Identity Merging and Identity Revision in Talmudic Logic: An outline paper

M. Abraham, I. Belfer, D. Gabbay and U. Schild

No Institute Given

Abstract. Suppose we are given a monadic theory $\mathbf{T}$ about the constants x and y. So $\mathbf{T}$ is built up in classical logic from monadic predicates $\{P_1, P_2, ...\}$ and the classical connectives and the quantifiers and possibly the equality symbol $=$. For example the theory $\mathbf{T}$ may have in it $P(x)$ and $\neg P(y)$. We now add to the theory the revision input $x = y$. The new theory may be inconsistent. We need a belief revision mechanism to revise $\mathbf{T}$ so that it is consistent with the input. This is a very specific form of input and belief revision, of the form which we are calling "identity merging". We present in outline how the Talmud deals with this type of revision.

1 Background and orientation

Suppose we are given a monadic theory $\mathbf{T}$ about the free variables x and y. If x and y are never quantified upon in $\mathbf{T}$ they can also be viewed as constants x and y. So $\mathbf{T}$ is built up in classical logic from monadic predicates $\{P_1, P_2, \ldots\}$ and the classical connectives $\{\neg, \wedge, \vee, \rightarrow\}$ and the quantifiers $\forall, \exists$ and possibly the equality symbol $=$.

For example the theory $\mathbf{T}$ may have in it $P(x)$ and $\neg P(y)$.

We now add to the theory the revision input $x = y$. The new theory is $\mathbf{T}_{x=y}$, $\mathbf{T} \cup \{x = y\}$ is inconsistent. We need a belief revision mechanism to revise $\mathbf{T}$ so that it is consistent with the input. We can of course use one of the many AGM approaches and algorithms for this case, but these are too general and we need a more specialised tailored approach for our case. This is a very specific form of input and belief revision, of the form which we are calling "identity merging". We have knowledge about two distinct individuals (so we believe) and we discover that they are the same individuals. Now we have to reconcile what we know. Another very common case is where two conflicting bodies of laws apply to the same individual case and we need to decide how to proceed. Such cases require specialised procedures possibly tailored for each application area (case study). So formally we have a theory $\mathbf{T}$ and two distinct constants or variables x and y and we add the input $x = y$. We need not necessarily deal with a language with identity. If we do not have "$=$" in the language, we can still take $\mathbf{T}$ and take a new variable z and substitute in $\mathbf{T}$ the variable z for every free occurrence of x or of y. We will get a new theory which we can denote by $\mathbf{T}(x = y = z)$ which is inconsistent (containing $P(z)$ and $\neg P(z)$) and in need for revision. Note on the notation side, we can regard x and y either as constants or as free variables, this does not matter as long as we do not apply any quantifiers to them. So in the sequel we might talk about constants a or b or variables x and y. The choice depends on stylistic reasons.

We have three options here for revision.

1. Use the well known AGM machinery, [1, 2], which is this case means we choose one of each atomic contradicting pair $\{P(z), \neg P(z)\}$;
2. Use some new approach taking advantage of the specific form of the revision problem.
3. Use the Talmudic Logic approach;

Let us give some examples before we continue.

Example 1. Consider a university sysem with a Rector x and Head of Department of Informatics y. The university has regulations which say among others that:

1. The Rector can offer a position to a candidate and this is legally binding.
2. A Head of Department can offer a position to a candidate (in his department) but it is not legally binding, but is subject to approval by the Rector. The Head of Department should use a standard letter form which makes this clear.

Suppose now that there is a big fight between the Head of Department and his professors and the Head resigns and there is no agreement about a successor. Someone has to run the day-to-day matters of the Department, and so the Rector becomes acting Head of this Department for the time being. The Rector in his capacity as Head, offers a position in the Department to a candidate c. The standard letter which one sends in such a case says that this offer still requires the approval of the Rector but that the Department and its Head are confident that the Rector will approve the offer.

In this case the Head, who writes the letter as Head, is also the Rector, who needs to approve the appointment. The question is:

Is this letter binding or not?[1]

We have:

Rector writes $\rightarrow$ binding
Heads writes $\rightarrow \neg$ binding

If we revise by the input Rector = Head, do we take binding or $\neg$ binding?

Commonsense says that this is a binding offer.

[1] This actually happened to D. Gabbay in 1972. The perceptive reader might wonder why the Rector was ambiguous in his letter? Well, it may have been deliberate or he may have wanted to follow clear procedural lines and first write as Head and follow it up as Rector. There is more to it than that. Can the recipient of the letter assume that for all practical purposes he/she actually got the job or is there still a practical possibility that the Rector would say "I approve the appointment as Head of Department but I do not approve it as Rector"?

Example 2. This is a real example recently discussed in the American press. It relates to the Boston terrorist bombing.[2] The terrorists were American citizens and so there were two options:

1. Viewed as terrorists, send them to military trial or even to Guantanamo Bay detention camp.
2. Treat them as American citizens and send them through the American legal system.

In principle what is happening here is that we have two bodies of laws and regulations:

$\mathbf{T}_1(a)$ = Rules for a, a typical terrorist
$\mathbf{T}_2(b)$ = Rules for b, a typical US citizen.

The theory is $\mathbf{T} = \mathbf{T}_1(a) \cup \mathbf{T}_2(b)$. The input, forced upon us by the real world, is $a = b$.

The aim of this paper is to formalise and introduce the Talmudic approach. The approach is general and can be used in AI for this case, as an alternative methodology to AGM.

The AGM approach would simply take out from $\mathbf{T}$ one of $\{P(z), \neg P(z)\}$ and restore consistency (assuming P is atomic).

The Talmudic approach will do something different. To introduce it, however, we begin with describing an intermediate non-monotonic approach which is not the Talmudic one, but has an independent interest and would lead into the Talmudic approach.

The non-monotonic approach (ANM vs. AGM) says that the language of $\mathbf{T}$ (i.e. $P_1, P_2, P_3, \ldots$) is only a surface language $\mathbb{M}$ and the fact that $\mathbf{T}$ contains $P(x)$ and $\neg P(y)$ stems from deeper level non-monotonic considerations in a deeper language $\mathbb{L}$. When we revise with $x = y = z$, we have to go to the deep level non-monotonic theory governing P, x and y and see what happens there and then decide whether to contract $P(x)$ or to contract $\neg P(y)$. Thus the non-monotonic approach is a refinement of the AGM approach, where the choice of which of the contradicting pair $\{P(z), \neg P(z)\}$ to take out is made using the extra non-monotonic machinery available in the system. This is best understood when actually defined. Let us propose an ANM model.

Definition 1.

1. *Let $\mathbb{M}$ be the monadic classical predicate language with unary predicates $\{P_1, P_2, \ldots\}$ and variables and constants $\{x, y, z, c_1, c_2, \ldots\}$. We say that $\{P_i, c_j\}$ are the predicates and constants of $\mathbb{M}$. Let $\mathbb{L}$ be an expansion of $\mathbb{M}$ with additional predicates and constants $\{A, B, \ldots\}$ and and $\{d_1, d_2, ...\}$.*
2. *With each constant c and predicate P of $\mathbb{M}$ we associate a family of predicates and constants from $\mathbb{L}$ (which include P and c). Let us use the notation $\mathbb{F}(P, c)$ for this family. For example, let*

$$\mathbb{F}(P, a) = \{A\} \cup \{P, a\}$$
$$\mathbb{F}(P, b) = \{B\} \cup \{P, b\}.$$

[2] https://en.wikipedia.org/wiki/Boston_Marathon_bombing, http://www.britannica.com/event/Boston-Marathon-bombing-of-2013.

(We will not repeat "$\{P, c\}$" any more.)

3. *Assume that we have a non-monotonic consequence $\Vdash$ governing the language $\mathbb{L}$ and a theory Δ of facts for the new predicates $\{A, B \ldots\}$ of $\mathbb{L}$ which contains $\mathbb{M}$.*
4. *Assume that our surface theory $\mathbf{T}$ is the result of Δ. Namely*

$$P(x) \in \mathbf{T}_{\Delta,\mathbb{F}} \textit{ iff } \Delta \restriction \mathbb{F}(P, x) \Vdash P(x).$$

5. *We say that $\mathbf{T} = \mathbf{T}_{\Delta,\mathbb{F}}$ is derived from Δ using $\Vdash$.*

Example 3. Consider the surface predicates and constants of $\mathbf{T}$ to be P, a, b.

Let

$$\mathbb{F}(P, a) = \{A\}$$
$$\mathbb{F}(P, b) = \{B\}$$

Assume our non-monotonic logic for $\mathbb{L}$ relies on more specificity and that we have a theory Δ with the following rules:

1. $A(a) \to P(a)$
2. $B(b) \to \neg P(b)$
3. $A(b) \wedge B(b) \to P(b)$
4. $A(a)$
5. $B(b)$

Our theory $\mathbf{T}$ will contain therefore $P(a)$ and $\neg P(b)$. This is because showing $P(a)$ we can use only clauses 1. and 4. and showing $P(b)$ we can use only clauses 2. and 5.

Now let us see what happens if we add the input $a = b$. This changes the language we consider from the separate $\mathbb{F}(P, a)$ and $\mathbb{F}(P, b)$ into the joint $\mathbb{F}' = \mathbb{F}(P, a) \cup \mathbb{F}(P, b)$. Now the clauses to consider from Δ are 1. to 5.

But now, because of more specificity

$$\Delta \restriction \mathbb{F}' \Vdash P(b)$$

and so we revise $\mathbf{T}$ in view of the input $a = b$ by contracting $\neg P(b)$. Thus we see that whereas ordinary AGM theory allowed for arbitrary choice in either contracting $P(b)$ or contracting $\neg P(b)$, the non-monotonic background theory Δ, recommended contracting $\neg P(b)$. Intuitively we asked ourselves (and asked Δ) where do $\mathbf{T} \vdash P(a)$ and $\mathbf{T} \vdash \neg P(b)$ come from and we made a decision based on the answer.

We realise that perhaps the non-monotonic system may not resolve the issue. We can rely on another level (i.e. another Δ' related to Δ in a similar way to the relation of Δ to $\mathbf{T}$) and language to resolve the issue. The details are not so important as the overall approach.

Let us now work towards giving a complete formal presentation of the above ideas.[3]

[3] The perceptive reader might ask: "Where does the richer language theory Δ come from? Isn't it a bit artificial, to introduce it just to solve the problem, like a special distance/choice function for AGM revision?

Definition 2. *Let $\mathbb{L}$ be a language containing $\neg$ and let $(\Delta, \Vdash)$ be a non-monotonic theory in $\mathbb{L}$. This means that Δ is a set of formulas of $\mathbb{L}$ and $\Vdash$ is a consequence relation of the form*

$$A_1, \ldots, A_n \Vdash B$$

where A_i, B are formulas of $\mathbb{L}$ and $\Vdash$ satisfies the following conditions:

1. *Reflexivity*
$$A_1, \ldots, A_n \Vdash B, \text{ if } B \in \{A_i\}$$
2. *Cut*
$$A_1, \ldots, A_n X \Vdash B$$
and
$$A_1, \ldots, A_n \Vdash X$$
imply
$$A_1, \ldots, A_n \Vdash B$$
3. *Restricted monotonicity*
$$A_1, \ldots, A_n \Vdash B$$
and
$$A_1, \ldots, A_n \Vdash C$$
imply
$$A_1, \ldots, A_n, B \Vdash C.$$
4. *Let $\theta', \theta \subseteq \Delta$ be two finite subsets of Δ.*
We may have $\theta \Vdash B$ but $\theta \cup \theta' \Vdash \neg B$ or alteratively $\theta \Vdash \neg B$ but $\theta \cup \theta' \Vdash B$.
5. *$(\Delta, \Vdash)$ is said to be consistent iff for no $B, \theta \subseteq \Delta, \theta$ finite, we have $\theta \Vdash B$ and $\theta \Vdash \neg B$.*
6. *θ is said to* decide *a wff B if we have either $\theta \Vdash B$ or $\theta \Vdash \neg B$ (as opposed to neither).*
7. *θ is said to be minimal theory deciding B, if θ decides B and no $\theta' \subsetneqq \theta$ decides B.*
8. *We say that $\Delta \Vdash B$ iff there is a $\theta \subseteq \Delta$, θ finite, which decides B and for every minimal θ which decides B we have $\theta \Vdash B$.*

Definition 3. *1. Let $\mathbf{T}$ be a classical consistent set of wff, (considered as a theory) in a language $\mathbb{M}$. Let Δ be a consistent non-monotonic set of wffs (considered a theory) in a richer langauge $\mathbb{L} \supseteq \mathbb{M}$. Let $\Vdash$ be its consequence relation. Let P be a unary atomic predicate of $\mathbb{M}$ and let $a_i, i = 1, \ldots, k$ be distinct constants of $\mathbb{M}$. Let $\mathbb{F}$ be a function giving for each*

I see how it works and where it comes from in the examples, but in the general case?".

Our answer to this is that indeed laws and regulations come from practical situations where in the background there are undesirable cases to be avoided. So for each specific $\mathbf{T}$ tailored for a specific application area there will be a corresponding Δ. For a general theory we must stipulate and study a general recursive hierarchy of $\mathbf{T}_n$, where $\mathbf{T}_{n+1}$ acts as the "Δ" of $\mathbf{T}_n$.

$\alpha = \{P, a_1, \ldots, a_k\}$ a sublanguge $\mathbb{F}(\alpha)$ of $\mathbb{L}$. We assume that $\alpha \subseteq \mathbb{F}(\alpha)$. Note that we may have

$$\Delta \restriction \mathbb{F}(\alpha) \Vdash \pm P(a)$$

but

$$\Delta \restriction \mathbb{F}(\beta) \Vdash \mp P(a),$$

for $\beta \supsetneqq \alpha$. This can happen because $\Vdash$ is non-monotonic.

2. *Let $\mathbf{T}$ and Δ be as in (1) above. We say that Δ supports $\mathbf{T}$ if the follwing (*) holds for each unary P and constant a:*

(*) $\mathbf{T} \vdash P(a)$ *iff* $\Delta \restriction \mathbb{F}(P, a) \Vdash P(a)$.

3. *Let a, b be two distinct constants of $\mathbf{T}$. Denote by $\mathbf{T}_{a/b}$ the theory obtained from $\mathbf{T}$ by replacing every occurrence of a by the constant b.*
4. *Similarly, let P, Q be two unary predicates, we let $\mathbf{T}_{P/Q}$ be the theory obtained by replacing every occurence of P by Q.*

Remark 1. Let $\mathbf{T}$ be a monadic theory with monadic predicates $\{P_1, \ldots, P_k\}$ and constants $\{c_1, \ldots, c_m\}$. Let $\mathbf{m}$ be a classical model of the language with $\{P_j, c_i, \neg, \wedge, \vee, \rightarrow, \forall, \exists, =\}$. For this language, any model $\mathbf{m}$ has to say the following

1. For each c_i and P_j $\mathbf{m}$ has to say whether $P_j(c_i)$ or $\neg P_j(c_i)$ holds.
2. For each vector ε of $\{0, 1\}$ values of length k, let

$$\alpha_\varepsilon(x) = \bigwedge_{j=1}^{k} P_j^{\varepsilon_j}(x)$$

where $P_j^1(x) = P_j(x)$ and $P_j^0(x) = \neg P_j(x)$, and x is a variable. Then the model $\mathbf{m}$ has to say whether $\exists x \alpha_\varepsilon(x)$ holds for each ε and, if equality is in the language, $\mathbf{m}$ has to say how many different such elements exist, $0, 1, 2\ldots$ or infinity, i.e. for each n, $\mathbf{m}$ must say whether $\exists x_1, \ldots, x_n \bigwedge_{i \neq j} x_i \neq x_j$.

If there is no equality in the language then a model $\mathbf{m}$ for the language can be characterised by a wff $\varphi_\mathbf{m}$ of the form

$$\varphi_\mathbf{m} = \bigwedge_\varepsilon \pm \exists x \alpha_\varepsilon(x) \wedge \bigwedge_{i,\varepsilon} \pm \alpha_\varepsilon(c_i).$$

Let us assume we do not have equality.

Then we can assume that any $\mathbf{T}$ has only a finite number of finite models.

Remark 2. Let $\mathbf{T}$ be a monadic theory without equality as in Remark 1. Let $\mathbf{m}$ be a model for the language of the form $\varphi_\mathbf{m}$ as in Remark 1.

We now investigate the effect of the additional revision/input information that c_1 and c_2 are the same (i.e. $c_1 = c_2$). We ask whether $\varphi_\mathbf{m}$ is still consistent under the substituation of c_2 for c_1 (or $c_1 = c_2$)?

This depends on what conjuncts appear in $\varphi_\mathbf{m}$. The critical ones to be watched are triples $\varepsilon, \varepsilon', P$ such that

$$\varphi_\mathbf{m} \vdash \alpha_\varepsilon(c_1) \wedge \alpha_{\varepsilon'}(c_2)$$

and such that

$$\alpha_\varepsilon(c_1) \vdash P(c_1) \text{ and } \alpha_{\varepsilon'}(c_2) \vdash \neg P(c_2).$$

In other words, the problem is that the model $\mathbf{m}$ says for some set of unary predicates $\{P_{i_1}, \ldots, P_{i_n}\}$ the opposing pair $\{\pm P_{i_r}(c_1) \text{ and } \mp P_{i_r}(c_2)\}$.

Since we are claiming $c_1 = c_2$, we need to choose only one of them, if we want to maintain consistency.

We have a similar problem if we input equality of two predicates, say $P_1 = P_2$. There may be some $c_{j_1}, \ldots, c_{j_n}$, which the model says

$$\pm P_1(c_{j_r}) \text{ and } \mp P_2(c_{j_r})$$

again, we have opposing pairs and again we need to choose one of them if we want to maintain consistency.

Our theory of identity merging will tell us how to choose one from each opposing pair and thus maintain consistency. Our identity merging theory is a refienment of AGM for this particular case. AGM does not care how we choose.

Definition 4. *Let $\mathbf{T}$ be a complete and consistent monadic theory with constants $\{c_i\}$ and unary predicates $\{P_j\}$, and let $(\Delta, \Vdash)$ be a supporting theory for $\mathbf{T}$ as in Definition 3. Let a, b be two distinct constants and consider $\mathbf{T}_{a/b}$ and assume that it is inconsistent. The merge revision of $\mathbf{T}_{a/b}$ is performed as follows.*

Since $\mathbf{T}$ is complete, $\mathbf{T}_{a/b}$ being inconsistent means that either $\mathbf{T} \vdash P(a) \wedge \neg P(b)$ or that $\mathbf{T} \vdash \neg P(a) \wedge P(b)$ for some predicates P.

Assume without loss of generality that the former holds. Then we have that

$$\Delta \restriction \mathbb{F}(\{P, a\}) \Vdash P(a)$$
$$\Delta \restriction \mathbb{F}(\{P, b\}) \Vdash \neg P(b)$$

Consider $\theta = \Delta \restriction \mathbb{F}(\{P, a, b\})$ we may have $\theta \Vdash P(a)$ or $\theta \Vdash \neg P(a)$ or neither (but not both!). Similarly we have for the case of $P(b)$. We may now have that θ proves $P(a)$ and θ does not prove $\neg P(b)$ or θ proves $\neg P(a)$ and does not prove $P(b)$ or θ proves $P(b)$ and does not prove $\neg P(a)$ or θ proves $\neg P(b)$ and does not prove $P(a)$. In each of these cases we know how to revise.

If we still have that θ proves $P(a)$ and $\neg P(b)$ or θ proves $\neg P(a)$ and $P(b)$ or that θ proves nothing, then we revise arbitrarily.

Following the discussion of Remark 2, we can make a choice of whether to take $+P(a)$ or $\neg P(a)$ for our revised model. If $\Vdash$ does not tell us which one to take we can make an arbitrary choice. The algorithm is as follows:

1. If $\theta \Vdash P(a)$, then delete all occurrences of $\pm P(b)$ from $\varphi_{\mathbf{m}}$ to get $\varphi'_{\mathbf{m}}$. If $\theta \Vdash \neg P(b)$ then delete all occurences of $\pm P(a)$ from $\varphi_{\mathbf{m}}$ to get $\varphi''_{\mathbf{m}}$. Otherwise delete $\pm P(b)$.
   Since we assume that $a = b$, we have that $\varphi'_{\mathbf{m}} = \varphi''_{\mathbf{m}}$ and this is our revised model. The revised theory $\mathbf{T}_{a/b}$ is the theory of this model.

Similar considerations will apply to $\mathbf{T}_{P/Q}$.

This completes our discussion of the non-monotonic identity merging method. This method is, however, not how the Talmud handles the case.

The above discussion of the obvious solution now has prepared us for the introduction of the Talmudic approach, as well as providing us with the means of comparison.

A theory can be revised by introducing new items of data which affect what it can prove. A theory can be revised also by cancelling or restricting the proof rules it can use. The latter method is used in resolving logical paradoxes. The data is fixed and leads to a paradox (inconsistency or unintuitive results). So one blocks some of the proofs and thus resolves the paradox.The Talmud revises by using a hierarchy of rules cancellations as we explain in the next section.

2 The Talmudic approach

Let us look at a well known example (x is universally quantified):

1. $\text{Bird}(x) \rightarrow \text{Fly}(x)$
2. $\text{Penguin}(x) \rightarrow \text{Bird}(x)$
3. $\text{Penguin}(x) \rightarrow \neg\ \text{Fly}\ (x)$
4. $\text{Penguin}\ (a)$

We say that in viewing the above data, since Penguin is a more specific bird, then it wins and so we deduce $\neg\text{Fly}(a)$.

Let us look now at the following data:

5. Aeroplane 747 Flight BA101 → Land at Heathrow
   $A \rightarrow L$.
6. Aeroplane 747 Flight BA101 and Bad Weather → ¬ Land at Heathrow
   $A \wedge W \rightarrow \neg L$
7. Aeroplane 747 Flight BA101 and Bad Weather and Short on Fuel → Land at Heathrow
   $A \wedge W \wedge F \rightarrow L$.

We may look at this again using the principle that the more specific assumptions (i.e. the antecedent of the rule contains more conjuncts than the other rule) win. So if we have only the information that an Aeroplane 747 Flight BA101 wants to land, we conclude that it can land. If we also add the conjunct that the weather is bad then it cannot land and if we even further add the conjunct that it is also short of fuel then it can land.

The Talmud looks at this differently as in Figure 1. W and F are meta-level principles. In the Figure ordinary arrow $\rightarrow$ means support and double arrow $\twoheadrightarrow$ means attack.

The basic principle is $A \rightarrow L$. The weather conditions involve a meta-level principle which cancel the arrow leading from A to L.[4]

The fuel shortage is involved in another meta-level principle which cancels the cancellation. So we are not dealing here with more specific knowledge but we are dealing with levels of meta-knowledge and a calculus of cancellations. The

[4] Think of it as a rule of wisdom based on experience. “Just do not land in bad weather”. Another such a rule is “If you are short of fuel land as soon as you can”.

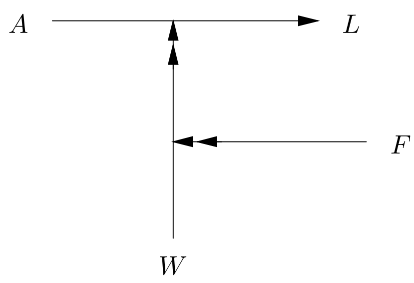


Fig. 1.

appropriate modelling of this is higher level attack and support (argumentation) networks.

So the Talmud uses a calculus of cancellations to resolve identity merging, as opposed to our previous proposal of non-monotonic support.

Let us give some examples.

Example 4. This example is really from Talmudic logic, recast in everyday modern situation.

1. The story runs as follows:
   We have a duty to maintain our homes. We also have the instinct to save money. We believe in professional people doing jobs for us, but if we can do it properly ourselves, then we do it ourselves, and not call the expert and thus save money.
   So, if the kitchen sink is blocked, we do not call a plumber to do the job but do it ourselves and save money (a plumber home visit costs about $50 just to come, independent of the job he does).
   If the problem is more serious, say a blocked toilet, then better call a plumber and not take the risk of doing the job yourself. This case does need an expert!
   We can write these rules in non-monotonic logic as follows
   (a) x is blocked $\rightarrow$ repair x yourself
   (b) x is blocked $\wedge x$ is a serious job $\rightarrow$ get John the plumber to repair x
   (c) sink is blocked
   (d) toilet is blocked
   (e) repairing the toilet is a serious job, but not the sink.
   The problem with the above is that it implies that we call John the plumber and he repairs the toilet while we repair the sink. Common sense dictates that since the plumber is available he should repair the sink as well! We could add a new clause (f) to help:
   f. x is blocked $\wedge$ John the plumber repairs $y \rightarrow$ get John the plumber to repair x.
   Clause (f) says that if x is blocked and there is any y which John the plumber repairs y[5] then John the plumber to repair x. The format of clause

[5] We tacitly assume here that they are all in the same, say, apartment building to be considered the same "call" by the plumber.

(f) is not the usual monadic one, and does not make the information on x more specific. We can artificially fix this by adding a dummy universal predicate $U(x, y)$ which relates any two elements (something like $((x = y) \bigvee \neg(x = y)))$ and write (f*)

f*. x is blocked $\bigwedge$ (John the plumber repairs $y \bigwedge U(x, y)) \rightarrow$ get John the plumber to repair x.

Now (f*) is more specific than (a) on account of the additional predicate $V(x) =$ (John the plumber repairs $y \bigwedge U(x, y))$ This is clearly a fiddle and it departs from the intuitive understanding of what is going on, which is clearly two meta-principles, namely save money but not at the expense of needed expertise!

Let us see how the Talmudic calculus of cancellations overcomes this problem.

Figures 2 and 3 describe these rules. The description is intuitive and not formal. The meaning of the nodes and arrows can be read intuitively from the figures.

The question arises what to do if both the sink and the toilet are blocked? If we just take the union of the two figures, (i.e. union of Figure 2 and Figure 3) i.e. update that the two plumbers a and b are equal, we will get that we call a plumber, the plumber does the toilet and at the same time we do the sink ourselves. It is more reasonable, however, since the plumber is already coming (and the \$50 call fee is to be paid anyway) to have the plumber do the sink as well.

Thus the "merging" of the two cases, i.e. merging of the two figures for the case that both the sink and the toilet are blocked is just a union of the graphs of the two figures. We will get Figure 4.

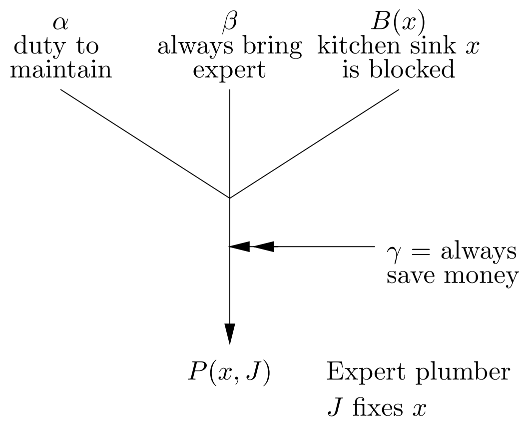
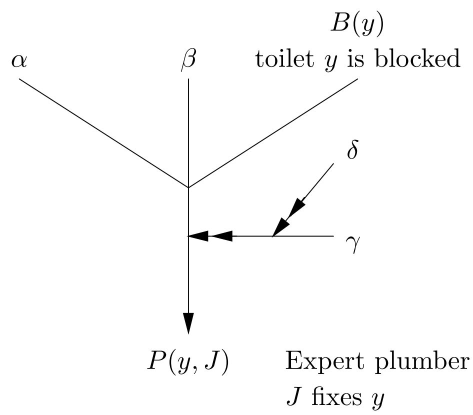


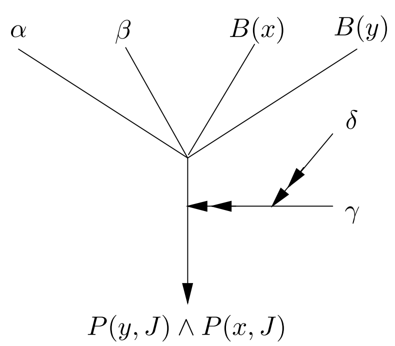
Fig. 2.

2. We now explain our notation.
   (a) $x, y, \ldots$ denote objects like $x =$ kitchen sink, $y =$ toilet.



B(y)
toilet y is blocked
α
β
δ
γ
P(y, J)
Expert plumber
J fixes y

Fig. 3.



α
β
B(x)
B(y)
δ
γ
P(y, J) ∧ P(x, J)

Fig. 4.

(b) B, P denote predicates which when applied to objects give states:
$B(x)$ = kitchen sink is blocked, $B(y)$ = toilet is blocked.
$P(x, z)$ = kitchen sink is repaired by plumber z, $P(y, z)$ = toilet is repaired by plumber z.

(c) α, β, γ are policies. For example:
α = policy to maintain your house
β = policy to always use experts
γ = policy to always save money
δ = policy to not take any risk for heavy maintenance jobs, if possible.

(d) A word about our notation: We denote the transition from one state to another by an arrow.
Figure 5 shows such notation. The π annotates the arrow. This means

$$B(x) \xrightarrow{\pi} P(x, a)$$

Fig. 5.

that because of policy π we take action and move from $B(x)$ to $P(x, a)$. It may be that several policies come together and are involved in motivating some action, or it may be the case that some policies may cancel or overrule other policies. So we allow for alternative notation which we can use as well, when there are lots of policies to denote.
Figure 5 can be equivalently presented as Figure 6 or as Figure 7.

$$\pi \text{ and } B(x) \rightarrow P(x, a)$$

Fig. 6. Alternative notation to Figure 5

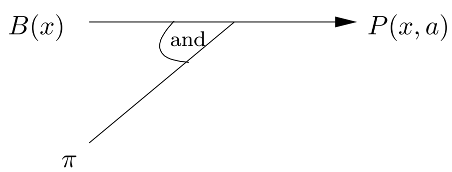


Fig. 7. Alternative notation to Figure 5

(e) Cancellation is done by double arrow.
Figure 8 shows some cancellations from some policies. It has no meaning, just a sample technical figure illustrating the notation.

i. π_1 and π_2 support together the move from $B(x)$ to $P(x,a)$.
ii. π_3 cancels the support of π_1 but allows the action to go forward on the basis of π_2.
iii. π_5 cancels the move to $P(x,a)$ no matter what, but also does not think that the support of π_4 to $Q(y)$ is a reason to cancel $\pi_2 \to P(x,a)$.

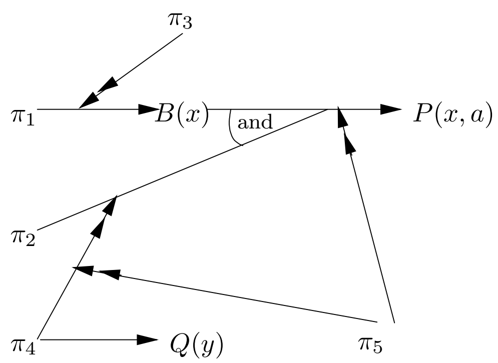


Fig. 8.

Remark 3. The perceptive reader might think that the model of arrow cancellations as presented in Figures 2, 3 and 4 is nothing special and is just a notational variant of defeasible logic with specificity. Thus using the notation of Example 4 we can write a defeasible database Δ with the following universal formulas clauses, with w, z universal variables.

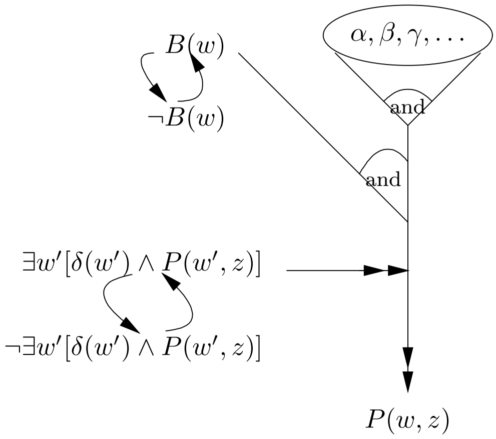
1. $B(w) \wedge \alpha \wedge \beta \wedge \gamma \to \neg P(w,z)$
2. $B(w) \wedge \alpha \wedge \beta \wedge \gamma \wedge \delta(w) \to P(w,z)$

If we instantiate (1) with $w = x, z = a$ and (2) with $w = y, z = b$ we get

1*. $B(x) \wedge \alpha \wedge \beta \wedge \gamma \to \neg P(x,a)$.
This is Figure 2 with $x =$ sink and $z =$ plumber a.
2*. $B(y) \wedge \alpha \wedge \beta \wedge \gamma \wedge \delta(y) \to P(y,b)$
This is Figure 3 with $y =$ toilet and $z =$ plumber b.

If we put (1*) and (2*) together in the same database and add the input $a = b = e$, then the database is consisent and the same plumber e will repair the toilet but not the sink. Defeasible logic based on specificity cannot tell us that because we have (2*) with $P(y,e)$ plumber e, we reverse and defeat (1*) and conclude $P(x,e)$ as well.

However, if we use the figures with the cancellation arrows, it is easier to model this feature. Figure 9 sums it all up. This is a predicate argumentation network involving joint attacks and higher order attacks, see [3].



$\alpha, \beta, \gamma, \ldots$
$B(w)$
$\neg B(w)$
and
and
$\exists w'[\delta(w') \wedge P(w', z)]$
$\neg\exists w'[\delta(w') \wedge P(w', z)]$
$P(w, z)$

Fig. 9.

3 Conclusion

We presented an outline paper showing how Talmudic logic uses a calculus of cancellation to execute identity merging. In this conclusion section we want to impress upon the reader the schematic advantage of the calculus of cancellation.

Suppose we have two clauses

1. $\alpha \wedge A \rightarrow \exists x C(x)$
2. $\beta \wedge B \rightarrow \exists x \neg D(x)$.

We want to put (1) and (2) together in the same database and

(a) maintain consistency
(b) have the existential quantifiers pick up the same element.

The mechanisms we use are

(i) to take the specificity formulas out of the clauses and consider them as meta-principles, which are subject to being prioritised and apply to them the calculus of cancellations.

So we have

3. $\{\alpha, \beta\} : A \wedge B \rightarrow \exists x C(x) \wedge \exists x \neg D(x)$.

(ii) Convert

$$\begin{array}{l} A \rightarrow \exists x C(x) \\ B \rightarrow \exists x \neg D(x) \end{array}$$

into respective figures and turn (3) into

4. $\{\alpha, \beta\}$: union of Figures.

In the process of taking union of Figures we get that $\exists x$ chooses the same x.

Acknowledgement

The authors are grateful to the referees for their most valuable comments.

References

1. C. E. Alchourròn, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50:510–530, 1985.
2. http://en.wikipedia.org/wiki/Belief_revision
3. D. Gabbay. Theory of Semi-instantiation in Abstract Argumentation. To appear *Logica Universalis*. http://arxiv.org/abs/1504.07020