University of Amsterdam
Master of Logic

Forcing with Urelements

Witness names and Yao's repair

Simone Testino
Supervisor Yurii Khomskii
Second semester 2025–2026
Last updated: 26 November 2026

SOURCE

Bokai Yao, *Set Theory with Urelements*
arXiv v3 (2023)

MATERIALS

GitHub LaTeX source
Checked Isabelle/HOL exercise



INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Outline

1 A forcing refresher
2 Why old names fail
3 Yao's repair and mixing
4 Collection and fullness

Appendix

Atoms, sets and the empty set

The language contains membership $\in$ and a primitive atom predicate $\mathcal{A}$.

$$\mathcal{A}(a) \implies \forall x\ (x \notin a), \qquad \mathrm{Set}(x) \equiv \neg\mathcal{A}(x).$$

Having no members does not determine an object

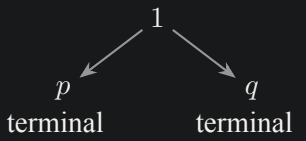
Two distinct atoms a, b and the set $\varnothing$ can all have no members. Extensionality identifies *sets* with the same members.

Yao's base theory requires neither the existence of an atom nor a set of all atoms.

Our example adds the explicit assumption $\mathcal{A}(a)$, $\mathcal{A}(b)$ and $a \neq b$.

Yao, §1.2, Axiom A and Definition 1. Printed pp. 4–5 / PDF pp. 11–12.

The ground model and the forcing order



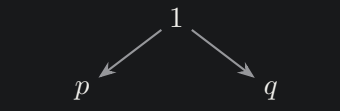
The *ground model* M satisfies set theory with urelements (ZFU_R). In M take distinct atoms a, b and a pure, three-condition poset $\mathbb{P} = \{1, p, q\}$. Its only strict comparisons are $p < 1$ and $q < 1$.

Two assumptions about M

Transitive means that if $x \in M$ and $y \in x$, then $y \in M$.
Countable means that, from outside M, we can enumerate its objects. The enumeration need not belong to M.

Yao, §§3.1–3.2 and Remark 55, specialised to three conditions. Printed pp. 44–48 / PDF pp. 51–55.

Which conditions can a filter contain?



Filter

A filter is a nonempty $G \subseteq \mathbb{P}$ with both properties.

Upward closure. If $r \in G$ and $r \leq s$ in $\mathbb{P}$, then $s \in G$.

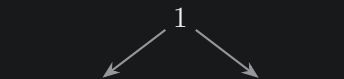
Directedness. For all $r, s \in G$ there is a $t \in G$ with $t \leq r$ and $t \leq s$.

Upward closure puts the weaker condition 1 in every filter. Since $p \perp q$, no condition strengthens both, so $\{1, p, q\}$ fails directedness. The filters are $\{1\}$, $\{1, p\}$ and $\{1, q\}$.

The *poset atoms* p, q are minimal conditions. The witnesses a, b are urelements.

Yao, §3.2 and Remark 55, worked finite example. Printed pp. 45–48 / PDF pp. 52–55.

Why there are exactly two generic filters



Testing $D = \{p, q\}$

start at r $\quad 1 \quad p \quad q$

choose s $\quad p \quad p \quad q$

Here s depends on r, and equality is allowed.

Density and genericity

A subset $D \subseteq \mathbb{P}$ is *dense* if

$$\forall r \in \mathbb{P}\ \exists s \in D\ (s \leq r).$$

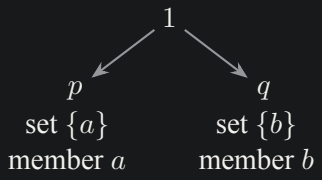
A filter G is M*-generic* if $G \cap D \neq \varnothing$ for every dense $D \in M$.

Since $\{p, q\} \in M$ is dense, a generic must contain p or q. Directedness rules out both at once.

Each terminal condition is its own only strengthening. Every dense set contains both, so G_p and G_q are generic.

Yao, §3.2 and Remark 55, worked finite example. Printed pp. 45–48 / PDF pp. 52–55.

The question on two branches



Suppose the set we get is $\{a\}$ in G_p and $\{b\}$ in G_q. Once the branch is known, we can point to its only member.

A forcing *name* σ is chosen in M. Its value after a filter G is chosen is written σ_G.

Can we choose one name before we know the branch?

We need the *same* σ to give $\sigma_{G_p} = a$ and $\sigma_{G_q} = b$.

Both filters already belong to M, so here $M[G] = M$. What varies is the value of a name inside that model.

Yao, Remark 55 and the old valuation, finite instance. Printed pp. 47–48 / PDF pp. 54–55.



Section 2

Can one old name pick the member on both branches?

Outline

1 A forcing refresher

2 **Why old names fail**

3 Yao's repair and mixing

4 Collection and fullness

Appendix

How to read an old name

In Yao's old system, each atom a is a name with $a_G = a$ on every branch. A *set-name* σ is a set of ordered pairs.

$$\Big\langle \underbrace{\tau}_{\text{an old name}} \;,\; \underbrace{r}_{\text{a condition}} \Big\rangle$$

To evaluate σ at G

Keep the pairs with $r \in G$. Evaluate each first entry at G and collect its value as a member of the resulting set.

$$\sigma_G = \{\tau_G \mid \exists r \in G\ (\langle \tau, r \rangle \in \sigma)\}.$$

For example, $\langle a, p \rangle$ contributes a whenever $p \in G$. If it is the only pair kept, the outer braces give $\sigma_G = \{a\}$.

Yao, Definition 51 and the old valuation. Printed pp. 46–47 / PDF pp. 53–54.

Evaluating the same set-name twice

Choose one old set-name $s_{\#} = \{\langle a, p\rangle, \langle b, q\rangle\}$.

Filter G	Pair kept	Value of $s_{\#}$	Its member
$G_p = \{1, p\}$	$\langle a, p\rangle$	$\{a\}$	a
$G_q = \{1, q\}$	$\langle b, q\rangle$	$\{b\}$	b

On the left, $p \in G_p$ and $q \notin G_p$. The one value to collect is $a_{G_p} = a$, giving $(s_{\#})_{G_p} = \{a\}$. On the right, only q survives the test, giving $\{b\}$.

Both filters contain 1, and both values have a member. In this example that is exactly what $1 \Vdash_{\#} \exists y\,(y \in s_{\#})$ means.

Yao, old valuation, Theorem 53 and Remark 55. Printed pp. 47–48 / PDF pp. 54–55.

Local witnesses and a uniform name

Write $M^{\mathbb{P}}_{\#}$ for the old names in M. On each branch there is a witness.

$$\forall G \in \{G_p, G_q\}\ \exists y\ (y \in (s_{\#})_G).$$

For example, the old name a works on G_p but fails on G_q, since $a \notin \{b\}$.

The uniform question quantifies over names first

$$\exists \sigma \in M^{\mathbb{P}}_{\#}\ \forall G \in \{G_p, G_q\}\ (\sigma_G \in (s_{\#})_G)?$$

The object y in the first formula can depend on G. In the second, one σ must give $\sigma_{G_p} = a$ and $\sigma_{G_q} = b$.

Yao, Remark 55; quantifier comparison for the finite example. Printed pp. 47–48 / PDF pp. 54–55.

Why every old witness name fails

Suppose an old name σ works on both branches. There are only two cases.

1. If σ is an atom, then $\sigma_{G_p} = \sigma_{G_q} = \sigma$. It cannot equal both a and b, since $a \neq b$.
2. If σ is a set-name, both values are sets. Neither value can equal the urelement required on its branch.

The failure is a failure of fullness

$$1 \Vdash_{\#} \exists y\, (y \in s_{\#}), \qquad \nexists \sigma \in M_{\#}^{\mathbb{P}}\; 1 \Vdash_{\#} \sigma \in s_{\#}.$$

Here forcing at 1 means truth on both generic branches. The old forcing theorem and truth lemma still hold.

Yao, Theorems 53–54 and Remark 55. Printed pp. 47–48 / PDF pp. 54–55.



Section 3

How can one name take two different atom values?

Outline

1 A forcing refresher

2 Why old names fail

3 Yao's repair and mixing

4 Collection and fullness

Appendix

Two roles for pairs in repaired names

Every repaired name is a *set* of pairs $\langle y, r \rangle$. Here y is either a urelement or another repaired name.

The first component determines the role

$y = a$ is an atom	proposes a as the value of the whole name
$y = \sigma$ is a name	proposes σ_G as a member of its set-value

The condition r still controls when the proposal is active.

Atoms themselves are no longer names. This opens the possibility of a set-name whose value is an atom.

Yao, Definition 57. Printed pp. 48 / PDF pp. 55.

Compatibility of repaired names

Compatibility requirement

For two pairs in one repaired name τ,

$$\left.\begin{array}{c}\langle a, r\rangle, \langle y, s\rangle \in \tau, \\ \mathcal{A}(a), \quad a \neq y\end{array}\right\} \implies r \perp s.$$

- Different atom-values must be attached to incompatible conditions.
- An atom-value and a proposed member must also be attached to incompatible conditions: a name y is a set, so $y \neq a$.

A filter can activate at most one atom-value. When it does, it activates no member proposal.

Yao, Definition 57, clause (ii). Printed pp. 48 / PDF pp. 55.

Evaluating a repaired name

Valuation at G

$$\tau_G = \begin{cases} a, & \text{if } \langle a, r\rangle \in \tau,\ \mathcal{A}(a),\ r \in G, \\ \{\sigma_G : \langle \sigma, r\rangle \in \tau,\ \sigma \text{ a name},\ r \in G\}, & \text{if no atom proposal is active.} \end{cases}$$

The compatibility rule makes the active atom unique and rules out active member proposals in that case.

If no atom proposal is active, valuation collects the active members in the familiar way. If no pair is active, the value is $\varnothing$.

Yao, Definition 63, using Definition 57. Printed pp. 48, 53 / PDF pp. 55, 60.

Names for an atom and its singleton

A name for the atom

$$\check{a} = \{\langle a, 1\rangle\}.$$

The first component is the atom a. Its pair proposes the whole value.

$$(\check{a})_G = a.$$

A name for its singleton

$$\{\check{a}\} = \{\langle \check{a}, 1\rangle\}.$$

The first component is the name $\check{a}$. Its pair proposes a member.

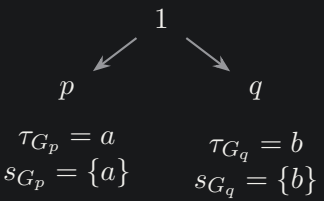
$$(\{\check{a}\})_G = \{a\}.$$

Replacing a by $\check{a}$ changes the role of the pair. The braces around a name alone do not tell us whether its value is an atom or a set.

Yao, Definitions 57 and 63. Printed pp. 48, 53 / PDF pp. 55, 60.

A repaired witness on both branches

$$\tau = \{\langle a, p\rangle, \langle b, q\rangle\}, \qquad s = \{\langle \check{a}, p\rangle, \langle \check{b}, q\rangle\}.$$



$p \perp q$ makes τ a valid repaired name. The name s has only member proposals.

On either branch, $\tau_G \in s_G$. Hence

$$1 \Vdash \tau \in s.$$

The displayed set of pairs for τ is the same as for $s_{\#}$, but its semantics has changed.

Yao, Definitions 57 and 63; finite instance of Lemma 60. Printed pp. 48–50, 53 / PDF pp. 55–57, 60.

Mixing extends this construction

Let $U^{\mathbb{P}}$ be the repaired names. On the antichain $D = \{p, q\}$, set

$$f(p) = \check{a}, \qquad f(q) = \check{b}.$$

The name τ is a mixture: $p \Vdash \tau = \check{a}$ and $q \Vdash \tau = \check{b}$.

Yao's Mixing Lemma, Lemma 60

For a given set function $f : D \to U^{\mathbb{P}}$, with D an antichain, there is a repaired name μ such that

$$\forall r \in D \qquad r \Vdash \mu = f(r).$$

Mixing starts with a set function f and needs neither Collection nor Choice. Gathering witness names is a separate step.

Yao, Lemma 60, statement and construction. Printed pp. 49–50 / PDF pp. 56–57.

Structure, renaming and representation

Swap a and b consistently, leaving our pure poset fixed. The same calculations give the same existence and impossibility results, with the atom labels exchanged.

A structuralist reading of the example

The phenomenon depends on the relation between branches, names and distinct witnesses, rather than on the labels of the atoms.

Here both constructions give $M[G] = M$ and use the same atoms. Changing the naming language changes which branch-dependent values admit one uniform name.

Yao extends permutations of a set of atoms to membership automorphisms. The interpretation in this slide is our reading of the example, not a philosophical theorem attributed to Yao.

Yao, Permutations of atoms; Remark 55; Definition 57. Printed pp. 6, 47–48 / PDF pp. 13, 54–55.



Section 4

When does every forced existential have a witness name?

Outline

1 A forcing refresher

2 Why old names fail

3 Yao's repair and mixing

4 **Collection and fullness**

Appendix

Atoms, classes and gathering witnesses

The formula $\mathcal{A}(x)$ defines the class of all atoms. It need not define a set.

- If some set contains all atoms, Separation gives the set of all atoms. That set may be infinite.
- If no set contains them all, the class of atoms is proper, relative to the model.

The gathering question is more specific

For each input x in a set w, suppose some witness y exists. Can one set contain a suitable witness for every x?

A proper class of atoms does not by itself refute this principle. The two-branch example needs only two distinct atoms; the general gathering problem is additional.

Yao, Definition 1; Definition 3 and Proposition 5. Printed pp. 5–6 / PDF pp. 12–13.

Replacement and Collection

For a set w and a fixed parameter u, Yao writes:

Replacement

$$\big(\forall x \in w\ \exists! y\ \varphi(x, y, u)\big) \implies \big(\exists v\ \forall x \in w\ \exists y \in v\ \varphi(x, y, u)\big).$$

Collection

$$\big(\forall x \in w\ \exists y\ \varphi(x, y, u)\big) \implies \big(\exists v\ \forall x \in w\ \exists y \in v\ \varphi(x, y, u)\big).$$

The same set v serves every x. It need not be unique and need not contain all possible witnesses. The formula's remaining parameters stay fixed.

Each displayed implication is universally quantified over w, u, and each admissible formula gives an instance of the scheme.

Yao, §1.2: Replacement at the end of p. 4; Collection at the start of p. 5. Printed pp. 4–5 / PDF pp. 11–12.

A common container for witnesses

Let $w = \{1, 2\}$ and let $\varphi(x, y)$ mean $y = 10x$ or $y = 10x + 1$.

x	possible witnesses	one witness in $v = \{10, 21\}$
1	$10, 11$	10
2	$20, 21$	21

- Every x has a witness, so Collection's antecedent holds.
- Witnesses are not unique, so Replacement's antecedent fails for this formula.
- The displayed container satisfies the common conclusion.

A false antecedent does not make the Replacement implication false. This example illustrates the schemes; it does not separate their strength.

Yao, §1.2; numerical illustration of the two schemes. Printed pp. 4–5 / PDF pp. 11–12.

Which scheme implies which?

Write U for unique witnesses, E for existing witnesses and C for a common witness container.

$$U \Longrightarrow E, \qquad \text{Replacement: } U \Longrightarrow C, \qquad \text{Collection: } E \Longrightarrow C.$$

Collection implies Replacement

Assume U. Then E holds. Collection gives C. Uniqueness was assumed, not inferred from Collection.

In pure ZF, Replacement also proves Collection using bounds on witness ranks. In Yao's setting, Replacement need not prove Collection, even with Choice. With a proper class of atoms, a rank bound alone need not provide a set container.

Yao, Definition 1 and following paragraph; Proposition 5. Printed pp. 5–6 / PDF pp. 12–13.

Fullness for a general forcing poset

Fullness

For each condition p, each formula and each tuple of name parameters,

$$p \Vdash \exists y\ \varphi(y, \vec{\tau}) \quad \Longrightarrow \quad \exists \sigma \in U^{\mathbb{P}}\ p \Vdash \varphi(\sigma, \vec{\tau}).$$

The name σ may depend on p and the formula. The condition p stays fixed.

The existential forcing clause gives witness names densely below p. Fullness asks for one name working already at p.

The repaired two-branch construction proves this one instance directly. It does not establish fullness for arbitrary formulas or posets.

Yao, Definition 58(8); Theorem 61. Printed pp. 49–52 / PDF pp. 56–59.

Yao's general equivalence

The base theory explicitly includes Choice:

$$\mathrm{ZFU}_R = \mathrm{ZU} + \text{Replacement}, \qquad \mathrm{ZFCU}_R = \mathrm{ZFU}_R + \mathrm{AC}.$$

Here ZU is Zermelo theory with atoms and Foundation.

Theorem 61, over ZFCU_R

$$\text{Collection} \iff \begin{array}{c}\text{for every forcing poset } \mathbb{P},\\ \text{the repaired forcing relation is full.}\end{array}$$

The quantifier over *all* posets matters. This is a general theorem about the repaired machinery, separate from the finite example.

Yao, Definition 1; Theorem 61. Printed pp. 5, 50–52 / PDF pp. 12, 57–59.

Collection, Choice and mixing

Suppose $p \Vdash \exists y\ \varphi(y)$, with fixed name parameters.

1. Choice gives a maximal antichain D of conditions below p admitting local witness names.
2. Collection puts enough of those names into one set.
3. Choice selects a name $f(r)$ for each $r \in D$.
4. Mixing combines the given function f into one name μ.
5. Density below p gives $p \Vdash \varphi(\mu)$.

The roles are distinct

Collection gathers names. Choice selects them. Mixing combines them.

For the finite example, the two local names were already explicitly available. The general argument must first obtain a set of names.

Yao, Theorem 61, forward implication. Printed pp. 50–51 / PDF pp. 57–58.

Fullness, truth and preservation

Fullness A forced existential has a witness name under the *same* condition.

Truth lemma Truth in one extension corresponds to forcing by *some* condition in its generic filter:

$$M[G] \models \varphi(\vec{\tau}_G) \iff \exists r \in G \; M \models r \Vdash \varphi(\vec{\tau}).$$

Preservation Axioms holding in the ground model continue to hold in the extension, under the relevant theorem's assumptions.

The old names have a truth lemma despite failing fullness. The repaired forcing theorem over ZFU_R does not assume Choice or Collection.

Yao, Theorems 53, 61 and 66; preservation in §3.2.4. Printed pp. 47, 50–62 / PDF pp. 54, 57–69.

Conclusions and the next question

The old names describe the singleton on either branch, but cannot name its varying atom-member uniformly. Yao's repaired names can do both.

Mixing generalises the repair. Over ZFCU_R, Collection gives exactly the additional gathering strength needed for fullness for all posets.

Source. Bokai Yao, *Set Theory with Urelements*, doctoral dissertation, University of Notre Dame, arXiv v3.

Further work. The Archive of Formal Proofs develops forcing over pure ZFC in Isabelle/ZF. Our proposed finite development would compare the two valuations in the presence of atoms.

Project materials Isabelle project scope



Appendix

Technical details and next steps

Outline

1 A forcing refresher
2 Why old names fail
3 Yao’s repair and mixing
4 Collection and fullness

Appendix

Appendix: the mixing construction

For a given set function $f : D \to U^{\mathbb{P}}$ on an antichain, define

$$\mu = \bigcup_{r \in D} \{\langle y, t\rangle : t \in \mathbb{P},\ \exists s\ (\langle y, s\rangle \in f(r)\ \wedge\ t \leq r, s)\}.$$

Pairs from $f(r)$ are restricted to common strengthenings with their assigned branch r.

- Within one branch, the original name's compatibility condition prevents conflicts.
- Across different branches, the antichain prevents conflicts.
- Under a branch $r \in D$, the resulting name agrees with $f(r)$.

Thus $r \Vdash \mu = f(r)$ for every $r \in D$.

Yao, Lemma 60, proof. Printed pp. 49–50 / PDF pp. 56–57.

Appendix: ranks and available atoms

For a *set* A of atoms, the hierarchy $V_\alpha(A)$ has set-sized levels. Proposition 5 proves restricted Collection: if every input has a witness in $V(A)$, one common set contains sufficient witnesses.

Why Replacement suffices in that restricted situation

Assign each input the least ordinal level containing a witness. Replacement gathers these ordinals; a sufficiently high level gives the container.

For pure ZF use $A = \varnothing$. For a proper class of atoms, there need not be one set A accommodating all required witnesses.

Here $\ker(x)$ is the set of atoms in the transitive closure of $\{x\}$. In the reverse direction of Theorem 61, the witness name's kernel together with the poset's kernel bounds the atoms needed. Proposition 5 gives Collection.

Yao, Definition 2; Proposition 5; Theorem 61, reverse implication. Printed pp. 5–6, 51–52 / PDF pp. 12–13, 58–59.

Appendix: what the extension preserves

For a countable transitive $M \models \mathrm{ZFU}_R$ and an M-generic G over set forcing $\mathbb{P} \in M$:

ground model	extension
ZFU_R	ZFU_R
ZFCU_R	ZFCU_R
$\mathrm{ZFCU}_R + \text{Collection}$	$\mathrm{ZFCU}_R + \text{Collection}$

There are no new atoms or ordinals. New sets of existing atoms may appear. The kernel bound $\ker(\tau_G) \subseteq \ker(\tau)$ limits which atoms a name can use.

These are preservation results, not a reformulation of fullness.

Yao, Lemmas 64–65; Theorems 67, 71 and 73. Printed pp. 53, 56–60 / PDF pp. 60, 63–67.

Appendix: when forcing really adds a set

For comparison, let $\mathbb{P} = 2^{<\omega}$ be finite binary strings, ordered by extension.

$$\langle 0, 1, 0 \rangle \leq \langle 0, 1 \rangle \leq \langle 0 \rangle.$$

For each n, strings of length at least n form a dense set. A generic filter determines

$$c = \bigcup G : \omega \longrightarrow \{0, 1\}.$$

For each ground-model binary sequence r, conditions disagreeing with r somewhere form a dense set in M. Thus $c \neq r$ for every such r.

This is an extension adding a new set. Our terminal two-branch poset was chosen to test the naming language without that additional complication.

Yao, Classical forcing background to §3.2; comparison example. Printed pp. 45–46 / PDF pp. 52–53.

Current work and next steps

A finite Isabelle/HOL exercise checks the two-branch example.

What Isabelle checks

- **Old names.** No single old name takes value a in G_p and b in G_q.
- **Yao's repair.** The selector $w = \{\langle a, p\rangle, \langle b, q\rangle\}$ is admissible, with $w_{G_p} = a$ and $w_{G_q} = b$.

My independent formalisation is ongoing. Yao's general theorem on Collection and fullness remains a reading topic.

Checked teaching exercise Research-code status

LaTeX source Bokai Yao's full thesis

Thank you for your attention.

Student project at the University of Amsterdam and ILLC. Contact the author.



Project page