The Implementation of UCLA OTP Constraints: Cases

Daniel M. Albro

May 15, 1999

1 Notes

1.1 Notation

A represents the antecedent of an implication constraint (a conjunction) or all of the arguments of a clash constraint (also a conjunction), B represents the consequent of an implication constraint (a disjunction), C represents "in or end A," and D represents "in or begin A." "In or end" means for each tier a_i in A, the expression $+]|a_1 \ldots +]|a_n$. If an expression is followed by prime (A'), this means that the expression has not undergone the appropriate transformations (see below).

1.2 Transformations

For all cases except the "unilevel" cases, A should be read as $A \wedge -[]|DEL \wedge -[]|INS \wedge -[]|RDEL$, where $-[]|DEL$ is added if A contains surface tiers, $-[]|INS$ is added if A contains underlying tiers, and $-[]|RDEL$ is added if A contains reduplicant tiers (so, if A contains tiers from some level of representation, the DEL tier is added in in this transformation.) In the "unilevel" cases, the above constituents of the form $-[]|t$ are replaced by constituents of the form $-[t$.

For mixed-level edge implication, edge-interval implication, multi-level edge implication, edge-interval clash, and multi-level edge clash, in cases where the reduplicant is to be avoided (these are cases where there is reduplication going on, and the constraint mentions some tier other than the surface tier), A is transformed further, beyond the above. If A contains [or |[elements, but no |,]|, or] elements, A should be read as $t(A) \wedge -]RED$, where $t(A)$ refers to the above transformation. If A contains] or]| elements, but no |, |[, or [elements, then A should be read as $t(A) \wedge -[RED$. Otherwise, A should be read as $t(A) \wedge -RED$.

In the situations where A is transformed to contain $-[]|t$ elements, B is transformed to read $(\vee(s(B)) \wedge -[]|DEL) \vee (\vee(u(B)) \wedge -[]|INS) \vee (\vee(r(B)) \wedge -[]|RDEL)$, where $s(B)$ is the surface constituents of B, and so forth. The constituents in the above of the form $-[]|t$ are replaced by $-[t$ in the unilevel situations. Finally, in the situations mentioned in the second paragraph, where

reduplication is being avoided, B is transformed to read

$$\begin{array}{c}(\vee(lf(s(B))) \wedge -[]|DEL \wedge -]RED)\vee \\ (\vee(rt(s(B))) \wedge -[]|DEL \wedge -[RED)\vee \\ (\vee(pl(s(B))) \wedge -[]|DEL \wedge -[]RED)\vee \\ (\vee(lf(u(B))) \wedge -[]|INS \wedge -]RED)\vee \\ (\vee(rt(u(B))) \wedge -[]|INS \wedge -[RED)\vee \\ (\vee(pl(u(B))) \wedge -[]|INS \wedge -[]RED)\vee \\ (\vee(lf(r(B))) \wedge -[]|RDEL \wedge -]RED)\vee \\ (\vee(rt(r(B))) \wedge -[]|RDEL \wedge -[RED)\vee \\ (\vee(pl(r(B))) \wedge -[]|RDEL \wedge -[]RED)\vee \end{array} \quad (1)$$

where $lf(B)$ is all constituents of B that have the symbol [in them somewhere, $rt(B)$ is all constituents that have] in them somewhere, and $pl(B)$ is all constituents that have + in them somewhere.

In the cases where A is transformed to $A \wedge -[]|t \ldots$, C and D undergo that same transformation.

1.3 Referenced Functions

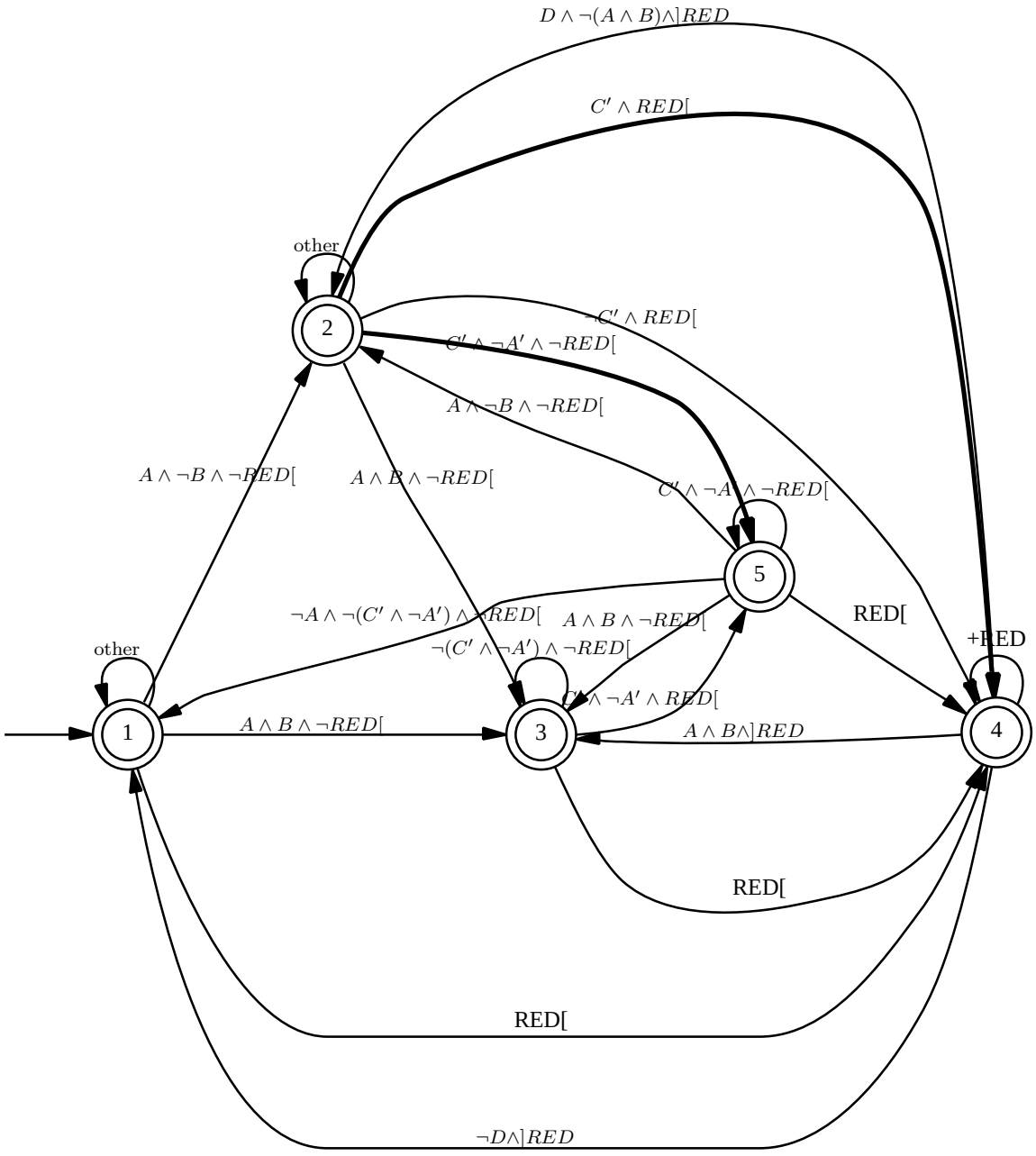
Note that A is the list of antecedents of an implication, or all arguments of a clash. It is implicitly considered to be linked with $\wedge$. B is the list of consequents of an implication, and is implicitly considered to be linked with $\vee$.

- u returns the underlying constituents of a list.
- s returns the surface constituents of a list.
- r returns the reduplicant constituents of a list.
- rs given a list of constituents, returns a new list as follows: If a constituent has], returns]. If a constituent has [, returns $-$]. If a constituent has +, returns +. Note that the initial constituents are all assumed to be from the set {+,]|, |[}.
- l given a list of constituents, returns a new list as follows: If a constituent has [, returns [. If a constituent has], returns $-$[. Otherwise no corresponding constituent is returned. Note that the initial constituents are all assumed to be from the set {+,]|, |[}.
- n is only defined upon lists of constituents that are all in the same level of representation. If those constituents are all surface, it returns the DEL tier, if underlying, returns the INS tier, and if reduplicant, it returns the $RDEL$ tier. That is, it returns the tier whose presence indicates that its operand doesn't exist.
- e given a list, returns the elements of that list that represent edges.

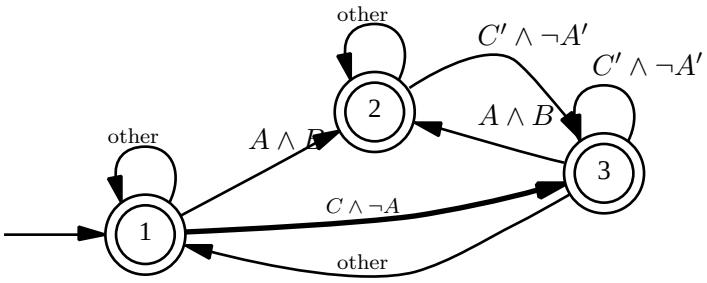
2 Interval Implication

This type of constraint is used for implications where A consists entirely of interiors.

2.0.1 Avoiding the Reduplicant



2.0.2 Not Avoiding the Reduplicant

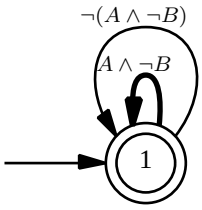


3 Edge Implication

These cases are used for implications where A contains at least one edge.

3.1 Unilevel

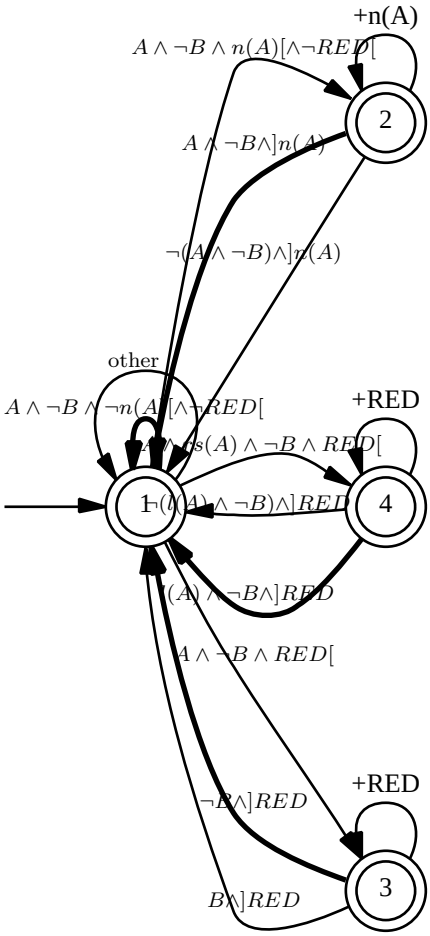
This case is used for implications where A contains an edge, and both A and B only mention tiers on a single level of representation.



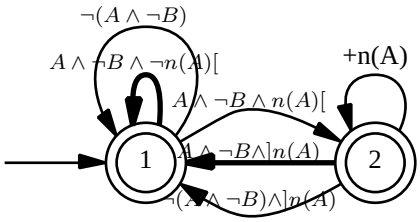
3.2 Mixed-Edge

These cases are used for implications where A contains an edge, and A only mentions tiers on a single level of representation, and B mentions some tier on a different level of representation.

3.2.1 Avoiding Reduplicant



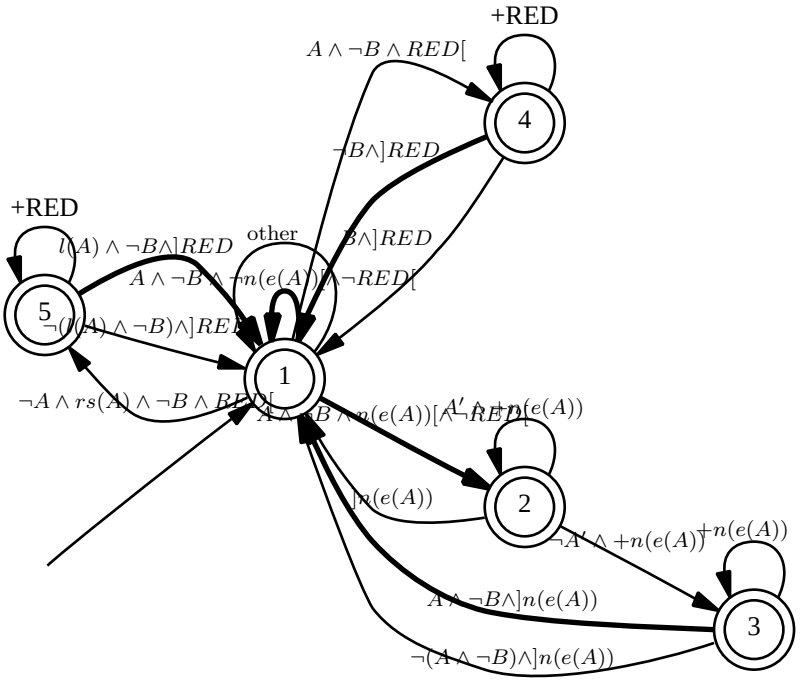
3.2.2 Not Avoiding Reduplicant



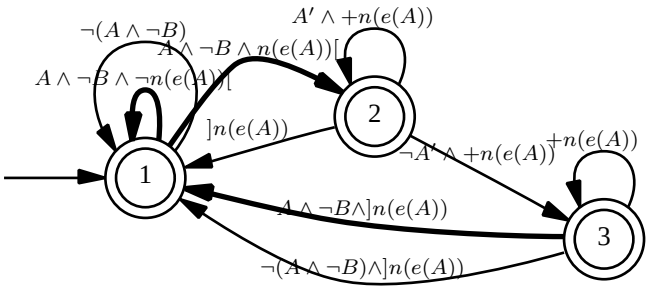
3.3 Edge-Interval Implication

This case is used when A mentions an edge, A mentions more than one level (no constraint may mention more than two levels, by the way), and on one of those levels all members of A are interiors.

3.3.1 Avoiding Reduplicant



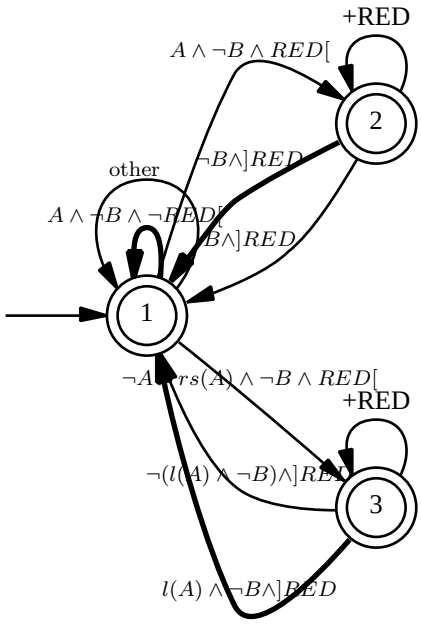
3.3.2 Not Avoiding Reduplicant



3.4 Multi-Level

This covers all other cases of implications in which A mentions an edge.

3.4.1 Avoiding the Reduplicant



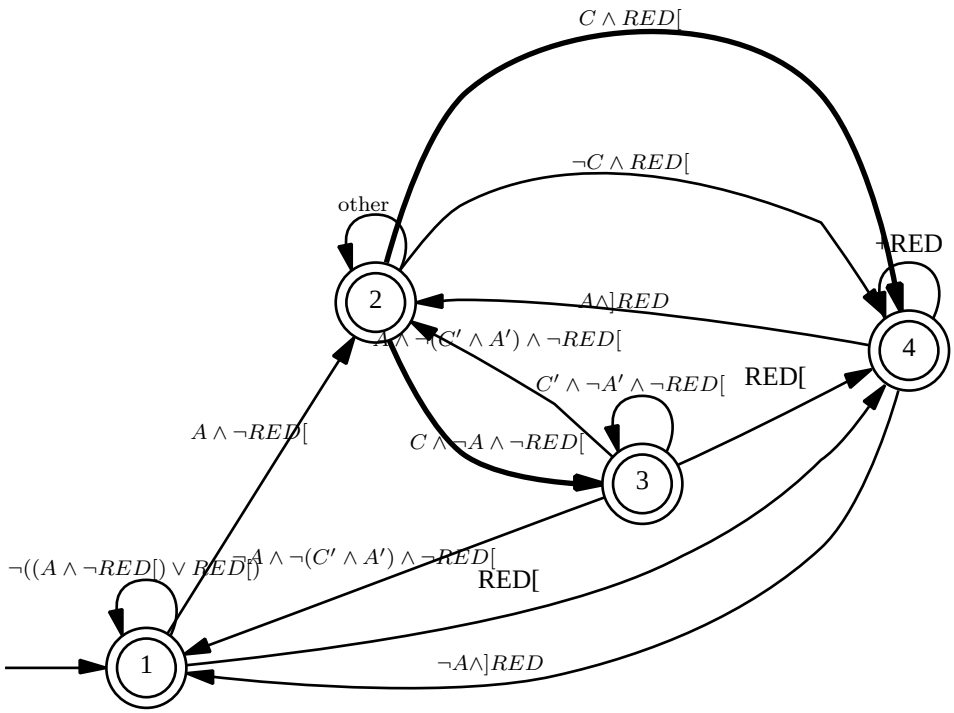
3.4.2 Not Avoiding the Reduplicant

Same as in Section 3.1, but with the multi-level transformations.

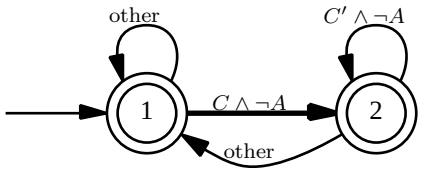
4 Interval Clash

This case covers the situation where the operation is clash, and A consists entirely of intervals.

4.0.3 Avoiding the Reduplicant



4.0.4 Not Avoiding the Reduplicant

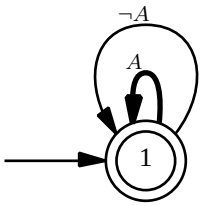


5 Edge Clash

These cases cover clashes where A mentions an edge.

5.1 Unilevel Edge Clash

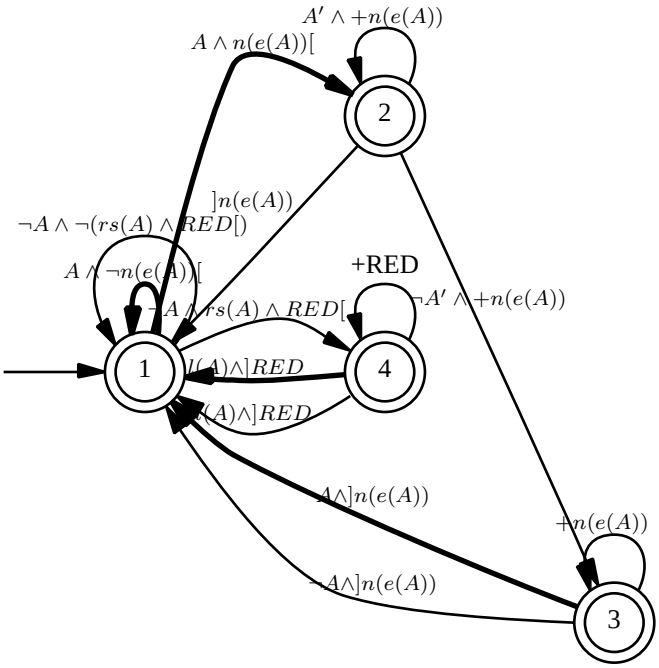
This case covers the situation where in a clash constraint A mentions an edge, and all of A is on a single level of representation.



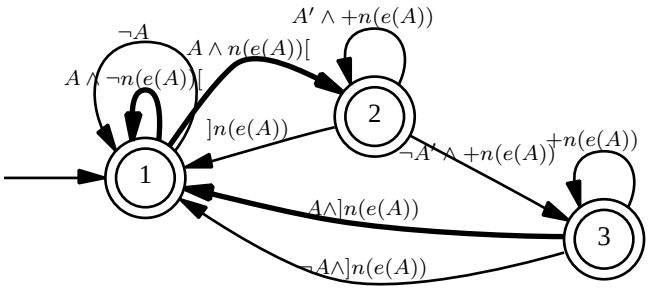
5.2 Edge-Interval Clash

This case covers the situation where in a clash constraint A mentions an edge, A mentions two different levels of representation, and on one of those levels of representation all elements of A are interiors.

5.2.1 Avoiding the Reduplicant



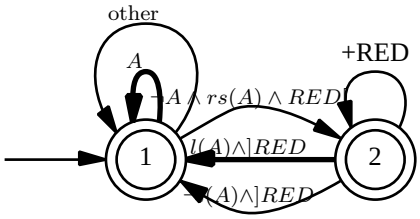
5.2.2 Not Avoiding the Reduplicant



5.3 (Other) Edge Clash

This covers all other clash constraints where A mentions an edge.

5.3.1 Avoiding the Reduplicant



5.3.2 Not Avoiding the Reduplicant

Like 5.1 but with the multi-level transformations.