Simplifying Simulation of Distributed Datastores Based on Statistical Estimating CAP-Constraint Violation

Viktor Sobol

School of Mathematics and Computer Science,
V.N. Karazin Kharkiv National University
4, Svobody Sqr., Kharkiv, 61022, Ukraine
viktor.pdt@gmail.com

Abstract. Running software in a distributed manner is a common practice nowadays. This approach produces a lot of new challenges which should be thought in advance. This paper is the next step on understanding systems such as distributed datastores by using statistical estimation for violation of guarantees from Brewer's Conjecture. The paper focuses on finding ways for simplification of theoretical and practical modelling of a system of a distributed datastore. Considering that real-world distributed datastore consists of nodes with a different distribution of fail and recovery time it is proposed to substitute every node of distributed datastore with nodes with one common distribution of fail and one common distribution of recovery time. The verification of the approach is done by modelling systems and statistically comparing their violation of guarantees from Brewer's Conjecture. The results allow us to define cases where we can substitute one system with another without losing perception of its behaviour.

Keywords: Distributed datastore · Partition-tolerance · CAP-theorem · statistic metrics.

1 Introduction

Modern databases store data with multiple copies and often replicate it to different geographical regions to support the efficient decision-making process. However, processing and storing data in a distributed environment produce a lot of new challenges and limitations which didn't exist in a single-computer world. Some of those limitations were formalized as Brewer's Conjecture [5] or more often it is referred as CAP-theorem. CAP-theorem is often used as a tool to reason about trade-offs in practical systems [6]. However such reasoning provides some ambiguities and was reviewed by Martin Kleppmann [6] showing that it isn't enough to justify system behaviour by relying only on CAP theorem and he recommends to avoid using it for making design decisions.

Paper [2] proposes an alternative probabilistic approach to measure CAP properties of a distributed datastore. The proposed way opens up the opportunities for studying CAP properties fulfilment to give a good assumption about

the behaviour of practical systems. As authors noted, for future research of the proposed approach it is needed to represent experimental results hence the model of a distributed datastore has to be simulated. From the definition, distributed datastore consists of a set of nodes connected by links. Each node has the probabilistic distribution of recovering time and time of failure. The mentioned above consideration was taken from [2]. Real-world datastores consist of different nodes following from that the distribution of fail and recovery time is different for every node. The way of simplification explored in this paper is to simulate a system of a distributed datastore with nodes having the same probabilistic distribution of fail and recovery time and compare statistical estimation for violation with a system which consists of nodes with different characteristics and to define cases where mentioned substitution won't change the whole system behaviour.

Obviously links that bind nodes may fail too but this is outside of the scope of the next analysis and it is assumed that every link if always works correctly.

2 Distributed datastore model

To conduct a simulation mathematical model of a distributed datastore defined in [2] was taken. The model by itself is a tuple (N, L, ∂, D, r), where:

N	is a finite set whose elements corresponds to nodes of a distributed datastore;
L	is a finite set whose elements corresponds to links of a distributed datastore;
$\partial\colon L \to 2^N$	is a mapping which associates each link with two nodes that it connects;
D	is a finite set whose elements corresponds to stored data units;
$r\colon D \to 2^N$	is a mapping which associates each data unit with a subset of nodes that store its replica;

Experiments in this paper focus on estimation of the third guarantee from Brewer's Conjecture — partition tolerance. The metric used to estimate partition tolerance is based on Definition 4 in [2] as a random variable that represent two possible events at a time point $t \geqslant 0$

$\zeta_t = 0$	there exist data unit which is not reachable by some node from subset of alive nodes;
$\zeta_t = 1$	all data units are reachable from any alive node;

Figure 1 is an example of $\zeta_t = 1$ as clearly every alive node is able to reach any other alive node (a non-alive node is represented by a red-filled circle). Figure 2 is an example of $\zeta_t = 0$. As node 1 is not able to reach any other alive node.

It is assumed that the distribution of fail and recovery time for every node obeys Poisson distribution. As we can treat time in our experiments as a discrete quantity, events — failure and recovery as a series of events with the known average time but the exact time is random. From practical examples, it may be

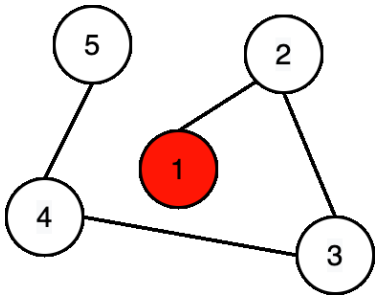

Fig. 1. System is in state *up*

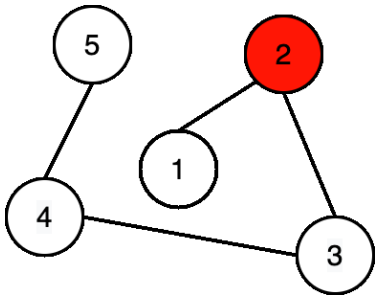

Fig. 2. System is in state *down*

the hardware failures, periodic releases of newer versions of software, etc. Saying that some node has a different distribution comparing with some other node it is understood as both of them are having Poisson distribution but parameters of Poisson distribution are different.

In order, to simplify future simulating of a distributed datastore and theoretical reasoning it is proposed to substitute probabilistic distribution of every node with one common distribution. Empirical results to answer a question — when ζ_t for a model of distributed datastore with every node having a different probabilistic distribution of fail and recovery time is not statistically different from the model with nodes having the same distribution are presented below.

3 Experiment description

As was mentioned above in the set of experiments, time is considered as a discrete quantity.

Required steps before every experiment run:

- Generation of graph (N, L) which corresponds to nodes and links of a distributed datastore:
- Generation of parameters for Poisson distribution of fail and recovery time for every node from graph (N, L) for system having different parameters of distribution — $S_{original}$
- Generation of parameters for Poisson distribution of fail and recovery time for every node from graph (N, L) for system having different parameters of distribution — S_{ideal}
- Define discrete time series — T

To generate a graph, WattsStrogatz model [3] was used. The model produces a random graph with small-world properties what is well represent real-world systems especially distributed across different geographical locations. WattsStrogatz model is using next parameter defined below to generate a graph:

Pr	probability of rewiring in WattsStrogatz model;
K	degree in WattsStrogatz model;

The next steps are taken to define parameters for both systems — $S_{original}$ and S_{ideal}:

- define $\mathsf{E}(fail)$ and $\mathsf{E}(recovery)$ for S_{ideal}. Mentioned values are the input data of an experiment;
- for every node from $S_{original}$ calculate random values — $deviation_{recovery}$ and $deviation_{fail}$ based on normal distribution $N(0, \mathsf{E}(recovery)/2)$ and $N(0, \mathsf{E}(fail) * (deviation)$ respectively. Define $\mathsf{E}(fail - deviation_{fail})$ and $\mathsf{E}(recovery - deviation_{recovery})$ for $S_{original}$. Where $deviation$ is an input parameter of an experiment;

During experiment run for every $t \in T$ the value for ζ is recorded based on Definition 4 from [2] for both system, $S_{original}$ and S_{ideal}.

The process is repeated for different randomly generated network structures with the same parameters and then values for random variable ζ is aggregated by the number of times when the system had $\zeta = 1$ and frequency of this systems. The mentioned aggregations are stored separately for $S_{original}$ and S_{ideal}. Pearson test(χ^2) with $p - value > 0.05$ is applied to obtained data to get a conclusion about the statistical difference in systems behaviour. Every aggregated result is stored in a bucket of size — S based on a number of times. Bucketing is done in order to eliminate inaccuracies in cases when time series is long enough and the difference between two values is small enough to be treated as one value for χ^2 test.

4 Experiment results

In total 5000 different systems were generated and run. Below are the results of the most notable cases.

4.1 Experiment 1

Parameters used in this experiment are shown in Table 1.

Distributions of a results are shown in Figure 3 and Figure 4. Shapiro-Wilk Test [4] was applied to these distributions with a positive outcome. The results were aggregated using different bucket size — S and χ^2 test was applied. The outcome is shown in Table 2

4.2 Experiment 2

Parameters for experiment 2 are presented in Table 3. Distributions of tested systems are shown in Figure 5 and Figure 6. Positive outcome is received on Shapiro-Wilk test. χ^2 test results are presented in Table 3. In this case it is possible to conclude that big $\mathsf{E}(fail)$ in comparison to $\mathsf{E}(recovery)$ highly contributes to χ^2 test results.

Table 1. Parameters for Experiment 1

Parameters	
Name	Value
Pr	0.5
K	5
$\mathsf{E}(fail)$	300
$\mathsf{E}(recovery)$	30
$deviation$	0.5

Table 2. Results for Experiment 1

Test results	
S	χ^2 results
10	0.0093
20	0.013
30	0.5

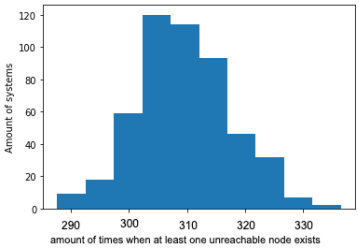

Fig. 3. Original system

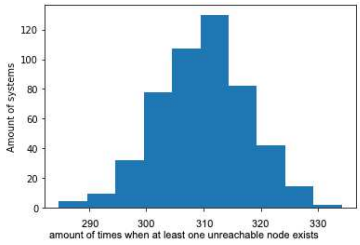

Fig. 4. Ideal system

4.3 Experiment 3

The parameters used in experiment 3 are shown in Table 5. Distribution results for this experiment are shown in Figure 7 and Figure 8. Shapiro-Wilk Test [4] was applied to the mentioned distribution as well and gave a positive outcome. χ^2 test results can be seen in Table 6. Good results from χ^2 can be explained by small *deviation* value.

5 Result interpretation

Based on experiment results, it is possible to specify the next parameters which results of χ^2 test depends on hence answer to a question — are the system statistically equal based on Definition 4 from [2].

– The ratio of expected values of fail and recovery time:

$$AR = \frac{\mathsf{E}(recovery)}{\mathsf{E}(fail)} \tag{1}$$

– The average deviations of all components considering that $\nu(N, L)$ - is a number of vertices in graph (N, L):

$$AD = \frac{\sum_1^{\nu(N,L)} deviation}{\nu(N, L)} \tag{2}$$

Table 3. Parameters for Experiment 2

Parameters	
Name	Value
Pr	0.3
K	5
$\mathsf{E}(fail)$	300
$\mathsf{E}(recovery)$	5
$deviation$	0.5

Table 4. Results for Experiment 2

Test results	
S	χ^2 results
10	0.3281
20	0.7203
30	0.82
40	0.94

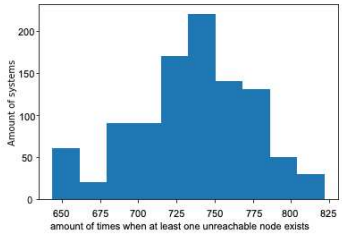

Fig. 5. Original system

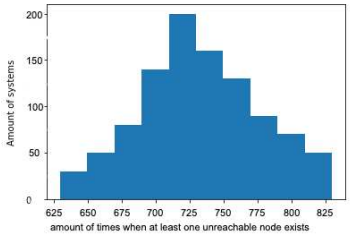

Fig. 6. Ideal system

– Graph clustering, the definition is taken as Barrat and Weigt [1] measure for clustering:

$$C'(\beta) \tag{3}$$

Experiments showed that the result of χ^2 test has an inverse relation to the specified above parameters. The bucket size — S has a direct relation to χ^2 test, as we treat close to each other values as one value for χ^2 test.

Based on relation each parameter has to the end result it is possible to define one proportion that represents the influence of each parameter.

$$\frac{S}{AR * AD * C'(\beta)} \tag{4}$$

6 Conclusion

The study described in the paper was focused on improving and simplifying tackling challenges set up in [2].

The proposed in the paper relation (4) based on empirical data gives a practically useful tool concerning to model a real-world system (referred in this paper as $S_{original}$) as a system with nodes having the same distribution (referred in the paper as S_{ideal}.

Table 5. Parameters for Experiment 3

Parameters	
Name	Value
Pr	0.5
K	5
$\mathsf{E}(fail)$	300
$\mathsf{E}(recovery)$	30
$deviation$	0.2

Table 6. Results for Experiment 3

Test results	
Step	χ^2 results
10	0.1293
20	0.3072
30	0.5412
40	0.6378

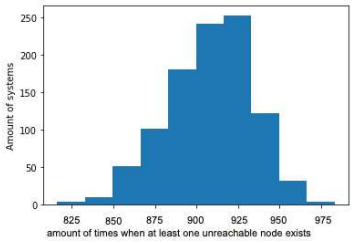

Fig. 7. Original system

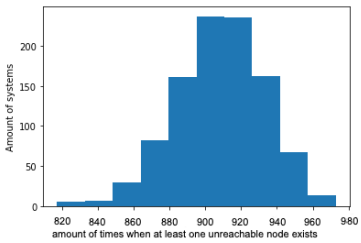

Fig. 8. Ideal system

This relation is grounded by a big set of experiments was conducted however it can be improved using coefficients in order to make the influence of each of the parameters more explicit.

We plan to use relation (4) for studying different mechanisms to estimate the degree of ensuring each CAP-guarantee applying metrics specified in [2].

Acknowledgement

The author thanks to Prof. Grygoriy Zholtkevych for his supervision and support.

References

1. Barrat, A., Weigt, M. On the properties of small-world network models. Eur. Phys. J. B 13, 547560 (2000). https://doi.org/10.1007/s100510050067
2. Rukkas, K., Zholtkevych, G. Distributed Datastores: Towards Probabilistic Approach for Estimation of Reliability. V.N. Karazin Kharkiv National University, 523-534 (2015) http://ceur-ws.org/Vol-1356/paper_51.pdf.
3. Watts, D., Strogatz, S. Collective dynamics of small-world networks. Nature 393, 440442 (1998). https://doi.org/10.1038/30918
4. Biometrika, Volume 52, Issue 3-4, December 1965, Pages 591611, https://doi.org/10.1093/biomet/52.3-4.591

5. Seth Gilbert and Nancy Lynch, "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services", ACM SIGACT News, Volume 33 Issue 2 (2002), pg. 5159. https://doi.org/10.1145/564585.564601
6. Martin Kleppmann, "A Critique of the CAP Theorem" https://arxiv.org/pdf/1509.05393.pdf
7. Karl Pearson F.R.S. X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling Pages 157-175 https://doi.org/10.1080/14786440009463897