On Redundancy vs Dependency Preservation in Normalization: An Information-Theoretic Study of 3NF

Solmaz Kolahi
University of Toronto
solmaz@cs.toronto.edu

Leonid Libkin
University of Toronto and University of Edinburgh
libkin@cs.toronto.edu and libkin@inf.ed.ac.uk

ABSTRACT

A recently introduced information-theoretic approach to analyzing redundancies in database design was used to justify normal forms like BCNF that completely eliminate redundancies. The main notion is that of an information content of each datum in an instance (which is a number in $[0, 1]$): the closer to 1, the less redundancy it carries. In practice, however, one usually settles for 3NF which, unlike BCNF, may not eliminate all redundancies but always guarantees dependency preservation.

In this paper we use the information-theoretic approach to prove that 3NF is the best normal form if one needs to achieve dependency preservation. For each dependency-preserving normal form, we define the *price of dependency preservation* as an information-theoretic measure of redundancy that gets introduced to compensate for dependency preservation. This is a number in the $[0, 1]$ range: the smaller it is, the less redundancy a normal form guarantees. We prove that for every dependency-preserving normal form, the price of dependency preservation is at least 1/2, and it is precisely 1/2 for 3NF. Hence, 3NF has the least amount of redundancy among all dependency-preserving normal forms. We also show that, information-theoretically, unnormalized schemas have at least twice the amount of redundancy than schemas in 3NF.

1. Introduction

In this paper we provide a justification for one of the most popular and commonly used normal forms, 3NF. We adopt a recently proposed information-theoretic framework for reasoning about database designs [4].

The problem of database normalization is one of the oldest and most researched in database theory and practice, with descriptions of well-known normal forms such as 3NF and BCNF appearing in practically all texts (see, e.g., [1, 16, 19]) and many practical tools existing for database design. Nonetheless, the question of what is it that makes a database design good had not been dealt with nearly as thoroughly, with texts typically offering a rather informal explanation based on the absence of update anomalies or elimination of redundancies. Papers that attempted a more formal evaluation of normal forms (e.g. [12, 13, 21]) still appealed to the notions of eliminating update anomalies.

To justify relational normal forms, and to provide a test of "goodness" of normal forms for other data models, [4] proposed an *information-theoretic framework* that is completely independent of the notions of update/query languages, and is based on the intrinsic properties of the data. The key concept of the framework is that of the *relative information content*, $\mathrm{RIC}_I(p|\Sigma)$, of a position p in a database instance I with respect to a set of constraints Σ. It is defined as a conditional entropy of a certain probability distribution, and is then normalized to the interval $[0, 1]$. Intuitively, if $\mathrm{RIC}_I(p|\Sigma) = 1$, then p carries the maximum possible amount of information: nothing about it can be inferred from the rest of the instance. Smaller values of $\mathrm{RIC}_I(p|\Sigma)$ say that positions carry some amount of redundancy, as some information about them can be inferred.

The notion of a *well-designed* normal form then says that in every instance I of a schema in that normal form, the relative information content $\mathrm{RIC}_I(p|\Sigma)$ of every position p is 1. That is, no redundancies are allowed. Characterizations of well-designed normal forms for different types of dependencies were obtained in [4]: for example, if Σ consists only of functional dependencies (FDs), then being well-designed is the same as being in the Boyce-Codd normal form (BCNF).

While this does justify a normal form that is perhaps the most popular one for database texts, BCNF is *not* the most common and popular normal form in practice – that role belongs to 3NF. For example, Oracle's "General Database Design FAQ" [23] defines designs that progressively achieve 1NF, 2NF, and 3NF, and then says that there are other normal forms but *"their definitions are of academic concern only, and are rarely required for practical purposes"*.

The main property possessed by 3NF but not BCNF is *de-*

PODS'06, June 26–28, 2006, Chicago, Illinois, USA.

pendency preservation: for every schema, there always exists a lossless decomposition into 3NF that preserves all the constraints. This is a very important property for integrity enforcement, as DBMSs provide a variety of mechanisms to ensure that integrity constraints are enforced during updates.

Therefore, if one needs to guarantee the integrity of the database, and uses a dependency-preserving normal form, *some* redundancy must be tolerated. A natural question is then whether 3NF is the right choice of a dependency-preserving normal form. More precisely, if we look at all normal forms that guarantee dependency preservation (which excludes BCNF) and apply the information-theoretic approach to measure the amount of redundancy they introduce, will 3NF be the one with the least amount of redundancy?

Here we give a positive answer to this question. Our two main results, stated informally, are:

1. Among all normalization conditions that guarantee dependency-preserving decompositions, 3NF has the least amount of redundancy. Thus, if dependency preservation is essential, 3NF is the best normal form.
2. 3NF has at least 50% less redundancy than unnormalized designs.

To state these formally, assume that $\mathcal{NF}$ is some dependency-preserving normal form. That is, every schema admits a lossless dependency-preserving decomposition into $\mathcal{NF}$. We then look at the *guaranteed information content* provided by $\mathcal{NF}$, i.e., the largest number $c \in [0, 1]$ such that every schema may be decomposed into $\mathcal{NF}$ in such a way that in all instances of the decomposed schema and all positions, the information content $\mathrm{RIC}(p|\Sigma)$ is at least c.

If the guaranteed information content equals c, then $1 - c$ is the *price of dependency preservation*, denoted by $\mathrm{PRICE}(\mathcal{NF})$: that is, the minimum amount of information content one must lose due to dependency preservation. We then prove the following.

Theorem A $\mathrm{PRICE}(3\mathrm{NF}) = 1/2$. *Furthermore, if $\mathcal{NF}$ is a dependency-preserving normal form, then $\mathrm{PRICE}(\mathcal{NF}) \geq 1/2$.*

In other words, 3NF achieves the smallest price one needs to pay to ensure dependency preservation, and thus among normal forms that guarantee dependency preservation it is the one with the least amount of redundancy. Moreover, it follows from the proof of Theorem A that 3NF designs produced by the standard synthesis algorithm [1, 6] are the ones that guarantee the smallest price of dependency preservation.

This last observation also motivates our second result. It has long been known [22, 25] that for some schemas already in 3NF, better 3NF designs can be produced by the standard synthesis algorithm (in fact [25] even proposed a different normal form for schemas that arise in such a way). Hence, an arbitrary 3NF schema may have quite a bit of extra redundancy. In fact, a first look at 3NF from the information-theoretic point of view was already taken in [17] with what looked like a rather discouraging result: for every $\varepsilon > 0$, one can find a 3NF schema with a set Σ of FDs, an instance of that schema, and a position p such that $\mathrm{RIC}_I(p|\Sigma) < \varepsilon$. Nonetheless, the example of [17] requires arbitrarily large sets of attributes and schemas that can be further decomposed into better 3NF designs.

This gives rise to the following question: what can be said about arbitrary 3NF schemas, not only the good ones that ensure the lower price of dependency preservation? Can they be as bad as arbitrary schemas? How do they compare to "good" 3NF designs?

To answer these, we introduce a *gain of normalization* function that allows us to compare different normal forms. For every $m \in \mathbb{N}$, and a condition $\mathcal{P}$ on schemas, we define the set of possible values of $\mathrm{RIC}_I(p|\Sigma)$ for m-attribute instances I of schemas satisfying $\mathcal{P}$:

$$\mathcal{POSS}_{\mathcal{P}}(m) = \{\mathrm{RIC}_I(p|\Sigma) \mid \begin{array}{l} I \text{ is an instance of } (R, \Sigma), \\ R \text{ has } m \text{ attributes}, \\ (R, \Sigma) \text{ satisfies } \mathcal{P}\}, \end{array}$$

and define the value $\inf \mathcal{POSS}_{\mathcal{P}}(m)$ (typically these sets are dense subsets of intervals $(\varepsilon, 1]$). For normal forms $\mathcal{NF}_1$ and $\mathcal{NF}_2$, the gain of normalization function is

$$\mathrm{GAIN}_{\mathcal{NF}_1/\mathcal{NF}_2}(m) \;=\; \frac{\inf \mathcal{POSS}_{\mathcal{NF}_1}(m)}{\inf \mathcal{POSS}_{\mathcal{NF}_2}(m)}.$$

That is, we measure the ratio of the least amount of information in instances of $\mathcal{NF}_1$- and $\mathcal{NF}_2$-schemas.

Our second main result formally confirms that some 3NF schemas may have more redundancy than others, but it also says that arbitrary 3NF schemas have at least twice the information content compared to unnormalized schemas. Here we use the notation $3\mathrm{NF}^+$ to refer to the schemas that arise as outputs of the standard 3NF synthesis algorithm, and All to refer to the class of all (unnormalized) schemas.

Theorem B *For every $m > 2$:*

- $\mathrm{GAIN}_{3\mathrm{NF}/\mathsf{All}}(m) = 2$;
- $\mathrm{GAIN}_{3\mathrm{NF}^+/3\mathrm{NF}}(m) = 2^{m-3}$.
- $\mathrm{GAIN}_{3\mathrm{NF}^+/\mathsf{All}}(m) = 2^{m-2}$;

These are our main results, but we prove others along the way, for example, comparing 3NF to BCNF, and computing exact values of $\inf \mathcal{POSS}_{\mathcal{NF}}$ for some conditions $\mathcal{NF}$.

In the next section we recall the basics of normalization, information theory, and review the $\mathrm{RIC}_I(p|\Sigma)$ measure. In Section 3 we prove Theorem A. In Section 4 we prove Theorem B. In Section 5 we give concluding remarks.

2. Background

2.1 Schemas and instances

In general R will stand for a relation name and S for a schema that consists of a set of relation names. With each R we associate a set of its attributes denoted by $attr(R)$. Instead of $attr(R) = \{A_1, \ldots, A_m\}$ we sometimes write $R(A_1, \ldots, A_m)$. Elements of database instances come from a countably infinite domain; to be concrete, we assume it to be $\mathbb{N}_+$, the set of positive integers. An instance I of S assigns to each m-attribute relation R in S a finite subset $I(R)$ of $\mathbb{N}_+^m$. We let $adom(I)$ stand for the *active domain* of I: the set of all elements of $\mathbb{N}_+$ that occur in I.

If I is an instance of S, the set of *positions* in I, denoted by $Pos(I)$, is the set $\{(R, t, A) \mid R \in S,\ t \in I(R)$ and $A \in attr(R)\}$.

Schemas may contain *integrity constraints*, in which case we refer to schemas (S, Σ), where S is a set of relation names and Σ is a set of constraints. We usually write (R, Σ) instead of the more formal $(\{R\}, \Sigma)$ in case of one relation. Since we are interested in 3NF, we deal with functional dependencies (FDs); we assume that FDs are of the form $X \to Y$ with both X and Y nonempty. If Σ is a set of FDs, then Σ^+ denotes the set of all FDs implied by it, and $inst(S, \Sigma)$ stands for the set of all instances of S satisfying Σ. We write $inst_k(S, \Sigma)$ for the set of instances $I \in inst(S, \Sigma)$ with $adom(I) \subseteq [1, k]$.

2.2 Normal forms

We review the most basic definitions and refer the reader to surveys [5, 15, 7] and texts [1, 16, 19] for additional information. A schema (S, Σ) is in BCNF if for every relation name R in it and every nontrivial FD $X \to Y$ over attributes of R, X is a key of R. Prime attributes are those that belong to a candidate (minimal) key. A schema (S, Σ) is in 3NF if for every relation name R in it and every nontrivial FD $X \to Y$ over attributes of R, either X is a key, or every attribute in $Y - X$ is prime.

Given a schema (R, Σ) and some normal form $\mathcal{NF}$, a set of schemas (R_j, Σ_j), $j \in J$, is called a (lossless) $\mathcal{NF}$-decomposition if each (R_j, Σ_j) is in $\mathcal{NF}$, and for every $I \in inst(R, \Sigma)$ we have $\pi_{attr(R_j)}(I) \models \Sigma_j$ and furthermore $I = \bowtie \{\pi_{attr(R_j)}(I) \mid j \in J\}$. Such a decomposition is called *dependency-preserving* if $\left(\bigcup_j \Sigma_j\right)^+ = \Sigma^+$. It is well-known that both 3NF and BCNF admit lossless decompositions, which in the case of 3NF can be guaranteed to be dependency-preserving. In the case of BCNF dependency preservation is not always possible (consider a schema with attributes A, B, C and FDs $AB \to C$ and $C \to A$).

3NF designs are often produced by a synthesis algorithm proposed initially in [6]. The algorithm works as follows (see, e.g., [1]): given a set of FDs Σ, it computes a minimal cover Σ_c. If there is an FD $X \to A$ in Σ_c (where A is an attribute) such that $X \cup \{A\}$ contain all attributes, it stops; otherwise for each $X \to A \in \Sigma_c$ it outputs a schema $(XA, X \to A)$, and it combines two schemas if one is contained in the other. Also, if none of the sets XA contains a key, a schema $(K, \emptyset)$ for some key K must be included. We shall refer to schemas produced by this algorithm as $3NF^+$ schemas.

2.3 Basics of information theory

The main concept of information theory is that of entropy, which measures the amount of information provided by a certain event. Assume that an event can have n different outcomes $s_1, \ldots, s_n$. Then for a probability space $\mathcal{A} = (\{s_1, \ldots, s_n\}, P_{\mathcal{A}})$, where $P_{\mathcal{A}}$ is a probability distribution, its entropy is defined as

$$H(\mathcal{A}) \;=\; \sum_{i=1}^{n} P_{\mathcal{A}}(s_i) \log \frac{1}{P_{\mathcal{A}}(s_i)}.$$

For probabilities that are zero, we adopt the convention that $0 \log \frac{1}{0} = 0$, since $\lim_{x \to 0} x \log \frac{1}{x} = 0$. It is known that $0 \leq H(\mathcal{A}) \leq \log n$, with $H(\mathcal{A}) = \log n$ only for the uniform distribution $P_{\mathcal{A}}(s_i) = 1/n$ [9].

We shall also need the concept of *conditional entropy*. For two probability spaces $\mathcal{A} = (\{s_1, \ldots, s_n\}, P_{\mathcal{A}})$, $\mathcal{B} = (\{s'_1, \ldots, s'_m\}, P_{\mathcal{B}})$ and, probabilities $P(s'_j, s_i)$ of all the events (s'_j, s_i) ($P_{\mathcal{A}}$ and $P_{\mathcal{B}}$ may not be independent), the conditional entropy of $\mathcal{B}$ given $\mathcal{A}$, denoted by $H(\mathcal{B} \mid \mathcal{A})$, gives the average amount of information provided by $\mathcal{B}$ if $\mathcal{A}$ is known [9]. If $P(s'_j \mid s_i) = P(s'_j, s_i)/P_{\mathcal{A}}(s_i)$ are conditional probabilities, then

$$H(\mathcal{B} \mid \mathcal{A}) = \sum_{i=1}^{n} \left(P_{\mathcal{A}}(s_i) \sum_{j=1}^{m} P(s'_j \mid s_i) \log \frac{1}{P(s'_j \mid s_i)} \right).$$

2.4 Relative information content

We now review the main definition of relative information content from [4] that was used to justify BCNF and other normal forms, and that we use here to justify 3NF. Unlike previously proposed information-theoretic measures [18, 8, 10, 20] that work only at the level of data, this measure takes into account both data and schema constraints.

Fix a schema S and a set Σ of constraints, and let $I \in inst(S, \Sigma)$. We want to define $\mathrm{RIC}_I(p \mid \Sigma)$, the relative information content of a position $p \in Pos(I)$ with respect to the set of constraints Σ. We want this value to be normalized to the interval $[0, 1]$. Since the maximum value of entropy for a discrete distribution on k elements is $\log k$, we shall define, for all k, a measure $\mathrm{RIC}_I^k(p \mid \Sigma)$ that works for instances $I \in inst_k(S, \Sigma)$, and take the limit of the ratio $\frac{\mathrm{RIC}_I^k(p|\Sigma)}{\log k}$ as $k \to \infty$.

Since this is a measure of the amount of redundancy, intuitively, we want to measure how much, on average, the value of position p is determined by any set of positions in I. For that, we take a set $X \subseteq Pos(I) - \{p\}$ and assume that the values in those positions X are lost, and then someone restores them from $[1, k]$. Then we measure (as the entropy of a suitably chosen distribution) how much information about the value in p this provides. The average such measure is $\text{RIC}_I^k(p \mid \Sigma)$.

Formally, we assume that I has n positions (which we enumerate as $1, \ldots, n$), and fix an n-element set of variables $\{v_i \mid 1 \le i \le n\}$. Fix a position $p \in Pos(I)$, and let $\Omega(I, p)$ be the set of all 2^{n-1} vectors $(a_1, \ldots, a_{p-1}, a_{p+1}, \ldots, a_n)$ such that for every $i \in [1, n] - \{p\}$, a_i is either v_i or the value in the i-th position of I. We make this into a probability space $\mathcal{A}(I, p) = (\Omega(I, p), P_u)$ with the uniform distribution $P_u(\bar{a}) = 2^{1-n}$.

We next define conditional probabilities $P_k(a \mid \bar{a})$ that show how likely a is to occur in position p, if values are removed from I according to the tuple $\bar{a} \in \Omega(I, p)$[1]. Let $I_{(a,\bar{a})}$ be obtained from I by putting a in position p, and a_i in position $i \neq p$. A substitution is a map $\sigma : \bar{a} \to [1, k]$ that assigns a value to each a_i which is a variable, and leaves other a_is intact. We let $\text{SAT}_\Sigma^k(I_{(a,\bar{a})})$ be the set of all substitutions σ such that $\sigma(I_{(a,\bar{a})}) \models \Sigma$ and $|\sigma(I_{(a,\bar{a})})| = |I|$ (the latter ensures that no two tuples collapse as the result of applying σ). Then $P_k(a \mid \bar{a})$ is defined as:

$$P_k(a \mid \bar{a}) \;=\; \frac{|\text{SAT}_\Sigma^k(I_{(a,\bar{a})})|}{\sum_{b \in [1,k]} |\text{SAT}_\Sigma^k(I_{(b,\bar{a})})|}.$$

With this, we define $\text{RIC}_I^k(p \mid \Sigma)$ as

$$\sum_{\bar{a} \in \Omega(I,p)} \left(\frac{1}{2^{n-1}} \sum_{a \in [1,k]} P_k(a \mid \bar{a}) \log \frac{1}{P_k(a \mid \bar{a})} \right).$$

Since $\sum_{a \in [1,k]} P_k(a \mid \bar{a}) \log \frac{1}{P_k(a|\bar{a})}$ measures the amount of information in p, given constraints Σ and some missing values in I, represented by the variables in $\bar{a}$, our measure $\text{RIC}_I^k(p \mid \Sigma)$ is the average such amount over all $\bar{a} \in \Omega(I, p)$.

To see that $\text{RIC}_I^k(p \mid \Sigma)$ is a conditional entropy, define

$$P_k'(a) \;=\; \frac{1}{2^{n-1}} \sum_{\bar{a} \in \Omega(I,p)} P_k(a \mid \bar{a}) \, .$$

It is a probability distribution on $[1, k]$ (intuitively, it says how likely an element from $[1, k]$ is to satisfy Σ when put in position p, given all possible interactions between p and sets of positions in I). If $\mathcal{B}_\Sigma^k(I, p)$ is the probability space $([1, k], P_k')$, then $\text{RIC}_I^k(p \mid \Sigma)$ is the conditional entropy:

$$\text{RIC}_I^k(p \mid \Sigma) \;=\; H(\mathcal{B}_\Sigma^k(I, p) \mid \mathcal{A}(I, p)).$$

[1]Technically, we should refer not to P_k but rather $P_{I,\Sigma,k}$ but I and Σ will always be clear from the context.

Since the domain of $\mathcal{B}_\Sigma^k(I, p)$ is $[1, k]$, we have $0 \le \text{RIC}_I^k(p \mid \Sigma) \le \log k$. To normalize this, we consider the ratio $\text{RIC}_I^k(p \mid \Sigma)/\log k$. The key observation of [4] is that for most reasonable constraints Σ (certainly for all definable in first-order logic), this sequence converges as $k \to \infty$, and we thus define

$$\text{RIC}_I(p|\Sigma) \;=\; \lim_{k \to \infty} \frac{\text{RIC}_I^k(p \mid \Sigma)}{\log k}.$$

The definition of being *well-designed* in [4] stated that (S, Σ) is well-designed if for every $I \in inst(S, \Sigma)$ and every $p \in Pos(I)$, $\text{RIC}_I(p \mid \Sigma) = 1$. It then showed that if Σ consists of FDs only, then (S, Σ) is well-designed iff it is in BCNF.

3. The price of dependency preservation and 3NF

Let $\mathcal{NF}$ be any dependency-preserving normal form: that is, for every relational schema (R, Σ), where Σ is a set of FDs, there is a lossless dependency-preserving decomposition of (R, Σ) into $(R_1, \Sigma_1), \ldots, (R_\ell, \Sigma_\ell)$, $\ell \ge 1$, such that each (R_i, Σ_i) satisfies $\mathcal{NF}$. We define the *guaranteed information content* for $\mathcal{NF}$-decompositions as the set of values $c \in [0, 1]$ such that for an arbitrary schema we can always guarantee an $\mathcal{NF}$-decomposition in which the information content in *all* positions is at least c. Formally, $\mathcal{G}(\mathcal{NF})$ is the set

$$\begin{array}{lll}\{c \in [0,1] & \mid & \forall\, (R, \Sigma),\ \forall\, I \in inst(R, \Sigma), \\ & & \exists\, \mathcal{NF}\text{-decomposition } \{(R_j, \Sigma_j)\}_{j=1}^{\ell} \text{ s.t.} \\ & & \forall j \le \ell\ \forall p \in Pos(I_j), \\ & & \text{RIC}_{I_j}(p|\Sigma_j) \ge c\},\end{array}$$

where I_j refers to $\pi_{attr(R_j)}(I)$. Using this, we define the price of dependency preservation for $\mathcal{NF}$ as the smallest amount of information content that is necessarily lost due to redundancies: that is, the smallest amount of redundancy one has to tolerate in order to have dependency preservation.

Definition 1. *For every dependency-preserving normal form $\mathcal{NF}$, the price of dependency preservation $\text{PRICE}(\mathcal{NF})$ is defined as $1 - \sup \mathcal{G}(\mathcal{NF})$.*

Clearly $\text{PRICE}(\mathcal{NF}) \le 1$. Since the FD-based normal form that achieves the maximum value 1 of $\text{RIC}_I(p|\Sigma)$ in all relations is BCNF [4], and BCNF does *not* ensure dependency preservation, $\text{PRICE}(\mathcal{NF}) > 0$ for any dependency-preserving normal form $\mathcal{NF}$.

Now we are ready to present the main result of this paper. It says that each normal form needs to pay at least $1/2$ in terms of redundancy to achieve dependency preservation, and this is exactly what 3NF pays.

Theorem A. $\text{PRICE}(\text{3NF}) = 1/2$. *If $\mathcal{NF}$ is a dependency-preserving normal form, then $\text{PRICE}(\mathcal{NF}) \ge 1/2$.*

In the rest of the section we prove this theorem. We say that a schema (R, Σ) is *indecomposable* if it has no lossless dependency-preserving decomposition. We are only interested in indecomposable schemas that are not in BCNF since BCNF already guarantees zero redundancy. The proof relies on two properties of indecomposable schemas presented in propositions below. Following [25], we say that a key X is *elementary* if there is an attribute $A \notin X$ such that $X' \to A \notin \Sigma^+$ for all $X' \subsetneq X$.

Proposition 1. *Let R have attributes $A_1, \ldots, A_m$, and let Σ be a non-empty set of FDs over R. Then (R, Σ) is indecomposable iff it has an $m-1$-attribute elementary candidate key.*

Proof: If (R, Σ) contains an $m-1$-attribute elementary candidate key, then every decomposition of it would lose this key; hence, it is indecomposable. Conversely, suppose (R, Σ) is indecomposable, and there is no elementary candidate key with $m-1$ attributes. Let Σ_c be an arbitrary minimal cover for Σ. Then for every FD $X \to A \in \Sigma_c$, we have $X \cup \{A\} \subsetneq attr(R)$. Hence the standard 3NF synthesis algorithm will produce a dependency-preserving decomposition of (R, Σ), a contradiction. $\square$

Let $\mathcal{ID}$ denote the property of being indecomposable. Recall (see the introduction) that $\mathcal{POSS}_{\mathcal{ID}}(m)$ is the set of possible values of $\mathrm{RIC}_I(p|\Sigma)$ for m-attribute instances of indecomposable schemas (R, Σ).

Proposition 2. $\inf \mathcal{POSS}_{\mathcal{ID}}(m) = 1/2$ *for all* $m > 2$.

Before we prove this proposition, we need a lemma. Let Σ be a set of FDs over a relation schema R, $I \in inst(R, \Sigma)$, $p \in Pos(I)$. We say that $\bar{a} \in \Omega(I, p)$ *determines* p if there exists $k_0 > 0$ such that for every $k > k_0$, we have $P(a|\bar{a}) = 1$ for some $a \in adom(I)$, and $P(b|\bar{a}) = 0$ for every $b \in [1, k] - \{a\}$. In other words, $\bar{a}$ determines p if one can specify a single value for p, given the values present in $\bar{a}$ and constraints Σ. We write $\Omega_0(I, p)$ for the set of all $\bar{a} \in \Omega(I, p)$ that determine p, and $\Omega_1(I, p)$ for the set of all $\bar{a} \in \Omega(I, p)$ that do not determine p. Let $n = |Pos(I)|$. Then:

Lemma 1. $\mathrm{RIC}_I(p|\Sigma) = |\Omega_1(I, p)|/2^{n-1}$.

Proof of Lemma 1. We show that the value of $\lim_{k\to\infty} \frac{1}{\log k} \sum_{a\in[1,k]} P_k(a|\bar{a}) \log \frac{1}{P_k(a|\bar{a})}$ is 0 if $\bar{a} \in \Omega_0(I, p)$ and it is 1 if $\bar{a} \in \Omega_1(I, p)$. Assume that $\bar{a}$ determines p. By definition there is $k_0 > 0$ such that for every $k > k_0$, it is the case that $P_k(a|\bar{a}) = 1$ for some $a \in adom(I)$, and $P_k(b|\bar{a}) = 0$ for all $b \in [1, k] - \{a\}$. Hence for all $k > k_0$ we have: $\sum_{a\in[1,k]} P_k(a|\bar{a}) \log \frac{1}{P_k(a|\bar{a})} = 0$. Note that $P_k(a|\bar{a}) \log \frac{1}{P_k(a|\bar{a})} = 0$ when $P_k(a|\bar{a}) = 0$, by definition. Then $\lim_{k\to\infty} \frac{1}{\log k} \sum_{a\in[1,k]} P_k(a|\bar{a}) \log \frac{1}{P_k(a|\bar{a})} = 0$.

Conversely, suppose $\bar{a}$ does not determine p. Then for every k_0 there is $k > k_0$ such that either $P_k(a|\bar{a}) = 0$ for all a, or $P_k(a_1|\bar{a}), P_k(a_2|\bar{a}) > 0$ for at least two different values a_1 and a_2. Since $I \models \Sigma$, we have $|\mathrm{SAT}^k_\Sigma(I_{(a,\bar{a})})| > 0$ for at least one $a \in adom(I)$, ruling out the first possibility. Since Σ contains only FDs, we conclude, by genericity, that $|\mathrm{SAT}^k_\Sigma(I_{(b,\bar{a})})| = |\mathrm{SAT}^k_\Sigma(I_{(b',\bar{a})})| > 0$ for all $b, b' \notin adom(I)$. Hence $P_k(b|\bar{a}) \leq 1/(k-n)$.

Next, expand $\bar{a}$ to $\bar{a}'$ by putting in a value for every position that is determined by $\bar{a}$ (which excludes p). Let r be the number of variables in $\bar{a}'$. Then for each $c \in [1, k]$ we have $|\mathrm{SAT}^k_\Sigma(I_{(c,\bar{a})})| \leq k^r$. Furthermore, for each $b \notin adom(I)$, any substitution σ that assigns to the r variables different values in $[1, k] - (adom(I) \cup \{b\})$ will be in $\mathrm{SAT}^k_\Sigma(I_{(b,\bar{a})})$; hence, we have $|\mathrm{SAT}^k_\Sigma(I_{(b,\bar{a})})| \geq (k-n-r)^r$. We thus have $P_k(b|\bar{a}) \geq \frac{(k-n-r)^r}{k\cdot k^r} = \frac{1}{k}\left(1 - \frac{n+r}{k}\right)^r$.

Let $\pi_i = P_k(a_i|\bar{a})$ for each $a_i \in adom(I)$. Then $\frac{1}{\log k} \sum_{a\in[1,k]} P_k(a|\bar{a}) \log \frac{1}{P_k(a|\bar{a})}$ is at least $\frac{1}{\log k}\Big(\sum\limits_{a_i\in adom(I)} \pi_i \log \frac{1}{\pi_i} + (k-n) \cdot \frac{\log(k-n)}{k} \cdot \left(1 - \frac{n+r}{k}\right)^r\Big)$. Since n and r are fixed, this implies that $\lim_{k\to\infty} \frac{1}{\log k} \sum_{a\in[1,k]} P_k(a|\bar{a}) \log \frac{1}{P_k(a|\bar{a})} \neq 0$, and by a result from [4], this limit always exists, and if it is not 0, then it must be equal to 1.

Now we can conclude the proof of Lemma 1:

$$
\begin{aligned}
& \mathrm{RIC}_I(p|\Sigma) \\
= \;& \lim_{k\to\infty} \frac{1}{\log k} \sum_{\bar{a}\in\Omega(I,p)} \frac{1}{2^{n-1}} \sum_{a\in[1,k]} P(a|\bar{a}) \log \frac{1}{P(a|\bar{a})} \\
= \;& \frac{1}{2^{n-1}} \sum_{\bar{a}\in\Omega_1(I,p)} \lim_{k\to\infty} \frac{1}{\log k} \sum_{a\in[1,k]} P(a|\bar{a}) \log \frac{1}{P(a|\bar{a})} \\
= \;& |\Omega_1(I,p)|/2^{n-1}.
\end{aligned}
$$

We now come back to the proof of Proposition 2. It consists of two parts. We prove that:

(a) For every $m > 2$ and $\varepsilon > 0$, there exists a schema (R, Σ), an instance $I \in inst(R, \Sigma)$, and a position $p \in Pos(I)$, such that $|attr(R)| = m$, (R, Σ) is indecomposable, and $\mathrm{RIC}_I(p|\Sigma) < 1/2 + \varepsilon$;

(b) For every indecomposable schema (R, Σ), every instance $I \in inst(R, \Sigma)$, and every position $p \in Pos(I)$, we have $\mathrm{RIC}_I(p|\Sigma) \geq 1/2$.

(a) Consider the relational schema $R(A_1, \ldots, A_m)$ with FDs $\Sigma = \{A_1A_2 \ldots A_{m-1} \to A_m, \;\; A_m \to A_1\}$ and the instance I of this schema shown in Figure 1. By Proposition 1, (R, Σ) is indecomposable. Let t_0 denote the first tuple in I, and let p denote the position of the gray cell.

Claim 1. *The information content of position p is*

$$\mathrm{RIC}_I(p|\Sigma) = \frac{1}{2} + \frac{1}{2}\left(\frac{3}{4}\right)^{k-1}.$$

A_1	A_2	A_3	$\ldots$	A_m
1	1	1	$\ldots$	1
1	2	1	$\ldots$	1
1	3	1	$\ldots$	1
$\vdots$	$\vdots$	$\vdots$		$\vdots$
1	k	1	$\ldots$	1

Figure 1: A database instance for the proofs of Propositions 2 and 4.

Proof of Claim 1. Let $\bar{a}$ be an arbitrary vector in $\Omega(I,p)$. Let $\bar{a}_{[t_0]}$ denote the subtuple in $\bar{a}$ corresponding to tuple $t_0 \in I$ and $\bar{a}_{[t_1]}$ denote the subtuple in $\bar{a}$ corresponding to an arbitrary tuple $t_1 \in I$. Each position in these subtuples contains either a variable (representing a missing value) or a constant, which equals the value that I has for that position.

Then $\bar{a}$ does not determine p iff

1. the subtuple $\bar{a}_{[t_0]}$ has a variable in the position corresponding to attribute A_m; or
2. the subtuple $\bar{a}_{[t_0]}$ has a constant in the position corresponding to attribute A_m, and for an arbitrary subtuple $\bar{a}_{[t_1]}$ in $\bar{a}$, $t_1 \neq t_0$:
   - 2.1. the subtuple $\bar{a}_{[t_1]}$ has a variable in the position corresponding to attribute A_m; or
   - 2.2. the subtuple $\bar{a}_{[t_0]}$ has a constant in the position corresponding to attribute A_m but a variable in the position corresponding to attribute A_1.

In Case 1, $\bar{a}$ can have either a variable or a constant in all other $n-2$ positions. Therefore, we can have 2^{n-2} such $\bar{a}$'s. In Case 2, $\bar{a}_{[t_0]}$ can have either a constant or a variable in the positions corresponding to $A_2,\ldots,A_{m-1}$. Furthermore, in Case 2.1, every such subtuple $\bar{a}_{[t_1]}$ can have either a constant or a variable in the positions corresponding to attributes $A_1,\ldots,A_{m-1}$, and in Case 2.2, it can have either a constant or a variable in the positions corresponding to $A_2,\ldots,A_{m-1}$. Therefore, the total number of $\bar{a}$'s satisfying conditions of Case 2 is $2^{m-2}(2^{m-1}+2^{m-2})^{k-1}$ since we have $k-1$ tuples other than t_0 in the instance.

Then $|\Omega_1(I,p)|$, or the total number of different $\bar{a}$'s in $\Omega(I,p)$ that do not determine p is

$$2^{n-2}+2^{m-2}(2^{m-1}+2^{m-2})^{k-1}.$$

By Lemma 1, $\mathrm{RIC}_I(p|\Sigma)$ can be obtained by dividing this number by $2^{n-1}=2^{mk-1}$:

$$\begin{aligned}\mathrm{RIC}_I(p|\Sigma) &= \frac{2^{mk-2}+2^{m-2}(2^{m-1}+2^{m-2})^{k-1}}{2^{mk-1}}\\ &= \frac{1}{2}+\frac{1}{2}\Big(\frac{3}{4}\Big)^{k-1},\end{aligned}$$

which proves the claim.

Thus for any $\varepsilon>0$, there is an instance of the form shown in Figure 1 and a position p in it such that the information content of p is less than $1/2+\varepsilon$: one needs to choose $k>1+\log_{4/3}(1/(2\varepsilon))$ and apply Claim 1.

(b) We need an easy observation (that will also be used in the proofs of the next section). For a key X, an attribute $A \notin X$ such that A does not occur in the right-hand side of any nontrivial FD, we have $\mathrm{RIC}_I(p|\Sigma)=1$ for any instance I of (R,Σ) and any position p corresponding to attribute A. Indeed, in this case $|\mathrm{SAT}^k_\Sigma(I_{(a,\bar{a})})| = |\mathrm{SAT}^k_\Sigma(I_{(b,\bar{a})})|$ for arbitrary $a,b \in [1,k]$ and hence $P(a|\bar{a})=1/k$, and thus $\mathrm{RIC}^k_I(p|\Sigma)=\log k$, and $\mathrm{RIC}_I(p|\Sigma)=\lim_{k\to\infty}\mathrm{RIC}^k_I(p|\Sigma)/\log k=1$.

Now let Σ be an arbitrary non-empty set of FDs over $R(A_1,\ldots,A_m)$ such that (R,Σ) is indecomposable, and $A_1,\ldots,A_{m-1}\rightarrow A_m \in \Sigma$ be the FD of the form described in Proposition 1: that is, $A_1\ldots A_{m-1}$ is an elementary candidate key. For any instance I of (R,Σ) and any position $p=(R,t,A_m)\in Pos(I)$ corresponding to attribute A_m, we have $\mathrm{RIC}_I(p|\Sigma)=1$ since p cannot have any redundancy due to a non-key FD.

Let $I \in inst(R,\Sigma)$, $p=(R,t_0,A_i)\in Pos(I)$, for some $i\in[1,m-1]$, and $\bar{a}\in\Omega(I,p)$. Let $\bar{a}_{[t_0]}$ denote the subtuple of $\bar{a}$ corresponding to t_0. It is easy to see that if $\bar{a}_{[t_0]}$ has a variable in the position corresponding to attribute A_m, then $\bar{a}$ does not determine p, no matter what the other positions in $\bar{a}$ contain. This is because there is no nontrivial FD $X\rightarrow A_i\in\Sigma^+$ such that $X\subseteq\{A_2,\ldots,A_{m-1}\}$. All other $n-2$ positions in $\bar{a}$ can therefore contain either a constant or a variable, so there are at least 2^{n-2} $\bar{a}$'s that do not determine p. Then using Lemma 1, we conclude that the information content of p is at least $\frac{2^{n-2}}{2^{n-1}}=1/2$. This proves Proposition 2.

Now we go back to prove Theorem A. The first part of the proof follows from Proposition 2: the information content of a position in an indecomposable instance can be arbitrarily close to $1/2$. Therefore, for every dependency-preserving normal form $\mathcal{NF}$ (which cannot further decompose an indecomposable instance), $\sup\mathcal{G}(\mathcal{NF})$ cannot exceed $1/2$. Therefore, $\mathrm{PRICE}(\mathcal{NF})\geq 1/2$.

To prove the second part, we notice that, by Proposition 1 and basic properties of 3NF, every indecomposable (R,Σ) is in 3NF. Furthermore, if (R,Σ) is decomposable, then the 3NF synthesis algorithm will decompose (R,Σ) into indecomposable schemas. Therefore, for every (R,Σ) and every $I\in inst(R,\Sigma)$, one can find a 3NF-decomposition in which the information content of every position is at least $1/2$ and sometimes exactly $1/2$. That is, $\sup\mathcal{G}(\text{3NF})=1/2$, and $\mathrm{PRICE}(\text{3NF})=1/2$. This concludes the proof. □

Notice that the proof of Theorem A implies that the guaranteed information content $1/2$ (which witnesses $\mathrm{PRICE}(\text{3NF})=1/2$) occurs in decompositions produced by the standard synthesis algorithm that generates a 3NF design from a minimal cover for Σ. Hence, our result justifies not only 3NF as the best dependency-preserving normal form, but also the standard algorithm for producing 3NF designs.

4. Comparing normal forms

In Section 3, we compared 3NF with other normal forms that guarantee dependency preservation, and proved that one can always guarantee a 3NF decomposition whose price would be less than or equal to the price of other normal form decompositions. As already mentioned, 3NF designs could be quite different: those that are produced by the standard synthesis algorithm (which we call 3NF$^+$ schemas) are the best, but others could be of lesser quality, as noticed in [22, 25]. So in this section we use the information-theoretic framework to compare different normal forms, in particular, 3NF, 3NF$^+$, and unnormalized schemas.

The measure for this comparison, described in the introduction, is the *gain of normalization* function defined as

$$\textsc{Gain}_{\mathcal{NF}_1/\mathcal{NF}_2}(m) \;=\; \frac{\inf \mathcal{POSS}_{\mathcal{NF}_1}(m)}{\inf \mathcal{POSS}_{\mathcal{NF}_2}(m)},$$

where $\mathcal{POSS}_{\mathcal{NF}}(m)$ is the set of all possible values $\textsc{Ric}_I(p|\Sigma)$ as (R,Σ) ranges over schemas with m attributes satisfying condition $\mathcal{NF}$. Recall that All refers to the class of *all* schemas.

We now prove that any 3NF schema, not necessarily indecomposable, is at least twice as good as some unnormalized schema. More precisely, the gain function for 3NF is constant 2 for all $m > 2$ (the case of $m \leq 2$ is special, as any nontrivial FD over two attributes is a key, and hence all schemas are in BCNF). We also show that 3NF$^+$ schemas could be significantly better than arbitrary 3NF schemas. That is,

Theorem B. *For every $m > 2$:*

- $\textsc{Gain}_{3\mathrm{NF}/\mathsf{All}}(m) = 2$;
- $\textsc{Gain}_{3\mathrm{NF}^+/3\mathrm{NF}}(m) = 2^{m-3}$;
- $\textsc{Gain}_{3\mathrm{NF}^+/\mathsf{All}}(m) = 2^{m-2}$.

In the proof of Theorem A we showed that $\inf \mathcal{POSS}_{3\mathrm{NF}^+}(m) = \inf \mathcal{POSS}_{\mathcal{ID}}(m) = 1/2$. Hence, the result will follow from these two propositions.

Proposition 3. $\inf \mathcal{POSS}_{\mathsf{All}}(m) = 2^{1-m}$ *for all* $m > 2$.

Proposition 4. $\inf \mathcal{POSS}_{3\mathrm{NF}}(m) = 2^{2-m}$ *for all* $m > 2$.

We now prove Proposition 3. We need to show that:

(a) For every $m > 2$ and $\varepsilon > 0$, there exists a schema (R,Σ) with $|attr(R)| = m$, an instance $I \in inst(R,\Sigma)$, and a position $p \in Pos(I)$ such that $\textsc{Ric}_I(p|\Sigma) < 2^{1-m} + \varepsilon$;

(b) For every (R,Σ) with $|attr(R)| = m$, every instance $I \in inst(R,\Sigma)$, and every position $p \in Pos(I)$, we have $\textsc{Ric}_I(p|\Sigma) \geq 2^{1-m}$.

A_1	A_2	A_3	$\ldots$	A_m
1	1	1	$\ldots$	1
1	2	1	$\ldots$	1
1	1	2	$\ldots$	1
$\vdots$	$\vdots$	$\vdots$		$\vdots$
1	1	1	$\ldots$	2
1	3	1	$\ldots$	1
1	1	3	$\ldots$	1
$\vdots$	$\vdots$	$\vdots$		$\vdots$
1	1	1	$\ldots$	3
$\vdots$	$\vdots$	$\vdots$		$\vdots$
1	k	1	$\ldots$	1
1	1	k	$\ldots$	1
$\vdots$	$\vdots$	$\vdots$		$\vdots$
1	1	1	$\ldots$	k

Figure 2: A database instance for the proof of Proposition 3.

(a) Consider $R(A_1,\ldots,A_m)$ and $\Sigma = \{A_2 \rightarrow A_1, A_3 \rightarrow A_1, \ldots, A_m \rightarrow A_1\}$. Consider the instance $I \in inst(R,\Sigma)$ shown in Figure 2. Let t_0 denote the first tuple in this table, and $p = (R, t_0, A_1)$ denote the position of the gray cell. Let t be the number of tuples minus 1, that is, $(m-1)(k-1)$.

Claim 2. *The information content of position p is*

$$\textsc{Ric}_I(p|\Sigma) = \frac{1}{2^{m+t-1}} \sum_{i=0}^{m-1} \binom{m-1}{i} (1+2^{-i})^t.$$

Proof of Claim 2. Let $\bar{a}$ be an arbitrary vector in $\Omega(I,p)$. Let $\bar{a}_{[t_0]}$ denote the subtuple of $\bar{a}$ corresponding to t_0, and suppose $\bar{a}_{[t_0]}$ has constants in positions corresponding to i attributes, and it has variables in the positions corresponding to the remaining $m-1-i$ attributes. Then $\bar{a}$ does not determine p iff for any arbitrary subtuple $\bar{a}_{[t_1]}$ of $\bar{a}$ corresponding to a tuple $t_1 \in I$, $t_1 \neq t_0$, we have:

1. the subtuple $\bar{a}_{[t_1]}$ has a variable in the position corresponding to A_1; or

2. the subtuple $\bar{a}_{[t_1]}$ has a constant in the position corresponding to A_1 but variables in the positions corresponding to the same i attributes for which $\bar{a}_{[t_0]}$ has constants.

In Case 1, $\bar{a}_{[t_1]}$ can have either a constant or a variable in every position corresponding to the other attributes $A_2,\ldots,A_m$, and therefore there are 2^{m-1} possibilities for such subtuples. In Case 2, $\bar{a}_{[t_1]}$ can have either a constant or a variable in every position corresponding to the other $m-1-i$ attributes, and therefore there are 2^{m-1-i} such subtuples. There are t tuples in I other than t_0, and i can range over $[0, m-1]$. Therefore, $|\Omega_1(I,p)|$ or the total number of different $\bar{a}$'s in $\Omega(I,p)$ that do not determine p

is

$$\sum_{i=0}^{m-1} \binom{m-1}{i} (2^{m-1} + 2^{m-1-i})^t.$$

The information content of p is then obtained by dividing this number by $2^{n-1} = 2^{m(t+1)-1}$:

$$\begin{aligned} &\mathrm{RIC}_I(p|\Sigma) \\ &= \frac{1}{2^{m(t+1)-1}} \sum_{i=0}^{m-1} \binom{m-1}{i} (2^{m-1} + 2^{m-1-i})^t \\ &= \frac{1}{2^{m+t-1}} \sum_{i=0}^{m-1} \binom{m-1}{i} (1 + 2^{-i})^t, \end{aligned}$$

which proves Claim 2.

The following shows that as long as $t > \log_{4/3}(1/\varepsilon)$ (that is, $k > (1 + \log_{4/3}(1/\varepsilon))/(m-1)$), for the instance in Figure 2 and position p of the gray cell, such that the information content of p is less than $2^{1-m} + \varepsilon$:

$$\begin{aligned} \mathrm{RIC}_I(p|\Sigma) &= \frac{1}{2^{m+t-1}} \sum_{i=0}^{m-1} \binom{m-1}{i} (1 + 2^{-i})^t \\ &= \frac{1}{2^{m+t-1}} \left(2^t + \sum_{i=1}^{m-1} \binom{m-1}{i} (1 + 2^{-i})^t \right) \\ &< 2^{1-m} + \frac{1}{2^{m+t-1}} \sum_{i=0}^{m-1} \binom{m-1}{i} (1 + 2^{-1})^t \\ &= 2^{1-m} + \left(\frac{3}{4}\right)^t \quad < \quad 2^{1-m} + \varepsilon. \end{aligned}$$

(b) Let Σ be an arbitrary set of FDs over a relational schema R, $I \in inst(R, \Sigma)$, $p = (R, t_0, A_1) \in Pos(I)$, and $\bar{a} \in \Omega(I, p)$. Let $\bar{a}_{[t_0]}$ denote the subtuple in $\bar{a}$ corresponding to t_0. It is easy to see that if $\bar{a}_{[t_0]}$ has variables in all positions corresponding to attributes $A_2, \ldots, A_m$, then $\bar{a}$ does not determine p, no matter what the other positions in $\bar{a}$ contain. All the other $n - m$ positions in $\bar{a}$ can therefore contain either a constant or a variable, so the number of $\bar{a}$'s that do not determine p is at least 2^{n-m}; that is, $|\Omega_1(I, p)| \geq 2^{n-m}$. Thus, using Lemma 1, the information content of p is at least $\frac{2^{n-m}}{2^{n-1}} = 2^{1-m}$. This proves Proposition 3.

Next, we prove Proposition 4. We need to show that:

(a) For an arbitrary $\varepsilon > 0$ and every $m > 2$, there exists a 3NF schema (R, Σ) with $|attr(R)| = m$, an instance $I \in inst(R, \Sigma)$, and a position $p \in Pos(I)$ such that $\mathrm{RIC}_I(p|\Sigma) < 2^{2-m} + \varepsilon$.

(b) For every (R, Σ) in 3NF with $|attr(R)| = m$, every instance $I \in inst(R, \Sigma)$, and every position $p \in Pos(I)$, we have $\mathrm{RIC}_I(p|\Sigma) \geq 2^{2-m}$.

(a) Consider $R(A_1, \ldots, A_m)$ and

$\Sigma = \{A_1A_2 \to A_3 \ldots A_m,\ A_3 \to A_1,\ \ldots,\ A_m \to A_1\}$.

Clearly (R, Σ) is in 3NF. Consider the instance $I \in inst(R, \Sigma)$ shown in Figure 1. Let t_0 denote the first tuple in this table, and $p = (R, t_0, A_1)$ denote the position of the gray cell.

Claim 3. *The information content of position p is*

$$\mathrm{RIC}_I(p|\Sigma) = \frac{1}{2^{m+k-3}} \sum_{i=0}^{m-2} \binom{m-2}{i} (1 + 2^{-i})^{k-1}.$$

Proof of Claim 3. Let $\bar{a}$ be an arbitrary vector in $\Omega(I, p)$. Let $\bar{a}_{[t_0]}$ denote the subtuple in $\bar{a}$ corresponding to t_0, and suppose that $\bar{a}_{[t_0]}$ has constants in the positions corresponding to i attributes among $A_3, \ldots A_m$, and it has variables in the positions corresponding to the remaining $m - 2 - i$ attributes. Then $\bar{a}$ does not determine p iff for any arbitrary subtuple $\bar{a}_{[t_1]}$ in $\bar{a}$ corresponding to a tuple $t_1 \in I$, $t_1 \neq t_0$, either

1. the subtuple $\bar{a}_{[t_1]}$ has a variable in the position corresponding to A_1; or
2. the subtuple $\bar{a}_{[t_1]}$ has a constant in the position corresponding to A_1 but variables in the positions corresponding to the same i attributes for which $\bar{a}_{[t_0]}$ has constants.

In Case 1, $\bar{a}_{[t_1]}$ can have either a constant or a variable in every position corresponding to attributes $A_2, \ldots, A_m$, and hence there could be 2^{m-1} such subtuples for every $t_1 \neq t_0$. In Case 2, $\bar{a}_{[t_1]}$ can have either a constant or a variable in every position corresponding to the $m - 1 - i$ attributes, and therefore there are 2^{m-1-i} possible such subtuples. There are $k - 1$ subtuples like $\bar{a}_{[t_1]}$, and i can range over $[0, m-2]$. So far we have not said anything about values corresponding to A_2 in t_0, but since A_1A_2 is a candidate key, in both cases, $\bar{a}_{[t_0]}$ can have either a constant or a variable in that position. Putting it all together, we see that $|\Omega_1(I, p)|$, the total number of different $\bar{a}$'s in $\Omega(I, p)$ that do not determine p is

$$2 \cdot \sum_{i=0}^{m-2} \binom{m-2}{i} (2^{m-1} + 2^{m-1-i})^{k-1}.$$

The information content of p can be obtained by dividing this number by $2^{n-1} = 2^{mk-1}$:

$$\begin{aligned} &\mathrm{RIC}_I(p|\Sigma) \\ &= \frac{1}{2^{mk-2}} \sum_{i=0}^{m-2} \binom{m-2}{i} (2^{m-1} + 2^{m-1-i})^{k-1} \\ &= \frac{1}{2^{m+k-3}} \sum_{i=0}^{m-2} \binom{m-2}{i} (1 + 2^{-i})^{k-1}. \end{aligned}$$

This proves Claim 3.

Now we need to show that for any $\varepsilon > 0$ there is an instance of the form shown in Figure 1 and a position p in it corresponding to the gray cell such that the information content of p is less than $2^{2-m} + \varepsilon$. Taking p to be the position used in Claim 3 we have

$$\begin{aligned}
\textsc{Ric}_I(p|\Sigma) &= \frac{1}{2^{m+k-3}}\sum_{i=0}^{m-2}\binom{m-2}{i}(1+2^{-i})^{k-1} \\
&= \frac{1}{2^{m+k-3}}\left(2^{k-1}+\sum_{i=1}^{m-2}\binom{m-2}{i}(1+2^{-i})^{k-1}\right) \\
&< 2^{2-m}+\frac{1}{2^{m+k-3}}\sum_{i=0}^{m-2}\binom{m-2}{i}(1+2^{-1})^{k-1} \\
&= 2^{2-m}+\left(\frac{3}{4}\right)^{k-1} \quad < \quad 2^{2-m}+\varepsilon,
\end{aligned}$$

as long as $k > 1+\log_{4/3}(1/\varepsilon)$.

(b) Let (R,Σ) be in 3NF, $I \in inst(R,\Sigma)$, $p = (R, t_0, A_1) \in Pos(I)$, and $\bar{a} \in \Omega(I,p)$. Let $\bar{a}_{[t_0]}$ denote the subtuple in $\bar{a}$ corresponding to t_0. We assume that A_1 is a prime attribute, but not a key itself, because otherwise $\textsc{Ric}_I(p|\Sigma) = 1$ since p would not have any redundancy due to a non-key FD.

It is easy to see that if $\bar{a}_{[t_0]}$ has variables in all positions corresponding to attributes $A_2,\ldots,A_m$, then $\bar{a}$ does not determine p, no matter what the other positions in $\bar{a}$ contain. All the other $n-m$ positions in $\bar{a}$ can therefore contain either a constant or a variable, so there are at least 2^{n-m} $\bar{a}$'s that do not determine p. Since A_1 is prime and not a key by itself, there is at least another attribute A_k such that A_1, A_k belong to a candidate key. If $\bar{a}_{[t_0]}$ has a constant in the position corresponding to A_k and variables in all positions corresponding to the other attributes, then $\bar{a}$ does not determine p since the FD $A_k \to A_1 \notin \Sigma^+$. Thus, there are at least another 2^{n-m} $\bar{a}$'s that do not determine p. Then using Lemma 1, the information content of p is at least $\frac{2^{n-m}+2^{n-m}}{2^{n-1}} = 2^{2-m}$, which completes the proof of Proposition 4 and thus of Theorem B. □

Combining [4] and Theorem B, we obtain the following comparisons of BCNF and 3NF:

Corollary 1. *For every $m > 2$:*

- $\textsc{Gain}_{\text{BCNF}/\text{3NF}^+}(m) = 2$;
- $\textsc{Gain}_{\text{BCNF}/\text{3NF}}(m) = 2^{m-2}$;
- $\textsc{Gain}_{\text{BCNF}/\mathsf{All}}(m) = 2^{m-1}$.

5. Conclusions

The main conclusion is that among normal forms that achieve dependency preservation, 3NF is the one that guarantees the least amount of redundancy: in fact, it is those 3NF designs that are produced by the standard synthesis algorithm that guarantee the least amount of redundancy. But even arbitrary 3NF schemas are still better than unnormalized ones, having at least twice the minimum information content.

There are several ways in which we would like to extend these results. First, one can think of a definition of the price of dependency preservation based not on the minimal guaranteed information content, but the average guaranteed information content. We would like to see how this different measure relates to 3NF.

Much of database theory and practice as of late has focused on transferring relational technology to XML [24]; in fact, since there are several approaches to XML design that appeared in the literature (e.g., [3, 11]), one of the motivations behind the information-theoretic approach was to provide a formal justification for normal forms for XML documents. We would like to use the information-theoretic approach to see what a natural analog of 3NF for XML is. Notice that the hierarchical structure of XML documents makes the interplay between redundancies and dependency preservation more intricate: for example, there are relational schemas that do not admit dependency-preserving BCNF decompositions, but can nonetheless be hierarchically represented in XML in a way that preserves all dependencies and has no redundancies [17].

We also would like to understand the relationship between the information-theoretic approach of [4] based on the concept of entropy, and the notion of information capacity of schemas [2, 14] based on the existence of mappings between schemas.

Acknowledgments We would like to thank Marcelo Arenas, Wenfei Fan, and Luc Segoufin for their comments.

6. References

[1] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, 1995.

[2] J. Albert, Y. Ioannidis, and R. Ramakrishnan. Equivalence of keyed relational schemas by conjunctive queries. *JCSS*, 58(3):512–534, 1999.

[3] M. Arenas and L. Libkin. A normal form for XML documents. *ACM TODS* 29 (2004), 195–232.

[4] M. Arenas and L. Libkin. An information-theoretic approach to normal forms for relational and XML data. *Journal of the ACM*, 52 (2005), 246–283. Extended abstract in *PODS'03*.

[5] C. Beeri, P. Bernstein, and N. Goodman. A sophisticate's introduction to database normalization theory. In *VLDB'78*, pages 113–124.

[6] P. Bernstein. Synthesizing third normal form relations from functional dependencies. *ACM TODS*, 1 (1976), 277–298.

[7] J. Biskup. Achievements of relational database schema design theory revisited. In *Semantics in Databases*, LNCS 1358, pages 29–54. Springer-Verlag, 1995.

[8] R. Cavallo and M. Pittarelli. The theory of probabilistic databases. In *VLDB'87*, pages 71–81.

[9] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.

[10] M. Dalkilic and E. Robertson. Information dependencies. In *PODS'00*, pages 245–253.

[11] D. W. Embley and W. Y. Mok. Developing XML documents with guaranteed "good" properties. In *ER'01*, pages 426–441.

[12] R. Fagin. Normal forms and relational database operators. In *SIGMOD'79*, pages 153–160.

[13] R. Fagin. A normal form for relational databases that is based on domains and keys. *ACM TODS* 6 (1981), 387–415.

[14] R. Hull. Relative information capacity of simple relational database schemata. *SIAM J. Comput.*, 15(3):856–886, 1986.

[15] P. Kanellakis. *Elements of Relational Database Theory*, In *Handbook of TCS, vol. B*, 1990, pages 1075–1144.

[16] M. Kifer, A. Bernstein, P. Lewis. *Database Systems: An Application-Oriented Approach*. Addison-Wesley, 2005.

[17] S. Kolahi. Dependency-preserving normalization of relational and XML data. In *DBPL'05*.

[18] T. T. Lee. An information-theoretic analysis of relational databases - Part I: Data dependencies and information metric. *IEEE Trans. on Software Engineering*, 13(10):1049–1061, 1987.

[19] M. Levene and G. Loizou. *A Guided Tour of Relational Databases and Beyond*. Springer, 1999.

[20] M. Levene and G. Loizou. Why is the snowflake schema a good data warehouse design? *Information Systems*, 28 (2003), 225–240.

[21] M. Levene and M. W. Vincent. Justification for inclusion dependency normal form. *IEEE TKDE*, 12(2):281–291, 2000.

[22] T.W. Ling, F. Tompa, T. Kameda. An improved third normal form for relational databases. *ACM TODS* 6 (1981), 329–346.

[23] Oracle's General Database Design FAQ. `http://www.orafaq.com/faqdesgn.htm`.

[24] V. Vianu. A Web Odyssey: from Codd to XML. In *PODS'01*, pages 1–15.

[25] C. Zaniolo. A new normal form for the design of relational database schemata. *ACM TODS* 7 (1982), 489–499.