Using QCA to Study Causal Order: Comment on Caren and Panofsky (2005)

Charles C. Ragin and Sarah Strand
Department of Sociology
University of Arizona
Tucson, Arizona 85721

The goal of qualitative comparative analysis (QCA) is to identify the different combinations of causally relevant conditions linked to an outcome. The researcher typically focuses on a qualitative outcome and seeks to identify the different conjunctural conditions that generate it. In this way QCA allows for causal complexity--for the possibility that no single cause may be either necessary or sufficient. Instead causes are viewed as *INUS* conditions: insufficient but necessary components of unnecessary but sufficient combinations of conditions (Mackie 1965). Caren and Panofsky (2005) seek to advance QCA by demonstrating that it can be used to study causal conditions that occur in sequences and introduce a technique they call TQCA (temporal qualitative comparative analysis). In their formulation the causal conjuncture is a *sequence* of conditions or events.

While we applaud their effort, in this comment we seek to clarify aspects of their analysis and to present a generalization of the approach that is more amenable to truth table analysis and use of existing software, fsQCA (Ragin 1987; 2000;

Ragin, Drass, and Davies 2006). Our first task is to correct what appear to be errors of omission in their analysis. Specifically, they seem to have stopped the process of logical minimization short of completion. We show that it is possible to produce a logically simpler solution than the one they present, while still remaining true to the principles they advocate. Our second task is to demonstrate how to use fsQCA to implement a generalization of their procedure. This procedure takes advantage of an under-utilized feature of fsQCA software, namely, the facility in crisp-set analyses to code a causal condition not only as "present" versus "absent," but also as "irrelevant." The coding of "irrelevant" is especially important in analyses of event sequences, where event order is relevant only if the events actually occur. Thus, the question, "Which came first, event A or event B?" is relevant only if both A and B are coded "present."

Brief Summary of Caren and Panofsky's TQCA

TQCA considers the temporal order of events that jointly contribute to an outcome. Consider conditions *A* and *B*, whose temporal order may have differential effects on the outcome. Notationally, we use a slash to indicate temporal order; *A/B* should be read as "A then B." Combinations *A/B*, *A/b*, *a/B*, *a/b*, and *B/A* are the five possible configurations for these two conditions. Because the order of events that *did not* occur is impossible to identify, in general TQCA only considers ordering as it applies to conditions that occurred. Thus, *b/A*, *b/a*, and *B/a* are temporally equivalent to *A/b*, *a/b*, and *a/B,* respectively. Furthermore, in order to limit the number of possible combinations for both substantive and practical concerns, the researcher may fix one or more conditions in specified temporal positions, such as always occurring first or always occurring last in a sequence.

TQCA minimization follows three basic steps and is similar to that of QCA. The researcher must first examine whether or not temporal boundaries are necessary. When configurations are identical except for the temporal ordering of two causal conditions, this ordering may be removed by replacing the slash (/) with an asterisk (*), indicating logical *and*. If both *A*/*B* and *B*/*A* lead to presence of the outcome, then the ordering must be irrelevant. The configuration may then be rewritten as *A***B* (or *B***A*; the two are equivalent). Second, the researcher utilizes QCA rules to minimize conditions within each temporal block. Some configurations will be identical, with the exception of the presence or absence of a single causal condition within a temporal block, such as *A/B*c* and *A/B*C*. In this instance, the varying attribute may be removed. Following the logic of QCA, if the same outcome is achieved regardless of a condition's presence or absence within a temporal block, the condition itself must be inconsequential. *A*/*B***c* and *A*/*B***C* may be minimized to *A*/*B*, because the outcome is the same regardless of *C*'s presence or absence. Finally, minimization cannot cross temporal blocks; this would render null the consideration of temporality by indicating that time order is immaterial.

However, conditions may be factored as long as temporal boundaries are respected. *A*/*B*/*D* and *A*/*B*/*C*/*d* can be factored into *A*/*B*/[(*C*/*d*) + *D*], which is read as *A*, then *B*, then (*C* then *d*) *or* then *D*. Although *B* is present in both original configurations, it may not be factored out by itself because its location within a temporal block is fixed.

To demonstrate their method, Caren and Panofsky (2005) use a hypothetical data set in which they evaluate the causal conditions leading to university recognition of a graduate student union (RECOGNITION--recognition). Their causal conditions include public verus private university (PUBLIC--public), graduate student elite allies (ELITE--elite), national union affiliation (AFFILIATE--affiliate), and strike or threat of strike (STRIKE--strike). PUBLIC--public is fixed in the constant initial position because this condition is more than likely established before all others; STRIKE--strike is fixed in the final position because it is probably a last step before university recognition. When both ELITE and AFFILIATE are present, they are allowed to vary between the second and third positions in the configuration; when one or both are absent, the temporal order does not matter and * (logical *and*) notation is used (as in elite*AFFILIATE). The introduction of time sequencing increases the number of logically possible configurations from sixteen to twenty. Cases are reassigned to configurations in order to proceed with minimization.

[Table 1 about here.]

Caren and Panofsky apply TQCA minimization techniques to the hypothetical data on recognition of graduate student unions, shown in Table 1. They summarize their results as follows:

RECOGNITION = PUBLIC / ELITE / STRIKE +
PUBLIC / ELITE / AFFILIATE / strike +
public / AFFILIATE / ELITE / STRIKE

In other words, recognition of graduate student unions is achieved under three sets of conditions: (1) when public university graduate students gain elite allies and then strike or threaten to strike; (2) when public university graduate students gain elite allies, then become affiliated with a national union, but then refrain from threatening a strike; or (3) when private university students become affiliated with a national union, then gain elite affiliates, and finally strike or threaten to strike.

Limitations of Their Analysis

While their results appear to follow directly from the procedures they advocate, further logical simplification is possible, yielding a more parsimonious solution. Specifically, Caren and Panofsky's second and third reduced paths each contain one more causal condition than is logically required. While they claim to follow QCA logic in their minimization process, they stop short of full simplification. Without explanation, Caren and Panofsky limit themselves by utilizing each original configuration only once in the minimization process. Given three configurations *AbC*, *Abc*, and *abC*, the first, *AbC,* may combine with *Abc* to create *Ab*; *AbC* may also combine with *abC* to create *bC* (see minimization example in Ragin 1987:94). Original configurations can and should certainly be used more than once in the minimization process if possible. Using Caren and Panofsky's hypothetical data, PUBLIC/AFFILIATE/ELITE/STRIKE can be combined not only with public/AFFILIATE/ELITE/STRIKE to generate AFFILIATE/ELITE/STRIKE; it also can be combined with PUBLIC/ELITE/AFFILIATE/STRIKE to form PUBLIC/ELITE*AFFILIATE/STRIKE. Allowing multiple uses of the original configurations results in a simpler solution. Thus, their two four-condition "final" configurations can each be minimized further, yielding a final solution with three three-condition configurations. We illustrate the contrast between our analysis and Caren and Panofsky's in Figures 1 and 2.

[Figures 1 and 2 about here.]

We summarize our results algebraically as follows:

RECOGNITION = PUBLIC / ELITE / AFFILIATE +

AFFILIATE / ELITE / STRIKE +

PUBLIC / ELITE / STRIKE

In other words, university recognition of graduate student unions occurs following three paths: (1) when public university students gain elite allies and then become affiliated with a union; (2) when graduate students (either public or private) become affiliated with a union, gain elite allies, and then strike or threaten to strike; or (3) when public university students gain elite allies and then threaten to strike. The first and third terms can be factored, following Caren and Panofsky's recommended procedures to show:

RECOGNITION = PUBLIC / ELITE / (AFFILIATE + STRIKE) +

AFFILIATE / ELITE / STRIKE

Generalization of TQCA Using Truth Table Analysis

Caren and Panofsky's general recommendations regarding the analysis of sequenced events are sound--if carried to their proper conclusion. One way to ensure that this is achieved is to utilize the existing fsQCA software for the analysis of the evidence and not rely on paper-and-pencil procedures. It is also important to note that the greater the number of causal conditions in a sequence, the greater the likelihood of human error.

The possibility of studying event sequences using QCA was first mentioned when the method was introduced. Ragin (1987:162) notes that it is possible to include information about event sequencing as causal conditions in a conventional truth table. For example, a causal condition could be coded as "true" (present) if event A occurred before event B; a coding of "false" (absent) in this instance would indicate that event B occurred before event A. Using Caren and Panofsky's hypothetical data set to illustrate, the order of AFFILIATE and ELITE can be addressed by creating an additional condition in the truth table, designated for convenience as E_BEFORE_A. This attribute is present (E_BEFORE_A) when ELITE occurs before AFFILIATE; it is absent (e_before_a) when AFFILIATE occurs before ELITE. Because the temporality of ELITE and AFFILIATE is not always an issue (i.e., when either condition is absent), E_BEFORE_A is assigned presence or absence only for relevant combinations--those with both conditions present. In the remaining combinations (i.e., when either condition is absent) the "don't care" coding is used, indicating that the condition is irrelevant. When coding a truth table in fsCQA, the dash ("-") is used to indicate the "don't care" coding. (Unfortunately, Caren and Panofsky use this code to indicate temporal sequence.)

[Table 2 about here.]

The reformulated data set is shown in Table 2, which is based directly on the information provided in Table 1. Note that the condition E_BEFORE_A is most commonly coded "don't care" (using the dash) because most cases do not have both ELITE and AFFILATE present. In fsQCA, this data set can be simplified to a logical equation showing the combinations of conditions linked to the outcome using the "crisp-set/Quine" procedure, with positive outcomes (union recognized) set to "true" and negative outcomes (union not recognized) set to "false" in the specification panel of the QUINE procedure. The results of this analysis show:

RECOGNITION = PUBLIC*ELITE*STRIKE +

ELITE*AFFILIATE*STRIKE*e_before_a +

PUBLIC*ELITE*AFFILIATE*E_BEFORE_A

This equation can be represented using sequencing notation as:

RECOGNITION = PUBLIC / ELITE / STRIKE +

AFFILIATE / ELITE / STRIKE +

PUBLIC / ELITE / AFFILIATE

This result is the same as the one we demonstrated using the pencil-and-paper procedure shown in Figure 2. This direct coding of event sequences (i.e., adding the condition E_BEFORE_A to the truth table) along with use of the "don't care" code greatly simplifies the examination of event sequences and reduces the problem of human error.

Please note that this procedure, like Caren and Panofsky's, becomes progressively unwieldy with every increment to the number of events to be sequenced. When there are three events (say, A, B, and C) to be sequenced, three event pairings are added to the truth table (A before B, B before C, and A before C). The possible codings of these three event pairings (with all three events present) and their associated sequences are shown in Table 3.

[Table 3 about here.]

With three events, only two out of the eight combinations yield intransitive (and therefore meaningless) sequences. With four events, the number of event pairings to be added to the truth table is six. (The formula for number of event pairings to be added to the truth table is (k*(k-1))/2, where k is the number of events.) However, adding six event pairings to a truth table greatly increases the number of logically possible combinations ($2^6 = 64$), and most of these combinations yield intransitive (i.e., meaningless) sequences. The number of combinations that are transitive, with four events (k = 4), is k! or 24. Thus, only 24 of the 64 logically possible combinations generated by the six event pairings (using four events) are empirically plausible. Analyzing the possible sequences of four events using QCA is difficult but not impossible; more than four would be very difficult indeed. Thus, the recommendation is to limit the number of sequenced events to two, three, or possibly four. The strategy of fixing events or conditions in specific locations (e.g., as first or last) as suggested by Caren and Panofsky is especially valuable and useful as a way to simplify the study of event sequences using QCA. It is also important to note that often all that matter is the timing of specific *pairs* of events, which is quite manageable with fsQCA, for each pair of events studied results in only a single new column in the truth table. A researcher, for example, might be interested in whether A preceded B or the reverse and whether C preceded D or the reverse, but uninterested in other possible event pairings (e.g., of B and C).

Conclusion

Caren and Panofsky offer some promising leads for the analysis of event sequences using QCA. However, they also demonstrate the hazards of relying on pencil-and-paper techniques. In this comment, we show how to correctly implement the analysis of simple sequence data using QCA. We also demonstrate the use of the "don't care" facility in QCA, an important but under-utilized feature, through which simple temporal sequences may be incorporated into analysis. Consideration of temporality adds another dimension of depth to QCA, and we encourage its appropriate use.

References

Caren, Neal and Aaron Panofsky. 2005. "TQCA: A Technique for Adding Temporality to Qualitative Comparative Analysis." *Sociological Methods & Research* 34(2): 147-172.

Mackie, John L. 1965. "Causes and Conditionals." *American Philosophical Quarterly* 2: 245-65.

Ragin, Charles C. 1987. *The Comparative Method: Moving Beyond Qualitative and Quantitative Strategies*. Berkeley, CA: University of California Press.

Ragin, Charles C. 2000. *Fuzzy-Set Social Science.* Chicago, IL: University Chicago Press.

Ragin, Charles C., Kriss A. Drass and Sean Davey. 2006. *Fuzzy-Set/Qualitative Comparative Analysis 2.0*. Tucson, Arizona: Department of Sociology, University of Arizona.

Table 1: TQCA Configurations for graduate student union recognition, from Caren and Panofsky (2005:160)

Number	Combination	Cases	Recognition
1	PUBLIC / ELITE / AFFILIATE / STRIKE	2	2/2
2	PUBLIC / AFFILIATE / ELITE / STRIKE	1	1/1
3	PUBLIC / ELITE/AFFILIATE / strike	2	2/2
4	PUBLIC / AFFILIATE/ELITE / strike	0	?
5	PUBLIC / ELITE*affiliate / STRIKE	1	1/1
6	PUBLIC/ ELITE*affiliate / strike	0	?
7	PUBLIC/ elite*AFFILIATE / STRIKE	2	0/2
8	PUBLIC/ elite*AFFILIATE / strike	0	?
9	PUBLIC/ elite*affiliate / STRIKE	3	0/3
10	PUBLIC/ elite*affiliate / strike	1	0/1
11	public / ELITE / AFFILIATE / STRIKE	0	?
12	public / AFFILIATE / ELITE / STRIKE	1	1/1
13	public / ELITE / AFFILIATE / strike	0	?
14	public / AFFILIATE / ELITE / strike	0	?
15	public / ELITE*affiliate / STRIKE	0	?
16	public / ELITE*affiliate / strike	1	0/1
17	public / elite*AFFILIATE / STRIKE	0	?
18	public / elite*AFFILIATE / strike	0	?
19	public / elite*affiliate / STRIKE	0	?
20	public / elite*affiliate / strike	3	0/3

Table 2: Truth table for Caren and Panofsky's Data, using "don't care" coding

Number	Condition					Cases	Recognition
	PUBLIC	*ELITE*	*AFFILIATE*	*STRIKE*	*E_BEFORE_A*		
1	1	1	1	1	1	2	1
2	1	1	1	1	0	1	1
3	1	1	1	0	1	2	1
4	1	1	1	0	0	0	?
5	1	1	0	1	-	1	1
6	1	1	0	0	-	0	?
7	1	0	1	1	-	2	0
8	1	0	1	0	-	0	?
9	1	0	0	1	-	3	0
10	1	0	0	0	-	1	0
11	0	1	1	1	1	0	?
12	0	1	1	1	0	1	1
13	0	1	1	0	1	0	?
14	0	1	1	0	0	0	?
15	0	1	0	1	-	0	?
16	0	1	0	0	-	1	0
17	0	0	1	1	-	0	?
18	0	0	1	0	-	0	?
19	0	0	0	1	-	0	?
20	0	0	0	0	-	3	0

Table 3: Event pairings and sequences for three sequenced events

A_BEFORE_B	*A_BEFORE_C*	*B_BEFORE_C*	Event Sequence
1	1	1	A/B/C
1	1	0	A/C/B
1	0	0	C/A/B
1	0	1	Intransitive
0	1	1	B/A/C
0	1	0	Intransitive
0	0	1	B/C/A
0	0	0	C/B/A

Figure 1: Caren and Panofsky's Minimization, using TQCA

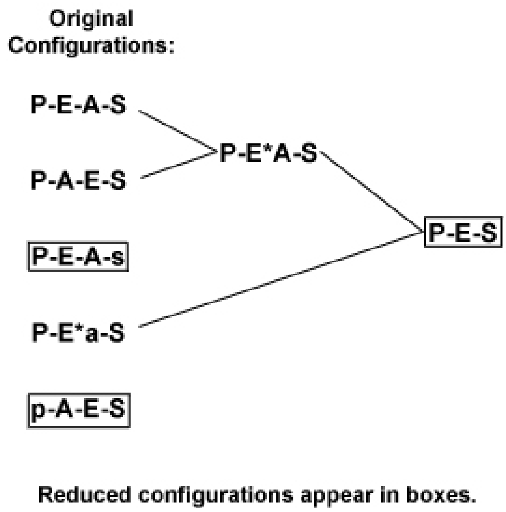
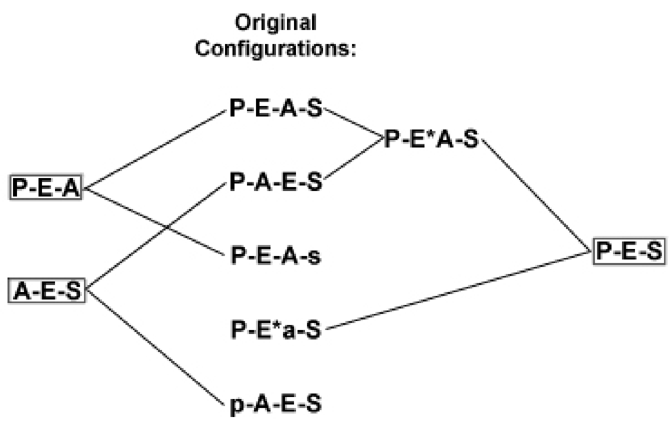


Figure 2: Our minimization, using TQCA



Reduced configurations appear in boxes.