

# “How-to” systematically compute probabilistic constraints such as optimal Boole-Bell type inequalities by assessing extreme situations

<http://tph.tuwien.ac.at/~svozil/publ/2020-b-pres.pdf>

Karl Svozil

ITP TU Wien, Vienna Austria  
svozil@tuwien.ac.at

Online presentation, summer of 2021

∞ *The big picture:  
strategies & tactics* ∞

# The big picture—Strategies & tactical moves

- S1 Take your observables from quantum mechanical configurations “inspired by” faithful orthogonal representations—as vectors in Hilbert space.

# The big picture—Strategies & tactical moves

- S1 Take your observables from quantum mechanical configurations “inspired by” faithful orthogonal representations—as vectors in Hilbert space.
- S2.1 Force upon these “logics” or propositional structures of observables a *quantum mechanical* interpretation; and make falsifiable predictions.

# The big picture—Strategies & tactical moves

- S1 Take your observables from quantum mechanical configurations “inspired by” faithful orthogonal representations—as vectors in Hilbert space.
- S2.1 Force upon these “logics” or propositional structures of observables a *quantum mechanical* interpretation; and make falsifiable predictions.
- S2.2 If possible, force upon these “logics” or propositional structures of observables a *classical* interpretation as two-valued truth value assignments (independent of co-occurrences: context independence); and make falsifiable predictions.

# The big picture—Strategies & tactical moves

- S1 Take your observables from quantum mechanical configurations “inspired by” faithful orthogonal representations—as vectors in Hilbert space.
- S2.1 Force upon these “logics” or propositional structures of observables a *quantum mechanical* interpretation; and make falsifiable predictions.
- S2.2 If possible, force upon these “logics” or propositional structures of observables a *classical* interpretation as two-valued truth value assignments (independent of co-occurrences: context independence); and make falsifiable predictions.
- S3 Observe discrepancies between quantum and classical predictions (if any) and falsify one or the other (or both ;-).

Tactical issue: since observable discrepancies occur only in cases involving complementarity—involving two or more contexts, aka Boolean subalgebras, maximal (simultaneously co-measurable) observables or blocks, these argument necessitate **counterfactual reasoning** (DOI [j.1746-8361.1960.tb00422.x](https://doi.org/10.1746-8361.1960.tb00422.x), [English translation](#)).

# The big picture—Strategies & **tactical moves**

- T1 Boole-Bell-type inequalities involving isolated/nonintertwined contexts;

# The big picture—Strategies & **tactical moves**

- T1 Boole-Bell-type inequalities involving isolated/nonintertwined contexts;
- T2 intertwined contexts which still have classical interpretations but those are “weird” (Cabello calls them “Hardy-type”);



# The big picture—Strategies & tactical moves

- T1 Boole-Bell-type inequalities involving isolated/nonintertwined contexts;
- T2 intertwined contexts which still have classical interpretations but those are “weird” (Cabello calls them “Hardy-type”);
- T3 intertwined contexts which still have classical interpretations but cannot be embedded faithfully in a classical realm; e.g., nonseparating or nonunital set of two-valued states;

# The big picture—Strategies & tactical moves

- T1 Boole-Bell-type inequalities involving isolated/nonintertwined contexts;
- T2 intertwined contexts which still have classical interpretations but those are “weird” (Cabello calls them “Hardy-type”);
- T3 intertwined contexts which still have classical interpretations but cannot be embedded faithfully in a classical realm; e.g., nonseparating or nonunital set of two-valued states;
- T4 intertwined contexts which fail to have classical interpretations (often called “Kochen-Specker-type”).

Tactical issue: the arbitrariness of the (finite) proofs involved yield inconsistent, contradictory classical predictions. This is due to the fact that classical predictions are contingent on the (finite) configurations involved.

## Tactics with regard to classical probability, subject to existence

TC1 Seek all “**extreme**” events/outcome that can happen/occur, and are **mutually exclusive in every single context**.

# Tactics with regard to classical probability, subject to existence

**TC1** Seek all “**extreme**” events/outcome that can happen/occur, and are **mutually exclusive in every single context**.

**TC2** This amounts to the construction of all two-valued states that assign

**TVS1** only one atom/elementary observable per context the value “1”,

**TVS2** all other/the remaining atoms/elementary observables are assigned the value “0”.

This is straightforward for single and multiple isolated contexts, but less trivial for intertwining ones.

# Tactics with regard to classical probability, subject to existence

**TC1** Seek all “**extreme**” events/outcome that can happen/occur, and are **mutually exclusive in every single context**.

**TC2** This amounts to the construction of all two-valued states that assign

**TVS1** only one atom/elementary observable per context the value “1”,

**TVS2** all other/the remaining atoms/elementary observables are assigned the value “0”.

This is straightforward for single and multiple isolated contexts, but less trivial for intertwining ones.

**TC3** Take a convex combination of all these “extreme” cases and identify these with the respective probability distributions.

## (Greechie-type) orthogonality hypergraphs

For configurations of (multiple) contexts(s) Greechie has proposed a kind of orthogonality hypergraph (DOI [10.1016/0097-3165\(71\)90015-X](https://doi.org/10.1016/0097-3165(71)90015-X) and [10.1007/978-3-319-00080-0](https://doi.org/10.1007/978-3-319-00080-0)):

1. Entire contexts (Boolean subalgebras, blocks) are drawn as smooth lines, such as straight (unbroken) lines, circles or ellipses;

## (Greechie-type) orthogonality hypergraphs

For configurations of (multiple) contexts(s) Greechie has proposed a kind of orthogonality hypergraph (DOI [10.1016/0097-3165\(71\)90015-X](https://doi.org/10.1016/0097-3165(71)90015-X) and [10.1007/978-3-319-00080-0](https://doi.org/10.1007/978-3-319-00080-0)):

1. Entire contexts (Boolean subalgebras, blocks) are drawn as smooth lines, such as straight (unbroken) lines, circles or ellipses;
2. the atomic propositions of the context are drawn as (small) circles overlaying the lines; and

## (Greechie-type) orthogonality hypergraphs

For configurations of (multiple) contexts(s) Greechie has proposed a kind of orthogonality hypergraph (DOI [10.1016/0097-3165\(71\)90015-X](https://doi.org/10.1016/0097-3165(71)90015-X) and [10.1007/978-3-319-00080-0](https://doi.org/10.1007/978-3-319-00080-0)):

1. Entire contexts (Boolean subalgebras, blocks) are drawn as smooth lines, such as straight (unbroken) lines, circles or ellipses;
2. the atomic propositions of the context are drawn as (small) circles overlaying the lines; and
3. contexts intertwining at a single atomic proposition are represented as non-smoothly connected lines, broken at that proposition.



## (Greechie-type) orthogonality hypergraphs

For configurations of (multiple) contexts(s) Greechie has proposed a kind of orthogonality hypergraph (DOI [10.1016/0097-3165\(71\)90015-X](https://doi.org/10.1016/0097-3165(71)90015-X) and [10.1007/978-3-319-00080-0](https://doi.org/10.1007/978-3-319-00080-0)):

1. Entire contexts (Boolean subalgebras, blocks) are drawn as smooth lines, such as straight (unbroken) lines, circles or ellipses;
2. the atomic propositions of the context are drawn as (small) circles overlaying the lines; and
3. contexts intertwining at a single atomic proposition are represented as non-smoothly connected lines, broken at that proposition.

In Hilbert space realizations, the straight lines or smooth curves depicting contexts represent orthogonal bases (or, equivalently, maximal observables, Boolean subalgebras or blocks), and points on these straight lines or smooth curves represent elements of these bases; that is, two points on the same straight line or smooth curve represent two orthogonal basis elements. From dimension three onwards, bases may intertwine in common elements.

∞ *Single context* ∞

# Classical probabilities on a single “classical” Boolean algebra (aka context/block/maximal observable)

We are dealing here with a *finite* number of observables.

The basic idea of classical probability theory are expressed by the Boole-Borel-Kolmogorov axioms; among them the most pertinent (DOI [10.1007/978-3-642-49888-6](https://doi.org/10.1007/978-3-642-49888-6)):

**K III. Nonnegativity:** The probability of an event  $a$  is a **nonnegative** real number  $P(a) \geq 0$ ;

# Classical probabilities on a single “classical” Boolean algebra (aka context/block/maximal observable)

We are dealing here with a *finite* number of observables.

The basic idea of classical probability theory are expressed by the Boole-Borel-Kolmogorov axioms; among them the most pertinent (DOI [10.1007/978-3-642-49888-6](https://doi.org/10.1007/978-3-642-49888-6)):

**K III. Nonnegativity:** The probability of an event  $a$  is a **nonnegative** real number  $P(a) \geq 0$ ;

**K IV. Inclusion/completeness and exclusion:** the probability of certainty, obtained by including all mutually exclusive outcomes (not one is missing), is one:  $P(\mathbb{1}) = 1$ .

The probability of the **excluded** absurdity “nothing happens” is zero:  $P(\emptyset) = 0$ .

# Classical probabilities on a single “classical” Boolean algebra (aka context/block/maximal observable)

We are dealing here with a *finite* number of observables.

The basic idea of classical probability theory are expressed by the Boole-Borel-Kolmogorov axioms; among them the most pertinent (DOI [10.1007/978-3-642-49888-6](https://doi.org/10.1007/978-3-642-49888-6)):

**K III. Nonnegativity:** The probability of an event  $a$  is a **nonnegative** real number  $P(a) \geq 0$ ;

**K IV. Inclusion/completeness and exclusion:** the probability of certainty, obtained by including all mutually exclusive outcomes (not one is missing), is one:  $P(\mathbb{1}) = 1$ .

The probability of the **excluded** absurdity “nothing happens” is zero:  $P(\emptyset) = 0$ .

**K V. Additivity from exclusivity:** the probability of occurrence of (either) one of the two mutually exclusive outcomes  $a$  and  $b$  (with  $a \wedge b = \emptyset$ ) is the sum of the probabilities; that is,  $P(a \vee b) = P(a) + P(b)$ . This generalizes to an arbitrary number of mutually exclusive outcomes. “Mutual exclusivity” is traditionally termed “incompatibility” but we reserve the latter term “incompatibility” to observables in different contexts).

# Classical probability distributions on a single “classical” context/Boolean algebra

There are  $n$  individual atomic “mutually exclusive” outcomes/events  $a_i$  associated with the Boolean algebra  $2^n$ .

# Classical probability distributions on a single “classical” context/Boolean algebra

There are  $n$  individual atomic “mutually exclusive” outcomes/events  $a_i$  associated with the Boolean algebra  $2^n$ .

## Nomenclature:

In what follows **contexts**/Boolean algebras will be written in terms of the union  $\mathcal{C} = \{a_1, \dots, a_n\}$  of their  $n$  “elementary” indecomposable atoms representing such individual atomic outcomes/events.

# Classical probability distributions on a single “classical” context/Boolean algebra

There are  $n$  individual atomic “mutually exclusive” outcomes/events  $a_i$  associated with the Boolean algebra  $2^n$ .

## Nomenclature:

In what follows **contexts**/Boolean algebras will be written in terms of the union  $\mathcal{C} = \{a_1, \dots, a_n\}$  of their  $n$  “elementary” indecomposable atoms representing such individual atomic outcomes/events.

## Note on expectations $\longleftrightarrow$ probabilities:

Expectation values  $E$  of dichotomic outcomes  $\in \{-1, 1\}$  are only an affine transformation—an addition followed by a squeeze—“away” from probabilities  $P$  of the occurrence of “1”, since

$$E = 2P - 1, \text{ or conversely, } P = \frac{1}{2}(E + 1).$$



## Note on expectations $\longleftrightarrow$ probabilities cntd.

In quantum mechanics and Hilbert spaces of Dimension greater than one, this generalizes to **Householder transformations**

$$\mathbf{U}_{\mathbf{x}} = \mathbb{1} - 2\mathbf{x}\mathbf{x}^\dagger,$$

where  $|\mathbf{x}\rangle$  is a unit vector.

Eigensystem of  $\mathbf{U}_{\mathbf{x}}$  with eigenvalues  $\pm 1$ :

−1:  $|\mathbf{x}\rangle$  is an eigenvector of  $\mathbf{U}_{\mathbf{x}}$  with eigenvalue  $-1$ .

+1: The remaining  $n - 1$  mutually orthogonal eigenvectors span the  $n - 1$  dimensional orthogonal subspace of  $|\mathbf{x}\rangle$ . Every vector in that subspace has eigenvalue  $+1$ . (For  $n > 2$  the spectrum is degenerate.)

Moreover, for any “complete” context represented by some orthonormal basis  $\mathfrak{B} = \{|\mathbf{e}_1\rangle, |\mathbf{e}_2\rangle, \dots, |\mathbf{e}_n\rangle\}$ ,

$$\mathbf{U}_{\mathbf{e}_1}\mathbf{U}_{\mathbf{e}_2}\cdots\mathbf{U}_{\mathbf{e}_n} = -\mathbb{1}.$$

## Classical probability distributions on a single “classical” context/Boolean algebra cntd.

- (1) **Truth assignments:** every experimental run only “renders” a *single* atomic outcome/event  $a_i$ —that is,  $a_i$  “occurs”—all other such possible outcomes/events  $a_j \in \mathcal{C} - a_i$  “do not occur”. (Think of an ideal  $n$ -port beam splitter.)

# Classical probability distributions on a single “classical” context/Boolean algebra cntd.

- (1) **Truth assignments:** every experimental run only “renders” a *single* atomic outcome/event  $a_i$ —that is,  $a_i$  “occurs”—all other such possible outcomes/events  $a_j \in \mathcal{C} - a_i$  “do not occur”. (Think of an ideal  $n$ -port beam splitter.)
- (2) **Encoding of truth assignments by two-valued state:**
  - (2.1) Define a binary two-valued state as a function  $v$  of the elements of the context/Boolean algebra such that  $v(a_i) = 1$  if the event/outcome  $a_i$  occurs, otherwise it vanishes.  
Check: There are  $n$  different two-valued states  $v_1, \dots, v_n$  on a context  $\mathcal{C}$ , namely  $v_i(a_j) = \delta_{ij}$  (modulo permutations).

# Classical probability distributions on a single “classical” context/Boolean algebra cntd.

- (1) **Truth assignments:** every experimental run only “renders” a *single* atomic outcome/event  $a_i$ —that is,  $a_i$  “occurs”—all other such possible outcomes/events  $a_j \in \mathcal{C} - a_i$  “do not occur”. (Think of an ideal  $n$ -port beam splitter.)
- (2) **Encoding of truth assignments by two-valued state:**
  - (2.1) Define a binary two-valued state as a function  $v$  of the elements of the context/Boolean algebra such that  $v(a_i) = 1$  if the event/outcome  $a_i$  occurs, otherwise it vanishes.  
Check: There are  $n$  different two-valued states  $v_1, \dots, v_n$  on a context  $\mathcal{C}$ , namely  $v_i(a_j) = \delta_{ij}$  (modulo permutations).
  - (2.2) **Convexity:** A general classical probability distribution  $P(a_i)$  on a context  $\mathcal{C}$  can be written as a convex sum of two-valued states:

$$P(x) = \lambda_1 v_1(x) + \dots + \lambda_n v_n(x),$$

with  $\lambda_1 + \dots + \lambda_n = 1$  and  $\lambda_j \in [0, 1]$ ,  $j \in \{1, \dots, n\}$ .

# One context “containing” two atoms, two-valued states and probability distributions

Two two-valued states, represented by the Travis matrix (Raymond David Travis, “The Logic of a Physical Theory”, Master’s Thesis under the supervision of David J. Foulis, Wayne State University, 1962):

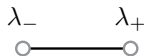
$\nu_1$ :     $\begin{array}{cc} - & + \\ \bullet & \circ \end{array}$

$\nu_2$ :     $\begin{array}{cc} - & + \\ \circ & \bullet \end{array}$

$T$	$-$	$+$
$\nu_1$	1	0
$\nu_2$	0	1

Probability distributions: subject to  $\lambda_-, \lambda_+ \geq 0$  and

$$\lambda_- + \lambda_+ = 1$$



∞ *Multiple nonintertwining contexts* ∞

# Classical probabilities on multiple yet isolated/nonintertwining contexts

From this situation on—by contemplating more than one of finitely many contexts/Boolean subalgebras/maximal observables/blocks—we are dealing with **complementarity**.

(1) **Isolation principle—autonomous probabilities on isolated contexts:**

In the case of isolated/nonintertwining—no common element(s)—contexts every context is treated separately, as isolated entity. Therefore, in particular, probability distributions can be treated as isolated entities.

# Classical probabilities on multiple yet isolated/nonintertwining contexts

From this situation on—by contemplating more than one of finitely many contexts/Boolean subalgebras/maximal observables/blocks—we are dealing with **complementarity**.

(1) **Isolation principle—autonomous probabilities on isolated contexts:**

In the case of isolated/nonintertwining—no common element(s)—contexts every context is treated separately, as isolated entity. Therefore, in particular, probability distributions can be treated as isolated entities.

(2) **Multiplicativity of joint probabilities of events from different contexts:**

The joint probabilities and expectations are treated multiplicatively.

In particular, for two contexts  $\mathcal{C}_1$  and  $\mathcal{C}_2$ , in order to obtain all joint two-valued states  $w$ , their respective two-valued states  $u$  and  $v$  can be multiplied:

$$v_{ij}(a_s \wedge b_t) \equiv v_{ij}^{1,2}(a_s, b_t) = v_i^1(a_s)v_j^2(b_t),$$

with  $a_s \in \mathcal{C}_1$ ,  $b_t \in \mathcal{C}_2$ , and  $v_i^1$  and  $v_j^2$  are two-valued states on those two contexts, respectively.

This can be understood either as consequence of the isolation principle, or in terms of the product of two-valued states on multiple isolated contexts as follows.



## Classical probabilities on multiple yet isolated/nonintertwining contexts cntd.

- (1) **Truth assignments:** per context every experimental run only “renders” a *single* atomic outcome/event. All other possible outcomes/events in these contexts “do not occur”.

## Classical probabilities on multiple yet isolated/nonintertwining contexts cntd.

- (1) **Truth assignments:** per context every experimental run only “renders” a *single* atomic outcome/event. All other possible outcomes/events in these contexts “do not occur”.
- (2) **Encoding of truth assignments by two-valued state:**
  - (2.1) Define a binary two-valued state as a function  $v$  on the elements of the contexts/Boolean algebras such that  $v(a_i) = 1$  if the event/outcome  $a_i$  occurs, otherwise the state  $v$  vanishes. Suppose an exhaustive search rendered  $n$  such possible two-valued states  $v_1, \dots, v_n$ .

## Classical probabilities on multiple yet isolated/nonintertwining contexts cntd.

- (1) **Truth assignments:** per context every experimental run only “renders” a *single* atomic outcome/event. All other possible outcomes/events in these contexts “do not occur”.
- (2) **Encoding of truth assignments by two-valued state:**
  - (2.1) Define a binary two-valued state as a function  $v$  on the elements of the contexts/Boolean algebras such that  $v(a_i) = 1$  if the event/outcome  $a_i$  occurs, otherwise the state  $v$  vanishes. Suppose an exhaustive search rendered  $n$  such possible two-valued states  $v_1, \dots, v_n$ .
  - (2.2) **Convexity:** A general classical probability distribution  $P$  on the contexts  $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_m$  can be written as a convex sum of two-valued states: let  $x \in \bigcup_{i=1}^m \mathcal{C}_i$

$$P(x) = \lambda_1 v_1(x) + \dots + \lambda_n v_n(x),$$

with  $\lambda_1 + \dots + \lambda_n = 1$  and  $\lambda_j \in [0, 1]$ ,  $j \in \{1, \dots, n\}$ .

# Example: Two isolated/nonintertwining contexts “containing” two atoms each, two-valued states and probability distributions

Two-valued states:

$$\begin{aligned}
 v_1: & \quad \overset{x_-}{\bullet} \overset{x_+}{\circ} \times \overset{y_-}{\bullet} \overset{y_+}{\circ} = \overset{x_-y_-}{\bullet} \overset{x_-y_+}{\circ} \overset{x_+y_-}{\circ} \overset{x_+y_+}{\circ} \\
 v_2: & \quad \overset{x_-}{\bullet} \overset{x_+}{\circ} \times \overset{y_-}{\circ} \overset{y_+}{\bullet} = \overset{x_-y_-}{\circ} \overset{x_-y_+}{\bullet} \overset{x_+y_-}{\circ} \overset{x_+y_+}{\circ} \\
 v_3: & \quad \overset{x_-}{\circ} \overset{x_+}{\bullet} \times \overset{y_-}{\bullet} \overset{y_+}{\circ} = \overset{x_-y_-}{\circ} \overset{x_-y_+}{\circ} \overset{x_+y_-}{\bullet} \overset{x_+y_+}{\circ} \\
 v_4: & \quad \overset{x_-}{\circ} \overset{x_+}{\bullet} \times \overset{y_-}{\circ} \overset{y_+}{\bullet} = \overset{x_-y_-}{\circ} \overset{x_-y_+}{\circ} \overset{x_+y_-}{\circ} \overset{x_+y_+}{\bullet}
 \end{aligned}$$

Probability distributions:  $\lambda_{x_-y_-}, \lambda_{x_-y_+}, \lambda_{x_+y_-}, \lambda_{x_+y_+} \geq 0$  and

$$\lambda_{x_-y_-} + \lambda_{x_-y_+} + \lambda_{x_+y_-} + \lambda_{x_+y_+} = 1$$

$T$	$x_-y_-$	$x_-y_+$	$x_+y_-$	$x_+y_+$
$v_1$	1	0	0	0
$v_2$	0	1	0	0
$v_3$	0	0	1	0
$v_4$	0	0	0	1



∞ *Multiple intertwining contexts* ∞

# Classical probabilities on multiple intertwining Boolean contexts

In these configurations we are still assuming that all pairs of atomic, elementary propositions formalized by unit vectors/orthogonal projection operators/one-dimensional subspaces are mutually *separable* by some two-valued measure; cf. the Kochen-Specker demarcation criterion Theorem 0, DOI [10.1512/iumj.1957.6.56050](https://doi.org/10.1512/iumj.1957.6.56050).

- (1) **Computing extreme states satisfying exclusivity:** Compute all two-valued states on the propositional structure.

# Classical probabilities on multiple intertwining Boolean contexts

In these configurations we are still assuming that all pairs of atomic, elementary propositions formalized by unit vectors/orthogonal projection operators/one-dimensional subspaces are mutually *separable* by some two-valued measure; cf. the Kochen-Specker demarcation criterion Theorem 0, DOI [10.1512/iumj.1957.6.56050](https://doi.org/10.1512/iumj.1957.6.56050).

- (1) **Computing extreme states satisfying exclusivity:** Compute all two-valued states on the propositional structure.
- (2) **Convex combination:** the probability distributions on a particular atomic, elementary proposition are convex combinations of the respective nonzero two-valued states on them. The convex sum of all the weights is normalized to 1.

## Classical probabilities on multiple intertwining Boolean contexts cntd.

- (1) **Truth assignments:** per context every experimental run only “renders” a *single* atomic outcome/event. All other possible outcomes/events in these contexts “do not occur”.



# Classical probabilities on multiple intertwining Boolean contexts cntd.

- (1) **Truth assignments:** per context every experimental run only “renders” a *single* atomic outcome/event. All other possible outcomes/events in these contexts “do not occur”.
- (2) **Encoding of truth assignments by two-valued state:**
  - (2.1) Define a binary two-valued state as a function  $v$  on the elements of the contexts/Boolean algebras such that  $v(a_i) = 1$  if the event/outcome  $a_i$  occurs, otherwise the state  $v$  vanishes. Suppose an exhaustive search rendered  $n$  such possible two-valued states  $v_1, \dots, v_n$ .

# Classical probabilities on multiple intertwining Boolean contexts cntd.

- (1) **Truth assignments:** per context every experimental run only “renders” a *single* atomic outcome/event. All other possible outcomes/events in these contexts “do not occur”.
- (2) **Encoding of truth assignments by two-valued state:**
  - (2.1) Define a binary two-valued state as a function  $v$  on the elements of the contexts/Boolean algebras such that  $v(a_i) = 1$  if the event/outcome  $a_i$  occurs, otherwise the state  $v$  vanishes. Suppose an exhaustive search rendered  $n$  such possible two-valued states  $v_1, \dots, v_n$ .
  - (2.2) **Convexity:** A general classical probability distribution  $P$  on the contexts  $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_m$  can be written as a convex sum of two-valued states: let  $x \in \bigcup_{i=1}^m \mathcal{C}_i$

$$P(x) = \lambda_1 v_1(x) + \dots + \lambda_n v_n(x),$$

with  $\lambda_1 + \dots + \lambda_n = 1$  and  $\lambda_j \in [0, 1]$ ,  $j \in \{1, \dots, n\}$ .

## Detour: a revisionist history of classical probabilities on intertwining contexts

- Q Cabello, in a **historic review @ 30m27s** of [https://youtu.be/5r6J-PQVo\\_s](https://youtu.be/5r6J-PQVo_s), states that: “... *it's a pity that Pitowsky who had all the background needed to go further, stopped by just noticing that Bell inequalities are a subset of the Boole inequalities.*”

## Detour: a revisionist history of classical probabilities on intertwining contexts

- Q Cabello, in a **historic review @ 30m27s** of [https://youtu.be/5r6J-PQVo\\_s](https://youtu.be/5r6J-PQVo_s), states that: “... *it's a pity that Pitowsky who had all the background needed to go further, stopped by just noticing that Bell inequalities are a subset of the Boole inequalities.*”
- R My response to that:
- R1 Pitowsky turned his attention to the (arguably) much stronger result of Kochen & Specker, and **indeterminacy**.

## Detour: a revisionist history of classical probabilities on intertwining contexts

Q Cabello, in a [historic review @ 30m27s of https://youtu.be/5r6J-PQVo\\_s](https://youtu.be/5r6J-PQVo_s), states that: “. . . *it's a pity that Pitowsky who had all the background needed to go further, stopped by just noticing that Bell inequalities are a subset of the Boole inequalities.*”

R My response to that:

R1 Pitowsky turned his attention to the (arguably) much stronger result of Kochen & Specker, and [indeterminacy](#).

R2 Why should one do this from a pragmatic point of view? Arguments involving intertwining contexts needed to study qudit or qudit systems with Hilbert spaces of dimension three and higher (no intertwined observables in  $d = 2$ ) that are difficult to operationalize even to this day.

## Detour: a revisionist history of classical probabilities on intertwining contexts

Q Cabello, in a [historic review @ 30m27s of https://youtu.be/5r6J-PQVo\\_s](https://youtu.be/5r6J-PQVo_s), states that: “. . . *it's a pity that Pitowsky who had all the background needed to go further, stopped by just noticing that Bell inequalities are a subset of the Boole inequalities.*”

R My response to that:

- R1 Pitowsky turned his attention to the (arguably) much stronger result of Kochen & Specker, and [indeterminacy](#).
- R2 Why should one do this from a pragmatic point of view? Arguments involving intertwining contexts needed to study qudit or qudit systems with Hilbert spaces of dimension three and higher (no intertwined observables in  $d = 2$ ) that are difficult to operationalize even to this day.
- R3 Indeed, on what grounds is it better to certify differences between quantum and classical predictions from counterfactual constructions of complementary intertwined rather than isolated contexts? Why is it not even sufficient for this purpose to consider the quantitative difference of a single quantum versus a classical expectation value? Why involve three (Suppes Zanotti) or four (CHSH) such expectation values?

## Detour to a revisionist history of classical probabilities on intertwining contexts cntd.

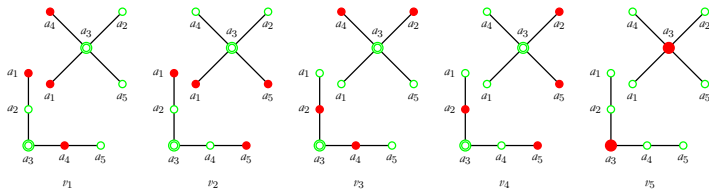
Trigger warning of self-promotion: nevertheless—because I fiddled around with partition logics—my 2001 arXiv paper “On generalized probabilities: correlation polytopes for automaton logic and generalized urn models . . .” @ [arXiv:quant-ph/0012066](https://arxiv.org/abs/quant-ph/0012066) considers the hull problem for the “Specker bug” (a true-implies-false gadget, see later discussion) and similar intertwining propositional structures already in its “modern form” via the Travis matrix. The paper was written for the Fifth Biannual IQSA Conference “Quantum Structures’ 2000” in Cesena, Italy, March 31 – April 5, 2001, but for personal reasons I never made it to Cesena. I even submitted it (to Greechie) for the proceedings but never heard of it afterwards.

I started contemplating generalized hull problems for intertwining contexts with the 2nd version of the paper stating that “*The (nonclassical) correlation polytope corresponding to  $L$  can be defined as the convex hull of all two-valued states thereon.*” Version 3 contains a hull computation by using a Viennese package bei Kreuzer and Skarke.

But, unlike the famous 2008 [Klyachko, Can, Binicioğlu and Shumovsky paper DOI: 10.1103/PhysRevLett.101.020403](https://arxiv.org/abs/quant-ph/0609088) I did not consider actual violations of the hull inequalities by quantum predictions.

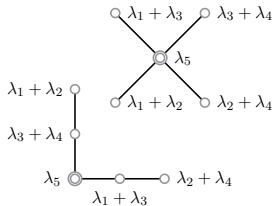
# Classical probabilities on two intertwining Boolean contexts with three atoms each (“two different tripods with one common leg”)

Two-valued states (two representations by hypergraphs, one meaning):



Probability distributions:  $\lambda_i \geq 0$  for  $i \in \{1, \dots, 5\}$  and  $\sum_{i=1}^5 \lambda_i = 1$ .

$T$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
$v_1$	1	0	0	1	0
$v_2$	1	0	0	0	1
$v_3$	0	1	0	1	0
$v_4$	0	1	0	0	1
$v_5$	0	0	1	0	0





❧ Boole's “*Conditions of Possible Experience*” ❧

# Boole's “Conditions of Possible Experience”

Geometric (re)interpretation/representation of probability distributions as convex polytope by Froissart DOI [10.1007/BF02903286](#), Garg & Mermin DOI [10.1007/BF00741645](#), Pitowsky DOI [10.1063/1.527066](#), [10.1007/BFb0021186](#), Tsirelson and others.

1. Form a “bouquet” of  $n$  “relevant” (that is up to you what you include/exclude) combinations of (joint) dichotomic elementary outcomes/events. Arrange these as  $n$ -tuples and consider them as “row vectors” wrt some orthonormal basis.

# Boole's “Conditions of Possible Experience”

Geometric (re)interpretation/representation of probability distributions as convex polytope by Froissart DOI [10.1007/BF02903286](#), Garg & Mermin DOI [10.1007/BF00741645](#), Pitowsky DOI [10.1063/1.527066](#), [10.1007/BFb0021186](#), Tsirelson and others.

1. Form a “bouquet” of  $n$  “relevant” (that is up to you what you include/exclude) combinations of (joint) dichotomic elementary outcomes/events. Arrange these as  $n$ -tuples and consider them as “row vectors” wrt some orthonormal basis.
2. Suppose there are  $k$  two-valued states on the pertinent contexts. Apply these valuations to the chosen  $n$  combinations of (joint) dichotomic elementary outcomes/events, and form  $k$  row vectors from the resulting  $n$ -tuples.

# Boole's "Conditions of Possible Experience"

Geometric (re)interpretation/representation of probability distributions as convex polytope by Froissart DOI [10.1007/BF02903286](#), Garg & Mermin DOI [10.1007/BF00741645](#), Pitowsky DOI [10.1063/1.527066](#), [10.1007/BFb0021186](#), Tsirelson and others.

1. Form a "bouquet" of  $n$  "relevant" (that is up to you what you include/exclude) combinations of (joint) dichotomic elementary outcomes/events. Arrange these as  $n$ -tuples and consider them as "row vectors" wrt some orthonormal basis.
2. Suppose there are  $k$  two-valued states on the pertinent contexts. Apply these valuations to the chosen  $n$  combinations of (joint) dichotomic elementary outcomes/events, and form  $k$  row vectors from the resulting  $n$ -tuples.
3. Interpret the  $n$ -tuples as vectors in an  $n$ -dimensional (real or complex) Hilbert space.  
"Stack" these row vectors formed in 2. on top of each other. The resulting matrix is a (kind of) Travis matrix.

## Boole's "Conditions of Possible Experience" cntd.

4. Consider the convex polytope formed by interpreting the  $k$   $n$ -tuples as *vertices* of the polytope.

## Boole's “Conditions of Possible Experience” cntd.

4. Consider the convex polytope formed by interpreting the  $k$   $n$ -tuples as *vertices* of the polytope.
5. According to the Farkas-Minkowski-Weyl “main” representation theorem (DOI [10.1007/978-1-4613-8431-1](https://doi.org/10.1007/978-1-4613-8431-1)) this convex polytope (aka convex cone or just cone) has an equivalent representation as
  - 5.1. its *vertices* (aka the cone is finitely generated; cf. Alexander Schrijver, “Theory of Linear and Integer Programming”, Wiley, 1986); as well as
  - 5.2. its *facets* or intersecting half-spaces (aka the cone is polyhedral) obtained by the *hull computation*.

## Boole's “Conditions of Possible Experience” cntd.

4. Consider the convex polytope formed by interpreting the  $k$   $n$ -tuples as *vertices* of the polytope.
5. According to the Farkas-Minkowski-Weyl “main” representation theorem (DOI [10.1007/978-1-4613-8431-1](https://doi.org/10.1007/978-1-4613-8431-1)) this convex polytope (aka convex cone or just cone) has an equivalent representation as
  - 5.1. its *vertices* (aka the cone is finitely generated; cf. Alexander Schrijver, “Theory of Linear and Integer Programming”, Wiley, 1986); as well as
  - 5.2. its *facets* or intersecting half-spaces (aka the cone is polyhedral) obtained by the *hull computation*.
6. These facet (in)equalities represent Boole-Bell type “conditions of possible (classical) experience”.

If one is dealing with expectation values then instead of the two-valued states the (affine) transformed expectation values have to be inserted for the Travis matrix.

# Programs to compute optimal Boole-Bell type inequalities (aka to solve the hull problem)

Komei Fukuda's “**cddlib** is an implementation of the Double Description Method [[. . . DOI [10.1007/3-540-61576-8\\_77](https://doi.org/10.1007/3-540-61576-8_77) . . .]] for generating all vertices (i.e. extreme points) and extreme rays of a general convex polyhedron given by a system of linear inequalities.

The program also supports the reverse operation (i.e. convex hull computation). This means that one can move back and forth between an inequality representation and a generator (i.e. vertex and ray) representation of a polyhedron with cdd. Also, it can solve a linear programming problem, i.e. a problem of maximizing and minimizing a linear function over a polyhedron.” (Quote from Matthias Troffaes’ <https://pypi.org/project/pycddlib/>)

An alternative package is **lrs** by David Avis.



## Two possible ways to access **cddlib**

There are two possible ways to access Komei Fukuda's **cddlib**:

1. by “direct” installation/compilation: go to [https://people.inf.ethz.ch/fukudak/cdd\\_home/](https://people.inf.ethz.ch/fukudak/cdd_home/) or, in particular, to <https://github.com/cddlib/cddlib> and don't forget to install **GMP**, a library for arbitrary precision arithmetic;

## Two possible ways to access **cddlib**

There are two possible ways to access Komei Fukuda's **cddlib**:

1. by “direct” installation/compilation: go to [https://people.inf.ethz.ch/fukudak/cdd\\_home/](https://people.inf.ethz.ch/fukudak/cdd_home/) or, in particular, to <https://github.com/cddlib/cddlib> and don't forget to install **GMP**, a library for arbitrary precision arithmetic;
2. by Matthias Troffaes' **pycddlib**, a Python wrapper for **cddlib**.

In what follows **pycddlib**, the second path to **cddlib**, will be implemented.

# How-to install **pycddlib**, a Python wrapper for **cddlib**

1. Go to <https://www.python.org/> or get Python “from the store”. Download & install Python.

# How-to install **pycddlib**, a Python wrapper for **cddlib**

1. Go to <https://www.python.org/> or get Python “from the store”. Download & install Python.
- 1.1. Check if Python is installed correctly; in particular, if environment “path” variables are set by typing from a command shell: **python**.

# How-to install **pycddlib**, a Python wrapper for **cddlib**

1. Go to <https://www.python.org/> or get Python “from the store”. Download & install Python.
- 1.1. Check if Python is installed correctly; in particular, if environment “path” variables are set by typing from a command shell: **python**. If you see three “arrow prompts” **>>>** then all is well, and we can exit the Python shell with **exit()**.

# How-to install **pycddlib**, a Python wrapper for **cddlib**

1. Go to <https://www.python.org/> or get Python “from the store”. Download & install Python.
- 1.1. Check if Python is installed correctly; in particular, if environment “path” variables are set by typing from a command shell: **python**. If you see three “arrow prompts” **>>>** then all is well, and we can exit the Python shell with **exit()**.
- 1.2 Check if “pip” is installed correctly by typing from a command shell: **pip**.

# How-to install **pycddlib**, a Python wrapper for **cddlib**

1. Go to <https://www.python.org/> or get Python “from the store”. Download & install Python.
- 1.1. Check if Python is installed correctly; in particular, if environment “path” variables are set by typing from a command shell: **python**. If you see three “arrow prompts” **>>>** then all is well, and we can exit the Python shell with **exit()**.
- 1.2 Check if “pip” is installed correctly by typing from a command shell: **pip**.
2. Go to <https://pypi.org/project/pycddlib/> to inform yourself of the package (optional) and

# How-to install **pycddlib**, a Python wrapper for **cddlib**

1. Go to <https://www.python.org/> or get Python “from the store”. Download & install Python.
  - 1.1. Check if Python is installed correctly; in particular, if environment “path” variables are set by typing from a command shell: **python**. If you see three “arrow prompts” **>>>** then all is well, and we can exit the Python shell with **exit()**.
  - 1.2 Check if “pip” is installed correctly by typing from a command shell: **pip**.
2. Go to <https://pypi.org/project/pycddlib/> to inform yourself of the package (optional) and
  - 2.1. from a command shell type **pip install pycddlib**



# How-to install **pycddlib**, a Python wrapper for **cddlib**

1. Go to <https://www.python.org/> or get Python “from the store”. Download & install Python.
  - 1.1. Check if Python is installed correctly; in particular, if environment “path” variables are set by typing from a command shell: **python**. If you see three “arrow prompts” **>>>** then all is well, and we can exit the Python shell with **exit()**.
  - 1.2 Check if “pip” is installed correctly by typing from a command shell: **pip**.
2. Go to <https://pypi.org/project/pycddlib/> to inform yourself of the package (optional) and
  - 2.1. from a command shell type **pip install pycddlib**
  - 2.2. we should see something like

... Successfully installed pycddlib-2.1.4

With this we should be ready to go!

# How-to use **pycddlib** for hull computations

For a documentation of how to implement polyhedron computations see [https:](https://pycddlib.readthedocs.io/en/latest/polyhedron.html)

[//pycddlib.readthedocs.io/en/latest/polyhedron.html](https://pycddlib.readthedocs.io/en/latest/polyhedron.html).

In this slide I am mainly quoting from there.

1. `import cdd` imports the package into Python.

# How-to use **pycddlib** for hull computations

For a documentation of how to implement polyhedron computations see [https:](https://pycddlib.readthedocs.io/en/latest/polyhedron.html)

[//pycddlib.readthedocs.io/en/latest/polyhedron.html](https://pycddlib.readthedocs.io/en/latest/polyhedron.html).

In this slide I am mainly quoting from there.

1. `import cdd` imports the package into Python.
2. `mat = cdd.Matrix([1, T])` defines Travis matrix with an appended column of “1s”—that is, the stacked matrix, represented by  $[[1, v_1], \dots, [1, v_k]]$ , of  $k$  vertex row vectors ( $n$ -tuples)  $v_1, \dots, v_k$  containing the values of the  $k$  two-valued states on all  $n$  “included” observables (e.g., elementary propositions).

# How-to use **pycddlib** for hull computations

For a documentation of how to implement polyhedron computations see [https:](https://pycddlib.readthedocs.io/en/latest/polyhedron.html)

[//pycddlib.readthedocs.io/en/latest/polyhedron.html](https://pycddlib.readthedocs.io/en/latest/polyhedron.html).

In this slide I am mainly quoting from there.

1. `import cdd` imports the package into Python.
2. `mat = cdd.Matrix([1, T])` defines Travis matrix with an appended column of “1s”—that is, the stacked matrix, represented by  $[[1, v_1], \dots, [1, v_k]]$ , of  $k$  vertex row vectors ( $n$ -tuples)  $v_1, \dots, v_k$  containing the values of the  $k$  two-valued states on all  $n$  “included” observables (e.g., elementary propositions).
3. `poly = cdd.Polyhedron(mat)`: For a polyhedron described as  $\text{poly} = \text{conv}(v_1, \dots, v_n)$  the vertex  $V$ -representation matrix is  $[1, T]$ , where  $\mathbf{1} = \underbrace{(1, \dots, 1)^T}_{k \text{ times}}$  ( $T$  stands for transposition) is the column vector with  $k$  ones, and  $T$  is the Travis matrix.

## How-to use **pycddlib** for hull computations cntd.

4. `ine = poly.get_inequalities()`: For a polyhedron described as  $\text{poly} = \{x \mid A \cdot x \leq b\}$ , the hull  $H$ -representation is the matrix  $[b, -A]$ .  
Thereby,  $x = \underbrace{(x_1, \dots, x_k)}_{k \text{ weights}}^T$  stands for the respective weights of the “included” atoms (e.g., elementary propositions).

## How-to use **pycddlib** for hull computations cntd.

4. **ine = poly.get\_inequalities()**: For a polyhedron described as  $\text{poly} = \{x \mid A \cdot x \leq b\}$ , the hull  $H$ -representation is the matrix  $[b, -A]$ .

Thereby,  $x = \underbrace{(x_1, \dots, x_k)}_{k \text{ weights}}^T$  stands for the respective weights of the “included” atoms (e.g., elementary propositions).

5. **print(ine)** prints out the results in terms of  $[b, -A]$ .  
One could say the  $[b, -A]$  is a sort of “inverted Travis matrix” and, by the Farkas-Minkowski-Weyl theorem, is equivalent to the Travis matrix.

```
f = open('name_of_outputfile', 'w')
print(ine, file=f)
f.close()
```

writes the result to **name\_of\_outputfile**.



*Instances and example computations I:  
single context''*



## Example hull computation: A single isolated context with two mutually exclusive binary observables

From now on, **color marked entries** are from the Travis matrix.

input

```
import cdd
mat = cdd.Matrix([
  [1, 1, 0],
  [1, 0, 1]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()

print(ine)
```

output

```
H - representation
linearity 1 3
begin
  3 3 rational
  1 - 1 0       $\Rightarrow P_- \leq 1$ 
  0 1 0       $\Rightarrow -P_- \leq 0 \Rightarrow P_- \geq 0$ 
  -1 1 1       $\Rightarrow -P_- - P_+ = -1 \Rightarrow P_- + P_+ = 1$ 
end
```

“**linearity 1 3**” means that one line—namely line three—should be interpreted as equality. This is consistent with the identifications  $P_- = \lambda_-$ ,  $P_+ = \lambda_+$ , and  $\lambda_-, \lambda_+ \geq 0$  and  $\lambda_- + \lambda_+ = 1$ , as mentioned earlier.



*Instances and example computations II:  
isolated/nonintertwining contexts*

## Example hull computation: Two isolated/nonintertwining contexts with two mutually exclusive binary observables

Cf. Pitowsky, Section 2.1 of DOI [10.1007/BFb0021186](https://doi.org/10.1007/BFb0021186).

Suppose the binary observables take on the values 0 and 1—that is,  $O_1, O_2 \in \{0, 1\}$ —and consider the “bouquet” of (joint) observables  $\{O_1, O_2, O_1 \cdot O_2\}$ . The dot “ $\cdot$ ” in “ $a \cdot b$ ” represents scalar multiplication of  $a$  and  $b$ . From now on it will be (mostly ;- ) ommitted.

Entries of the Travis matrix are obtained from all combinations of values as

	$O_1$	$O_2$	$O_1 O_2$
$v_1$	0	0	$0 \cdot 0 = 0$
$v_2$	0	1	$0 \cdot 1 = 0$
$v_3$	1	0	$1 \cdot 0 = 0$
$v_4$	1	1	$1 \cdot 1 = 1$

## Example hull computation: Two isolated/nonintertwining contexts with two mutually exclusive binary observables cntd.

input

```
import cdd
mat = cdd.Matrix([
  [1, 0, 0, 0],
  [1, 0, 1, 0],
  [1, 1, 0, 0],
  [1, 1, 1, 1]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()

print(ine)
```

output

```
H - representation
begin
  4 4 rational
  1 - 1 - 1 1     $\Rightarrow P_{O_1=1} + P_{O_2=1} - P_{(O_1=1) \wedge (O_2=1)} \leq 1$ 
  0 1 0 - 1      $\Rightarrow -P_{O_1=1} + P_{(O_1=1) \wedge (O_2=1)} \leq 0 \Rightarrow P_{(O_1=1) \wedge (O_2=1)} \leq P_{O_1=1}$ 
  0 0 1 - 1      $\Rightarrow -P_{O_2=1} + P_{(O_1=1) \wedge (O_2=1)} \leq 0 \Rightarrow P_{(O_1=1) \wedge (O_2=1)} \leq P_{O_2=1}$ 
  0 0 0 1        $\Rightarrow -P_{(O_1=1) \wedge (O_2=1)} \leq 0 \Rightarrow P_{(O_1=1) \wedge (O_2=1)} \geq 0$ 
end
```

Nothing exciting here!

## Example hull computation: Two isolated/nonintertwining contexts with two mutually exclusive binary observables

Same as before, but suppose the binary observables take on the “affine transformed” values  $-1$  and  $1$ —that is,  $E_1, E_2 \in \{-1, 1\}$ —and consider the “bouquet” of (joint) observables  $\{E_1, E_2, E_{12} = E_1 \cdot E_2\}$ . Entries of the Travis matrix are obtained from all combinations of values as

	$E_1$	$E_2$	$E_{12}$
$v_1$	-1	-1	$-1 \cdot (-1) = 1$
$v_2$	-1	1	$-1 \cdot 1 = -1$
$v_3$	1	-1	$1 \cdot (-1) = -1$
$v_4$	1	1	$1 \cdot 1 = 1$

## Example hull computation: Two isolated/nonintertwining contexts with two mutually exclusive binary observables cntd.

input

```
import cdd
mat = cdd.Matrix([
  [1, -1, -1, 1],
  [1, -1, 1, -1],
  [1, 1, -1, -1],
  [1, 1, 1, 1]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()

print(ine)
```

output

```
H - representation
begin
  4 4 rational
  1 -1 -1 1     $\Rightarrow E_1 + E_2 - E_{12} \leq 1$ 
  1 1 -1 -1     $\Rightarrow -E_1 + E_2 + E_{12} \leq 1$ 
  1 -1 1 -1     $\Rightarrow E_1 - E_2 + E_{12} \leq 1$ 
  1 1 1 1       $\Rightarrow -E_1 - E_2 - E_{12} \leq 1$ 
end
```

Nothing exciting here!

## Example hull computation: Two isolated/nonintertwining contexts with two mutually exclusive binary observables cntd.

Note that, from the parameterization obtained earlier, with  $\lambda_{x_+y_-} = \lambda_{+-}$  et cetera, we have

$$E_1 = -\lambda_{--} - \lambda_{-+} + \lambda_{+-} + \lambda_{++},$$

$$E_2 = -\lambda_{--} + \lambda_{-+} - \lambda_{+-} + \lambda_{++},$$

$$E_{12} = \lambda_{--} - \lambda_{-+} - \lambda_{+-} + \lambda_{++},$$

so that, in consistency with the probability distributions,

$$\begin{aligned} E_1 + E_2 - E_{12} &= -3\lambda_{--} + \lambda_{-+} + \lambda_{+-} + \lambda_{++} \\ &= -4\lambda_{--} + 1 \leq 1 \Rightarrow \lambda_{--} \geq 0, \end{aligned}$$

$$\begin{aligned} -E_1 + E_2 + E_{12} &= \lambda_{--} + \lambda_{-+} - 3\lambda_{+-} + \lambda_{++} \\ &= -4\lambda_{+-} + 1 \leq 1 \Rightarrow \lambda_{+-} \geq 0, \end{aligned}$$

$$\begin{aligned} E_1 - E_2 + E_{12} &= \lambda_{--} - 3\lambda_{-+} + \lambda_{+-} + \lambda_{++} \\ &= -4\lambda_{-+} + 1 \leq 1 \Rightarrow \lambda_{-+} \geq 0, \end{aligned}$$

$$\begin{aligned} -E_1 - E_2 - E_{12} &= \lambda_{--} + \lambda_{-+} + \lambda_{+-} - 3\lambda_{++} \\ &= -4\lambda_{++} + 1 \leq 1 \Rightarrow \lambda_{++} \geq 0. \end{aligned}$$

## Three isolated/nonintertwining contexts with two mutually exclusive binary observables (Suppes-Zanotti inequalities)

Suppes & Zanotti DOI [10.1007/BF01063886](https://doi.org/10.1007/BF01063886), Khrennikov DOI

[10.1007/s10773-020-04666-z](https://doi.org/10.1007/s10773-020-04666-z), KS DOI [10.1007/s10773-021-04850-9](https://doi.org/10.1007/s10773-021-04850-9)

Same as before, but suppose three binary observables take on the “affine transformed” values  $-1$  and  $1$ —that is,  $E_1, E_2, E_3 \in \{-1, 1\}$ —and consider the “bouquet” of joint observables

$$\{E_{12} = E_1 \cdot E_2, E_{13} = E_1 \cdot E_3, E_{23} = E_2 \cdot E_3\}.$$

Entries of the Travis matrix are obtained from all combinations of values, and (optionally) by eliminating redundancies—that is, identical row vectors  $(E_{12}, E_{13}, E_{23})$ —resulting in a “reduced”  $4 \times 3$  Travis matrix:

	$E_1$	$E_2$	$E_3$	$E_{12}$	$E_{13}$	$E_{23}$
$v_1$	1	1	1	1	1	1
$v_2$	1	1	-1	1	-1	-1
$v_3$	1	-1	1	-1	1	-1
$v_4$	1	-1	-1	-1	-1	1
$v_5$	-1	1	1	-1	-1	1
$v_6$	-1	1	-1	-1	1	-1
$v_7$	-1	-1	1	1	-1	-1
$v_8$	-1	-1	-1	1	1	1

$$\Rightarrow \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{pmatrix}.$$

# Suppes-Zanotti cntd.

input of the Suppes-Zanotti hull computation:

```
import cdd
mat = cdd.Matrix([
[1, 1, 1, 1],
[1, 1, -1, -1],
[1, -1, 1, -1],
[1, -1, -1, 1]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()

print(ine)
```

output of the Suppes-Zanotti hull computation:

```
H - representation
begin
4 4 rational
1 -1 -1 1       $\Rightarrow E_{12} + E_{13} - E_{23} \leq 1$ 
1 1 -1 -1       $\Rightarrow -E_{12} + E_{13} + E_{23} \leq 1$ 
1 -1 1 -1       $\Rightarrow E_{12} - E_{13} + E_{23} \leq 1$ 
1 1 1 1         $\Rightarrow -E_{12} - E_{13} - E_{23} \leq 1$ 
end
```

Something exciting here! Please see earlier references.



# Four isolated/nonintertwining contexts with two mutually exclusive binary observables [Clauser-Horn-Shimony-Holt (CHSH), 1969]

CHSH DOI [10.1103/PhysRevLett.23.880](https://doi.org/10.1103/PhysRevLett.23.880)

From now on, Travis matrix entries will **not** be color marked.

Same as before, but suppose *four* binary observables—“two per side”—take on the “affine transformed” values  $-1$  and  $1$ —that is,  $E_1, E_2, E_3, E_4 \in \{-1, 1\}$ —and consider the “bouquet” of joint observables

$$\{E_{13}, E_{14}, E_{23}, E_{24}\}.$$

For the explicit enumeration of all valuations and full details see the *supplement* of KS DOI [10.3390/e22060602](https://doi.org/10.3390/e22060602)

## CHSH cntd.: Input of the CHSH hull computation

Identical vertices can be eliminated.

```
import cdd
mat = cdd.Matrix([
    [1, 1, 1, 1, 1],
    [1, 1, -1, 1, -1],
    [1, -1, 1, -1, 1],
    [1, -1, -1, -1, -1],
    [1, 1, 1, -1, -1],
    [1, 1, -1, -1, 1],
    [1, -1, 1, 1, -1],
    [1, -1, -1, 1, 1]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()
print(ine)
```

# CHSH cntd.: Output of the CHSH hull computation

H – representation

begin

16 5 rational

$$2 - 1 \ 1 \ 1 \ 1 \quad \Rightarrow E_{13} - E_{14} - E_{23} - E_{24} \leq 2$$

$$1 \ 0 \ 0 \ 1 \quad \Rightarrow -E_{24} \leq 1$$

$$1 \ 0 \ 0 \ 1 \ 0 \quad \Rightarrow -E_{23} \leq 1$$

$$2 \ 1 - 1 \ 1 \ 1 \quad \Rightarrow -E_{13} + E_{14} - E_{23} - E_{24} \leq 2$$

$$2 \ 1 \ 1 \ 1 - 1 \quad \Rightarrow -E_{13} - E_{14} - E_{23} + E_{24} \leq 2$$

$$1 \ 1 \ 0 \ 0 \ 0 \quad \Rightarrow -E_{13} \leq 1$$

$$2 \ 1 \ 1 - 1 \ 1 \quad \Rightarrow -E_{13} - E_{14} + E_{23} - E_{24} \leq 2$$

$$1 \ 0 \ 1 \ 0 \ 0 \quad \Rightarrow -E_{14} \leq 1$$

$$2 \ 1 - 1 - 1 - 1 \quad \Rightarrow -E_{13} + E_{14} + E_{23} + E_{24} \leq 2$$

$$1 \ 0 \ 0 \ 0 - 1 \quad \Rightarrow E_{24} \leq 1$$

$$1 \ 0 \ 0 - 1 \ 0 \quad \Rightarrow E_{23} \leq 1$$

$$2 - 1 \ 1 - 1 - 1 \quad \Rightarrow E_{13} - E_{14} + E_{23} + E_{24} \leq 2$$

$$1 \ 0 - 1 \ 0 \ 0 \quad \Rightarrow E_{14} \leq 1$$

$$2 - 1 - 1 \ 1 - 1 \quad \Rightarrow E_{13} + E_{14} - E_{23} + E_{24} \leq 2$$

$$2 - 1 - 1 - 1 \ 1 \quad \Rightarrow E_{13} + E_{14} + E_{23} - E_{24} \leq 2$$

$$1 - 1 \ 0 \ 0 \ 0 \quad \Rightarrow E_{13} \leq 1$$

end

More exciting things here! Please see earlier references.

## Six isolated/nonintertwining contexts with two mutually exclusive binary observables [Pitovsky-Svozil (PS), 2000]

PS DOI [10.1103/PhysRevA.64.014102](https://doi.org/10.1103/PhysRevA.64.014102)

See also follow-up papers by Sliwa DOI

[10.1016/S0375-9601\(03\)01115-0](https://doi.org/10.1016/S0375-9601(03)01115-0), Colins and N. Gisin DOI

[10.1088/0305-4470/37/5/021](https://doi.org/10.1088/0305-4470/37/5/021), and Avis, Imai, Ito and Sasaki DOI

[10.1088/0305-4470/38/50/007](https://doi.org/10.1088/0305-4470/38/50/007).

Same as before, but suppose *six* binary observables—“three per side”—take on the “affine transformed” values  $-1$  and  $1$ —that is,  $E_1, E_2, \dots, E_6 \in \{-1, 1\}$ —and consider the “bouquet” of joint observables

$$\{E_1, E_2, \dots, E_6, E_{14}, E_{15}, E_{16}, E_{34}, E_{35}, E_{36}\}.$$

For the explicit enumeration of all valuations and full details see the *supplement* of KS DOI [10.3390/e22060602](https://doi.org/10.3390/e22060602)

## PS cntd.: Input of the PS hull computation

```
import cdd
mat = cdd.Matrix([
[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
[1, 1, 1, 1, 1, 1, -1, 1, 1, -1, 1, 1, -1, 1, 1],
[1, 1, 1, 1, 1, -1, 1, 1, -1, 1, 1, -1, 1, 1, -1],
...
[1, -1, -1, -1, -1, 1, -1, 1, -1, 1, 1, -1, 1, 1, -1],
[1, -1, -1, -1, -1, -1, 1, 1, 1, -1, 1, 1, -1, 1, 1],
[1, -1, -1, -1, -1, -1, -1, 1, 1, 1, 1, 1, 1, 1, 1]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()
print(ine)
```

# PS cntd.: Output of the PS hull computation

H – representation

begin

684 16 rational

...

4 0 - 1 1 - 1 - 1 0 1 - 1 0 1 1 1 - 1 - 1 1

$$\Rightarrow -4 \leq -E_2 + E_3 - E_4 - E_5 \\ + E_{14} - E_{15} + E_{24} + E_{25} + E_{26} - E_{34} - E_{35} + E_{36}$$

...

4 1 1 0 1 1 0 1 1 1 1 1 - 1 1 - 1 0

$$\Rightarrow -4 \leq E_1 + E_2 + E_4 + E_5 \\ + E_{14} + E_{15} + E_{16} + E_{24} + E_{25} - E_{26} + E_{34} - E_{35}$$

...

end

Often one assumes equibalanced single outcomes, for which  $E_i = 0$ .

More exciting things here! Please see earlier references.



*Instances and example computations III:  
intertwining contexts''*



## Two intertwining contexts with three mutually exclusive observables per context: Input of the firefly-in-a-box (FFB) hull computation

For the Travis matrix of this configuration see its earlier enumeration; or, for instance, DOI [10.5169/seals-116747](https://doi.org/10.5169/seals-116747).

Consider the “bouquet” of joint probabilities

$$\{p_1, p_2, p_3, p_4, p_5, p_{14}, p_{15}, p_{24}, p_{25}\}.$$

```
import cdd
mat = cdd.Matrix([
  [1, 1, 0, 0, 1, 0, 1, 0, 0, 0],
  [1, 1, 0, 0, 0, 1, 0, 1, 0, 0],
  [1, 0, 1, 0, 1, 0, 0, 0, 1, 0],
  [1, 0, 1, 0, 0, 1, 0, 0, 0, 1],
  [1, 0, 0, 1, 0, 0, 0, 0, 0, 0]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()
print(ine)
```



## PS cntd.: Output of the FFP hull computation

H — representation

linearity 5 6 7 8 9 10

begin

10 10 rational

1 -1 -1 0 0 0 0 0 0

0 1 0 0 0 -1 0 0 0

0 0 1 0 -1 0 1 0 0 0

0 0 0 0 1 0 -1 0 0 0

0 0 0 0 0 0 1 0 0 0

-1 1 1 1 0 0 0 0 0 0

0 -1 -1 0 1 1 0 0 0 0

0 -1 0 0 0 0 1 1 0 0

0 0 0 0 -1 0 1 0 1 0

0 0 -1 0 1 0 -1 0 0 1

$$\Rightarrow 1 = p_1 + p_2 + p_3$$

$$\Rightarrow p_1 + p_2 = p_4 + p_5$$

$$\Rightarrow p_1 = p_{14} + p_{15}$$

$$\Rightarrow p_4 = p_{14} + p_{24}$$

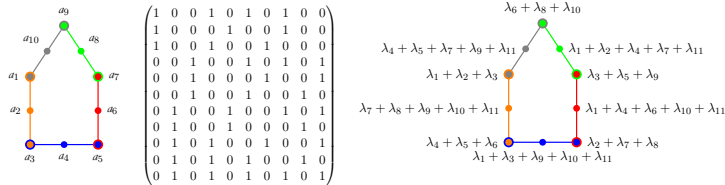
$$\Rightarrow 0 = p_2 \underbrace{-p_4 + p_{14}}_{-p_{24}} - p_{25} \quad \Rightarrow p_2 = p_{24} + p_{25}$$

end

Note: The polytope spanned by the Travis matrix is not full-dimensional—that is, five vertices cannot span a nine-dimensional space.

# Five cyclically intertwining contexts with three mutually exclusive observables (pentagon/pentagram/house shaped hypergraph)

For the Travis matrix of this configuration see, for instance, Wright, 1978 DOI [10.1016/B978-0-12-473250-6.50015-7](https://doi.org/10.1016/B978-0-12-473250-6.50015-7), resulting in the following classical probabilities:



The sum of the probabilities of the “intertwining” observables  $a_1, a_3, a_5, a_7, a_9$ —no hull computation is required for this ad hoc estimate—yields the Bub & Stairs inequality DOI [10.1007/s10701-009-9307-8](https://doi.org/10.1007/s10701-009-9307-8):

$$\lambda_1 + 2\lambda_2 + 2\lambda_3 + \lambda_4 + 2\lambda_5 + 2\lambda_6 + \lambda_7 + 2\lambda_8 + \lambda_9 + \lambda_{10} \leq 2 \sum_{i=1}^{11} \lambda_i = 2.$$

## Five cyclically intertwining contexts: affine shifted values

Consider now, with Klyachko, Can, Binicioglu, and Shumovsky DOI [10.1103/PhysRevLett.101.020403](https://doi.org/10.1103/PhysRevLett.101.020403), the affine shifted value assignments:

$$v(a_i) \rightarrow \mathcal{A}_1 = 2v(a_i) - 1,$$

or, more explicitly,

$$v = 0 \mapsto \mathcal{A} = -1 \text{ and } v = 1 \mapsto \mathcal{A} = 1.$$

The associated “shifted” Travis matrix is the original one with 0s substituted by  $-1$ s.

Consider the “bouquet” of joint probabilities (only joint observables contribute and are thus marked blue on yellow):

$$\{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}, A_{13} = A_1 A_3, \\ A_{35} = A_3 A_5, A_{57} = A_5 A_7, A_{79} = A_7 A_9, A_{91} = A_9 A_1\}.$$

# Five cyclically intertwining contexts: affine shifted values cntd.

The affine shifted Travis matrix then is (please check yourself ;-):

$$\begin{pmatrix} 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 \\ -1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 \\ -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Input for the Hull computation:

```
import cdd
mat = cdd.Matrix([
[1, -1, 1, 1, 1, -1],
[1, -1, -1, -1, 1, -1],
[1, -1, 1, -1, -1, -1],
[1, -1, -1, 1, 1, 1],
[1, -1, -1, -1, -1, 1],
[1, -1, -1, 1, -1, -1],
[1, 1, -1, -1, 1, 1],
[1, 1, -1, -1, -1, -1],
[1, 1, 1, -1, -1, 1],
[1, 1, 1, 1, -1, -1],
[1, 1, 1, 1, 1, 1]])
poly = cdd.Polyhedron(mat)
ine = poly.get_inequalities()

print(ine)
```

# Hull computation for the affine shifted values of propositions arranged in five cyclically intertwining contexts: the “pentagram inequality”

```
H — representation
begin
  11 6 rational
  1 0 0 0 1 0
  1 0 0 0 0 1
  1 0 1 0 0 0
  3 1 1 1 1 1
     $\Rightarrow 3 \geq -A_{13} - A_{35} - A_{57} - A_{79} - A_{91}$ 
     $\Rightarrow A_{13} + A_{35} + A_{57} + A_{79} + A_{91} \geq -3$ 
  1 1 0 0 0 0
  1 0 0 1 0 0
  1 1 - 1 1 - 1 - 1
  1 - 1 1 - 1 - 1 1
  1 1 - 1 - 1 1 - 1
  1 - 1 1 - 1 1 - 1
  1 - 1 - 1 1 - 1 1
end
```

The art here, as in the Bub-Stairs inequality, and indeed, all Boole-Bell type inequalities, is to figure out “bouquets” of (joint) probabilities that are violated for at least some quantum states.

∞ *Leaving the hull problem* ∞  
*for deriving classical bounds on probabilities . . .*

## Insert: Cabello's ad hoc “contextual inequalities” are not from hull computations

One can “ease” the rules for the two-valued states and just consider **any** binary value assignment—provided that it is not context dependent—so this value assignment should be global but otherwise arbitrary. Cabello did a respective brute-force or exhaustive search computation in DOI [10.1016/0375-9601\(96\)00134-X](#), for a Kochen-Specker type logical structure introduced by Cabello, Estebaranz, and García-Alcaine in DOI [10.1016/0375-9601\(96\)00134-X](#), which has no “classical” two-valued state.

This logic involves 18 binary observables in 9 contexts and thus results in  $2^{18} = 262144$  vertices. Cabello in DOI [10.1103/PhysRevLett.101.210401](#) considered the 9 (from the 9 contexts involved) 4th order joint expectations corresponding to the products of all binary- $\{-1, 1\}$  observables in the 9 contexts involved.

## Insert: Cabello's “contextual inequalities” are not from hull computations

A brute-force or exhaustive search computation finds that, classically, if all 9 (from the 9 contexts involved) such products from the 4th order joint expectations are added together, the minimum is  $-7$ . Indeed, this can be readily checked (please do with a computer program ;-): there are 9216 of 262144 value assignments rendering  $-7$ , and no  $-8$  or  $-9$  instances.

However, quantum mechanically, a straightforward calculation shows that, for binary- $\{-1, 1\}$  observables  $2|a_i\rangle\langle a_i| - \mathbb{1}$ , and for vectors  $|a_i\rangle$  of an arbitrary orthomodular basis,

$$\prod_{i=1}^4 (2|a_i\rangle\langle a_i| - \mathbb{1}) = -2 \left( \sum_{i=1}^4 |a_i\rangle\langle a_i| \right) + \mathbb{1} = (-2 + 1)\mathbb{1} = -\mathbb{1}.$$

Therefore, quantum mechanics requires these 4th order joint expectations to be always  $-9$ , regardless of the state.

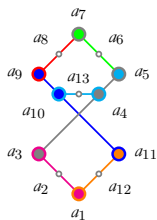
It is important to emphasize that this is no hull computation, and the bound from below of  $-7$  is no Boole-type inequality.



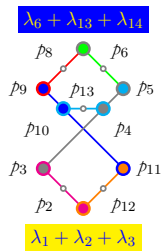
## ❧ *Quantum gadgets* ❧

# True-implies-false (TIF) gadget: Seven bi-intertwining contexts with three mutually exclusive observables: “Specker bug”/Pitowsky’s “cat’s cradle”

Kochen & Specker, 1965 DOI [10.1007/978-3-0348-9259-9\\_19](https://doi.org/10.1007/978-3-0348-9259-9_19) re-discovered and analyzed by various authors:



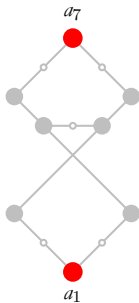
1	0	0	1	0	1	0	0	1	0	0	0	0	0
1	0	0	0	1	0	0	1	0	1	0	0	0	0
1	0	0	0	1	0	0	0	1	0	0	0	0	1
0	1	0	1	0	1	0	1	0	0	1	0	0	0
0	1	0	1	0	1	0	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0	0	0	1	0	0
0	1	0	0	1	0	0	1	0	1	0	1	0	0
0	1	0	0	1	0	0	0	1	0	0	1	0	1
0	1	0	0	1	0	0	0	0	1	0	0	1	1
0	0	1	0	0	1	0	1	0	1	0	1	0	0
0	0	1	0	0	1	0	1	0	0	1	0	1	0
0	0	1	0	0	1	0	0	1	0	0	1	0	1
0	0	1	0	0	0	1	0	0	1	0	0	1	1
0	0	1	0	0	0	0	1	0	0	1	0	1	0
0	0	1	0	0	0	0	1	0	0	0	1	0	1



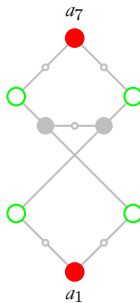
This “Specker bug” (Specker’s “Käfer”) can be employed as a classical (non-contextual) true-implies-false (TIF) gadget, as the two “terminal points”  $a_1$  and  $a_7$  cannot both be “true” (value 1) at the same time. Nevertheless one can be true (have value 1) and the other one false (value 0). Also they can both be false (value 0): their respective probabilities  $\lambda_1 + \lambda_2 + \lambda_3$  and  $\lambda_6 + \lambda_{13} + \lambda_{14}$  are mutually exclusive!

# Specker bug as TIF gadget continued: proof by contradiction

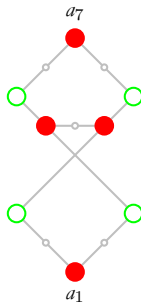
From DOI [10.1103/PhysRevA.103.022204](https://doi.org/10.1103/PhysRevA.103.022204)



“wrong” assumption: both terminals  $a_1$  and  $a_7$  have value 1



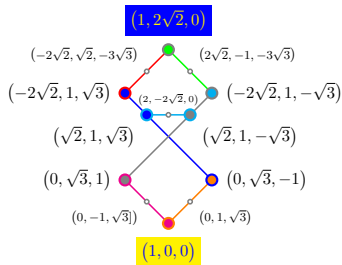
consequence



“wrong” consequence: only one atom in a context can have value 1

Note: TIF gadgets are symmetric with respect to an exchange of their terminal points!

# Specker bug as TIF gadget continued: quantum predictions



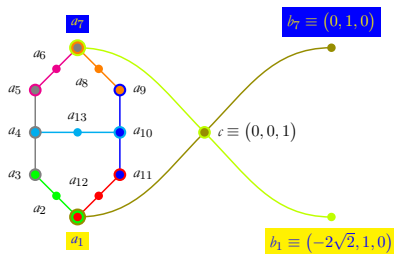
And yet quantum joint probabilities allow predictions of co-occurrences of both  $a_1$  and  $a_7$ —“prepare  $a_1$  and measure  $a_7$ ”—with frequencies/expectations up to  $\frac{1}{9}$ . That is, the pure state associated with the vector  $|a_1\rangle \equiv (1, 0, 0)$  and the proposition associated with the dyadic product of  $|a_7\rangle \equiv \frac{1}{3} (1, 2\sqrt{2}, 0)$  results in the joint quantum probability

$$|\langle a_7 | a_1 \rangle|^2 = \frac{1}{9} > 0$$

of  $|a_7\rangle$  given  $|a_1\rangle$  (and vice versa); and the Specker bug has an associated orthogonal representation of all entities involved.

# Extending the TIF gadget to a true-implies-true (TTT) gadget

From Kochen & Specker, 1967, DOI [10.1512/iumj.1968.17.17004](https://doi.org/10.1512/iumj.1968.17.17004)

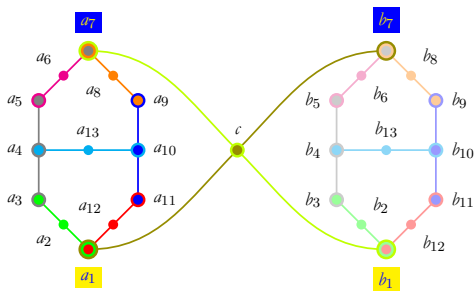


If  $a_1$  is true (has value 1) then  $b_1$  has to be true (has value 1). Likewise, if  $a_7$  is true (has value 1) then  $b_7$  has to be true (has value 1). Just as for the Specker bug, a proof is either direct by enumerating all two-valued states (Travis matrix), or by contradiction.

Kochen & Specker used an faithful orthogonal representation allowing a serial composition of several TTT gadgets—actually, five TTT gadgets with “aperture”, that is, angle between terminal points  $a_1$  and  $b_1$  or  $a_7$  and  $b_7$  of  $\frac{\pi}{10}$  or  $(\frac{90}{5})^\circ = 18^\circ$ —to arrive at a compound TTT gadget with “aperture”  $\frac{\pi}{2}$  or  $90^\circ$ , which is forbidden, as it is within a new context.

# Extending two intertwining TTT gadgets to a combo realizing classical propositional inseparability and non-embeddability

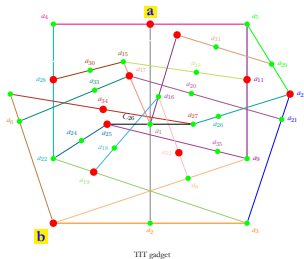
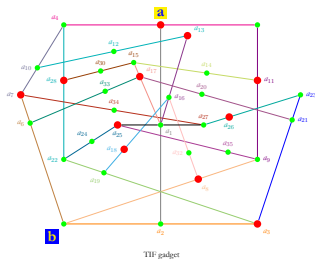
From Kochen & Specker, 1967, DOI [10.1512/iumj.1968.17.17004](https://doi.org/10.1512/iumj.1968.17.17004)



If  $a_1$  is true (has value 1) then  $b_1$  has to be true (has value 1), and vice versa. Likewise, if  $a_7$  is true (has value 1) then  $b_7$  has to be true (has value 1), and vice versa. Therefore,  $a_1$  cannot be classically separated from  $b_1$ , and  $a_7$  cannot be classically separated from  $b_7$ . Just as for the Specker bug, a proof is either direct by enumerating all two-valued states (Travis matrix), or by contradiction. Non-embeddability follows from non-separability, see Kochen & Specker's Theorem 0.

# A true-implies-false gadget and a true-implies-true gadget with identical terminal points

DOI [10.3390/e20060406](https://doi.org/10.3390/e20060406) based on Abbott, Calude, and KS DOI [10.1063/1.4931658](https://doi.org/10.1063/1.4931658)



If  $a$  is true (has value 1) then,

according to the TIF gadget:  $b$  cannot be true (cannot have value 1);

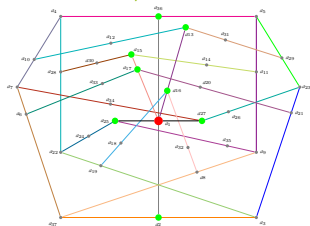
according to the TTT gadget:  $b$  cannot be false (cannot have value 0).

Therefore, if  $a$  is true (has value 1),  $b$  can neither be true nor false (has neither the values 0 nor 1): it needs to be **value indefinite/indeterminate**.

See also Pitowsky DOI [10.1063/1.532334](https://doi.org/10.1063/1.532334).

Quantum propositional structures whose classical interpretation requires certain observables to be true and others false (nonunitarity)

DOI 10.1007/978-3-030-34316-3\_24 based on Abbott, Calude, and KS

DOI [10.1063/1.4931658](https://doi.org/10.1063/1.4931658)[illegible]

Proposition  $a_1$  must be true (value 1) all the time.

Propositions  $a_2, a_{13}, a_{15}, a_{16}, a_{17}, a_{25}, a_{27}, a_{36}$  must be false (value 0) all the time.

Note: one can always change the coordinate system / basis and rotate a state or a dichotomic elementary proposition into  $a_1$  or  $a_2$ .

What does such an outcome signify? Cf. DOI

10.3390/quantum2020018



Quantum propositional structures whose classical interpretation requires certain observables to be true and others false (nonunitarity) cntd.

The configuration of observables, if interpreted classically, forces one observable to be true all the time and thus some adjacent, connected ones to be false all the time.

In operational terms this is equivalent of stating that the classical prediction is that no observable can be prepared to be in any of its two particular states: it can neither be true nor false, because depending on one value I can always rotate my basis such that my state coincides with the “one” or “zero” value states, respectively.

This, in my opinion, may well be the eclipse—or the end—of this kind of graph-theoretic considerations in quantum mechanics: one can derive an immediate, almost trivially testable, contradiction relative to the assumption of classical value definiteness.

Thank you for your attention!

