Under what reducibilities are KLR and MLR Medvedev equivalent?

David J. Webb

Department of Mathematics, University of Hawaii at Manoa

Midwest Computability Seminar

KLR and MLR

Open Question

Do Martin-Löf randomness (MLR) and Kolmogorov-Loveland randomness (KLR) coincide?

Known: If $A \in \mathrm{MLR}$, then $A \in \mathrm{KLR}$. So $\mathrm{KLR} \leq_s \mathrm{MLR}$.

Theorem 1 (Merkle et al.)

If $A = A_0 \oplus A_1$ is KL-random, then at least one of the A_i is ML-random.

Corollary: As mass problems, $\mathrm{MLR} \leq_w \mathrm{KLR}$.

Question: (Miyabe) Is there a uniform reduction?

KLR and MLR

Open Question

Do Martin-Löf randomness (MLR) and Kolmogorov-Loveland randomness (KLR) coincide?

Known: If $A \in \text{MLR}$, then $A \in \text{KLR}$. So $\text{KLR} \leq_s \text{MLR}$.

Theorem 1 (Merkle et al.)

If $A = A_0 \oplus A_1$ is KL-random, then at least one of the A_i is ML-random.

Corollary: As mass problems, $\text{MLR} \leq_w \text{KLR}$.

Question: (Miyabe) Is there a uniform reduction?

KLR and MLR

Open Question

Do Martin-Löf randomness (MLR) and Kolmogorov-Loveland randomness (KLR) coincide?

Known: If $A \in$ MLR, then $A \in$ KLR. So KLR $\leq_s$ MLR.

Theorem 1 (Merkle et al.)

If $A = A_0 \oplus A_1$ is KL-random, then at least one of the A_i is ML-random.

Corollary: As mass problems, MLR $\leq_w$ KLR.

Question: (Miyabe) Is there a uniform reduction?

KLR and MLR

Open Question

Do Martin-Löf randomness (MLR) and Kolmogorov-Loveland randomness (KLR) coincide?

Known: If $A \in \text{MLR}$, then $A \in \text{KLR}$. So $\text{KLR} \leq_s \text{MLR}$.

Theorem 1 (Merkle et al.)

If $A = A_0 \oplus A_1$ is KL-random, then at least one of the A_i is ML-random.

Corollary: As mass problems, $\text{MLR} \leq_w \text{KLR}$.

Question: (Miyabe) Is there a uniform reduction?

KLR and MLR

Open Question

Do Martin-Löf randomness (MLR) and Kolmogorov-Loveland randomness (KLR) coincide?

Known: If $A \in \text{MLR}$, then $A \in \text{KLR}$. So $\text{KLR} \leq_s \text{MLR}$.

Theorem 1 (Merkle et al.)

If $A = A_0 \oplus A_1$ is KL-random, then at least one of the A_i is ML-random.

Corollary: As mass problems, $\text{MLR} \leq_w \text{KLR}$.

Question: (Miyabe) Is there a uniform reduction?

KLR and MLR

Theorem 2

As mass problems, $\mathrm{MLR} \leq_s \mathrm{KLR}$.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in \mathrm{MLR}$ iff $\exists c \, \forall n \; K(A_i \restriction n) \geq n - c$.
- Approximate $K(A_i \restriction n)$ from above by $K_s(A_i \restriction n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \restriction n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

$\Phi(A)$ A_0

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact $\mathrm{MLR} \leq_{s,tt} \mathrm{KLR}$.

KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ K(A_i{\upharpoonright}n) \geq n - c$.
- Approximate $K(A_i{\upharpoonright}n)$ from above by $K_s(A_i{\upharpoonright}n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i{\upharpoonright}n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ K(A_i \upharpoonright n) \geq n - c$.
- Approximate $K(A_i \upharpoonright n)$ from above by $K_s(A_i \upharpoonright n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \upharpoonright n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

$\Phi(A)$

A_0

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\, \forall n\; K(A_i \restriction n) \geq n - c$.
- Approximate $K(A_i \restriction n)$ from above by $K_s(A_i \restriction n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \restriction n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i \upharpoonright n) \geq n - c$.
- Approximate $K(A_i \upharpoonright n)$ from above by $K_s(A_i \upharpoonright n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \upharpoonright n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i{\upharpoonright}n) \geq n - c$.
- Approximate $K(A_i{\upharpoonright}n)$ from above by $K_s(A_i{\upharpoonright}n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i{\upharpoonright}n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i \upharpoonright n) \geq n - c$.
- Approximate $K(A_i \upharpoonright n)$ from above by $K_s(A_i \upharpoonright n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \upharpoonright n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

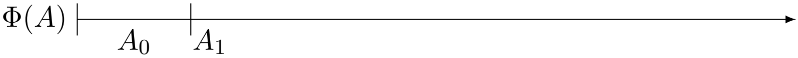
KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i \restriction n) \geq n - c$.
- Approximate $K(A_i \restriction n)$ from above by $K_s(A_i \restriction n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s + 1$, an $n \leq s + 1$ has $K_{s+1}(A_i \restriction n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.



Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i \upharpoonright n) \geq n - c$.
- Approximate $K(A_i \upharpoonright n)$ from above by $K_s(A_i \upharpoonright n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \upharpoonright n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.

Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

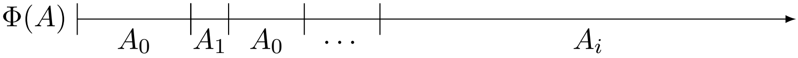
KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i \restriction n) \geq n - c$.
- Approximate $K(A_i \restriction n)$ from above by $K_s(A_i \restriction n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \restriction n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.



Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

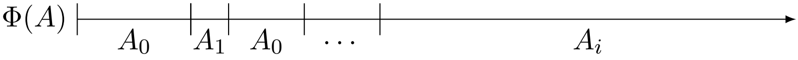
KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i \upharpoonright n) \geq n - c$.
- Approximate $K(A_i \upharpoonright n)$ from above by $K_s(A_i \upharpoonright n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s + 1$, an $n \leq s + 1$ has $K_{s+1}(A_i \upharpoonright n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.



Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

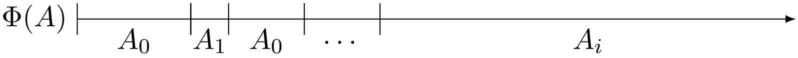
KLR and MLR

Theorem 2

As mass problems, MLR $\leq_s$ KLR.

Proof Idea: Output bits from A_i, switching whenever A_i "doesn't seem random".

- $A_i \in$ MLR iff $\exists c\ \forall n\ \forall s\ K_s(A_i \upharpoonright n) \geq n - c$.
- Approximate $K(A_i \upharpoonright n)$ from above by $K_s(A_i \upharpoonright n)$.
- Test values of c, starting at $c = 0$.
- If at a stage $s+1$, an $n \leq s+1$ has $K_{s+1}(A_i \upharpoonright n) < n - c_s$, switch to outputting A_{1-i} and set $c_{s+1} = c_s + 1$.



Only $2n$ bits of A are needed to compute $\Phi^A(n)$. So in fact MLR $\leq_{s,tt}$ KLR.

Notation

Definition

Φ^X is a truth-table reduction if there is a computable function f such that for each n and X, $n \in \Phi^X$ iff $X \models \sigma_{f(n)}$.

- $\{\sigma_n \mid n \in \omega\}$ is a uniformly computable list of all the finite propositional formulas in variables $v_1, v_2, \ldots$.
- The variables in σ_n are $v_{n_1}, \ldots, v_{n_d}$, where d depends on n.
- $X \models \sigma_n$ if σ_n is true with $X(n_1), \ldots, X(n_d)$ substituted for $v_{n_1}, \ldots, v_{n_d}$.

Question

For what reducibilities $*$ is it true that MLR $\leq_{s,*}$ KLR?

Definition

$\text{Either}(\mathcal{C}) = \{A \oplus B : A \in \mathcal{C} \text{ or } B \in \mathcal{C}\}$

Notation

Definition

Φ^X is a truth-table reduction if there is a computable function f such that for each n and X, $n \in \Phi^X$ iff $X \models \sigma_{f(n)}$.

- $\{\sigma_n \mid n \in \omega\}$ is a uniformly computable list of all the finite propositional formulas in variables $v_1, v_2, \ldots$.
- The variables in σ_n are $v_{n_1}, \ldots, v_{n_d}$, where d depends on n.
- $X \models \sigma_n$ if σ_n is true with $X(n_1), \ldots, X(n_d)$ substituted for $v_{n_1}, \ldots, v_{n_d}$.

Question

For what reducibilities $*$ is it true that MLR $\leq_{s,*}$ KLR?

Definition

$\text{Either}(\mathcal{C}) = \{A \oplus B : A \in \mathcal{C} \text{ or } B \in \mathcal{C}\}$

Notation

Definition

Φ^X is a truth-table reduction if there is a computable function f such that for each n and X, $n \in \Phi^X$ iff $X \models \sigma_{f(n)}$.

- $\{\sigma_n \mid n \in \omega\}$ is a uniformly computable list of all the finite propositional formulas in variables $v_1, v_2, \ldots$.
- The variables in σ_n are $v_{n_1}, \ldots, v_{n_d}$, where d depends on n.
- $X \models \sigma_n$ if σ_n is true with $X(n_1), \ldots, X(n_d)$ substituted for $v_{n_1}, \ldots, v_{n_d}$.

Question

For what reducibilities $*$ is it true that MLR $\leq_{s,*}$ KLR?

Definition

$\text{Either}(\mathcal{C}) = \{A \oplus B : A \in \mathcal{C} \text{ or } B \in \mathcal{C}\}$

Notation

Definition

Φ^X is a truth-table reduction if there is a computable function f such that for each n and X, $n \in \Phi^X$ iff $X \models \sigma_{f(n)}$.

- $\{\sigma_n \mid n \in \omega\}$ is a uniformly computable list of all the finite propositional formulas in variables $v_1, v_2, \ldots$.
- The variables in σ_n are $v_{n_1}, \ldots, v_{n_d}$, where d depends on n.
- $X \models \sigma_n$ if σ_n is true with $X(n_1), \ldots, X(n_d)$ substituted for $v_{n_1}, \ldots, v_{n_d}$.

Question

For what reducibilities $*$ is it true that MLR $\leq_{s,*}$ KLR?

Definition

$\text{Either}(\mathcal{C}) = \{A \oplus B : A \in \mathcal{C} \text{ or } B \in \mathcal{C}\}$

Notation

Definition

Φ^X is a truth-table reduction if there is a computable function f such that for each n and X, $n \in \Phi^X$ iff $X \models \sigma_{f(n)}$.

- $\{\sigma_n \mid n \in \omega\}$ is a uniformly computable list of all the finite propositional formulas in variables $v_1, v_2, \ldots$.
- The variables in σ_n are $v_{n_1}, \ldots, v_{n_d}$, where d depends on n.
- $X \models \sigma_n$ if σ_n is true with $X(n_1), \ldots, X(n_d)$ substituted for $v_{n_1}, \ldots, v_{n_d}$.

Question

For what reducibilities $*$ is it true that MLR $\leq_{s,*}$ KLR?

Definition

$\text{Either}(\mathcal{C}) = \{A \oplus B : A \in \mathcal{C} \text{ or } B \in \mathcal{C}\}$

Notation

Definition

Φ^X is a truth-table reduction if there is a computable function f such that for each n and X, $n \in \Phi^X$ iff $X \models \sigma_{f(n)}$.

- $\{\sigma_n \mid n \in \omega\}$ is a uniformly computable list of all the finite propositional formulas in variables $v_1, v_2, \ldots$.
- The variables in σ_n are $v_{n_1}, \ldots, v_{n_d}$, where d depends on n.
- $X \models \sigma_n$ if σ_n is true with $X(n_1), \ldots, X(n_d)$ substituted for $v_{n_1}, \ldots, v_{n_d}$.

Question

For what reducibilities $*$ is it true that $\mathrm{MLR} \leq_{s,*} \mathrm{Either}(\mathrm{MLR})$?

Definition

$\mathrm{Either}(\mathcal{C}) = \{A \oplus B : A \in \mathcal{C} \text{ or } B \in \mathcal{C}\}$

Reducibilities

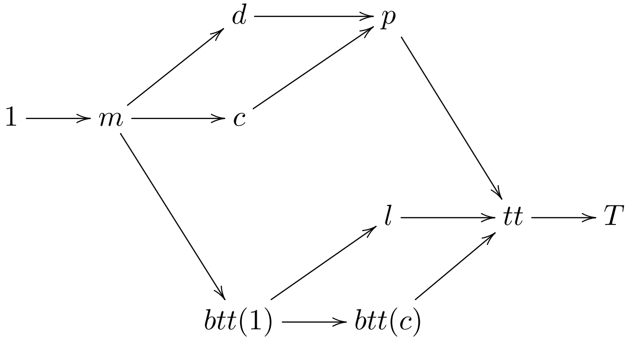


Figure: Reducibilities related to tt

Theorem

Positive, linear, and bounded truth-table reductions do not witness MLR $\leq_s$ Either(MLR)

Reducibilities

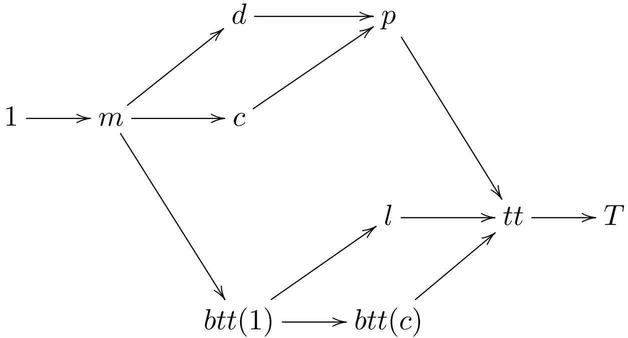


Figure: Reducibilities related to tt

Theorem

Positive, linear, and bounded truth-table reductions do not witness MLR $\leq_s$ Either(MLR)

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either(MLR)}$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either(MLR)}$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 1:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an even literal.

- Let $A = 1 \oplus R$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 2: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only odd literals.

- Set $A = R \oplus 0$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Positive Reducibility Cannot Be Used

Here each $\sigma_{f(n)}$ is a CNF of the form $\bigwedge_{k=1}^{t_n} \bigvee_{i=1}^{m_k} v_{f(n),i,k}$.

Proof: **Case 3:** There are infinitely many tables $\sigma_{f(n)}$ such that every $\bigvee_{i=1}^{m_k}$ contains an odd literal.

- Let $A = R \oplus 1$, so $A \in \text{Either}(\text{MLR})$.
- $A \models \sigma_{f(n)}$, as every disjunct in such a $\sigma_f(n)$ is true.
- It is computable to determine if $\sigma_{f(n)}$ is of this form.
- $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Case 4: For almost all tables $\sigma_{f(n)}$, there is a $\bigvee_{i=1}^{m_k}$ containing only even literals.

- Set $A = 0 \oplus R$.
- $A \not\models \sigma_{f(n)}$, as some disjunct is false.
- $|\Phi^A| < \infty$, so Φ^A is computable.

Linear Reducibility Cannot Be Used

Now each $\sigma_{f(n)}$ is of the form $\bigoplus_{k=1}^{t_n} v_{f(k)}$.

Definition

Let v_{n_i} appear in $\sigma_{f(n)}$. Say that n_i is a *fresh* bit if for $m < n$, v_{n_i} does not appear in $\sigma_{f(m)}$.

Proof: If Φ^X only queries finitely many bits, it is computable regardless of X. So suppose a fresh bit can always be found.

- Without loss of generality, infinitely many of these are even.
- Changing a single bit of any $\sigma_{f(n)}$ changes the output of the table.
- For fresh even n_i, ensure $\bigoplus_{k=1}^{t_n} v_{f(k)} = 1$.
- Set the odd bits of A to be random.
- It is computable to search for fresh even bits, so $\Phi^A \notin \mathrm{IM} \supseteq \mathrm{MLR}$.

Linear Reducibility Cannot Be Used

Now each $\sigma_{f(n)}$ is of the form $\bigoplus_{k=1}^{t_n} v_{f(k)}$.

Definition

Let v_{n_i} appear in $\sigma_{f(n)}$. Say that n_i is a *fresh* bit if for $m < n$, v_{n_i} does not appear in $\sigma_{f(m)}$.

Proof: If Φ^X only queries finitely many bits, it is computable regardless of X. So suppose a fresh bit can always be found.

- Without loss of generality, infinitely many of these are even.
- Changing a single bit of any $\sigma_{f(n)}$ changes the output of the table.
- For fresh even n_i, ensure $\bigoplus_{k=1}^{t_n} v_{f(k)} = 1$.
- Set the odd bits of A to be random.
- It is computable to search for fresh even bits, so $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Linear Reducibility Cannot Be Used

Now each $\sigma_{f(n)}$ is of the form $\bigoplus_{k=1}^{t_n} v_{f(k)}$.

Definition

Let v_{n_i} appear in $\sigma_{f(n)}$. Say that n_i is a *fresh* bit if for $m < n$, v_{n_i} does not appear in $\sigma_{f(m)}$.

Proof: If Φ^X only queries finitely many bits, it is computable regardless of X. So suppose a fresh bit can always be found.

- Without loss of generality, infinitely many of these are even.
- Changing a single bit of any $\sigma_{f(n)}$ changes the output of the table.
- For fresh even n_i, ensure $\bigoplus_{k=1}^{t_n} v_{f(k)} = 1$.
- Set the odd bits of A to be random.
- It is computable to search for fresh even bits, so $\Phi^A \notin \mathrm{IM} \supseteq \mathrm{MLR}$.

Linear Reducibility Cannot Be Used

Now each $\sigma_{f(n)}$ is of the form $\bigoplus_{k=1}^{t_n} v_{f(k)}$.

Definition

Let v_{n_i} appear in $\sigma_{f(n)}$. Say that n_i is a *fresh* bit if for $m < n$, v_{n_i} does not appear in $\sigma_{f(m)}$.

Proof: If Φ^X only queries finitely many bits, it is computable regardless of X. So suppose a fresh bit can always be found.

- Without loss of generality, infinitely many of these are even.
- Changing a single bit of any $\sigma_{f(n)}$ changes the output of the table.
- For fresh even n_i, ensure $\bigoplus_{k=1}^{t_n} v_{f(k)} = 1$.
- Set the odd bits of A to be random.
- It is computable to search for fresh even bits, so $\Phi^A \notin \text{IM} \supseteq \text{MLR}$.

Linear Reducibility Cannot Be Used

Now each $\sigma_{f(n)}$ is of the form $\bigoplus_{k=1}^{t_n} v_{f(k)}$.

Definition

Let v_{n_i} appear in $\sigma_{f(n)}$. Say that n_i is a *fresh* bit if for $m < n$, v_{n_i} does not appear in $\sigma_{f(m)}$.

Proof: If Φ^X only queries finitely many bits, it is computable regardless of X. So suppose a fresh bit can always be found.

- Without loss of generality, infinitely many of these are even.
- Changing a single bit of any $\sigma_{f(n)}$ changes the output of the table.
- For fresh even n_i, ensure $\bigoplus_{k=1}^{t_n} v_{f(k)} = 1$.
- Set the odd bits of A to be random.
- It is computable to search for fresh even bits, so $\Phi^A \notin \mathrm{IM} \supseteq \mathrm{MLR}$.

Linear Reducibility Cannot Be Used

Now each $\sigma_{f(n)}$ is of the form $\bigoplus_{k=1}^{t_n} v_{f(k)}$.

Definition

Let v_{n_i} appear in $\sigma_{f(n)}$. Say that n_i is a *fresh* bit if for $m < n$, v_{n_i} does not appear in $\sigma_{f(m)}$.

Proof: If Φ^X only queries finitely many bits, it is computable regardless of X. So suppose a fresh bit can always be found.

- Without loss of generality, infinitely many of these are even.
- Changing a single bit of any $\sigma_{f(n)}$ changes the output of the table.
- For fresh even n_i, ensure $\bigoplus_{k=1}^{t_n} v_{f(k)} = 1$.
- Set the odd bits of A to be random.
- It is computable to search for fresh even bits, so $\Phi^A \notin \mathrm{IM} \supseteq \mathrm{MLR}$.

Proof Ideas

Definition

$Q(n) = \{v_{n_1}, \ldots, v_{n_d}\}$, the set of variables in $\sigma_{f(n)}$

Definition

$C \subseteq Q(n)$ *controls* $\sigma_{f(n)}$ if some truth assignment of C ensures $\sigma_{f(n)} = \top$.

Example: $\{p, q\}$ controls $(p \vee q) \rightarrow r$ via the assignment $p = q = \bot$. $\{r\}$ can also control the formula via $r = \top$.

The proof strategy for $btt(c)$ relies on two ideas that appeared in earlier proofs:

- Computably search for $\sigma_{f(n)}$ with fresh $Q(n)$
- Try to assign them to control $\sigma_{f(n)}$

Proof Ideas

Definition

$Q(n) = \{v_{n_1}, \ldots, v_{n_d}\}$, the set of variables in $\sigma_{f(n)}$

Definition

$C \subseteq Q(n)$ *controls* $\sigma_{f(n)}$ if some truth assignment of C ensures $\sigma_{f(n)} = \top$.

Example: $\{p, q\}$ controls $(p \vee q) \to r$ via the assignment $p = q = \bot$. $\{r\}$ can also control the formula via $r = \top$.

The proof strategy for $btt(c)$ relies on two ideas that appeared in earlier proofs:

- Computably search for $\sigma_{f(n)}$ with fresh $Q(n)$
- Try to assign them to control $\sigma_{f(n)}$

Proof Ideas

Definition

$Q(n) = \{v_{n_1}, \ldots, v_{n_d}\}$, the set of variables in $\sigma_{f(n)}$

Definition

$C \subseteq Q(n)$ *controls* $\sigma_{f(n)}$ if some truth assignment of C ensures $\sigma_{f(n)} = \top$.

Example: $\{p, q\}$ controls $(p \vee q) \to r$ via the assignment $p = q = \bot$. $\{r\}$ can also control the formula via $r = \top$.

The proof strategy for $btt(c)$ relies on two ideas that appeared in earlier proofs:

- Computably search for $\sigma_{f(n)}$ with fresh $Q(n)$
- Try to assign them to control $\sigma_{f(n)}$

Proof Ideas

Definition

$Q(n) = \{v_{n_1}, \ldots, v_{n_d}\}$, the set of variables in $\sigma_{f(n)}$

Definition

$C \subseteq Q(n)$ *controls* $\sigma_{f(n)}$ if some truth assignment of C ensures $\sigma_{f(n)} = \top$.

Example: $\{p, q\}$ controls $(p \vee q) \to r$ via the assignment $p = q = \bot$. $\{r\}$ can also control the formula via $r = \top$.

The proof strategy for $btt(c)$ relies on two ideas that appeared in earlier proofs:

- Computably search for $\sigma_{f(n)}$ with fresh $Q(n)$
- Try to assign them to control $\sigma_{f(n)}$

Bounded Truth-Table Reducibility Fails

There are at most c variables in each $\sigma_{f(n)}$.

Proof Sketch: We may assume $\sigma_{f(n)}$ is constant (i.e. $\top$ or $\bot$) only finitely often. Induct on c.

For the base case $c = 1$, each table queries at most one variable. So controlling these tables is easy!

If only finitely many tables query a fresh bit, Φ^X is computable. Instead assume Φ^X infinitely often queries a fresh bit - without loss, an even bit. Control these. Set the odd bits of A to be random.

The computable search for fresh even bits always finds another, so Φ^A has an infinite computable subset and is not immune.

Bounded Truth-Table Reducibility Fails

There are at most c variables in each $\sigma_{f(n)}$.

Proof Sketch: We may assume $\sigma_{f(n)}$ is constant (i.e. $\top$ or $\bot$) only finitely often. Induct on c.

For the base case $c = 1$, each table queries at most one variable. So controlling these tables is easy!

If only finitely many tables query a fresh bit, Φ^X is computable. Instead assume Φ^X infinitely often queries a fresh bit - without loss, an even bit. Control these. Set the odd bits of A to be random.

The computable search for fresh even bits always finds another, so Φ^A has an infinite computable subset and is not immune.

Bounded Truth-Table Reducibility Fails

There are at most c variables in each $\sigma_{f(n)}$.

Proof Sketch: We may assume $\sigma_{f(n)}$ is constant (i.e. $\top$ or $\bot$) only finitely often. Induct on c.

For the base case $c = 1$, each table queries at most one variable. So controlling these tables is easy!

If only finitely many tables query a fresh bit, Φ^X is computable. Instead assume Φ^X infinitely often queries a fresh bit - without loss, an even bit. Control these. Set the odd bits of A to be random.

The computable search for fresh even bits always finds another, so Φ^A has an infinite computable subset and is not immune.

Bounded Truth-Table Reducibility Fails

There are at most c variables in each $\sigma_{f(n)}$.

Proof Sketch: We may assume $\sigma_{f(n)}$ is constant (i.e. $\top$ or $\bot$) only finitely often. Induct on c.

For the base case $c = 1$, each table queries at most one variable. So controlling these tables is easy!

If only finitely many tables query a fresh bit, Φ^X is computable. Instead assume Φ^X infinitely often queries a fresh bit - without loss, an even bit. Control these. Set the odd bits of A to be random.

The computable search for fresh even bits always finds another, so Φ^A has an infinite computable subset and is not immune.

Bounded Truth-Table Reducibility Fails

There are at most c variables in each $\sigma_{f(n)}$.

Proof Sketch: We may assume $\sigma_{f(n)}$ is constant (i.e. $\top$ or $\bot$) only finitely often. Induct on c.

For the base case $c = 1$, each table queries at most one variable. So controlling these tables is easy!

If only finitely many tables query a fresh bit, Φ^X is computable. Instead assume Φ^X infinitely often queries a fresh bit - without loss, an even bit. Control these. Set the odd bits of A to be random.

The computable search for fresh even bits always finds another, so Φ^A has an infinite computable subset and is not immune.

$btt(c)$ fails: Induction Step

Now assume that for any $btt(d)$ reduction with $d < c$, there is an $B \in \text{Either}(\text{MLR})$ that defeats it.

The Greedy Algorithm for Fresh Bits

Search for indicies n such that $\sigma_{f(n)}$ only queries fresh bits as follows:

- $n_0 = 0$
- n_{i+1} is the least n such that $Q(n) \cap \bigcup_{k<i} Q(n_k) = \emptyset$.

If this search succeeds, we have an infinite computable set whose tables we can try to control. But what if the search fails?

$btt(c)$ fails: Induction Step

Now assume that for any $btt(d)$ reduction with $d < c$, there is an $B \in \text{Either(MLR)}$ that defeats it.

The Greedy Algorithm for Fresh Bits

Search for indicies n such that $\sigma_{f(n)}$ only queries fresh bits as follows:

- $n_0 = 0$
- n_{i+1} is the least n such that $Q(n) \cap \bigcup_{k<i} Q(n_k) = \emptyset$.

If this search succeeds, we have an infinite computable set whose tables we can try to control. But what if the search fails?

$btt(c)$ fails: Induction Step

Now assume that for any $btt(d)$ reduction with $d < c$, there is an $B \in \text{Either}(\text{MLR})$ that defeats it.

The Greedy Algorithm for Fresh Bits

Search for indicies n such that $\sigma_{f(n)}$ only queries fresh bits as follows:

- $n_0 = 0$
- n_{i+1} is the least n such that $Q(n) \cap \bigcup_{k<i} Q(n_k) = \emptyset$.

If this search succeeds, we have an infinite computable set whose tables we can try to control. But what if the search fails?

$btt(c)$ fails: IS: Search Fails

- If the search fails, then for some N, all greater n have that $Q(n) \cap \bigcup_{k<N} Q(n_k) \neq \emptyset$.

- So Φ is using information from $H = \bigcup_{k<N} Q(n_k)$ over and over again. Fixing H, Φ acts as a $btt(d)$-reduction, $d < c$.

- $\sigma_g(n)$ is the table $\sigma_f(n)$ with all $v_{n_i} \in H$ replaced by $\perp$. This defines a Ψ^X that acts as Φ^X on reals A with $A \cap H = \emptyset$.

- Each table $\sigma_{g(n)}$ has $|Q(n)| < c$. Use the induction hypothesis to get $B \in \text{Either(MLR)}$ with $\Psi^B \notin \text{MLR}$.

- Define $A = B \setminus H$. $A \in \text{Either(MLR)}$ as MLR is closed under finite differences. $\Phi^A = \Psi^B \notin \text{MLR}$.

$btt(c)$ fails: IS: Search Fails

- If the search fails, then for some N, all greater n have that $Q(n) \cap \bigcup_{k<N} Q(n_k) \neq \emptyset$.

- So Φ is using information from $H = \bigcup_{k<N} Q(n_k)$ over and over again. Fixing H, Φ acts as a $btt(d)$-reduction, $d < c$.

- $\sigma_g(n)$ is the table $\sigma_f(n)$ with all $v_{n_i} \in H$ replaced by $\perp$. This defines a Ψ^X that acts as Φ^X on reals A with $A \cap H = \emptyset$.

- Each table $\sigma_{g(n)}$ has $|Q(n)| < c$. Use the induction hypothesis to get $B \in$ Either(MLR) with $\Psi^B \notin$ MLR.

- Define $A = B \setminus H$. $A \in$ Either(MLR) as MLR is closed under finite differences. $\Phi^A = \Psi^B \notin$ MLR.

$btt(c)$ fails: IS: Search Fails

- If the search fails, then for some N, all greater n have that $Q(n) \cap \bigcup_{k<N} Q(n_k) \neq \emptyset$.

- So Φ is using information from $H = \bigcup_{k<N} Q(n_k)$ over and over again. Fixing H, Φ acts as a $btt(d)$-reduction, $d < c$.

- $\sigma_g(n)$ is the table $\sigma_f(n)$ with all $v_{n_i} \in H$ replaced by $\perp$. This defines a Ψ^X that acts as Φ^X on reals A with $A \cap H = \emptyset$.

- Each table $\sigma_{g(n)}$ has $|Q(n)| < c$. Use the induction hypothesis to get $B \in \text{Either(MLR)}$ with $\Psi^B \notin \text{MLR}$.

- Define $A = B \setminus H$. $A \in \text{Either(MLR)}$ as MLR is closed under finite differences. $\Phi^A = \Psi^B \notin \text{MLR}$.

$btt(c)$ fails: IS: Search Fails

- If the search fails, then for some N, all greater n have that $Q(n) \cap \bigcup_{k<N} Q(n_k) \neq \emptyset$.

- So Φ is using information from $H = \bigcup_{k<N} Q(n_k)$ over and over again. Fixing H, Φ acts as a $btt(d)$-reduction, $d < c$.

- $\sigma_g(n)$ is the table $\sigma_f(n)$ with all $v_{n_i} \in H$ replaced by $\perp$. This defines a Ψ^X that acts as Φ^X on reals A with $A \cap H = \emptyset$.

- Each table $\sigma_{g(n)}$ has $|Q(n)| < c$. Use the induction hypothesis to get $B \in \text{Either(MLR)}$ with $\Psi^B \notin \text{MLR}$.

- Define $A = B \setminus H$. $A \in \text{Either(MLR)}$ as MLR is closed under finite differences. $\Phi^A = \Psi^B \notin \text{MLR}$.

$btt(c)$ fails: IS: Search Fails

- If the search fails, then for some N, all greater n have that $Q(n) \cap \bigcup_{k<N} Q(n_k) \neq \emptyset$.

- So Φ is using information from $H = \bigcup_{k<N} Q(n_k)$ over and over again. Fixing H, Φ acts as a $btt(d)$-reduction, $d < c$.

- $\sigma_g(n)$ is the table $\sigma_f(n)$ with all $v_{n_i} \in H$ replaced by $\perp$. This defines a Ψ^X that acts as Φ^X on reals A with $A \cap H = \emptyset$.

- Each table $\sigma_{g(n)}$ has $|Q(n)| < c$. Use the induction hypothesis to get $B \in \text{Either(MLR)}$ with $\Psi^B \notin \text{MLR}$.

- Define $A = B \setminus H$. $A \in \text{Either(MLR)}$ as MLR is closed under finite differences. $\Phi^A = \Psi^B \notin \text{MLR}$.

$btt(c)$ fails: IS: Search Succeeds

Finally, assume we have access to a computable set of indices n_i such that the $Q(n_i)$ are disjoint.

Case 1: There are infinitely many n_i such that some $C(n_i)$ containing only even bits controls $\sigma_{f(n_i)}$.

- By assumption, the $C(n_i)$ are disjoint, so we may set their bits without issue to get $A \models \sigma_{f(n_i)}$.
- Set the odd bits of A to be random.
- The greedy algorithm guarantees Φ^A is not immune.

Case 2: Almost all of the $\sigma_{f(n_i)}$ cannot be controlled by subset $C(n) \subseteq Q(n)$ containing only even bits.

- Set the **even** bits of A to be random.
- For almost all n_i, no matter how the even bits are fixed, we can assign odd bits in $Q(n_i)$ so that $A \not\models \sigma_{f(n_i)}$.
- The greedy algorithm guarantees Φ^A is not co-immune.

$btt(c)$ fails: IS: Search Succeeds

Finally, assume we have access to a computable set of indices n_i such that the $Q(n_i)$ are disjoint.

Case 1: There are infinitely many n_i such that some $C(n_i)$ containing only even bits controls $\sigma_{f(n_i)}$.

- By assumption, the $C(n_i)$ are disjoint, so we may set their bits without issue to get $A \models \sigma_{f(n_i)}$.
- Set the odd bits of A to be random.
- The greedy algorithm guarantees Φ^A is not immune.

Case 2: Almost all of the $\sigma_{f(n_i)}$ cannot be controlled by subset $C(n) \subseteq Q(n)$ containing only even bits.

- Set the **even** bits of A to be random.
- For almost all n_i, no matter how the even bits are fixed, we can assign odd bits in $Q(n_i)$ so that $A \not\models \sigma_{f(n_i)}$.
- The greedy algorithm guarantees Φ^A is not co-immune.

$btt(c)$ fails: IS: Search Succeeds

Finally, assume we have access to a computable set of indices n_i such that the $Q(n_i)$ are disjoint.

Case 1: There are infinitely many n_i such that some $C(n_i)$ containing only even bits controls $\sigma_{f(n_i)}$.

- By assumption, the $C(n_i)$ are disjoint, so we may set their bits without issue to get $A \models \sigma_{f(n_i)}$.
- Set the odd bits of A to be random.
- The greedy algorithm guarantees Φ^A is not immune.

Case 2: Almost all of the $\sigma_{f(n_i)}$ cannot be controlled by subset $C(n) \subseteq Q(n)$ containing only even bits.

- Set the **even** bits of A to be random.
- For almost all n_i, no matter how the even bits are fixed, we can assign odd bits in $Q(n_i)$ so that $A \not\models \sigma_{f(n_i)}$.
- The greedy algorithm guarantees Φ^A is not co-immune.

$btt(c)$ fails: IS: Search Succeeds

Finally, assume we have access to a computable set of indices n_i such that the $Q(n_i)$ are disjoint.

Case 1: There are infinitely many n_i such that some $C(n_i)$ containing only even bits controls $\sigma_{f(n_i)}$.

- By assumption, the $C(n_i)$ are disjoint, so we may set their bits without issue to get $A \models \sigma_{f(n_i)}$.
- Set the odd bits of A to be random.
- The greedy algorithm guarantees Φ^A is not immune.

Case 2: Almost all of the $\sigma_{f(n_i)}$ cannot be controlled by subset $C(n) \subseteq Q(n)$ containing only even bits.

- Set the **even** bits of A to be random.
- For almost all n_i, no matter how the even bits are fixed, we can assign odd bits in $Q(n_i)$ so that $A \not\models \sigma_{f(n_i)}$.
- The greedy algorithm guarantees Φ^A is not co-immune.

$btt(c)$ fails: IS: Search Succeeds

Finally, assume we have access to a computable set of indices n_i such that the $Q(n_i)$ are disjoint.

Case 1: There are infinitely many n_i such that some $C(n_i)$ containing only even bits controls $\sigma_{f(n_i)}$.

- By assumption, the $C(n_i)$ are disjoint, so we may set their bits without issue to get $A \models \sigma_{f(n_i)}$.
- Set the odd bits of A to be random.
- The greedy algorithm guarantees Φ^A is not immune.

Case 2: Almost all of the $\sigma_{f(n_i)}$ cannot be controlled by subset $C(n) \subseteq Q(n)$ containing only even bits.

- Set the **even** bits of A to be random.
- For almost all n_i, no matter how the even bits are fixed, we can assign odd bits in $Q(n_i)$ so that $A \not\models \sigma_{f(n_i)}$.
- The greedy algorithm guarantees Φ^A is not co-immune.

$btt(c)$ fails: IS: Search Succeeds

Finally, assume we have access to a computable set of indices n_i such that the $Q(n_i)$ are disjoint.

Case 1: There are infinitely many n_i such that some $C(n_i)$ containing only even bits controls $\sigma_{f(n_i)}$.

- By assumption, the $C(n_i)$ are disjoint, so we may set their bits without issue to get $A \models \sigma_{f(n_i)}$.
- Set the odd bits of A to be random.
- The greedy algorithm guarantees Φ^A is not immune.

Case 2: Almost all of the $\sigma_{f(n_i)}$ cannot be controlled by subset $C(n) \subseteq Q(n)$ containing only even bits.

- Set the **even** bits of A to be random.
- For almost all n_i, no matter how the even bits are fixed, we can assign odd bits in $Q(n_i)$ so that $A \not\models \sigma_{f(n_i)}$.
- The greedy algorithm guarantees Φ^A is not co-immune.

$btt(c)$ fails: IS: Search Succeeds

Finally, assume we have access to a computable set of indices n_i such that the $Q(n_i)$ are disjoint.

Case 1: There are infinitely many n_i such that some $C(n_i)$ containing only even bits controls $\sigma_{f(n_i)}$.

- By assumption, the $C(n_i)$ are disjoint, so we may set their bits without issue to get $A \models \sigma_{f(n_i)}$.
- Set the odd bits of A to be random.
- The greedy algorithm guarantees Φ^A is not immune.

Case 2: Almost all of the $\sigma_{f(n_i)}$ cannot be controlled by subset $C(n) \subseteq Q(n)$ containing only even bits.

- Set the **even** bits of A to be random.
- For almost all n_i, no matter how the even bits are fixed, we can assign odd bits in $Q(n_i)$ so that $A \not\models \sigma_{f(n_i)}$.
- The greedy algorithm guarantees Φ^A is not co-immune.

Final Comments

These tt-reduction generalizes easily to any number of finite columns $A_0 \oplus A_1 \oplus \cdots \oplus A_n$ or even the infinite case $\bigoplus_{i=1}^{\infty} A_i$ (where only one column is random).

Our proofs are really about bi-immunity:

Theorem

If $* \in \{p, l, btt(c)\}$, then BIM $\not\leq_{s,*}$ Either(BIM).

Corollary

If $* \in \{p, l, btt(c)\}$, then 1G $\not\leq_{s,*}$ Either(1G).

Questions:

- Does $\leq_s$ hold in either of these cases?
  - Does $\leq_{s,tt}$?
- What techniques could strengthen MLR $\leq_{s,*}$ KLR to other reducibilities?

Final Comments

These tt-reduction generalizes easily to any number of finite columns $A_0 \oplus A_1 \oplus \cdots \oplus A_n$ or even the infinite case $\bigoplus_{i=1}^{\infty} A_i$ (where only one column is random).

Our proofs are really about bi-immunity:

Theorem

If $* \in \{p, l, btt(c)\}$, then BIM $\not\leq_{s,*}$ Either(BIM).

Corollary

If $* \in \{p, l, btt(c)\}$, then 1G $\not\leq_{s,*}$ Either(1G).

Questions:

- Does $\leq_s$ hold in either of these cases?
  - Does $\leq_{s,tt}$?
- What techniques could strengthen MLR $\leq_{s,*}$ KLR to other reducibilities?

Final Comments

These tt-reduction generalizes easily to any number of finite columns $A_0 \oplus A_1 \oplus \cdots \oplus A_n$ or even the infinite case $\bigoplus_{i=1}^{\infty} A_i$ (where only one column is random).

Our proofs are really about bi-immunity:

Theorem

If $* \in \{p, l, btt(c)\}$, then BIM $\not\leq_{s,*}$ Either(BIM).

Corollary

If $* \in \{p, l, btt(c)\}$, then 1G $\not\leq_{s,*}$ Either(1G).

Questions:

- Does $\leq_s$ hold in either of these cases?
  - Does $\leq_{s,tt}$?
- What techniques could strengthen MLR $\leq_{s,*}$ KLR to other reducibilities?

Final Comments

These *tt*-reduction generalizes easily to any number of finite columns $A_0 \oplus A_1 \oplus \cdots \oplus A_n$ or even the infinite case $\bigoplus_{i=1}^{\infty} A_i$ (where only one column is random).

Our proofs are really about bi-immunity:

Theorem

If $* \in \{p, l, btt(c)\}$, then BIM $\not\leq_{s,*}$ Either(BIM).

Corollary

If $* \in \{p, l, btt(c)\}$, then 1G $\not\leq_{s,*}$ Either(1G).

Questions:

- Does $\leq_s$ hold in either of these cases?
  - Does $\leq_{s,tt}$?
- What techniques could strengthen MLR $\leq_{s,*}$ KLR to other reducibilities?

Final Comments

These tt-reduction generalizes easily to any number of finite columns $A_0 \oplus A_1 \oplus \cdots \oplus A_n$ or even the infinite case $\bigoplus_{i=1}^{\infty} A_i$ (where only one column is random).

Our proofs are really about bi-immunity:

Theorem

If $* \in \{p, l, btt(c)\}$, then $\text{BIM} \not\leq_{s,*} \text{Either}(\text{BIM})$.

Corollary

If $* \in \{p, l, btt(c)\}$, then $\text{1G} \not\leq_{s,*} \text{Either}(\text{1G})$.

Questions:

- Does $\leq_s$ hold in either of these cases?
  - Does $\leq_{s,tt}$?
- What techniques could strengthen $\text{MLR} \leq_{s,*} \text{KLR}$ to other reducibilities?

References

Bjørn Kjos-Hanssen and David J. Webb.
KL-randomness and effective dimension under strong reducibility.
In *Connecting with Computability*, pages 457–468, Cham, 2021. Springer International Publishing.

Wolfgang Merkle, Joseph S. Miller, André Nies, Jan Reimann, and Frank Stephan.
Kolmogorov-Loveland randomness and stochasticity.
Ann. Pure Appl. Logic, 138(1-3):183–210, 2006.

Kenshi Miyabe.
Muchnik degrees and Medvedev degrees of randomness notions.
In *Proceedings of the 14th and 15th Asian Logic Conferences*, pages 108–128. World Sci. Publ., Hackensack, NJ, 2019.

Piergiorgio Odifreddi.
Strong reducibilities.
Bulletin (New Series) of the American Mathematical Society, 4(1):37 – 86, 1981.