The Collision Security of Tandem-DM in the Ideal Cipher Model

Jooyoung Lee[1], Martijn Stam[2,⋆], and John Steinberger[3,⋆⋆]

[1] Faculty of Mathematics and Statistics, Sejong University, Seoul, Korea
jlee05@sejong.ac.kr
[2] Department of Computer Science, University of Bristol, Bristol, United Kingdom
stam@cs.bris.ac.uk
[3] Institute of Theoretical Computer Science, Tsinghua University, Beijing, China
jpsteinb@gmail.com

Abstract. We prove that Tandem-DM, which is one of the two "classical" schemes for turning a blockcipher of $2n$-bit key into a double block length hash function, has birthday-type collision resistance in the ideal cipher model. A collision resistance analysis for Tandem-DM achieving a similar birthday-type bound was already proposed by Fleischmann, Gorski and Lucks at FSE 2009 [3]. As we detail, however, the latter analysis is wrong, thus leaving the collision resistance of Tandem-DM as an open problem until now. Our analysis exhibits a novel feature in that we introduce a trick not used before in ideal cipher proofs.

1 Introduction

The Tandem-DM compression function is a $3n$-bit to $2n$-bit compression function based on two applications of a blockcipher of $2n$-bit key and n-bit word length (Fig. 1). While Tandem-DM was proposed by Lai and Massey in 1992 [8] the first proof of collision security for Tandem-DM (in the ideal cipher model, as is usual for all such proofs) was only proposed in 2009 by Fleischmann, Gorski and Lucks [3]. Unfortunately, as we detail in Section 3, the "FGL proof" (as we shall refer to it) has a number of serious flaws which make it false and nonobvious to repair. The purpose of this paper is to offer a correct collision resistance analysis of Tandem-DM. We show that, as claimed in [3], Tandem-DM does indeed have birthday-type collision security (necessitating at least $2^{120.8}$ queries to break when the output length is $2n = 256$ bits). A nice feature of our work is that the analysis is relatively simple compared to typical results in this area. This simplicity is afforded by a new trick we introduce, apparently not used before in ideal cipher analyses.

⋆ Part of the work performed while at LACAL, École Polytechnique Fédérale de Lausanne, Switzerland.

⋆⋆ Supported in part by the National Basic Research Program of China Grant 2007CB807900, 2007CB807901, the National Natural Science Foundation of China Grant 61033001, 61061130540, 61073174 and by NSF grant CNS 0904380.

P. Rogaway (Ed.): CRYPTO 2011, LNCS 6841, pp. 561–577, 2011.

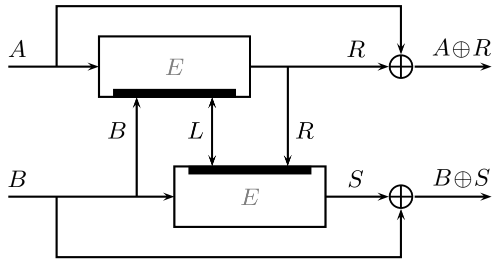

Fig. 1. The Tandem-DM compression function. All wires carry n-bit values. The top and bottom blockciphers are the same. Each has a $2n$-bit key and n-bit input/output. The wire marked L is an input to the compression function (along with A and B).

Related work on double block length constructions. Another classical scheme for turning a $2n$-bit key blockcipher into a $3n$-bit to $2n$-bit compression function is Abreast-DM, pictured in Fig. 2, which was proposed by Lai and Massey in the same paper as Tandem-DM [8]. The collision resistance of Abreast-DM was independently resolved by Fleischmann, Gorski and Lucks [4] and Lee and Kwon [9], who both showed birthday-type collision resistance for Abreast-DM. Before that, Hirose [5] had given a collision resistance analysis for a general class of compression functions that included Abreast-DM as a special case, but under the assumption that the top and bottom blockciphers of the diagram be distinct (this considerably simplifies the analysis). The work by Hirose was further generalized by Özen and Stam [14], who additionally discuss schemes that are only secure in the iteration.

Another $3n$-bit to $2n$-bit compression function making two calls to a blockcipher of $2n$-bit key was proposed by Hirose [6], who proved birthday-type collision resistance for his construction in the ideal cipher model. Hirose's construction (Fig. 3) is simpler than either Abreast-DM or Tandem-DM and in particular uses a single keying schedule for the top and bottom blockciphers. It is noteworthy that while Hirose introduced his construction over 10 years after Abreast-DM and Tandem-DM his collision resistance analysis pre-dates similar collision resistance analyses for Abreast-DM and Tandem-DM.

It is also possible to achieve birthday-type collision resistance for a $3n$-bit to $2n$-bit compression functions making only a single call to a $2n$-bit key blockcipher [22, 21, 13, 19, 20, 14, 10]; however these constructions have considerable overhead (typically comparable to a blockcipher call itself).

Comparison. Of the three well-known $3n$-bit to $2n$-bit compression functions making two calls to a $2n$-bit key blockcipher—those being Tandem-DM, Abreast-DM and Hirose's construction—the two constructions whose collision resistance has been successfully resolved (Hirose and Abreast-DM) share the feature that the inputs to the top and bottom blockcipher are bijectively related. For example, for Abreast-DM, if the top blockcipher call is $E_{B\|L}(A)$ then the bottom blockcipher call (for the same input $A\|B$) is $E_{L\|A}(\overline{B})$, where $\overline{B}$ denotes bit complementation of B; thus the inputs to the top and bottom blockciphers are

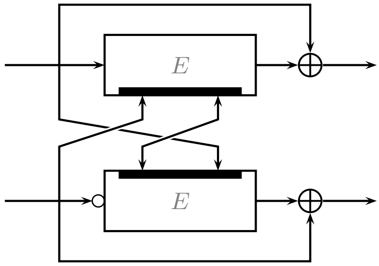

Fig. 2. The Abreast-DM compression function. The empty circle at bottom left denotes bit complementation.

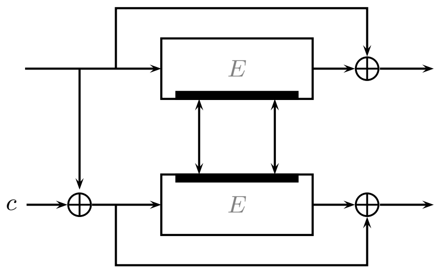

Fig. 3. Hirose's compression function. The bottom left-hand wire is not an input; it carries an arbitrary nonzero constant c.

related by the permutation $\pi : \{0,1\}^{3n} \to \{0,1\}^{3n}$, $\pi(X\|Y\|Z) = \overline{Y}\|Z\|X$. (Here the last $2n$ bits are the key.) In Hirose's construction, the inputs to the top and bottom blockciphers are related by the permutation $\pi' : \{0,1\}^{3n} \to \{0,1\}^{3n}$, $\pi'(X\|Y\|Z) = X \oplus c\|Y\|Z$.

By contrast, Tandem-DM exhibits a more subtle relationship between the inputs of the top and bottom blockciphers, as an *output* of the top blockcipher is used to key the bottom blockcipher. It is the presence of this "feedback" within the construction, it seems, that has complicated efforts to prove a collision resistance bound. On the other hand, Tandem-DM still has the agreeable feature that the top and bottom blockcipher calls uniquely determine each other in the following sense: given the key $B\|L$ and output R of the top cipher one can determine the key $L\|R$ and the input B of the bottom cipher, and vice-versa. This contrasts with constructions such as MDC-2 which use two calls to a blockcipher of n-bit key, and in which the top and bottom blockcipher calls do not uniquely determine each other. Typically, collision resistance analyses are much harder for the latter kind of compression functions. (MDC-2 can only be proved nontrivially collision resistant in the iteration, and the current best bound of $O(2^{\frac{3}{5}n})$ queries due to Steinberger [18] is undoubtedly suboptimal.)

We note that the permutations π and π' discussed above share the common feature of having *small cycle lengths*—all cycles of π have length 6 and all cycles of π' have length 2—which constitutes another strong similarity between Abreast-DM and Hirose's scheme. In fact, due to this reason, Hirose's collision resistance proof and the Abreast-DM collision resistance proof can be seen as special cases of the same framework, as noted in [4, 9]. Building on this observation, Fleischmann et al. [4] defined a general class of compression functions called 'Cyclic-DM' that are amenable to collision resistance analyses and that include Hirose's scheme and Abreast-DM as special cases. Similarly, one can define collision-resistant generalizations of Tandem-DM by isolating those properties of Tandem-DM that are used in our proof. While defining the most all-encompassing possible collision resistant generalization of Tandem-DM is not the goal of our work, we do briefly discuss these key properties and the corresponding collision-resistant generalizations of Tandem-DM in the paper's full version [11].

We mention that Fleischmann et al. [2] also provided a comprehensive generalization of their earlier works [3, 4]. In particular, a new and tighter collision resistance claim for Tandem-DM is made. Unfortunately, this second analysis has many similar flaws to the first, which are fatal to the integrity of the argument and to the final bound (in particular, the crucial "Argument B" of [2] is incorrect).

Full version contents. The proof of collision resistance that we provide in this paper is very slick, but slightly mysterious in its efficacy because it relies on a subtle trick that cuts out a large portion of the case analysis that "would have been there" in a more standard proof. As a kind of pedagogical bonus, the full version of this paper [11] contains a second collision resistance proof which does not utilize this trick. This proof is more straightforward but much longer, and the bound obtained is slightly worse (but still birthday).

Fleischmann et al. [3] also provide a preimage resistance proof for Tandem-DM which, unfortunately, suffers from similar flaws as their collision resistance proof. The full version of this paper [11] contains a description of these flaws, as well as a corrected preimage analysis. This preimage analysis shows $\Omega(2^n)$ queries are necessary to invert Tandem-DM on a random range point. We note, however, that dramatic progress has recently been made in this area, and it is now known that $\Omega(2^{2n})$ queries are necessary to invert Tandem-DM as well as Abreast-DM and Hirose's scheme [7, 12].

Further possible improvements. We note that our collision resistance has the form $\tilde{O}(q/(2^n - q))$ rather than $\tilde{O}(q^2/(2^n - q)^2)$. Both bounds reach constant values when $q = \Omega(2^n)$, however $q^2/(2^n - q)^2$ grows slower than $q/(2^n - q)$ since our bound is (only) "linear birthday" rather than true "quadratic birthday". We leave it as an open problem to prove "quadratic birthday"-type collision resistance for Tandem-DM (as exists for Abreast-DM and Hirose's scheme).

2 Definitions

A blockcipher is a function $E : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}^n$ such that $E(K, \cdot)$ is a permutation of $\{0,1\}^n$ for each $K \in \{0,1\}^m$. We call m the *key size* and

n the *word size* of the blockcipher. It is customary to write $E_K(X)$ instead of $E(K, X)$ for $K \in \{0,1\}^m$, $X \in \{0,1\}^n$. The function $E_K^{-1}(\cdot)$ denotes the inverse of $E_K(\cdot)$ (as $E_K(\cdot)$ is a permutation).

Given a blockcipher $E : \{0,1\}^{2n} \times \{0,1\}^n \to \{0,1\}^n$ we define the Tandem-DM compression function $TDM^E : \{0,1\}^{3n} \to \{0,1\}^{2n}$ by

$$TDM^E(A\|B\|L) = (A \oplus R)\|(B \oplus S)$$

where

$$R = E_{B\|L}(A) \qquad \text{and} \qquad S = E_{L\|R}(B).$$

In the collision resistance experiment, a computationally unbounded adversary $\mathcal{A}$ is given oracle access to a blockcipher E uniformly sampled among all blockciphers of key length $2n$ and word length n. We allow $\mathcal{A}$ to query both E and E^{-1}. After q queries to E, the *query history* of $\mathcal{A}$ is the set of triples $\mathcal{Q} = \{(X_i, K_i, Y_i)\}_{i=1}^q$ such that $E_{K_i}(X_i) = Y_i$ and $\mathcal{A}$'s i-th query is either $E_{K_i}(X_i)$ or $E_{K_i}^{-1}(Y_i)$ for $1 \le i \le q$. We let $\mathcal{Q}_i = \{(X_j, K_j, Y_j)\}_{j=1}^i$ be the first i elements of the query history; thus $\mathcal{Q} = \mathcal{Q}_q$. We say $\mathcal{A}$ *succeeds* or *finds a collision* after its first i queries if there exist distinct $3n$-bit values, $A\|B\|L$, $A'\|B'\|L'$ such that $TDM^E(A\|B\|L) = TDM^E(A'\|B'\|L')$ and such that $\mathcal{Q}_i$ contains both the queries necessary to compute $TDM^E(A\|B\|L)$ and $TDM^E(A'\|B'\|L')$. More formally—and see Fig. 4—we define this event by a predicate $\mathsf{Coll}(\mathcal{Q}_i)$, which is true if and only if there exist n-bit values A, B, L, R, S, A', B', L', R', S' such that

$$A\|B\|L \neq A'\|B'\|L', \qquad A \oplus R = A' \oplus R', \qquad B \oplus S = B' \oplus S' \tag{1}$$

and such that

$$(A, B\|L, R), (B, L\|R, S), (A', B'\|L', R'), (B', L'\|R', S') \in \mathcal{Q}_i. \tag{2}$$

We denote by

$$\mathbf{Adv}_{TDM}^{\text{coll}}(q)$$

the maximum chance of an adversary making q queries causing $\mathsf{Coll}(\mathcal{Q})$ to become true. The probability occurs over the uniform choice of E and over $\mathcal{A}$'s coin tosses, if any. Also note that n is a hidden parameter.

The "XOR-output" of a query (X_i, K_i, Y_i) is the quantity $X_i \oplus Y_i$. Another predicate which plays an important part in both our proof and the FGL proof is the "many queries with the same XOR-output" predicate $\mathsf{Xor}(\mathcal{Q})$, defined on a query history $\mathcal{Q} = \{(X_i, K_i, Y_i)\}_{i=1}^q$ by

$$\mathsf{Xor}(\mathcal{Q}) \iff \max_{Z \in \{0,1\}^n} |\{i : X_i \oplus Y_i = Z\}| > \alpha.$$

Here α is a free parameter of the analysis which appears in the final collision resistance bound. (In [3] this predicate is named $\text{LUCKY}(\mathcal{Q})$; in [18] a similar predicate is named $\mathsf{Win0}(\mathcal{Q})$.) Without going into details at this point, we mention that the FGL collision resistance proof—and ours, essentially, as well—upper bounds $\Pr[\mathsf{Coll}(\mathcal{Q})]$ by $\Pr[\mathsf{Xor}(\mathcal{Q})] + \Pr[\mathsf{Coll}(\mathcal{Q}) \wedge \neg\mathsf{Xor}(\mathcal{Q})]$. A larger α implies a lower value for $\Pr[\mathsf{Xor}(\mathcal{Q})]$ and a higher value for $\Pr[\mathsf{Coll}(\mathcal{Q}) \wedge \neg\mathsf{Xor}(\mathcal{Q})]$. The best value of α can be found numerically for a given value of n and q. Generally, readers may think of α as some small constant value (e.g. for $n = 128$ and $q = 2^{120.87}$, $\alpha = 16$).

So far, we have described "infrastructure" that is common to both proofs. We shall now introduce some material proper to our proof. Note a query history $\mathcal{Q} = \{(X_i, K_i, Y_i)\}_{i=1}^{q}$ does not record whether each triple (X_i, K_i, Y_i) was obtained by the adversary through a forward query $E_{K_i}(X_i)$ or a backward query $E_{K_i}^{-1}(Y_i)$. For this, we maintain two arrays $\mathsf{Fwd}[\cdot]$ and $\mathsf{Bwd}[\cdot]$ where $\mathsf{Fwd}[i] = 1$ if and only if the adversary's i-th query is a forward query and $\mathsf{Bwd}[i] = 1$ if and only if the adversary's i-th query is a backward query. We then define an additional predicate

$$\mathsf{FB}(\mathcal{Q}) \iff \max_{Z \in \{0,1\}^n} |\{i : (Y_i = Z \wedge \mathsf{Fwd}[i] = 1) \vee (X_i = Z \wedge \mathsf{Bwd}[i] = 1)\}| > \alpha. \tag{3}$$

('FB' stands for "Forward Backward".) Here α is the same free parameter as above. Note that $\neg\mathsf{FB}(\mathcal{Q})$ implies that

$$\max\nolimits_{Z \in \{0,1\}^n} |\{i : Y_i = Z \wedge \mathsf{Fwd}[i] = 1\}| \leq \alpha, \tag{4}$$

$$\max\nolimits_{Z \in \{0,1\}^n} |\{i : X_i = Z \wedge \mathsf{Bwd}[i] = 1\}| \leq \alpha. \tag{5}$$

It is really consequences (4) and (5) of $\neg\mathsf{FB}(\mathcal{Q})$ that interest us, though we define $\mathsf{FB}(\mathcal{Q})$ via (3) because this makes it slightly easier to bound $\Pr[\mathsf{FB}(\mathcal{Q})]$. We will use the bound

$$\begin{aligned} \Pr[\mathsf{Coll}(\mathcal{Q})] &\leq \Pr[\mathsf{Xor}(\mathcal{Q})] + \Pr[\mathsf{Coll}(\mathcal{Q}) \wedge \neg\mathsf{Xor}(\mathcal{Q})] \\ &\leq \Pr[\mathsf{Xor}(\mathcal{Q})] + \Pr[\mathsf{FB}(\mathcal{Q})] + \Pr[\mathsf{Coll}(\mathcal{Q}) \wedge \neg\mathsf{Xor}(\mathcal{Q}) \wedge \neg\mathsf{FB}(\mathcal{Q})]. \end{aligned} \tag{6}$$

One should thus think of $\mathsf{FB}(\mathcal{Q})$ and $\mathsf{Xor}(\mathcal{Q})$ as bad events whose nonoccurrence helps bound the probability of $\mathsf{Coll}(\mathcal{Q})$ occurring. We warn that (6) constitutes a slightly oversimplified encapsulation of our proof's high-level structure. We refer to Section 4 for more details.

3 The FGL Collision Resistance Proof

Since the interest of our paper would be substantially diminished (though not nullified, since our proof is much shorter) if the FGL collision resistance proof were correct, we detail here some of our objections to [3]. This material also serves as a good introduction to our own proof, and will give the reader more intuition about Tandem-DM.

Starting with a q-query collision-finding adversary $\mathcal{A}$, FGL first make the standard assumption that $\mathcal{A}$ never makes a query to which it already knows the answer (this could occur two ways: $\mathcal{A}$ could make the exact same query twice, or $\mathcal{A}$ could query (say) $E_K^{-1}(Y)$ after having received Y as an answer beforehand to a query $E_K(X)$). This ensures each answer $\mathcal{A}$ receives comes uniformly at

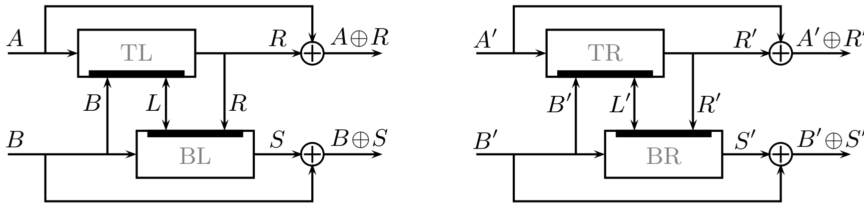

Fig. 4. The collision diagram for Tandem-DM. The adversary must find blockcipher queries to fit both sides of the diagram such that $A \oplus R = A' \oplus R'$, $B \oplus S = B' \oplus S'$ and $A\|B\|L \neq A'\|B'\|L'$. More precisely, the adversary must find four queries of the form $E_{B\|L}(A) = R$, $E_{L\|R}(B) = S$, $E_{B'\|L'}(A') = R'$, $E_{L'\|R'}(B') = S'$ such that the above conditions hold. Each query could either be learned through a forward query (to E) or through a backward query (to E^{-1}). The four queries in the diagram are labeled 'TL', 'BL', 'TR', 'BR' for 'Top Left', 'Bottom Left', etc.

random from a set of size at least $2^n - q$ (since $E_K(\cdot)$ is a random permutation for each K). Moreover, after $\mathcal{A}$ makes i queries its query history will contain exactly i distinct elements.

Say $\mathcal{A}$ *succeeds* at the i-th query if $\mathsf{Coll}(\mathcal{Q}_i)$ holds but neither $\mathsf{Coll}(\mathcal{Q}_{i-1})$ nor $\mathsf{Xor}(\mathcal{Q}_{i-1})$ holds. By upper bounding the probability that $\mathcal{A}$ ever succeeds we upper bound $\Pr[\mathsf{Coll}(\mathcal{Q}) \wedge \neg\mathsf{Xor}(\mathcal{Q})]$. (Upper bounding $\Pr[\mathsf{Xor}(\mathcal{Q})]$ is an easy probability exercise that we overlook for the purposes of this proof sketch.) A good analogy is to view $\mathcal{A}$ as trying to complete a puzzle where each element of its query history is a puzzle piece it can use to complete the collision diagram of Fig. 4. We use the expressions "$\mathcal{A}$ succeeds", "$\mathcal{A}$ finds a [puzzle] solution" or "$\mathcal{A}$ completes a collision" interchangeably (and we will rarely remind that the condition $\neg\mathsf{Xor}(\mathcal{Q}_{i-1})$ must hold for $\mathcal{A}$ to succeed). We refer to the four queries (in any hypothetical puzzle solution (a.k.a. collision)) as 'TL', 'BL', 'TR' and 'BR'; see Fig. 4.

Note the constraint $A\|B\|L \neq A'\|B'\|L'$ does not imply that the queries TL, BL, TR, BR are all distinct. For example, one could have TL = BR (in which case $(A, B\|L, R) = (B', L'\|R', S')$, so $A = B'$, $B = L'$, $L = R'$ and $R = S'$) or TL = BL (in which case we have the dramatic consequence that $A = B = L = R = S$, as is easy to check). This gives rise to several combinatorially distinct cases to consider; $\mathcal{A}$'s chance of obtaining a solution of each kind is upper bounded separately, and these probabilities are added together to form a final upper bound on $\mathcal{A}$'s chance of success. (Oddly, FGL include the cases TL = TR and BL = BR in their analysis, while these are impossible since they imply $A\|B\|L = A'\|B'\|L'$. This oversight, however, does not imply an incorrect proof in itself.)

We shall restrict our critique to FGL's analysis of the "generic" case when the queries TL, BL, TR, BR are all distinct. We note that, in these types of analyses, the generic case is usually the hardest to handle as $\mathcal{A}$'s job typically grows harder when additional constraints are placed on its solution. (The possibility of reusing the same query in two different positions of the collision diagram does, however, occasionally prove useful to $\mathcal{A}$, depending on the construction, so all cases must always be considered.) We call a puzzle solution in which TL, BL, TR, BR are distinct a "generic solution."

If $\mathcal{A}$ succeeds in finding a generic solution there is a smallest i such that a generic solution can be assembled from the queries in $\mathcal{Q}_i$. The i-th query is then called the "last query" of $\mathcal{A}$'s solution. To upper bound $\mathcal{A}$'s chance of obtaining a generic solution, FGL consider two cases. The first case is the event that $\mathcal{A}$'s last query can be used in position TL of the puzzle solution and the second case is the event that $\mathcal{A}$'s last query can be used in position BL (one of these two cases must occur). We shall focus on the first of these two cases, which is also the first analyzed in the order of the FGL proof. We call it the "TL generic" case.

One would typically consider two subcases for the TL generic case (or any other) depending on whether $\mathcal{A}$'s last query is a forward query to E or an inverse query to E^{-1}, but FGL lump their analysis into a single argument claiming that the two types of queries can be handled the same (in fact, they make this claim for every case in their proof, and never distinguish between forward and backward queries to E). For clarity, however, we shall restrict ourselves to considering the case of a forward query to E, and discuss how their argument specializes to that case. We also choose to specifically consider the forward query case because this is where FGL's analysis seems to be the most problematic.

The task at hand is thus to upper bound $\mathcal{A}$'s chance of completing a generic solution by making a forward query to E that can be used as query TL of such a solution. The usual approach for this, and the one used by FGL, is to consider any given forward query $E_{K_i}(X_i)$ made by $\mathcal{A}$ and to upper bound the probability that the answer Y_i to this query is such that the query history element (X_i, K_i, Y_i) can be used in the desired manner; one then multiplies this probability by q since $\mathcal{A}$ can make q queries total. With foresight on how we wish to use the query $E_{K_i}(X_i)$ it is convenient to rename K_i as $B\|L$ and X_i as A; thus the query is $E_{B\|L}(A)$. To proceed, one would typically upper bound the number of values $R \in \{0,1\}^n$ such that, if we had $E_{B\|L}(A) = R$, the query $(A, B\|L, R)$ could be used in position TL of a generic solution together with previous elements of the query history, and divide this number by $2^n - q$, since the answer to the query $E_{B\|L}(A)$ will come uniformly at random from a set of size at least $2^n - q$. In turn, the standard, formal way of bounding the number of such R's would be to upper bound the possible number of query triples (BL, BR, TR) already in the query history that could potentially be used with the query $E_{B\|L}(A)$ to form a generic solution, as the number of such triples is an upper bound for the number of R's. Note such a triple must have the form BL $= (B, L\|R, S)$, BR $= (B', L'\|R', S')$, TR $= (A', B'\|L', R')$ where $B \oplus S = B' \oplus S'$ (and note that A, B and L are fixed here by the last query).

FGL, however, do not adopt[1] this approach for bounding the number of good R's. Rather, they make the following argument: take the value of R, whatever it is, that is returned by the query $E_{B\|L}(A)$; because $\neg\mathsf{Xor}(\mathcal{Q}_{i-1})$ there will be

[1] Neither do we, in fact. Using a careful trick, we manage to upper bound the number of good R's by only considering the possibilities for the query BL rather than by considering the possible triples (BL, TR, BR). In the full version [11], however, we give for comparison the "brute force" proof which uses the method of upper bounding the number of triples (BL, TR, BR).

at most α queries TR $= (A', B'\|L', R')$ in the query history such that $A \oplus R = A' \oplus R'$; as the TR query uniquely determines the BR query, there are at most α possibilities for the BR query; now "give the query BL $= (B, L\|R, S)$ for free to the adversary"; then since there are at most α possibilities for the query BR $= (B', L'\|R', S')$ there is chance at most $\alpha/(2^n - q)$ that $B \oplus S = B' \oplus S'$ for one of the queries BR, so total chance at most $q\alpha/(2^n - q)$ that the adversary ever obtains a TL-generic solution with a forward query, there being at most q queries total.

The fallacy in the above argument can be succinctly summarized by pointing out that *the query* BL $= (B, L\|R, S)$ *may already be in the query history, in which case there is no randomness left in the value* $B \oplus S$. However, let us review in detail the argument in two different cases: when the query BL $= (B, L\|R, S)$ is already in the query history prior to the last query, and when it isn't. (Note that query BL only depends on R (besides B and L which are fixed by the last query), and not on which queries are "chosen" for TR and BR.) In the latter case, when BL $= (B, L\|R, S)$ is not yet in the query history at the i-th query, then A's last query can in any case not succeed in completing a generic TL collision since the query BL is missing; thus there is no need to bound anything (and no need even to "give the query BL for free"). In the case when query BL is already in the query history, on the other hand, all randomness is lost once R is revealed. FGL successfully argue that, for a given value of R, there will be at most α possibilities for the pair (TR , BR), but this does not in any way imply the *non-existence* of such queries TR, BR.

Note also that nothing in the FGL argument precludes the possibility that, when the adversary makes its i-th query $E_{B\|L}(A)$, there is not some very large number of distinct values of R—say $2^{0.5n}$—for which there exists a triplet of queries (BL, TR, BR) of the form BL $= (B, L\|R, S)$, BR $= (B', L'\|R', S')$, TR $= (A', B'\|L', R')$ where $B \oplus S = B' \oplus S'$, and such that R does not yet appear as the third coordinate of any query in the query history with key $B\|L$. Certainly, there being such a large number of values of R does not contradict $\neg\mathsf{Xor}(\mathcal{Q}_{i-1})$. Also certainly, the i-th query would have chance $2^{0.5n}/(2^n - q)$ of making the adversary succeed if such a large number of values of R existed, and not chance $\alpha/(2^n - q)$. In other words, one can infer something is wrong with the FGL argument because it simply does not address the main difficulty of the case at hand—that being the potential existence of a large number of triples (BL, BR, TR) that may fit with the query $E_{B\|L}(A)$.

Other issues are raised by FGL's casual comment that the query BL $= (B, L\|R, S)$ is simply "given for free" to the adversary. Indeed, if this query is not yet present, is it added to the query history before or after the i-th query itself? Is this query only made after the value of R is revealed, or is it somehow inserted into the query history before the value of R is revealed? The former might be all right; the latter not, since it would (drastically) alter R's distribution conditioned on the query history, i.e. R would no longer come uniformly at random from a set of size $\geq 2^n - q$. Most importantly, since this free query becomes part of the query history, one should account for the possibility that *this query* (not the i-th query) causes the adversary to succeed (and not necessarily by being used in position BL of a generic solution). Indeed, we are forced to give such credit to the adversary, since we have required the adversary never to make a query to which it already knows the answer, and since the adversary may have wished to subsequently make this query itself; this means the case analysis should be applied recursively to the free query, but if the case analysis requires other queries to be "given for free", then we bite our tail and end up giving an astronomical number of free queries to the adversary (e.g., nearly all possible queries).

While we singled out the TL generic case for examination, the same kinds of problems recur throughout the FGL case analysis, essentially invalidating the entire proof. Moreover, since the FGL proof sidesteps the most crucial challenges posed by an analysis of Tandem-DM (see the paragraph before last), it leaves little for any subsequent analysis to build on. We note that the FGL preimage resistance proof suffers from very similar flaws as the collision resistance proof, as discussed in the full version of this paper [11].

4 Main Result: Collision Resistance of Tandem-DM

It will be easier to explain the form of the probability bound in our main theorem if we explain a few high-level ideas from the proof beforehand. The proof starts by considering an arbitrary q-query collision-finding adversary $\mathcal{A}$ for Tandem-DM. We then construct an adversary $\mathcal{A}'$ as follows: $\mathcal{A}'$ simulates $\mathcal{A}$, but after each forward query $E_{V\|W}(U)$ made by $\mathcal{A}$, $\mathcal{A}'$ makes the backward query $E^{-1}_{U\|V}(W)$ *if* it does not already know[2] the answer to this query, and after each backward query $E^{-1}_{U\|V}(W)$ made by $\mathcal{A}$, $\mathcal{A}'$ makes the forward query $E_{V\|W}(U)$ *if* it does not already know[3] the answer to this query. (To better understand the relation of these instructions to Tandem-DM, view U, V, W as B, L, R.) Moreover if $\mathcal{A}$ ever makes a query to which $\mathcal{A}'$ already knows the answer from its query history, $\mathcal{A}'$ ignores this query. Thus $\mathcal{A}'$ never makes a query to which it knows the answer.

Let $\mathcal{Q}'$ be the query history of $\mathcal{A}'$ and $\mathcal{Q}$ be the query history of $\mathcal{A}$. Then $\mathcal{Q} \subseteq \mathcal{Q}'$ and $|\mathcal{Q}'| \leq 2q$. Since $\mathcal{Q} \subseteq \mathcal{Q}'$ we have

$$\begin{aligned} &\Pr[\mathsf{Coll}(\mathcal{Q})] \\ &\leq \Pr[\mathsf{Coll}(\mathcal{Q}')] \\ &\leq \Pr[\mathsf{Xor}(\mathcal{Q}')] + \Pr[\mathsf{FB}(\mathcal{Q}')] + \Pr[\mathsf{Coll}(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')]. \qquad (7) \end{aligned}$$

Our proof uses the inequality above to bound $\Pr[\mathsf{Coll}(\mathcal{Q})]$. We point out that if we construct an adversary $\mathcal{A}''$ from $\mathcal{A}'$ the same way $\mathcal{A}'$ is constructed from $\mathcal{A}$, then $\mathcal{A}''$ and $\mathcal{A}'$ will have the same query history, as is not difficult to see. In other words, *every* forward query $E_{V\|W}(U)$ made by $\mathcal{A}'$ (including its "own"

[2] More formally, if its query history does not contain any triple of the form $(\cdot, U\|V, W)$.
[3] More formally, if its query history does not contain any triple of the form $(U, V\|W, \cdot)$.

queries) is followed by the query $E^{-1}_{U\|V}(W)$ unless $\mathcal{A}'$ already knows this query, and likewise *every* backward query $E^{-1}_{U\|V}(W)$ made by $\mathcal{A}'$ is followed by the forward query $E_{V\|W}(U)$ unless $\mathcal{A}'$ already knows the answer to this query. The use of the augmented adversary $\mathcal{A}'$ may seem superficially similar to Fleischmann et al.'s idea of "giving away a query for free". However, it will become clear from our case analysis that we exploit the added structure of $\mathcal{Q}'$ entirely differently from the way Fleischmann et al. exploit their free queries. We also point out that the added structure of $\mathcal{Q}'$ enables the main interesting trick of our analysis, to be found in case 'TL Forward' of Proposition 3 below.

We can now more easily discuss our main result:

Theorem 1. *Let $N = 2^n$, $q < N/2$, $N' = N - 2q$ and let α be an integer, $1 \le \alpha \le 2q$. Then*

$$\mathbf{Adv}^{\mathrm{coll}}_{TDM}(q) \le 2N\left(\frac{2eq}{\alpha N'}\right)^{\alpha} + \frac{4q\alpha}{N'} + \frac{4q}{N'}.$$

The term $2N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}$ in Theorem 1 is an upper bound for $\Pr[\mathsf{Xor}(\mathcal{Q}')]+\Pr[\mathsf{FB}(\mathcal{Q}')]$. In fact $\Pr[\mathsf{Xor}(\mathcal{Q}')] \le N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}$ and $\Pr[\mathsf{FB}(\mathcal{Q}')] \le N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}$. The two remaining terms $4q\alpha/N' + 4q/N'$ are an upper bound for $\Pr[\mathsf{Coll}(\mathcal{Q}')\wedge\neg\mathsf{Xor}(\mathcal{Q}')\wedge\neg\mathsf{FB}(\mathcal{Q}')]$. To bound $\mathbf{Adv}^{\mathrm{coll}}_{TDM}(q)$ for a given value of n and q one should optimize α numerically. For example, for $n = 128$, Theorem 1 yields that $\mathbf{Adv}^{\mathrm{coll}}_{TDM}(2^{120.87}) < \frac{1}{2}$ using $\alpha = 16$. Asymptotically, Theorem 1 yields the following result:

Corollary 1. $\lim_{n\to\infty} \mathbf{Adv}^{\mathrm{coll}}_{TDM}(N/n) = 0$.

Proof. Let $q = N/n$ and $\alpha = n/\log n$, where the logarithm takes base 2. Since $N' > N/2$ for $n > 4$, we have

$$\begin{aligned}\mathbf{Adv}^{\mathrm{coll}}_{TDM}(q) &\le 2N\left(\frac{2eq}{\alpha N'}\right)^{\alpha} + \frac{4q\alpha}{N'} + \frac{4q}{N'} \le 2N\left(\frac{4eq}{\alpha N}\right)^{\alpha} + \frac{8q\alpha}{N} + \frac{8q}{N}\\ &\le 2N\left(\frac{4e\log n}{n^2}\right)^{\frac{n}{\log n}} + \frac{8}{\log n} + \frac{8}{n} = 2\left(\frac{4e\log n}{n}\right)^{\frac{n}{\log n}} + \frac{8}{\log n} + \frac{8}{n}.\end{aligned}$$

The last expression obviously goes to zero as $n \to \infty$. □

In particular, $\lim_{n\to\infty} \mathbf{Adv}^{\mathrm{coll}}_{TDM}\left(2^{(1-\varepsilon)n}\right) = 0$ for any fixed $\varepsilon > 0$.

The proof of Theorem 1 uses refinements $\mathsf{Coll}_1(\mathcal{Q})$, $\mathsf{Coll}_2(\mathcal{Q})$, $\mathsf{Coll}_3(\mathcal{Q})$ of the collision predicate $\mathsf{Coll}(\mathcal{Q})$, defined as follows:

$\mathsf{Coll}_1(\mathcal{Q})$ occurs if $\mathcal{Q}$ contains a collision with TL, BL, TR, BR distinct.
$\mathsf{Coll}_2(\mathcal{Q})$ occurs if $\mathcal{Q}$ contains a collision with either TL = BL or TR = BR.
$\mathsf{Coll}_3(\mathcal{Q})$ occurs if $\mathcal{Q}$ contains a collision with either TL = BR or BL = TR.

For example, $\mathsf{Coll}_2(\mathcal{Q})$ occurs if there exist values $A, B, L, R, S, A', B', L', R', S'$ such that (1)–(2) hold and such that $(A, B\|L, R) = (B, L\|R, S)$. Since BL $\neq$ BR and TL $\neq$ TR in any collision, we have the following proposition.

Proposition 1. $\mathsf{Coll}(\mathcal{Q}) \implies \mathsf{Coll}_1(\mathcal{Q}) \vee \mathsf{Coll}_2(\mathcal{Q}) \vee \mathsf{Coll}_3(\mathcal{Q})$ *for any query history* $\mathcal{Q}$.

In view of proving Theorem 1, let $\mathcal{A}$ be an arbitrary q-query adversary for Tandem-DM, and let $\mathcal{A}'$ be obtained from $\mathcal{A}$ as outlined above; let $\mathcal{Q}$ be the query history of $\mathcal{A}$ and $\mathcal{Q}'$ be the query history of $\mathcal{A}'$. Then by (7) it suffices to show that

$$\Pr[\mathsf{Xor}(\mathcal{Q}')] \leq N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}$$

$$\Pr[\mathsf{FB}(\mathcal{Q}')] \leq N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}$$

$$\Pr[\mathsf{Coll}(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq \frac{4q\alpha}{N'} + \frac{4q}{N'}$$

since the sum of the above probabilities is an upper bound for $\Pr[\mathsf{Coll}(\mathcal{Q})]$. Moreover, by Proposition 1, $\Pr[\mathsf{Coll}(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')]$ can be upper bounded by finding upper bounds for $\Pr[\mathsf{Coll}_i(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')]$ for $i = 1, 2, 3$ and taking the sum of these. We now upper bound these various probabilities in a series of propositions. For these propositions q, N and α are as in Theorem 1, and $\mathcal{Q}'$ is the query history of any adversary $\mathcal{A}'$ as just specified. We emphasize that $|\mathcal{Q}'| \leq 2q$ and that probabilities are taken over the random cipher E and over the coins of $\mathcal{A}'$, if any (it inherits these from $\mathcal{A}$).

Proposition 2. $\Pr[\mathsf{Xor}(\mathcal{Q}')] \leq N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}$ *and* $\Pr[\mathsf{FB}(\mathcal{Q}')] \leq N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}$.

Proof. Without loss of generality, we can assume that A' always makes exactly $2q$ queries. Let $\mathcal{Q}' = \{(X'_i, K'_i, Y'_i)\}_{i=1}^{2q}$ denote the query history of $\mathcal{A}'$. Since

$$\Pr[|\{i : X'_i \oplus Y'_i = Z\}| > \alpha] \leq \binom{2q}{\alpha}\left(\frac{1}{N'}\right)^{\alpha},$$

for each $Z \in \{0,1\}^n$, we have

$$\Pr[\mathsf{Xor}(\mathcal{Q}')] \leq N\binom{2q}{\alpha}\left(\frac{1}{N'}\right)^{\alpha} \leq N\left(\frac{2q \cdot e}{\alpha}\right)^{\alpha}\left(\frac{1}{N'}\right)^{\alpha} \leq N\left(\frac{2eq}{\alpha N'}\right)^{\alpha}.$$

$\Pr[\mathsf{FB}(\mathcal{Q}')]$ can be bounded similarly. □

Proposition 3. $\Pr[\mathsf{Coll}_1(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq 4q\alpha/N'$.

Proof. Let

$$\mathsf{Success}_1(\mathcal{Q}'_i) = \mathsf{Coll}_1(\mathcal{Q}'_i) \wedge \neg\mathsf{Coll}_1(\mathcal{Q}'_{i-1}) \wedge \neg\mathsf{Xor}(\mathcal{Q}'_{i-1}) \wedge \neg\mathsf{FB}(\mathcal{Q}'_{i-1})$$

for $i = 1 \ldots 2q$. Then $\Pr[\mathsf{Coll}_1(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq \sum_{i=1}^{2q} \Pr[\mathsf{Success}_1(\mathcal{Q}'_i)]$ and $\Pr[\mathsf{Success}_1(\mathcal{Q}'_i)] \leq \Pr[\mathsf{Coll}_1(\mathcal{Q}'_i) | \neg\mathsf{Coll}_1(\mathcal{Q}'_{i-1}) \wedge \neg\mathsf{Xor}(\mathcal{Q}'_{i-1}) \wedge \neg\mathsf{FB}(\mathcal{Q}'_{i-1})]$.

Fix a value of i, $1 \leq i \leq 2q$. We call the i-th query made by $\mathcal{A}'$ the *last query*. If $\mathsf{Success}_1(\mathcal{Q}'_i)$ occurs then either the adversary (i.e. $\mathcal{A}'$) can use its last query as query TL or as query BL of a collision in which TL, BL, TR and BR are distinct, by symmetry. Moreover the last query could either be a forward query or a backward query. This gives rise to four possible cases, and we bound $\Pr[\mathsf{Success}_1(\mathcal{Q}'_i)]$ for each separately. (We note the very first case, 'TL forward', is the case we discussed in Section 3.) For each case, we call the last query *successful* if this query completes a collision with TL, BL, TR, BR distinct and where the last query is used in the position stipulated by that case (e.g., for the case 'TL forward', the last query must be used in position TL).

TL forward: Let the last query be $E_{B\|L}(A)$. Call a value R *good* if there exists a query of the form $(B, L\|R, \cdot)$ in $\mathcal{Q}'$ that was obtained by $\mathcal{A}'$ as a backward query. We note that because of (5), $\neg\mathsf{FB}(\mathcal{Q}'_{i-1})$ implies there are at most α good R's.

We claim that for the last query to be successful the value R returned as an answer to the query must be good. Indeed, let R be the value returned; then a prerequisite for the query to be successful is that there be a query of the form $(B, L\|R, \cdot)$ in $\mathcal{Q}'_{i-1}$. We claim that this query must have been obtained as a backward query. Indeed, assume that the query $(B, L\|R, \cdot)$ was obtained as a forward query $E_{L\|R}(B)$ by $\mathcal{A}'$. Then, by construction, $\mathcal{A}'$ would have immediately followed this query by the query $E^{-1}_{B\|L}(R)$ unless $\mathcal{A}'$ already knew the answer to $E^{-1}_{B\|L}(R)$. Either way, $\mathcal{A}'$ would have the query $(A, B\|L, R)$ in its query history *prior* to the i-th (forward) query $E_{B\|L}(A)$, a contradiction since $\mathcal{A}'$ never makes a query to which it knows the answer. Thus the value R returned as an answer to the query $E_{B\|L}(A)$ must be good for the query to be successful.

Since there are at most α good values of R and since $\mathcal{A}'$ makes at most $2q$ queries, the probability that the last query is successful is therefore at most $\alpha/(2^n - 2q) = \alpha/N'$.

TL backward: Let the last query be $E^{-1}_{B\|L}(R)$. For the last query to be successful, there must be a (necessarily unique) query BL $= (B, L\|R, S) \in \mathcal{Q}'_{i-1}$, for some value $S \in \{0,1\}^n$. From the condition $B \oplus S = B' \oplus S'$ and from $\neg\mathsf{Xor}(\mathcal{Q}'_{i-1})$ there are at most α possibilities for the query BR. As each query BR uniquely determines the query TR, there are at most α possibilities for the query TR as well, and thus at most α possibilities for the value $A' \oplus R'$. Thus the value A returned by the last query has chance at most α/N' that $A \oplus R$ will be equal to $A' \oplus R'$ for one of these values $A' \oplus R'$, and so the last query has chance at most α/N' of being successful.

BL forward: A 180° rotation of the collision diagram shows this case is symmetric to the case TL backward. The chance of success in this case is therefore at most α/N'.

BL backward: A 180° rotation of the collision diagram shows this case is symmetric to the case TL forward. The chance of success in this case is therefore at most α/N'.

The chance a forward last query is successful is therefore at most $2\alpha/N'$ (adding the TL and BL forward cases) and likewise the chance that a backward last query is successful is at most $2\alpha/N'$. Thus $\Pr[\mathsf{Success}_1(\mathcal{Q}'_i)] \leq 2\alpha/N'$ for all i and $\sum_{i=1}^{2q} \Pr[\mathsf{Success}_1(\mathcal{Q}'_i)] \leq 4q\alpha/N'$. □

Proposition 4. $\Pr[\mathsf{Coll}_2(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq 2q/N'$.

Proof. Note that when TL = BL, $B\|L = L\|R$, so $B = L = R$; moreover $R = S$ and $A = B$, so $A = B = L = R = S$. For the adversary to obtain a collision with TL = BL, therefore, it must obtain a query of the form $(U, U\|U, U)$. The same argument applies to the case TR = BR. The chance of a query $E_{U\|U}(U)$ or of a query $E^{-1}_{U\|U}(U)$ being answered by U is at most[4] $1/N'$. Thus, since $2q$ queries are made total, $\Pr[\mathsf{Coll}_2(\mathcal{Q}')] \leq 2q/N'$. □

Proposition 5. $\Pr[\mathsf{Coll}_3(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq 2q\alpha/N' + 2q/N'$.

Proof. Note that in a collision with TL = BR we must have TL ≠ BL and $A \oplus R = B \oplus S$ (since $B \oplus S = B' \oplus S' = A \oplus R$, using TL = BR). Say the event $\mathsf{Coll}'_3(\mathcal{Q}'_i)$ occurs if there exist distinct queries $(A, B\|L, R)$, $(B, L\|R, S)$ in $\mathcal{Q}'_i$ such that $A \oplus R = B \oplus S$. With the same argument applied to the case BL = TR, we have $\mathsf{Coll}_3(\mathcal{Q}'_i) \implies \mathsf{Coll}'_3(\mathcal{Q}'_i)$. Therefore it suffices to show $\Pr[\mathsf{Coll}'_3(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq 2q\alpha/N' + 2q/N'$.

The analysis now proceeds rather similarly to Proposition 3. Let

$$\mathsf{Success}'_3(\mathcal{Q}'_i) = \mathsf{Coll}'_3(\mathcal{Q}'_i) \wedge \neg\mathsf{Coll}'_3(\mathcal{Q}'_{i-1}) \wedge \neg\mathsf{Xor}(\mathcal{Q}'_{i-1}) \wedge \neg\mathsf{FB}(\mathcal{Q}'_{i-1}).$$

We have $\Pr[\mathsf{Coll}'_3(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq \sum_{i=1}^{2q} \Pr[\mathsf{Success}'_3(\mathcal{Q}'_i)]$.

Fix a value of i, $1 \leq i \leq 2q$, and call the i-th query made by A' the *last query.* If $\mathsf{Success}'_3(\mathcal{Q}'_i)$ occurs then either the adversary (i.e. $\mathcal{A}'$) can use its last query as query TL or as query BL of its Coll'_3-solution. This gives rise to four possible cases given that the last query could be either forward or backward. In each case, we call the last query *successful* if $\mathsf{Success}'_3(\mathcal{Q}'_i)$ occurs and if the last query can be used in the position prescribed by that case (either TL or BL) in the Coll'_3-solution.

TL forward: We can use exactly the same analysis as in the case 'Forward TL' of Proposition 3. The probability that the last query is successful is therefore at most α/N'.

TL backward: Let $E^{-1}_{B\|L}(R)$ be the last query. For the last query to be successful, there must be a (necessarily unique) query of the form $(B, L\|R, S) \in \mathcal{Q}'_{i-1}$, for some $S \in \{0,1\}^n$. Since the answer A to the last query must be such that $A \oplus R = B \oplus S$ (as per the definition of Coll'_3) and $B \oplus S$ is uniquely determined, the last query has chance at most $1/N'$ of success.

[4] Since for each key there is only one relevant query, the tighter $1/N$ could be used as well.

BL forward: A 180° rotation of the collision diagram shows this case is symmetric to the case TL backward. The chance of success in this case is therefore at most $1/N'$.

BL backward: A 180° rotation of the collision diagram shows this case is symmetric to the case TL forward. The chance of success in this case is therefore at most α/N'.

The chance a forward last query is successful is therefore at most $(\alpha+1)/N'$ (adding the TL and BL forward cases) and likewise the chance that a backward last query is successful is at most $(\alpha+1)/N'$. Thus $\Pr[\mathsf{Success}'_3(\mathcal{Q}'_i)] \leq (\alpha+1)/N'$ for all i and $\sum_{i=1}^{2q} \Pr[\mathsf{Success}_1(\mathcal{Q}'_i)] \leq 2q\alpha/N' + 2q/N'$. (In fact, we even have $\Pr[\mathsf{Coll}_3(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq 2q\alpha/N' + 2q/N'$ since $\neg\mathsf{Xor}(\mathcal{Q}')$ was never used in the above.) □

Taking the sum of the bounds of Propositions 3, 4 and 5 one obtains that

$$\Pr[\mathsf{Coll}(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq \frac{6q\alpha}{N'} + \frac{4q}{N'}.$$

However, cases TL forward, BL backward and cases TL forward, BL backward of Propositions 3 and 5 reference the same events (the adversary is successful in case TL forward of Proposition 3 if and only if it is successful in case TL forward of Proposition 5, and likewise for the BL backward cases), which results in an "overcounting" of the adversary's probability of success by $2q\alpha/N'$. A more careful accounting of the adversary's probability of success thus shows

$$\Pr[\mathsf{Coll}(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')] \leq \frac{4q\alpha}{N'} + \frac{4q}{N'}. \tag{8}$$

Here we have not established (8) entirely formally, though this is the bound we use for $\Pr[\mathsf{Coll}(\mathcal{Q}') \wedge \neg\mathsf{Xor}(\mathcal{Q}') \wedge \neg\mathsf{FB}(\mathcal{Q}')]$ in Theorem 1. Establishing (8) formally would require dividing the event $\mathsf{Coll}(\mathcal{Q})$ into a different, less intuitive set of events than $\mathsf{Coll}_1(\mathcal{Q})$, $\mathsf{Coll}_2(\mathcal{Q})$, $\mathsf{Coll}_3(\mathcal{Q})$, events that are directly based on those that occur in the case analyses of Propositions 3–5. (For example, one of these events would be the event that the adversary ever obtains a "good R" through a forward or backward query, as defined for forward queries in case TL forward of Proposition 3 and implicitly defined (by symmetry) for backward queries in case BL backward of Proposition 3; another event would cover the cases TL backward and BL forward of Proposition 5; and so on.) The current form of the proof is our best compromise between readability and formality. In any case, the difference between $4q\alpha/N'$ and $6q\alpha/N'$ is relatively minor.

Summing (8) with the bounds of Proposition 2 and using (7), we obtain

$$\Pr[\mathsf{Coll}(\mathcal{Q})] \leq 2N \left(\frac{2eq}{\alpha N'}\right)^{\alpha} + \frac{4q\alpha}{N'} + \frac{4q}{N'}. \tag{9}$$

Since (9) holds for an arbitrary q-query adversary $\mathcal{A}$, this establishes Theorem 1.

5 Conclusion

In this work, we have shown that an earlier work concerning the security of Tandem-DM was incorrect. However, with a new proof (exploiting new ideas) we have shown that, in the ideal-cipher model, Tandem-DM is collision resistant almost up to the birthday bound and (provably) preimage resistant essentially up to the birthday bound (leaving considerable room for improvement for the latter).

On a high level, our proof of collision resistance adheres to a (by now) standard framework. We first modify the collision-finding adversary by giving it several "free" queries and subsequently we bound the modified adversary's chance of success using a case analysis. This approach allows to easily bound both the number of free queries and the probability of a query (free or not) causing a collision.

In contrast, the FGL proof directly uses a case analysis and subsequently uses free queries within the case analysis. This ad hoc addition of free queries (and its binding to a particular case) is problematic, as it does not allow proper accounting of the free queries. In particular, if a free query is fresh it might cause a collision (or other bad event) elsewhere whereas if the free query has actually been asked before, no new randomness can be extracted from it.

Thus, apart from having established the security of Tandem-DM, we hope that our work also serves as a useful reminder to some of the subtleties involved in ICM proofs and as a guideline on how to avoid certain pitfalls.

References

1. Dodis, Y., Steinberger, J.: Message Authentication Codes from Unpredictable Block Ciphers. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 267–285. Springer, Heidelberg (2009), Full version available at `http://people.csail.mit.edu/dodis/ps/tight-mac.ps`
2. Fleischmann, E., Forler, C., Gorski, M., Lucks, S.: Collision resistant double-length hashing. In: Heng, S.-H., Kurosawa, K. (eds.) ProvSec 2010. LNCS, vol. 6402, pp. 102–118. Springer, Heidelberg (2010)
3. Fleischmann, E., Gorski, M., Lucks, S.: On the security of TANDEM-DM. In: Dunkelman, O. (ed.) FSE 2009. LNCS, vol. 5665, pp. 84–103. Springer, Heidelberg (2009)
4. Fleischmann, E., Gorski, M., Lucks, S.: Security of Cyclic Double Block Length Hash Functions. In: Parker, M.G. (ed.) Cryptography and Coding 2009. LNCS, vol. 5921, pp. 153–175. Springer, Heidelberg (2009)
5. Hirose, S.: Provably secure double-block-length hash functions in a black-box model. In: Park, C.-s., Chee, S. (eds.) ICISC 2004. LNCS, vol. 3506, pp. 330–342. Springer, Heidelberg (2005)
6. Hirose, S.: Some plausible constructions of double-block-length hash functions. In: Robshaw, M.J.B. (ed.) FSE 2006. LNCS, vol. 4047, pp. 210–225. Springer, Heidelberg (2006)
7. Krause, M., Armknecht, F., Fleischmann, E.: Preimage resistance beyond the birthday bound: double-length hashing revisited. Preprint, `http://eprint.iacr.org/2010/519`

8. Lai, X., Massey, J.: Hash functions based on block ciphers. In: Rueppel, R.A. (ed.) EUROCRYPT 1992. LNCS, vol. 658, pp. 55–70. Springer, Heidelberg (1993)
9. Lee, J., Kwon, D.: The security of Abreast-DM in the ideal cipher model. IEICE Transactions 94-A(1), 104–109 (2011), `http://eprint.iacr.org/2009/225.pdf`
10. Lee, J., Steinberger, J.: Multi-property-preserving Domain Extension Using Polynomial-Based Modes of Operation. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 573–596. Springer, Heidelberg (2010)
11. Lee, J., Stam, M., Steinberger, J.: The collision security of Tandem-DM in the ideal-cipher model. Full version of this paper, `http://eprint.iacr.org/2010/409`
12. Lee, J., Stam, M., Steinberger, J.: The preimage security of double-block-length compression functions. Preprint, `http://eprint.iacr.org/2011/210`
13. Lucks, S.: A collision-resistant rate-1 double-block-length hash function. In: Symmetric Cryptography. Dagstuhl Seminar Proceedings, 07021 (2007)
14. Özen, O., Stam, M.: Another Glance at Double-Length Hashing. In: Parker, M.G. (ed.) Cryptography and Coding 2009. LNCS, vol. 5921, pp. 176–201. Springer, Heidelberg (2009)
15. Rogaway, P., Shrimpton, T.: Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 371–388. Springer, Heidelberg (2004)
16. Rogaway, P., Steinberger, J.: Constructing cryptographic hash functions from fixed-key blockciphers. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 433–450. Springer, Heidelberg (2008)
17. Shrimpton, T., Stam, M.: Building a collision-resistant compression function from non-compressing primitives. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part II. LNCS, vol. 5126, pp. 643–654. Springer, Heidelberg (2008)
18. Steinberger, J.P.: The Collision Intractability of MDC-2 in the Ideal-Cipher Model. In: Naor, M. (ed.) EUROCRYPT 2007. LNCS, vol. 4515, pp. 34–51. Springer, Heidelberg (2007)
19. Stam, M.: Beyond Uniformity: Better Security/Efficiency Tradeoffs for Compression Functions. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 397–412. Springer, Heidelberg (2008)
20. Stam, M.: Blockcipher-based hashing revisited. In: Dunkelman, O. (ed.) FSE 2009. LNCS, vol. 5665, pp. 67–83. Springer, Heidelberg (2009)
21. Wagner, D.: Cryptanalysis of the Yi-Lam Hash. In: Okamoto, T. (ed.) ASIACRYPT 2000. LNCS, vol. 1976, pp. 483–488. Springer, Heidelberg (2000)
22. Yi, X., Lam, K.-Y.: A new hash function based on block cipher. In: Mu, Y., Pieprzyk, J.P., Varadharajan, V. (eds.) ACISP 1997. LNCS, vol. 1270, pp. 139–146. Springer, Heidelberg (1997)