Review of the paper "**Epidemic Algorithm for Replicated Database Maintenance**" by "Alan Demers et al."

Reviewer: Dayuan Tan

1. Comprehension.

- Motivation and problem considered

  Maintain consistency of a replicated key-value database (eg. domain names to machine identities) so that to reduce network traffic due to replica consistency.

- Innovations/contributions made

  There are two innovations:

  - The proposed algorithms only rely on *eventual delivery* of repeated messages, while previous approaches ask for guaranteed underlying communication protocols and consistent distributed control structures.
  - The proposed algorithms are *randomized* while the previous algorithms are *deterministic*. This makes the probability of non-converged information decreases exponentially with time, and only simple data structure is needed to implement.

- Methods/arguments used or developed

  （Prioritize: focus on most important arguments/methods: eg anti-entropy and rumor morgering; deletion is useful, but kinda of an after thought.）

  1. Anti-entropy:

Each node (site) uniformly randomly chooses another node (site) and solve the difference between them. The site with newer timestamp will send its data to overwrite another site and update its timestamp.

Stop until all are updated.

The anti-entropy is expensive, so it is executed only periodically on each site with a carefully selected time window, and usually work as a back-up distribution method.

2. Rumor mongering:

Each updated (infective) site (called hot rumor) randomly chooses another site and ensure it to be updated, and then repeats.

Those hot rumors, whose recently connected sites are already updated, become inactive. The system stops when all sites are updated or inactive.

3. Difference between anti-entropy and Rumor mongering:

- **Initiator**: any site can push or pull to compare with another site and solve the difference if there is in anti-entropy; but only hot rumor (updated site) take initiative in rumor mongering.
- **Number of each site's connected sites**: each site connects to all other sites in anti-entropy; but hot rumor only connects with a limited number of other sites before it becomes inactive.
- **Traffic load**: If there are n sites and start with one updated site, anti-entropy needs n(n-1) times comparisons and sends log2(n) + ln(n) + O(1) copies of data when using push; while rumor mongering needs to send log2(n) + w copies where w is determined by how often a rumor hot become inactive.
- **Reliable**: Anti-entropy is reliable while rumor mongering is not (when k = 1 there are 20% sites miss the update while k = 2 there are 6%

Other reference:

- Distribution
  1. Direct mail (used)
     - Each node broadcast updates to all others.
     - Can be considered as broadcast.

    2. Anti-entropy (used)
        - Each node randomly chooses another node and fix the difference between each other.
        - Can be considered as random unicast.
    3. Rumor mongering (proposed)
        - Each node randomly chooses multiple nodes and share the updates.
        - Can be thought as random multicast.
    4. Rumor mongering backed up by anti-entropy (proposed)
        - Each node randomly chooses multiple nodes and share the updates. (a.k.a., initial distribution.)
        - Then do anti-entropy to ensure no susceptible nodes left. If found, then rumor mongering again (a.k.a., redistribution).
- Deletion
    1. Dormant death certificate with two time thresholds (t1, t2) and retention site list (proposed)
        - If first time threshold is reached (> t1), most server delete its death certificate. Only those servers which are on retention site list don't delete the death certificates, instead, save them as dormant death certificates.
        - If second time threshold is reached (> t1 + t2), dormant death certificates are discarded.
    2. Dormant death certificate with two time thresholds (t1, t2), retention site list and activation timestamp (proposed)
        - If original timestamp >= activation timestamp, then as same as above
        - Else:
            - If activation timestamp < t1, death certificates can be reactivated.
            - Else if activation timestamp >= t1, death certificates will be deleted, dormant death certificates keep.
            - Else if activation timestamp >= t1 + t2, dormant death certificates will be deleted.
        - Anti-entropy and rumor mongering work equally well for this activation timestamp mechanism.
- Spatial distribution
    1. A spatial distribution (1/Q(d^2)) with anti-entropy was used for Clearinghouse service.
    2. Anti-entropy is more robust than rumor mongering against changes in spatial contribution.

- Conclusions of the paper
    - Using randomized approaches for replicated database consistency problem, replacing deterministic approaches, works and has better performance in achieving consistency and reducing network traffic.
    - A well-chosen spatial distribution with anti-entropy reduces average link traffic.
    - Using rumor mongering as the initial update distribution algorithm, backed up by anti-entropy to cover those nodes which didn't receive the update in the first-round distribution, has a good performance for achieving consistency for CIN.
    - Death certificate is necessary for solving the obsolete data recurrence problem and dormant death certificate is a good trade-off between death certificates retention time and consumed space.

2. Critique.

Discuss 2-3 important points of disagreement with the paper. Carefully articulate your point of view, and provide support with appropriate technical/logical arguments and evidence.

- 1st point

    In section 1.3 anti-entropy, the bottleneck of anti-entropy approach is that it asks for comparing the two databases which consumes many networks traffic. The author provided two solutions for this problem, i.e., checksum method with well-chosen time window and inverted index of database timestamps.

    For the first solution, how to choose a good time window is still a problem

    For the second solution, it is very expensive to maintain the inverted index of database timestamps in practice.

    So, this bottleneck is still not solved properly.

- 2nd point

    In section 1.4 complex epidemics – variations – counter vs coin, the author claimed that counters and feedback improve the delay while counters playing a more significant role than feedback. But no related simulation results were displayed.

Similarly, no results comparison was displayed when the authors claim the counter and feedback variations improve the residue and counters are more important.

- 3rd point (optional)

  Results in Table 6 and Table 4, 5 cannot be compared directly.

- 4th point

  The spatial distribution discussed in the paper is only mesh structure distribution. How about other spatial distributions?

3. Synthesis.

Propose 1-2 directions that the work can be further extended. Provide arguments to support your proposal. [eg motivate the proposal, discuss the significance of the contributions of the extension if successful, outline methods/approaches you would use to materialize your proposal]

- 1st proposal
  - The paper mentioned that there is a critical link between Europe and North American. There are tens of sites in Europe and hundreds of sites in North American. What about we consider them as two sub-networks? We ensure the consistency inside each sub-network and then only one traffic is needed to synchronize the two sub-networks?
- 2nd proposal (optional)
  - Discussion about other spatial distribution may be needed.