A Dimensionless Graceful Degradation Metric for Quantifying Resilience

Jacob Beal
Raytheon BBN Technologies; Cambridge, MA, USA
Email: jakebeal@bbn.com

***Abstract*—Many self-* properties are variations on the same theme: resilience of a system to changes in itself or the conditions under which it operates. Quantifying resilience is difficult, however: there are no metrics of resilience that are readily comparable across systems, and the space of possible changes is typically prohibitively large. To address this problem, I propose a quantitative measure of graceful degradation that is independent of the units, scales, and number of system parameters. Although this metric is typically intractable to compute precisely, it can be approximated by perturbation surveys, and the quality of approximation is likely to be improved by a random perturbation approach based on recent advances in manifold learning.**

Keywords*-engineered self-organization; graceful degradation; perturbation analysis; self-adaptability; self-

I. Motivation

When we evaluate the self-* properties of an engineered system, what exactly should we be trying to quantify? There are many different self-* properties—self-adaptation, self-healing, self-configuration, self-organization, self-stabilization, etc.—and the definitions for many of them are still rather hazy, subject to dispute, and frequently overlapping.[1] Most of these properties, however, may be boiled down to the same general theme: the resilience of a system to changes, whether to itself or to the conditions in which it operates.

The problem here is that the space of possible changes is typically staggeringly immense, due to the large number of possible system states and operating conditions, and the potential for interaction between them. Analytic solutions are typically not feasible, meaning we must seek empirical approaches based on sampling system behavior at a limited number of points. To date, much of the work on evaluating self-* systems has been construction of taxonomies (e.g., [1], [2], [3], [4]) identifying qualitative classes of resilience. A number of possible metrics have been proposed, such as the entropy-based approaches in [5], [6], [7], the control autonomy measure in [8], and the trajectory divergence measure in [9]. All, however, are tightly dependent on the particular parameters and/or units used to describe a system and its operating conditions, making the metrics incomparable even for closely related systems. Many are also infeasible to compute. Perhaps it is for this reason that [10] found that most publications on self-* systems present little or no evidence for their claims.

To address this problem, I propose a new quantitative measure for graceful degradation, constructed to be independent of units, scales, and number of system parameters. Although this metric is typically intractable to compute precisely, it can be approximated by perturbation surveys. Moreover, the quality/cost trade-off for such perturbation surveys can likely be greatly improved by a random perturbation approach based on recent advances in manifold learning.

[1]Perhaps this haziness is why they are often lumped together as “self-*”?

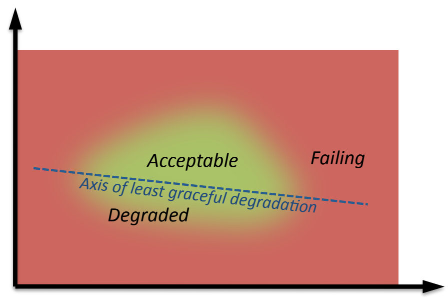

Fig. 1. Diagrammatic representation of a system’s domain: each axis is a configuration parameter or external condition; system performance is interpreted qualitatively as acceptable (solid green), failing (solid red), or degraded (mixtures). Graceful degradation is measured as the minimum ratio between degraded and acceptable behavior on any axis.

II. Formal Definition of Graceful Degradation

To quantify the degree to which a given system exhibits graceful degradation, we need a formal definition of:

- The domain D of possible configurations and operating conditions for the system.
- A set of performance metrics μ_i for system behavior.
- An interpretation function $\mathcal{I}_i$ for each metric, classifying performance as acceptable, degraded, or failing.
- A metric $\mathcal{G}$ that maps the interpreted performance over the domain to a number quantifying graceful degradation.

We will consider each of these in turn, building up to a definition of $\mathcal{G}$. Critically, our definition for $\mathcal{G}$ will produce a dimensionless number that is independent of the relative scaling of the parameters used to define D. This means that we need not be concerned about units—even such odd bedfellows as microfarads, meters, and gigawatts may be safely combined in a parameter space without any one inappropriately dominating the metric.

A. Domain and Performance Metrics

Consider an arbitrary engineered system. This system has some number n_p of configuration parameters p_1, p_2, ... p_{n_p}, each of which has a (potentially infinite) set of values P_i that it can take on. Let us assume that for each parameter p_i, there is a homeomorphic map from the values P_i onto a subset of $\mathbb{R}^k$. Let us also assume that the external conditions under which the system operates can be decomposed into a set of n_c parameters c_1, c_2, ... c_{n_c}, each of which can take on a set of of values C_j that similarly map onto real numbers.

The domain D of the system may then be expressed as the set of all combinations of the potential values for these variables, whether valid or not:

$$D = \prod_{i \leq n_p} P_i \times \prod_{j \leq n_c} C_j$$

Performance metrics map points in this domain to quantitative measurements. Typically, there are many aspects of a systems performance that are of interest. We thus consider a set of n_m performance metrics, μ_1, μ_2, ... μ_{n_m}, each of which is a function mapping $\mu_i : D \rightarrow \mathbb{R}$.

B. Interpretation of Performance

Our understanding of how resilient a system is depends critically on defining what it is that we want to achieve with the system. Consider, for example, the time delay for a message to propagate from source to destination through a network. This aspect of system performance can obviously be measured in seconds, but the number of seconds that is acceptable depends on the application. For email, it is typically quite reasonable for minutes to elapse, while a phone call becomes difficult if the delay is more than a few tenths of a second. The exact same system may therefore have different effective resiliences when considered for different purposes.

We thus need an interpretation function $\mathcal{I}_i$ for each performance metric μ_i. For this, I propose a mapping from the original quantitative measure onto the set $\{\mathbf{A}, \mathbf{D}, \mathbf{F}\}$, for the qualitative categorizations of "acceptable," "degraded," or "failing."[2] For many metrics, this can be implemented simply as a normalization of μ_i such that $\mathcal{I}_i \circ \mu_i(D) \geq 1$ is "acceptable," intermediate values of $0 < \mathcal{I}_i \circ \mu_i(D) < 1$ are "degraded", and $\mathcal{I}_i \circ \mu_i(D) \leq 0$ is "failing."

When the system's performance has failed according to any metric, the system as a whole may be considered to have failed. For example, it does not matter how reliable and secure message delivery is if the messages arrive too slowly to be useful. Likewise, degradation according to any metric may be considered to be degradation of the whole. Thus, the set of interpreted metrics may be combined with logical disjunction to produce a function:

$$\bigvee_{i \in n_m} \mathcal{I}_i \circ \mu_i : D \rightarrow \{\mathbf{A}, \mathbf{D}, \mathbf{F}\}$$

that interprets each point in the system domain as "acceptable," "degraded," or "failing."

[2]This notion is similar to the "nonstressed," "stressed," and "nonviable" space for functional blueprints [11].

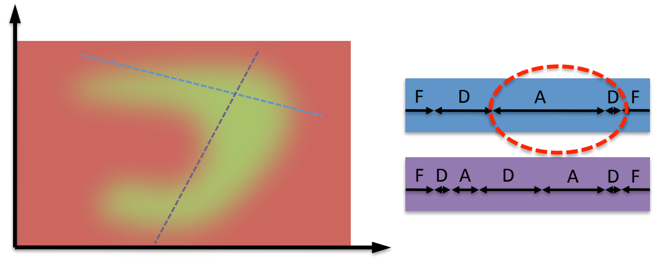

Fig. 2. Graceful degradation along an axis is quantified as the minimum ratio between the measure of adjacent "degraded" to "acceptable" sets along that axis. For example, the two example axes through the domain shown on the left have the interpreted performance sets shown on the right, with the sets producing the minimum graceful degradation for these two axes indicated by a dashed red circle.

C. Graceful Degradation

We now come to the crux of the question: given an interpretation of system performance, how can measure resilience? The volume of the acceptable and degraded spaces seems an obvious choice, and is one that has previously been proposed in various forms (e.g., [8]). Volume, however, depends on the particular choice of configuration parameters and their units, making it effectively impossible to make compare measurements of even closely related systems.

Instead, let us define our graceful degradation metric $\mathcal{G}$ as a dimensionless quantity that compares the ranges of the acceptable and degraded regions. Intuitively, this metric will indicate how difficult it is to navigate the system domain without the system failing.

Let us build this metric up by first considering a one-dimensional domain. Every point in this domain is interpreted as $\mathbf{A}$, $\mathbf{D}$, or $\mathbf{F}$, so we may consider the domain as a collection of intervals of the three qualities. The graceful degradation of this domain is then:

$$\mathcal{G}(D_1) = \min_{I_D, I_A} \frac{|I_D|}{|I_A|}$$

where each possible value of I_A is an acceptable interval and I_D is a degraded interval separating it from a failing interval (there are also special cases: $\mathcal{G}(D_1)$ is zero if acceptable and failing intervals are ever directly adjacent or there are no acceptable intervals, and infinity if no point is failing).

We may then generalize to arbitrary spaces by letting $\mathcal{G}(D)$ be the infimum of the one-dimensional metric for all lines passing through $\mathbf{A}$. In other words, the graceful degradation for the whole domain is the least graceful degradation that we get by starting at any acceptable point and heading toward failing in any direction. Figure 1 illustrates this for a two-dimensional domain, and Figure 2 shows examples of evaluating the metric along lines in a concave acceptable space.

Importantly, the value produced by $\mathcal{G}$ is dimensionless, having no relationship to the physical units of the domain parameters. Due to the method of calculation, $\mathcal{G}$ is unaffected by changes in scale, or in fact any linear transformation of the domain. Likewise, the number of dimensions in the parameter space will typically have little effect on the value produced by $\mathcal{G}$: adding irrelevant dimensions does not change the value of $\mathcal{G}$, because it is determined entirely by the single worst axis in the entire parameter space—in effect, the metric is looking for the system's weakest point.

Nonlinear transformations of the domain, however, can affect the metric: For example, consider a square block that is acceptable when its edge length is less than 10 units, degrades from 10 to 20 units, and fails when the edge is greater than 20 units. Its $\mathcal{G}$ is 1 (10/10), but switching from edge to area gives a $\mathcal{G}$ of 3 (300/100). Logarithmic versus linear scales is another simple example. In such cases, it is likely best to select the units in which errors are likely to occur: for example, if the square is produced by a manufacturing process that makes two cuts, then it is better to use edge length than area.

This definition does have some counter-intuitive consequences. For example, if the region of degraded behavior has equal width along two different axes, then then the axis with the *greater* interval of acceptable behavior will be judged to degrade *less* gracefully. Why shouldn't it be more graceful, since there is more acceptable behavior? Remember, however, that the relative "length" of the interval depends on choice of units. The metric is instead showing the degree of tension between quickly shifting a parameter through its acceptable range and at the same time keeping the system from failing.

D. Non-Numerical and Indeterminate-Size Parameters

Everything presented so far has assumed that it is possible to reasonably map any configuration parameter or external conditions onto a finite-dimensional interval of numbers (discrete or continuous). This may not always be the case, however.

Consider, for example, a system where one of the configurations parameters is a free choice of search algorithm. While this might possibly be mapped onto a sequence of real numbers, doing so may destroy the relations between elements that we need to be able to understand. How should non-numerical objects like breadth-first search, depth-first search, hill-climbing search, and branch-and-bound search be mapped onto numbers and compared?

While this problem may not be able to be resolved for all such non-numerical parameters, in many cases it can be mapped onto a continuous space by considering mixtures of values. For example, in the case of searches, consider each step of the search flipping a weighted coin to determine which of the possible types of step it will take next. We thus have a search algorithm under control of a small number of parameters, each a real number in the range $[0, 1]$, which blend smoothly between the possible types of search (at the cost of greatly increasing the number of dimensions in the domain).

Another problem can arise when the dimensionality of a parameter cannot be given a reasonable limit *a priori*. For example, one parameter may set the number of instances of a second parameter that exist, each of which is capable of being set independently. For example, in informed flocking [12] each informed member of the flock has a preferred direction of motion, so the number of potential "preferred direction" parameters scales linearly with the number of informed members. In such cases, however, it is often the case that as the number of parameters increases, the collective behavior approaches a limit based on the statistical distribution of values rather than individual elements. The number of parameters then reduces to the number required to describe the distribution, no matter the number of individuals.

There are likely cases that cannot be addressed by these strategies, or indeed that cannot be mapped onto this graceful degradation metric in any reasonable manner. If, however, a broad enough range of applications can be reasonable mapped or approximated for this definition, then it will still be useful.

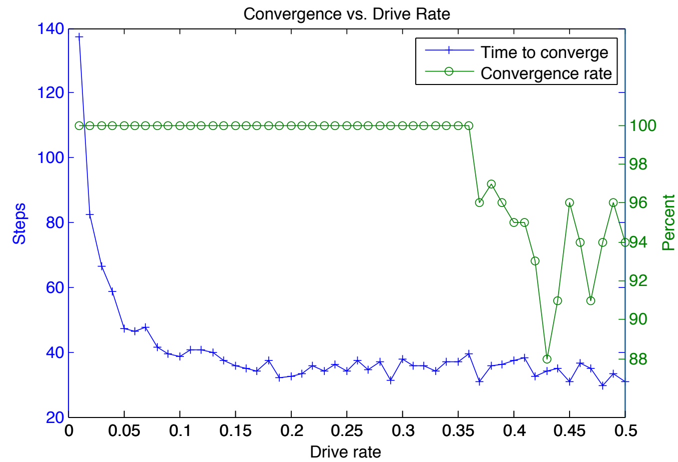

Fig. 3. Example of a perturbation survey from which graceful degradation can be computed. This survey, from [13], varied one configuration parameter of a self-adaptive design assistant and measured using two metrics; the range from 0.1 to 0.35 is acceptable; below 0.02 and above 0.4 are failing.

III. Computing Graceful Degradation

So far, we have formulated a mathematical definition of graceful degradation, but not yet considered how it can be practically calculated. Brute force approaches are infeasible for all but the simplest systems, due to the number of possible combinations of parameter values that must be scanned. We can reduce the cost to tractable levels, however, by conducting perturbation response surveys instead, in which scans are performed along selected vectors rather than the entire space.

A. Orthogonal Perturbation Survey

The high costs of parameter scans come from considering many dimensions at once. An alternate approach is to select a reference point r in the configuration domain D and conduct perturbation surveys around this point. For each parameter i in D, we take the basis vector $\vec{i}$ and measure the performance of $r + \Delta\vec{i}$ for incrementally varying values of Δ.[3] This radically reduces the number of performance evaluations to be made.

Figure 3 shows an example of a perturbation survey along one dimension, taken from [13], of a self-adaptive design

[3] Small combinations of perturbations can be used, but that way lies the combinatorial problems we are trying to escape.

assistant, measured on two metrics. This one-dimensional slice shows a classic profile of graceful degradation: when the parameter is too high or too low the system fails, but a broad range in the middle is acceptable and the transition from acceptable to failing is not sharp. Taking the acceptable range as 0.1 to 0.35 and the failing range as above 0.4 and below 0.02, we can compute apply the graceful degradation metric $\mathcal{G}$ developed above. The acceptable interval is size 0.25, with the upper and lower degraded regions having sizes 0.05 and 0.08 respectively. In this case, $\mathcal{G}(D)$ is thus equal to 0.2, indicating that there is a reasonable margin of error available for adjusting this parameter.

This approach depends critically on the assumption that for a large region around the reference point r there is an approximately linear relation between perturbation responses. In other words, it is assumed that:

$$\mu_i(r + \Delta\vec{j} + \Delta'\vec{k}) \approx \mu_i(r + \Delta\vec{j}) + \mu_i(r + \Delta'\vec{k}) - \mu_i(r)$$

where μ_i is any of the performance measures and Δ and Δ' are the size of perturbations away from reference point r along basis vectors $\vec{j}$ and $\vec{k}$ of D. If this is not the case, then it is likely for the resilience of a system to be overestimated, as several small changes interact to cause an unexpected failure—exactly the problem that self-* systems are intended to prevent! Surveying combinations of perturbations, however, returns to the combinatoric explosions that make parameter scans infeasible.

B. Random Perturbation Survey

There is an approach, however, that should be able, with high probability, to combine the thoroughness of parameter scans with the tractable cost of orthogonal perturbation surveys. Recently, the machine learning community has developed methods for "manifold learning" based on selection of random vectors [14]. The key insight is this: in a high dimensional space, any random vector is likely to have a significant component in the direction of the principle eigenvector of the space. Thus, choosing several random vectors and evaluating their dot products with a data set produces a good lightweight approximation of principle component analysis. The method can then recurse to refine its approximation of the data manifold by picking further random vectors on subsets of the data.

This same principle should be applicable for evaluating $\mathcal{G}$. As with orthogonal perturbation surveys, this approach would begin with a reference point r. Rather than perturb along a single parameter, however, this approach would perturb in a random direction, potentially combining all parameters. If the performance gradients are relative smooth, then a few such perturbations should be sufficient to detect the parameter combinations with the most restricted operating range.

Although the results of such an approach will be less intuitive to interpret, they offer a potentially major advantage in that they depend only on approximate smoothness of the performance metrics, rather than difficult to analyze system properties. At the same time, it is possible that the number of perturbations required to evaluate a system might actually be lower than for an orthogonal perturbation survey, since many parameters might be tested by the same random vector.

IV. Contributions and Open Problems

I have proposed a new dimensionless metric for graceful degradation, which has the advantage of being readily comparable across self-* systems. Although the metric is generally intractable to compute precisely, it can be approximated via perturbation surveys, and I have additionally proposed a new random perturbation approach to allow surveys to detect parameter interactions. These are, of course, just the beginning of what needs to be done to provide a practical ability to quantify the resilience of complicated self-* systems. A clear next step is to fully elaborate the random perturbation approach and to test it against well-known systems such as flocking. A mathematical analysis will need to be made, as well, of the conditions under which it can and cannot be relied upon to work. If these ventures are successful, however, the graceful degradation metric proposed in this paper may at last provide a practical and reliable means of quantifying self-* properties.

References

[1] J. A. McCann and M. C. Huebscher, "Evaluation issues in autonomic computing," in *Conference on Grid and Cooperative Computing Workshops*, 2004, pp. 597–608.

[2] E. Gjorven, F. Eliassen, and J. O. Aagedal, "Quality of adaptation," in *International Conference on Autonomic and Autonomous Systems*, 2006, pp. 9–14.

[3] G. Muhl, M. Werner, M. A. Jaeger, K. Herrmann, and H. Parzyjegla, "On the definitions of self-managing and self-organizing systems," in *KiVS Workshop: Selbstorganisierende, Adaptive, Kontextsensitive verteilte Systeme (SAKS07)*. VDE Verlag, 2007, pp. 291–301.

[4] N. Villegas, H. Muller, G. Tamura, L. Duchien, and R. Casallas, "A framework for evaluating quality-driven self-adaptive software systems," in *6th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS 2011)*. ACM, 2011.

[5] C. R. Shalizi, K. L. Shalizi, and R. Haslinger, "Quantifying self-organization with optimal predictors," *Physical Rev. Lett.*, vol. 93, pp. 1–4, 2004.

[6] F. Heylighen, "The science of self-organization and adaptivity," in *The Encyclopedia of Life Support Systems*, 1999, pp. 253–280.

[7] T. De Wolf, G. Samaey, T. Holvoet, and D. Roose, "Decentralized automatic computing: Analysing self-organising emergent behavior using advanced numerical methods." in *2nd International Conference on Automatic Computing (ICAC05)*, 2005, pp. 52–63.

[8] H. Schmeck, C. Muller-Schloer, E. Cakar, M. Mnif, and U. Richter, "Adaptivity and self-organization in organic computing systems," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 5, no. 3, p. Article 10, September 2010.

[9] Y. Chaaban, C. Muller-Schloer, and J. Hahner, "Measuring robustness in hybrid central/self-organising multi-agent systems," in *COGNITIVE 2012 : The Fourth International Conference on Advanced Cognitive Technologies and Applications*, 2012.

[10] M. Naqvi, "Claims and supporting evidence for self-adaptive systems a literature review," Master's thesis, Linnaeus University, 2012.

[11] J. Beal, "Functional blueprints: An approach to modularity in grown systems," *Swarm Intelligence*, vol. 5, no. 3, 2011.

[12] I. Couzin, J. Krause, N. Franks, and S. Levin, "Effective leadership and decision making in animal groups on the move." *Nature*, vol. 433, pp. 513–516, 2005.

[13] A. Adler, F. Yaman, J. Cleveland, and J. Beal, "Morphogenetically assisted design variation," in *2nd International Conference on Morphological Computation*, September 2011.

[14] Y. Freund, S. Dasgupta, M. Kabra, and N. Verma, "Learning the structure of manifolds using random projections," in *Advances in Neural Information Processing Systems*, vol. 20, 2007, pp. 473–480.