UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR INFORMATIONSSYSTEME

Özgür L. Özçep

Ontology Change 2

Lecture 10: Revision for Ontology Change

27 January, 2016

Foundations of Ontologies and Databases for Information Systems

CS5130 (Winter 2015)

Recap of/Continuing Lecture 9

AGM Belief revision

- Considered postulates and concrete operators for change operators on belief-sets
  - Belief-Sets = logically closed sets over given language
  - change operators: expansion (just adding and closing), contraction (eliminating), revision (adding and consistency)
  - Different ways to construct operators: we considered partial-meet based operators

- Criticisms: discussed recovery, minimality, success, Ramsey test (see next exercise) etc.

- Need for extensions and adaptations from ontology change perspective
  - Finiteness: (Finite) Belief Bases instead of Belief sets
  - Discussed last time syntax sensitive revision; continue here.

Semantical Belief-Base Revision

- Semantical belief-revision demands syntax insensitivity in both arguments: trigger and also the belief base

- In this scenario: belief bases = **knowledge bases**

Schema for semantical belief revision

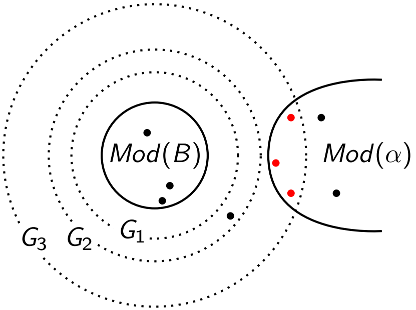
$$B * \alpha = \mathit{FinRepr}(\mathit{Mod}(B) *_{\mathit{sem}} \mathit{Mod}(\alpha))$$

- $\mathit{Mod}(X)$ = Models of X
- $*_{\mathit{sem}}$ a semantical revision operator operating on pairs of sets of models
- $\mathit{FinRep}(M)$ = Formulate or finite set of formulae that hold in all models in M

Approach 1 to Semantical Revision: Generalization

- Generalize (weaken) your belief base B' minimally s.t. enlarged set of models G_i intersects with Models of trigger

- Dalal's approach
  - Defined for propositional logic models • •
  - G_i = models with Hamming distance $\leq i$ to models in $Mod(B)$

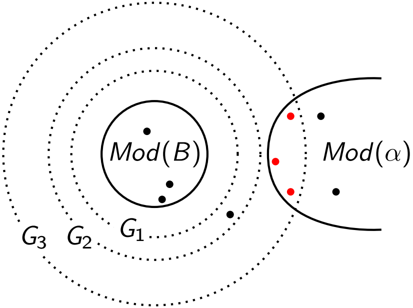


Lit: M. Dalal. Investigations into a theory of knowledge base revision: preliminary report. In AAAI-88, pages 475–479, 1988.

Approach 1 to Semantical Revision: Generalization

- Generalize (weaken) your belief base B' minimally s.t. enlarged set of models G_i intersects with Models of trigger

- Groves's approach: spheres
  - Defined on possible worlds • •
  - Possible world = maximally consistent set w.r.t. logic $(\mathcal{L}, Cn)$
  - G_i = sphere = set of possible worlds



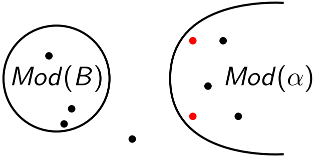
- **Note:** Maximal consistent sets correspond to models
- "Semantics" also possible in logics defined by $(\mathcal{L}, Cn)$

Lit: A. Grove. Two modellings for theory change. Journal of Philosophical Logic, 17:157–170, 1988.

Approach 2 to Semantical Revision: Minimal Distance

- Dual but more general approach to generalization: minimality
- Find trigger models • with "minimal distance" to $Mod(B)$
  $B * \alpha = FinRep\big(Min_{\leq_{Mod(B)}}(Mod(\alpha))\big)$

- Various ways to specify minimal distance
  - incorporating order, cardinality, etc.



Lit: K. Satoh. Nonmonotonic reasoning by minimal belief revision. In FGCS-88, 455–462, 1988.

Lit: A. Borgida. Language features for flexible handling of exceptions in information systems. ACM Trans. Database Syst., 10(4):565–603, 1985.

Lit: A. Weber. Updating propositional formulas. In Expert Database Conf., pp. 487–500, 1986.

Lit: M. Winslett. Updating Logical Databases. Cambridge University Press, 1990.

Lit: K. D. Forbus. Introducing actions into qualitative simulation. In IJCAI-89, 1273–1278, 1988.

Complexity of Revision

- Need feasibility of testing: $B * \alpha \models \beta$.
- No feasibility even if for finite propositional belief bases as:
- Consistency testing is hard & all subsets are culprit candidates
- Complexity, roughly, in polynomial hierarchy for propositional revision operators (so in PSPACE)

Reminder: Polynomial hierarchy using oracle speak

- $Cmplx1^{Cmplx2}$ = Problems solvable in $Cmplx1$ if one uses problems in $Cmplx2$ as oracle
  - $\Delta_0^P := \Sigma_0^P := \Pi_0^P := P$
  - $\Delta_{i+1}^P := P^{\Sigma_i^P}$ $\quad \Sigma_{i+1}^P := NP^{\Sigma_i^P}$ $\quad \Pi_{i+1}^{\mathrm{P}} := coNP^{\Sigma_i^P}$
- Example: $\Pi_2^P = coNP^{\Sigma_1^P} = coNP^{NP}$

Lit: T. Eiter and G. Gottlob. On the complexity of propositional knowledge base revision, updates, and counterfactuals. Artif. Intell., 57:227–270, October 1992.

How to cope with this modest complexities?

- Restrict logic to be used (not always helpful: see horn-revision)
- Restrict the set of culprits: E.g., allow only culprits in ABox; otherwise ignore them.
- Restrict other relevant parameters: treewidth, common variables

  Lit: A. Pfandler et al. On the parameterized complexity of belief revision. In IJCAI-15, pages 3149–3155, 2015.

End of Lecture 9 contents

Ontology Change

Classification of Ontology Change

- Group 1 (“Overcome Heterogeneity”)
  - Approaches where the main purpose is to resolve heterogeneity of ontologies by bridging between
  - Ontologies are not changed (directly)
  - But mappings may change
  - Examples: ontology mapping, o. alignment, o. morphisms etc.

- Group 2 (“Combine ontologies”)
  - Want to achieve a new ontology
  - Example: ontology merge (same domain), ontology integration (similar domain)

- Group 3(“Ontology modification”)
  - Change ontologies (not necessarily caused by other ontologies)
  - Examples: ontology debugging, ontology repair, ontology evolution

Lit: G. Flouris et al. Ontology change: classification and survey. The Knowledge Engineering Review, 23(2):117–152, 2008.

Classification of Ontology Change

- Group 1 (“Overcome Heterogeneity”)
  - Approaches where the main purpose is to resolve heterogeneity of ontologies by bridging between
  - Ontologies are not changed (directly)
  - But mappings may change
  - Examples: ontology mapping, o. alignment, o. morphisms etc.
- Group 2 (“Combine ontologies”)
  - Want to achieve a new ontology
  - Example: ontology merge (same domain), ontology integration (similar domain)
- Group 3(“Ontology modification”)
  - Change ontologies (not necessarily caused by other ontologies)
  - Examples: ontology debugging, ontology repair, ontology evolution

Lit: G. Flouris et al. Ontology change: classification and survey. The Knowledge Engineering Review, 23(2):117–152, 2008.

Classification of Ontology Change

- Group 1 (“Overcome Heterogeneity”)
  - Approaches where the main purpose is to resolve heterogeneity of ontologies by bridging between
  - Ontologies are not changed (directly)
  - But mappings may change
  - Examples: ontology mapping, o. alignment, o. morphisms etc.

- Group 2 (“Combine ontologies”)
  - Want to achieve a new ontology
  - Example: ontology merge (same domain), ontology integration (similar domain)

- Group 3(“Ontology modification”)
  - Change ontologies (not necessarily caused by other ontologies)
  - Examples: ontology debugging, ontology repair, ontology evolution

Lit: G. Flouris et al. Ontology change: classification and survey. The Knowledge Engineering Review, 23(2):117–152, 2008.

Classification of Ontology Change

- Group 1 (“Overcome Heterogeneity”)
  - Approaches where the main purpose is to resolve heterogeneity of ontologies by bridging between
  - Ontologies are not changed (directly)
  - But mappings may change
  - Examples: ontology mapping, o. alignment, o. morphisms etc.
- Group 2 (“Combine ontologies”)
  - Want to achieve a new ontology
  - Example: ontology merge (same domain), ontology integration (similar domain)
- Group 3(“Ontology modification”)
  - Change ontologies (not necessarily caused by other ontologies)
  - Examples: ontology debugging, ontology repair, ontology evolution

Lit: G. Flouris et al. Ontology change: classification and survey. The Knowledge Engineering Review, 23(2):117–152, 2008.

Requirements due to Ontology Merge (and others)

Ontology Merge (Flouris et al. 08)

Purpose:	Fuse knowledge from ontologies over same domain
Input:	Two ontologies (from identical domains)
Output:	An ontology
Properties:	Fuse knowledge to describe domain more accurately

Requirements for OC operators

- Trigger by itself is a belief base: **multiple revision**
- Belief base formulated in non-FOL (such as DLs)
  - Notion of AGM compliant revision

    Lit: G. Flouris, D. Plexousakis, and G. Antoniou. Generalizing the AGM postulates: preliminary results and applications. NMR-04, pp. 171–179, 2004.
  - Different postulates (to capture e.g. minimality):

    Lit: M. M. Ribeiro and R. Wassermann. Minimal change in AGM revision for non-classical logics. In KR-14, 2014.

Requirements due to Ontology Mapping

Ontology Mapping (Flouris et al. 08)

Purpose:	Heterogeneity resolution, interoperability of ontologies
Input:	Two (heterogeneous) ontologies
Output:	A mapping between the ontologies' vocabularies
Properties:	The output identifies related vocabulary entities

Requirements for OC operators

- Mappings should not lead to inconsistencies
- Change of mappings in design time or due to change in ontologies
- **Lit:** C. Meilicke and H. Stuckenschmidt. Reasoning support for mapping revision. Journal of Logic and Computation, 2009.
- **Lit:** G. Qi, Q. Ji, and P. Haase. A conflict-based operator for mapping revision. In DL-09, volume 477 of CEUR Workshop Proceedings, 2009.

Mappings for Ontologies

- Data exchange provided mappings between schemata
- Here consider mappings between mappable "elements" of an ontology
- No unique representation format for ontology mappings

Definition (Mappings according to (Meilicke et al. 09))

$$(e_1\ ,\ e_2\ ,\ c\ ,\ deg)$$

- $e_1 \in$ mappable elements of first ontology $\mathcal{O}_1$ (e.g. concept symbols of $\mathcal{O}_1$)
- $e_2 \in$ mappable elements of second ontology $\mathcal{O}_2$
- c: type of mapping (e.g. c is equivalence or subsumption if e_i concepts)
- deg : degree of trust in the mapping

Example (Incompatible ontologies)

$\mathcal{O}_A$

A1 $Article_A \equiv \exists publ_A.Journal_A$

A2 $Journal_A \sqsubseteq \neg Proceedings_A$

A3 $(func\ publ_A)$

$\mathcal{O}_B$

B1 $Article_B \equiv \exists publ_B.Journal_B \sqcup Proceedings_B$

B2 $publish_B(ab, procXY)$

B3 $Proceedings_B(procXY)$

- Following set of mappings $\mathcal{M}_1$ is not-consistent with ontologies
  - $(Article_A, Article_B, \equiv, 1)$
  - $(Journal_A, Journal_B, \equiv, 1)$
  - $(Proceedings_A, Proceedings_B, \equiv, 1)$
  - $(publ_A, publ_B, \equiv, 1)$

Example (Incompatible ontologies)

$\mathcal{O}_A$

A1 $Article_A \equiv \exists publ_A.Journal_A$

A2 $Journal_A \sqsubseteq \neg Proceedings_A$

A3 $(func\ publ_A)$

$\mathcal{O}_B$

B1 $Article_B \equiv \exists publ_B.Journal_B$
$\sqcup Proceedings_B$

B2 $publish_B(ab, procXY)$

B3 $Proceedings_B(procXY)$

- Following set of mappings $\mathcal{M}_2$ is consistent with ontologies
  - $(Article_A, Article_B, \subseteq, 1)$
  - $(Journal_A, Journal_B, \equiv, 1)$
  - $(Proceedings_A, Proceedings_B, \equiv, 1)$
  - $(publ_A, publ_B, \equiv, 1)$

Example (Incompatible ontologies)

$\mathcal{O}_A$

- A1 $Article_A \equiv \exists publ_A.Journal_A$
- A2 $Journal_A \sqsubseteq \neg Proceedings_A$
- A3 $(func\ publ_A)$

$\mathcal{O}_B$

- B1 $Article_B \equiv \exists publ_B.Journal_B$ $\sqcup Proceedings_B$
- B2 $publish_B(ab, procXY)$
- B3 $Proceedings_B(procXY)$

- Following set of mappings $\mathcal{M}_2$ is consistent with ontologies
  - $(Article_A, Article_B, \subseteq, 1)$
  - $(Journal_A, Journal_B, \equiv, 1)$
  - $(Proceedings_A, Proceedings_B, \equiv, 1)$
  - $(publ_A, publ_B, \equiv, 1)$

$\Longrightarrow$ Can use revision on mappings to get from $\mathcal{M}_1$ to $\mathcal{M}_2$.

Requirements due to Ontology Evolution

Ontology Evolution (Flouris et al. 08)

Purpose: Respond to a change in the domain or its conceptualization

Input: An ontology and a (set of) change operation(s)

Output: An ontology

Properties: Implements a (set of) change(s) to the source ontology

Requirements for OC operators

- Change in domain may be temporal change: **update vs. revision**
- Evolution calls for **iterative revision**

Requirements due to Ontology Learning

Ontology Learning (my addition)

Purpose:	Respond to new bits of information from
Input:	A start ontology and a potentially infinite sequence of information
Output:	An ontology (sequence)
Properties:	Learns an ontology from a sequence

- Related to evolution: but emphasis on change of informedness and potential infinity
- **Requirements for OC operators**
  - Informed iterated revision on potentially infinite sequences
  - Notion of learning success (e.g. stabilization, reliability)

    Lit: D. Zhang and N. Y. Foo. Convergency of learning process. In AI-02, vol 2667 of LNCS, pp. 547?556, 2002.

    Lit: K. T. Kelly. Iterated belief revision, reliability, and inductive amnesia. Erkenntnis, 50:11–58, 1998.

Update vs. Revision

- Early CS work related to BR in Database Theory

  Lit: A. M. Keller and M. Winslett. On the use of an extended relational model to handle changing incomplete information. IEEE Transactions on Software Engineering, 11(7):620–633, 1985.

- Problem: Preserve integrity constraints when DB is **updated**
- Two main differences to BR
  - In DB : Not a theory to update but a structure
  - Update operators $\diamond$ fulfill different postulates
- Reason is: different conflict diagnostics
  - Revision: Conflict caused by false information
  - Update: Conflict caused by outdated information
  - In ontology change even a third diagnostics is possible: different terminology

Lit: H. Katsuno and A. Mendelzon. On the difference between updating a knowledge base and revising it. In KR-91, pages 387–394,1991.

Example (Winslett 1988)

- Input belief set: There is either a book on the table or a magazine

$$Cn(\alpha \leftrightarrow \neg\beta))$$

- Trigger information: A book is put on the table α
- Apply revision operator fulfilling Postulates (R3) and (R4)

  (R3): $K * \alpha \subseteq K + \alpha$

  (R4): If $\neg\alpha \notin K$, then $K + \alpha \subseteq K * \alpha$. (Vacuity)

- Output belief set: There is a book on the table and no magazine.

$$Cn(\alpha \leftrightarrow \neg\beta) \cup \{\alpha\}) = Cn(\alpha \wedge \neg\beta)$$

- Alternative postulate instead of vacuity

  If $\alpha \in K$, then $K \diamond \alpha = K$

Lit: M. Winslett. Reasoning about action using a possible models approach. In Proc. of the 7th National Conference on Artificial Intelligence (AAAI-88), pp. 89–93, 1988.

Iterated Belief Revision

Iterating

- Aim: Apply change operators on sequence of triggers $\alpha_1, \alpha_2, \ldots$

- **Static approach**: same operator in every step on revision result

$$(\ldots((B * \alpha_1) * \alpha_2) * \ldots,) * \alpha_n)$$

- **Dynamic Approach**
  - operator my change depending on history

$$(\ldots((B *_1 \alpha_1) *_2 \alpha_2) *_3 \ldots,) *_n \alpha_n)$$

  - Belief Base may encode history

Iterated AGM Revision

- AGM BR not tailored towards iteration:
- Considers only postulates for arbitrary but fixed belief set
- Only interesting result for iterated AGM revision

Proposition

If $*$ fulfills all AGM revision postulates (R1)–(R8), then it fulfills

$$\textit{If } \neg\beta \notin K * \alpha, \ \textit{then } (K * \alpha) * \beta = K * (\alpha \wedge \beta)$$

- In words: If second trigger compatible, then revising with both triggers same as revising with conjunction

Need for Iteration Postulates

- Systematic study of iterated revision started in 1994

  Lit: A. Darwiche and J. Pearl. On the logic of iterated belief revision. In TARK-94, 5–23, 1994.

Example (Darwiche, Pearl 94)

- Agent hears an animal X barking like a dog
- So he thinks X is not a bird and cannot fly.

$$K \equiv \neg Bird \wedge \neg Flies$$

- But if he were told that X is a bird, he would assume that it flies.

$$K * Bird \equiv Bird \wedge Flies$$

- If agent were to know beforehand that X can fly, then he should still believe: If X were a bird, then X would fly.
- But one can construct AGM conform $*$ with

$$(K * Flies) * Bird \equiv Bird$$

Iteration Postulates (First Try)

DP1 If $\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"If second trigger stronger than first, then second trigger overrides effects of first".

DP2 If $\neg\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"For incompatible triggers the second one overrides the first one's effects"

DP3 If $\alpha \in K * \beta$, then $\alpha \in (K * \alpha) * \beta$.
"If revision only by second trigger entails first trigger, then revision with both triggers does too."

DP4 If $\neg\alpha \notin K * \beta$, then $\neg\alpha \notin (K * \alpha) * \beta$.
"If revision only by second trigger is compatible with first trigger, then revision with both triggers is too."

Iteration Postulates (First Try)

DP1 If $\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"If second trigger stronger than first, then second trigger overrides effects of first".

DP2 If $\neg\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"For incompatible triggers the second one overrides the first one's effects"

DP3 If $\alpha \in K * \beta$, then $\alpha \in (K * \alpha) * \beta$.
"If revision only by second trigger entails first trigger, then revision with both triggers does too."

DP4 If $\neg\alpha \notin K * \beta$, then $\neg\alpha \notin (K * \alpha) * \beta$.
"If revision only by second trigger is compatible with first trigger, then revision with both triggers is too."

Iteration Postulates (First Try)

DP1 If $\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"If second trigger stronger than first, then second trigger overrides effects of first".

DP2 If $\neg\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"For incompatible triggers the second one overrides the first one's effects"

DP3 If $\alpha \in K * \beta$, then $\alpha \in (K * \alpha) * \beta$.
"If revision only by second trigger entails first trigger, then revision with both triggers does too."

DP4 If $\neg\alpha \notin K * \beta$, then $\neg\alpha \notin (K * \alpha) * \beta$.
"If revision only by second trigger is compatible with first trigger, then revision with both triggers is too."

Iteration Postulates (First Try)

DP1 If $\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"If second trigger stronger than first, then second trigger overrides effects of first".

DP2 If $\neg\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.
"For incompatible triggers the second one overrides the first one's effects"

DP3 If $\alpha \in K * \beta$, then $\alpha \in (K * \alpha) * \beta$.
"If revision only by second trigger entails first trigger, then revision with both triggers does too."

DP4 If $\neg\alpha \notin K * \beta$, then $\neg\alpha \notin (K * \alpha) * \beta$.
"If revision only by second trigger is compatible with first trigger, then revision with both triggers is too."

Wake-Up-Question

Which one of the DP Postulates rules out the bird example?

DP1 If $\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.

DP2 If $\neg\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.

DP3 If $\alpha \in K * \beta$, then $\alpha \in (K * \alpha) * \beta$.

DP4 If $\neg\alpha \notin K * \beta$, then $\neg\alpha \notin (K * \alpha) * \beta$.

Example (Darwiche, Pearl 94)

- $K \equiv \neg \mathit{Bird} \wedge \neg \mathit{Flies}$
- $K * \mathit{Bird} \equiv \mathit{Bird} \wedge \mathit{Flies}$
- $(K * \mathit{Flies}) * \mathit{Bird} \equiv \mathit{Bird}$

Wake-Up-Question

Which one of the DP Postulates rules out the bird example?

DP1 If $\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.

DP2 If $\neg\alpha \in Cn(\beta)$, then $(K * \alpha) * \beta = K * \beta$.

DP3 If $\alpha \in K * \beta$, then $\alpha \in (K * \alpha) * \beta$.

DP4 If $\neg\alpha \notin K * \beta$, then $\neg\alpha \notin (K * \alpha) * \beta$.

Example (Darwiche, Pearl 94)

- $K \equiv \neg Bird \wedge \neg Flies$
- $K * Bird \equiv Bird \wedge Flies$
- $(K * Flies) * Bird \equiv Bird$

Need More Information

- (DP2) cannot be fulfilled by any AGM revision operator for belief sets

  Lit: M. Freund and D. J. Lehmann. Belief revision and rational inference. Computing Research Repository (CoRR), cs.AI/0204032, 2002.
- Reason is mainly: AGM allows for inconsistent belief sets
- Reaction by Darwiche and Pearl: consider postulates with epistemic states Ψ instead of belief sets

  Lit: A. Darwiche and J. Pearl. On the logic of iterated belief revision. Artificial intelligence, 89:1?29, 1997.
- Allows dynamic (state-based) iteration: history encoded in state Ψ and not captured by logic
  - Every state Ψ induces belief set $BS(\Psi)$
  - But revision depends on state Ψ not induced belief set
  - In particular: $\Psi_1 * \alpha \neq \Psi_2 * \alpha$ possible even if $BS(\Psi_1) = BS(\Psi_2)$.

Dynamic Operators

- Other approaches stick to belief sets (or belief bases) but allow dynamic revision operators.
- **Lit:** D. J. Lehmann. Belief revision, revised. In IJCAI-95, 1534–1540, 1995.
- **Lit:** A. C. Nayak, M. Pagnucco, and A. Sattar. Changing conditional beliefs unconditionally. In TARK-96, 119–135, 1996.

Infinite Iteration

Formal Learning Theory for Infinite Revision

- Iterable revision operators applied to potentially infinite sequence of triggers
- Define principles (postulates) that describe adequate behaviour
- The minimality ideas and relevant principles of BR not sufficient
- Let you guide by principles of inductive learning and **formal learning theory**
- Indeed, we need good principles for induction :)

  `http://www.der-postillon.com/2015/10/autofahrer-entlarvt-geheimen.html`

The Scientist-Nature-Scenario

- Class of possible worlds (one of them the real world = nature)
- Scientist has to answer queries regarding the real world
- He gets stream of data compatible with the real world
- Conjectures according to some strategy at every new arrival of trigger a hypothesis on the correct answer
- Success: Sequence of answers stabilizes to a correct hypothesis.

The Scientist-Nature-Scenario for Orders

- Class of possible worlds
- Scientist answers query regarding the real world (problem)
- He gets stream of data compatible with the real world
- Conjectures according to some strategy at every new arrival of trigger a hypothesis on the correct answer
- Success: Sequence of answers stabilizes to a correct hypothesis.

Example (Component of Order Example)

Strict orders $<$ on $\mathbb{N}$

- 0,1,2,3, ...
- 1,0,2,3, ...
- ...3,2,1, 0
- 0,2,4,6, ..., 1,3,5,7

The Scientist-Nature-Scenario for Orders

- Class of possible worlds
- Scientist answers query regarding the real world (problem)
- He gets stream of data compatible with the real world
- Conjectures according to some strategy at every new arrival of trigger a hypothesis on the correct answer
- Success: Sequence of answers stabilizes to a correct hypothesis.

Example (Component of Order Example)

Stream of dat made up by facts (called environments)

- R(2,3), R(1,2), R(0,2), R(1,4) ...
  (for world: 0,1,2,3, ...)
- R(4,3), R(5,2), ...
  (for world: ...3,2,1, 0)

The Scientist-Nature-Scenario for Orders

- Class of possible worlds
- Scientist answers query regarding the real world (problem)
- He gets stream of data compatible with the real world
- Conjectures according to some strategy at every new arrival of trigger a hypothesis on the correct answer
- Success: Sequence of answers stabilizes to a correct hypothesis.

Example (Component of Order Example)

Problem set: orders isomorphic to $\omega \cup \omega^*$

- 0,1,2,3, ... is isomporhic to ω
- ... 3,2,1, 0 is isomorphic to ω^*.
- Problem query: Has order a least element?

The Scientist-Nature-Scenario for Orders

- Class of possible worlds
- Scientist answers query regarding the real world (problem)
- He gets stream of data compatible with the real world
- Conjectures according to some strategy at every new arrival of trigger a hypothesis on the correct answer
- Success: Sequence of answers stabilizes to a correct hypothesis.

Example (Component of Order Example)

Scientist solves problem P iff for every $<\in P$ and every environment e:

- If $<$ has least element, then cofinitely often scientist says yes on $e(n)$ (on n-prefix of environment)
- If $<$ has no least element, then for cofinitely many n scientist says no on $e(n)$

The Scientist-Nature-Scenario for Orders

- Class of possible worlds
- Scientist answers query regarding the real world (problem)
- He gets stream of data compatible with the real world
- Conjectures according to some strategy at every new arrival of trigger a hypothesis on the correct answer
- Success: Sequence of answers stabilizes to a correct hypothesis.

Example (Component of Order Example)

Problem $P = \{< \in \omega \cup \omega^* \mid < \text{has least element}\}$ is solvable

- Consider L-score: For any finite sequence it is the smallest number not occurring in right argument of R
- G-score: smallest number not occurring in first argument of R
- Scientist: If L-score lower than G-score on given prefix, say yes, otherwise no.

Choosing Revision as Strategy

- Kelly investigates learning based on various revision operators defined for epistemic states
- Hypotheses = sentences in the belief sets
- Main (negative) result in (Kelly 98)

Theorem

Revision operators implementing a minimal (one-step) revision suffer from ***inductive amnesia****: If and only if some of the past is forgotten, stabilization is guaranteed.*

Lit: K. T. Kelly. Iterated belief revision, reliability, and inductive amnesia. Erkenntnis, 50:11–58, 1998.

Choosing Revision as Strategy

- Martin/Osherson investigate learning based revision operators defined for finite sequences
- So their revision operators have always the whole history within the trigger
- This leads to positive results

Theorem

Revision operators provide ideal learning strategies:There is a revision operator a scientist can use to solve every (solvable) problem.

Lit: E. Martin and D. Osherson. Scientific discovery based on belief revision. Journal of Symbolic Logic, 62(4):1352?1370, 1997.

Exercise 8: Bonus Exercises

Exercise 8.1 (4 Bonus points)

Belief Revision has strong connections to Non-monotonic reasoning: For any (say consistent) belief set K one can define an entailment relation $\vDash_K$ as follows:

$$\alpha \vDash_K \beta \textbf{ iff } \beta \in K * \alpha$$

Answer the question whether $\vDash_K$ is a monotonic entailment relation, i.e., whether it fulfills:

$$\text{If } X \vDash_K \alpha \text{ and } Y \subseteq Y, \text{ then } Y \vDash_K \alpha$$

Exercise 8.2 (4 Bonus points)

An alleged weakness of AGM belief revision is dealt under the term "Ramsey Test". Inform yourself on this test and explain how it works.

Exercise 8.3 (4 Bonus Points)

Consider the following postulate for belief bases B:

(R) If $\beta \in B$ and $\beta \notin B * \alpha$, then there is some B' with

1. $B * \alpha \subseteq B' \subseteq B \cup \{\alpha\}$
2. B' is consistent
3. $B' \cup \{\beta\}$ is inconsistent

First describe this postulates in natural language. What would be a good name for this postulate (which was invented following a criticisms of AGM revision)?