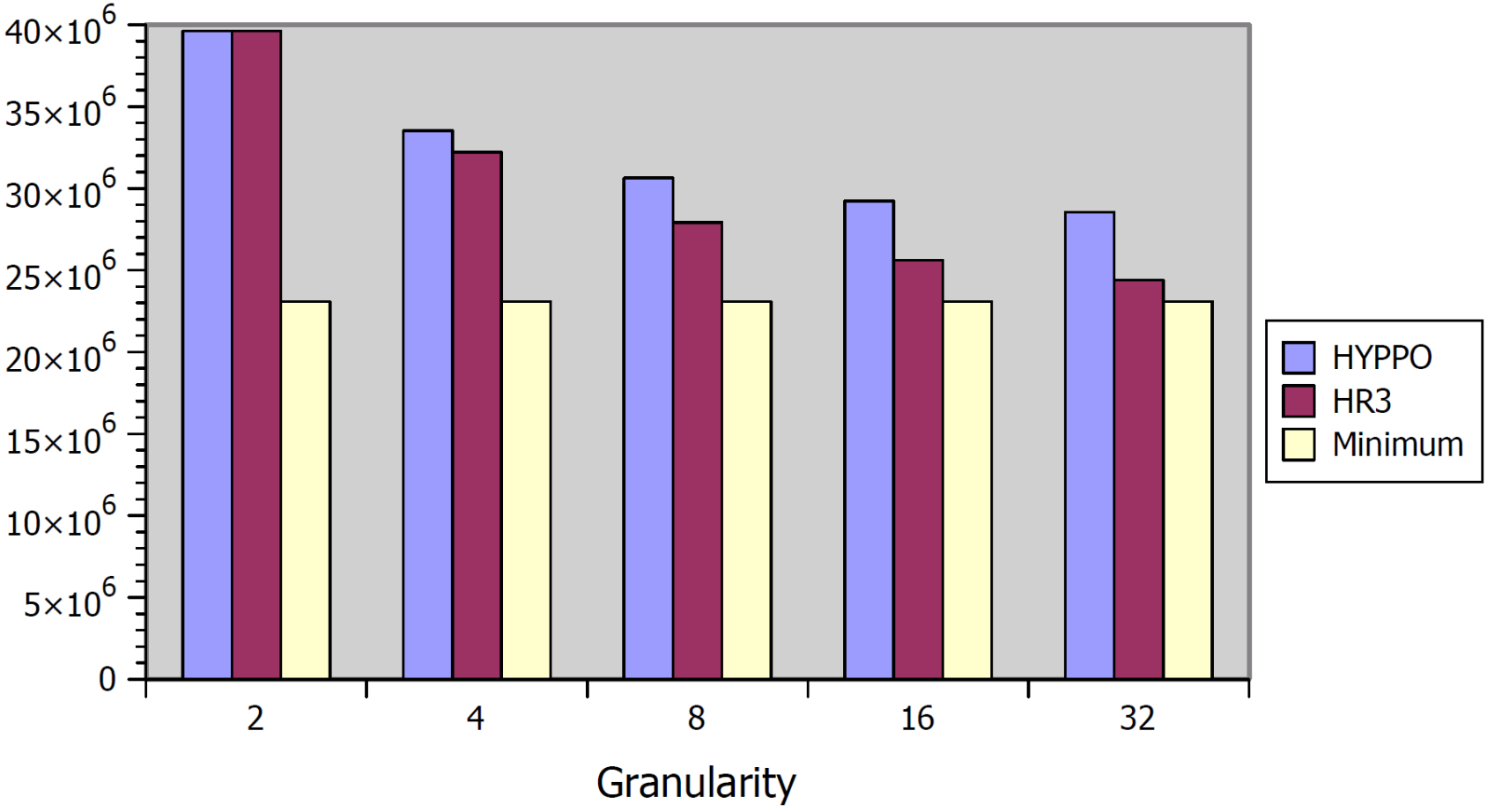
$\mathcal{HR}^3$: Experiments (Comparisons)

- Experiment 2: Deduplicating DBpedia places, $\theta = 99m$
- 0.64×10^6 less comparisons



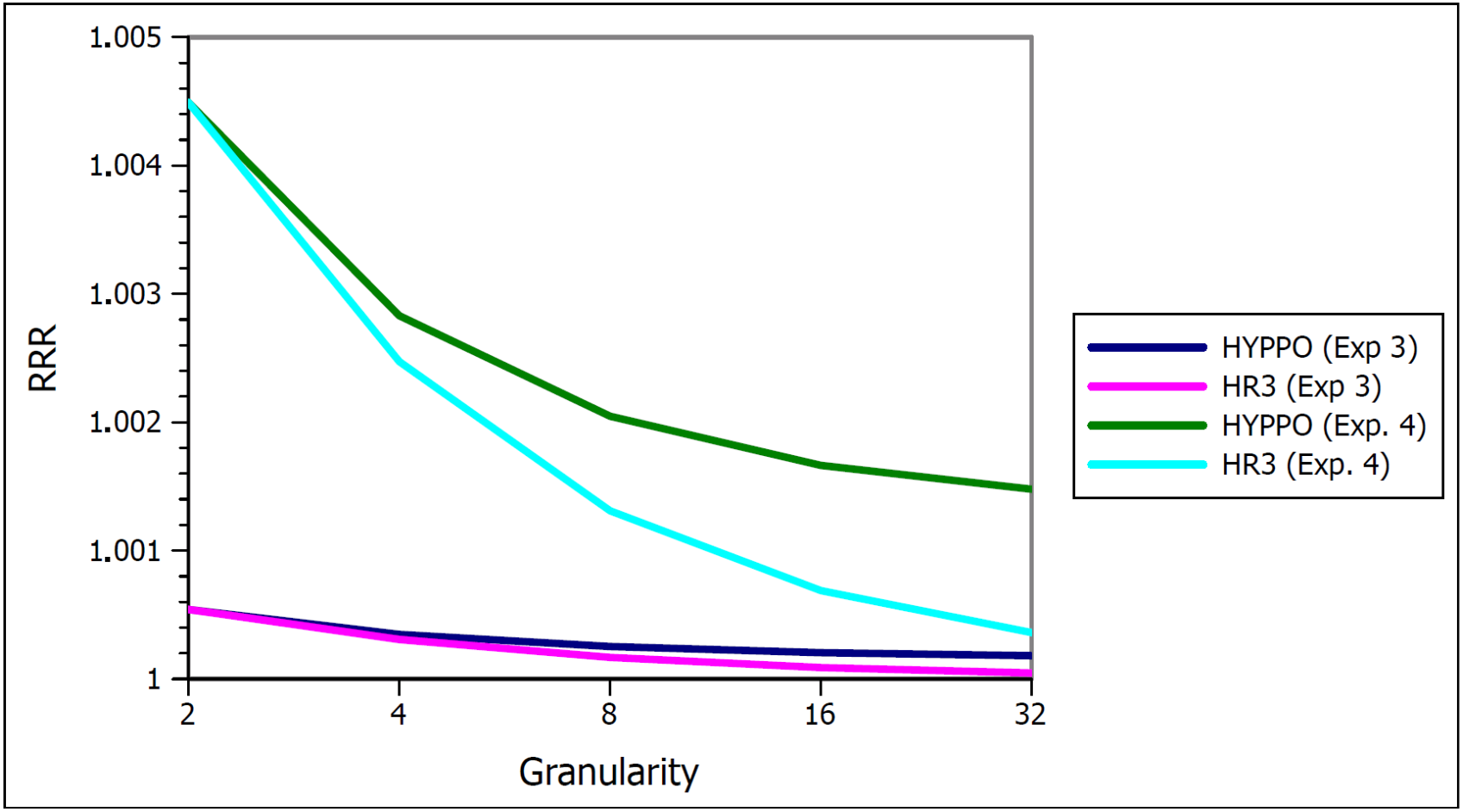
$\mathcal{HR}^3$: Experiments (Comparisons)

- Experiment 4: Linking Geonames and LinkedGedData, $\theta = 9°$
- 4.3×10^6 less comparisons



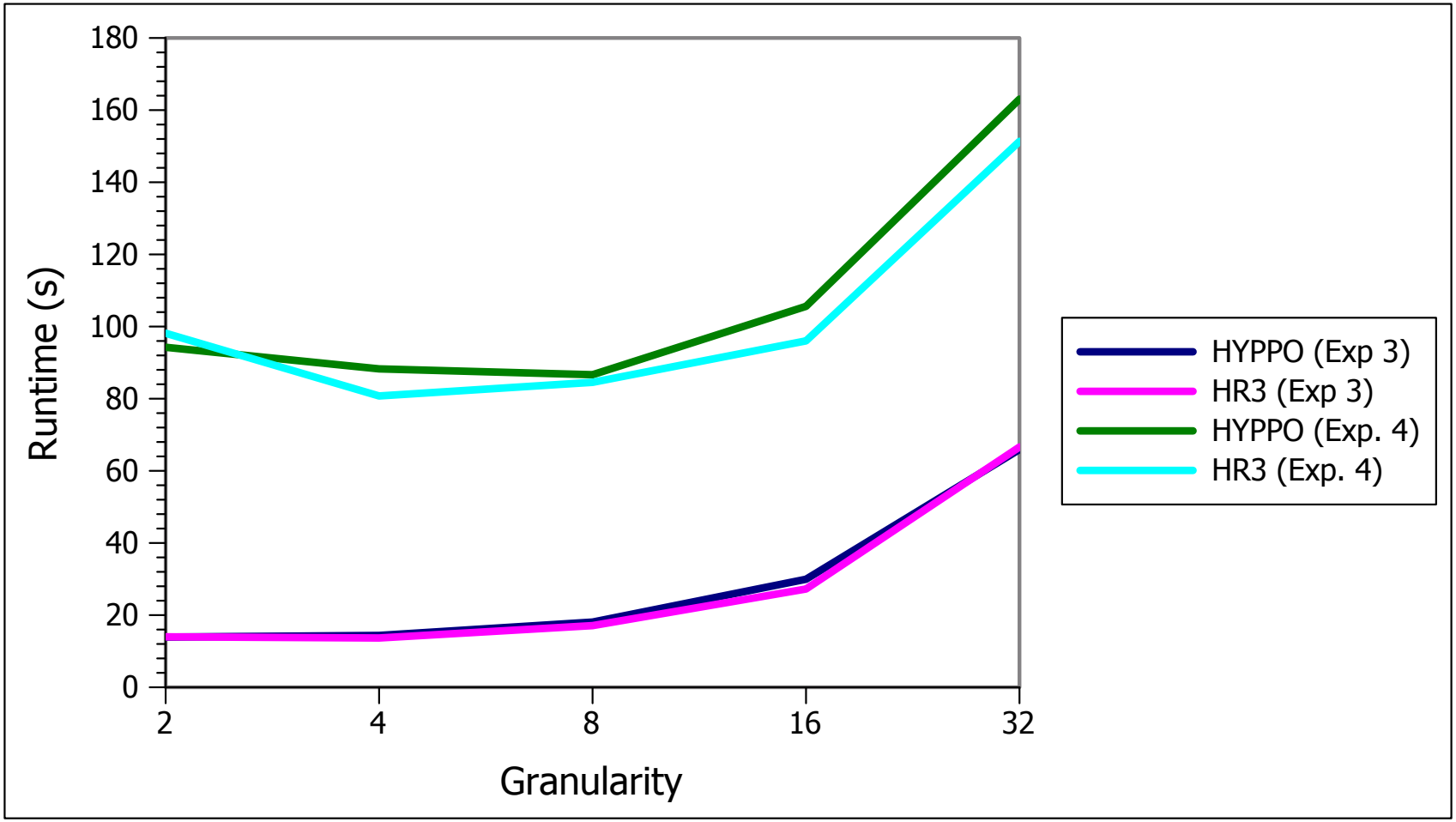
$\mathcal{HR}^3$: Experiments (RRR)

- Experiment 3,4: Geonames and LGD, $\theta = 1,9°$



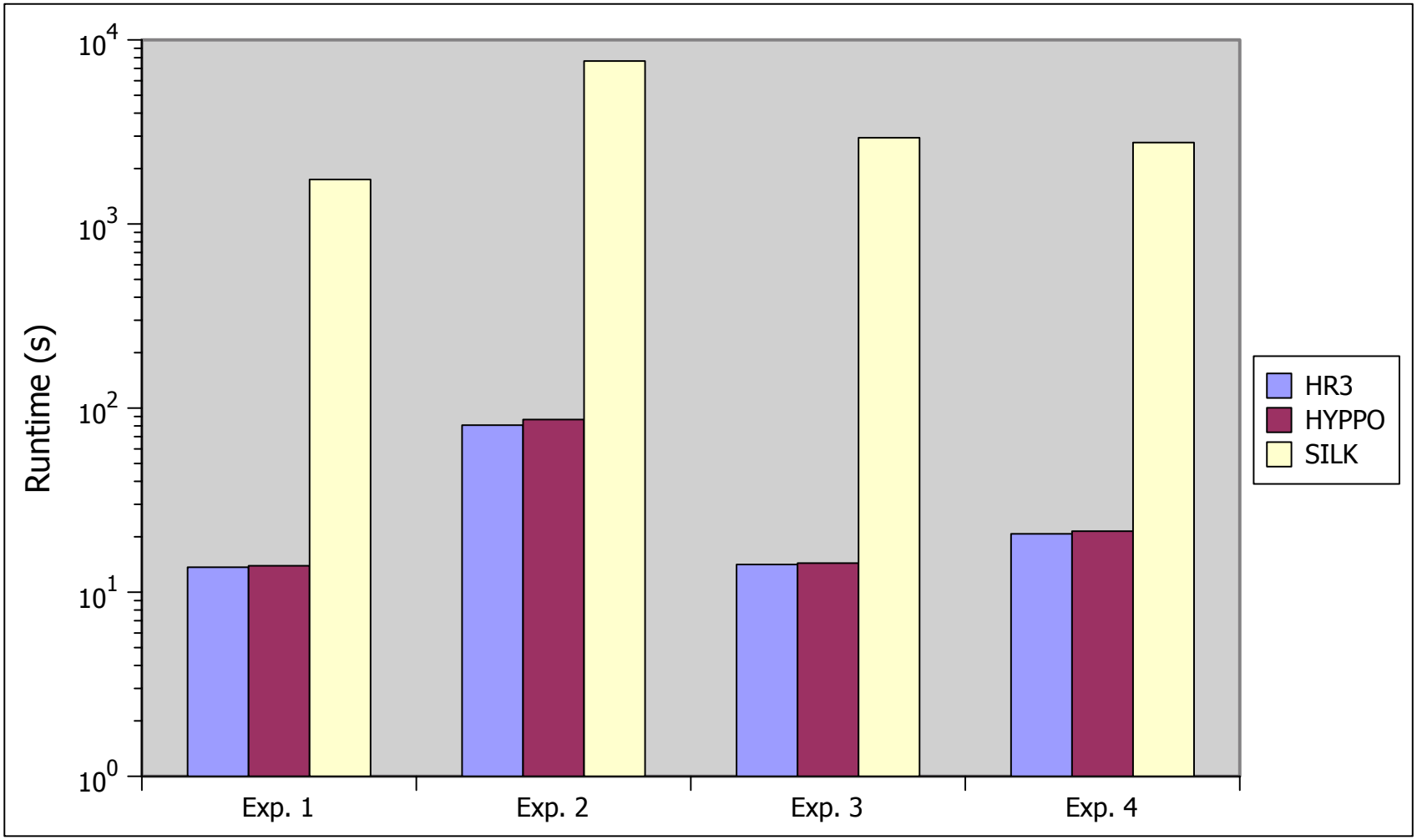
$\mathcal{HR}^3$: Experiments (Runtime)

- Experiment 3,4: Geonames and LGD, $\theta = 1,9°$



$\mathcal{HR}^3$: Experiments (Runtime)

- Experiment 1, 2: DBpedia, $\theta = 49,99m$
- Experiment 3, 4: Geonames and LGD, $\theta = 1,9°$



$\mathcal{HR}^3$: Conclusion

Mission

- New category of algorithms for link discovery

$\mathcal{HR}^3$: Conclusion

Mission

- New category of algorithms for link discovery

- Presented $\mathcal{HR}^3$
  - Link discovery in affine spaces with Minkowski measures
  - Outperforms the state of the art (runtime, comparisons)
  - Integrated in LIMES.

- Future Work
  - Combine $\mathcal{HR}^3$ with multi-indexing approach
  - Devise resource management approach
  - Develop other algorithms (esp. for strings) with the same/similar theoretical guarantees

Thank You!

Questions?

Axel Ngonga
Augustusplatz 10
D-04109 Leipzig
ngonga@informatik.uni-leipzig.de
http://bis.uni-leipzig.de/AxelNgonga
http://limes.sf.net

Technical Details

- Discretization: $|S \cup T|$
- Index calls: $|C|((2\alpha + 1)^n - 1)$
- Comparisons: $\sum_C |C \cap S| \left(\sum_{f(C) \cap T} |f(C)| \right)$