



Dynamic Fitness Landscape Analysis

Erik Pitzer^{1,2(✉)}, Bernhard Werth^{1,2}, and Johannes Karder^{1,2}

¹ Department Software Engineering, University of Applied Sciences Upper Austria,
Softwarepark 11, 4232 Hagenberg, Austria

{erik.pitzer,bernhard.werth,johannes.karder}@fh-hagenberg.at

² Institute for Formal Models and Verification, Johannes Kepler University,
Altenbergerstr 68, 4040 Linz, Austria

Abstract. Dynamic optimization problems pose a big challenge for classic optimization algorithms. They could simply be viewed as a series of related optimization problems. In particular the aspect of time-linkage has not been well studied yet. In this work we are analyzing an artificial problem based on real-world data to elucidate the potential of fitness landscape analysis methods to discover problem difficulty and follow along the changes of dynamic problems and how these changes can be measures and might be exploited by enabling algorithm introspection.

Keywords: Fitness landscape analysis · Dynamic optimization · Dynamic vehicle routing

1 Introduction

The suitability of automatic fitness landscape analysis (FLA) for the comparison of static optimization problem instances [14] and for the characterization of static problem classes [15, 16] has been demonstrated in the past. Moreover, it has been used to help with algorithm selection and parameter tuning. However, for static optimization problems the usual approach is to determine (with or without FLA) a suitable algorithm and suitable parameters for the given problem class and use it solve new problems as well. Dynamic optimization problems (DOPs) [1] pose an additional challenge: The problems can and do change over time, and therefore also the suitability of optimization methods. In this paper we are analyzing the extent to which such dynamic changes are reflected in fitness landscape analysis measurements.

Dynamic optimization problems themselves have already been researched previously [1, 4, 10] and even their fitness landscape characteristics have been analyzed [17]. Mostly, the aspect of time-linkage, however, has not been explicitly studied [10]. Therefore, this paper will study a problem with time-linkage, i.e. where previous decision have an impact on the later development of the problem and, hence on the available choices later during the optimization. Many real-world applications exhibit similar characteristics and this paves the way for transfer of theoretical knowledge towards more practical examples.

The Dynamic Vehicle Routing Problem (DVRP) is a well studied optimization problem [6] and makes for an excellent benchmark problem, as it is both flexible and simple [9]. Additionally, we have added an aspect of time linkage in our application scenario: The simulation assumes fixed locations that have to be serviced, and once serviced will need servicing again, drawing from a known but random distribution. In this case the mean time between failures (MTBF) is a tunable, than can be used to simulate scenarios of varying difficulties to study the effect on various fitness landscape measures. Time-linkage is achieved by linking subsequent service requirements to the previous service time, so that locations that have been serviced sooner, will also require re-servicing sooner, on average.

In this work we are aiming to establish a base-line of what can be seen with the help of generic fitness landscape analysis on these dynamic problems. In particular, we are going to examine the effect of difficulty on these measures as this has been an important intention for the design of many FLA measures. This information could in the future be used to give algorithms some more leverage for introspection and determine whether they are still well tuned to the current situation or need to change parameters or switch to another algorithm entirely.

2 State of the Art

2.1 Fitness Landscape Analysis

Fitness Landscape Analysis takes a very generic view on optimization problems. Only the bare minimum, that any optimization problem has to have, is used [5]. The fitness landscape \mathcal{F} can be defined as a triple consisting of the following:

- solution space S
- fitness function, e.g. $f : S \rightarrow \mathbb{R}$
- neighborhood, e.g. $N : S \rightarrow \mathcal{P}(S)$ or distance, e.g. $d : S \times S \rightarrow \mathbb{R}$

Surprisingly, a large number of possible measurements can be taken with this simple formulation [15] and allow deep insights into problem classes and also allow comparison of instances. Several studies have been performed to completely understand the fitness landscape, most notably, e.g. in [18], and in [2] by elementary decomposition, or in [3] by exhaustively determining local optima and analyzing their structure. On the other hand, quick and even superficial analysis can also be a good source of information, supporting algorithm and parameter decisions by offering a glimpse into the landscape and its characteristics pertaining to optimization algorithms as detailed e.g. in [5] or [15]. These methods usually sample only a tiny fraction of the solution space, most often using trajectory based sampling in the form of different “walks”. Best-known is the random walk that selects a random neighbor and provides an unbiased view of what a trajectory-based algorithm might observe. However, most algorithms do not “walk” randomly and therefore other walk types have been devised, such as an up-down walk, that alternatively maximizes towards a local maximum and

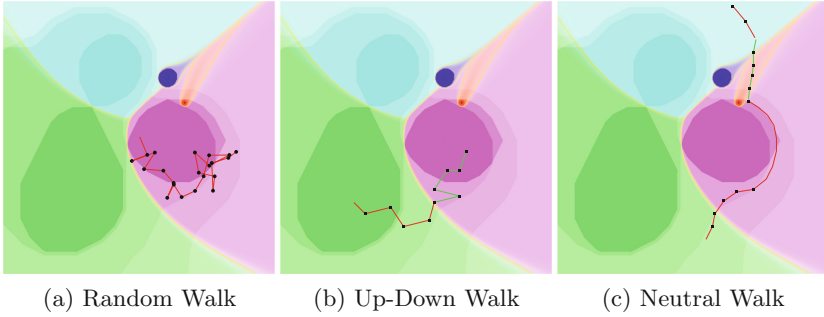


Fig. 1. Different walk types (based on [13])

then minimized towards a local minimum (ideally avoiding going back and forth) or a neutral walk that tries to stay within an equal or similar fitness range to estimate the size of (quasi) plateaus. Figure 1 illustrates some examples. Based on the obtained samples different measures have been defined. Mostly with the aim to quantify aspects of a problem’s “difficulty”. Most prominently, the auto-correlation [22] tries to capture the self-similarity of consecutive samples in a walk and is used to measure a trajectory’s *ruggedness*. A more complex approach introduced in [20] describes several measures for a trajectory’s entropy, or more informally, its “interestingness” by calculating different combinations of consecutive slope frequencies. Using other walk types other aspects can be described, such as, measures for a problem instance’s basin size using up-down walk lengths or frequencies of plateaus using neutral walk lengths and neutral neighbor fractions, i.e. the average number of neighbors with the same (or similar) fitness.

2.2 Dynamic Vehicle Routing Problems

A dynamic optimization problem [1] is defined as a problem that can change over time. In its simplest interpretation, this can be seen as series of problems over time. In a more complex view, the previous optimization decisions can have an influence on future problem states (time-linkage). On the other hand, this series of problems often does have a correlation to previous problem states which can be exploited by algorithms that have a long-term state, such as e.g. a population.

In this work, we selected a well studied dynamic optimization problem, the Dynamic Vehicle Routing Problem (DVRP) [6] to test the applicability of fitness landscape analysis and elucidate the potential for dynamic control on optimization algorithms. In its simplest formulation, the Vehicle Routing Problem (VRP) [19], can be defined as the problem to find the shortest path for l vehicles, to service n locations $\{v_1, \dots, v_n\}$ starting from a depot location v_0 , given a distance matrix D with driving times between all locations. Many other variants exist, e.g. adding capacity restrictions to the vehicles or a time window to the visit time at each location [7]. The Dynamic Vehicle Routing Problem (DVRP)

as a simple extension can be seen as a series of VRPs where each VRP has a certain lifetime before being replaced with the next instance. In realistic scenarios these consecutive VRPs are often closely related differing, e.g. only in a single location that has been added or removed.

3 Materials and Methods

3.1 Dynamic Servicing of Power Grid Facilities

We chose to implement a variant of the uncapacitated DVRP, modeling the servicing of power grid facilities which adds an interesting aspect to DVRP that gives rise to time-linkage between consecutive states influenced by the choices of the algorithm and allows efficient pre-calculation of travel cost between locations: Fig. 2 shows the distribution of power grid facilities in Upper Austria. In our simulation, these locations are activated and re-activated based on their own normal distribution, each of these normal distributions is, in turn, parameterized by a discrete distribution to allow different characteristics between locations but giving rise to clustering as the parameters are drawn from a limited set of possibilities. The range of these parameters can be manually tuned to give a predictable amount of *difficulty*. The influence of difficulty on the distribution of failure probabilities is calculated as $\mu_{\text{eff}} = \mu/\text{difficulty}$, effectively elongating the times between failures for lower difficulties.

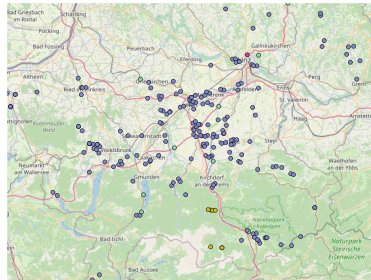


Fig. 2. Power grid facilities in Upper Austria

The sooner locations are re-activated, the harder it will be to service them in a timely manner or at all. Moreover, each location has a certain service time the vehicle needs to remain at rest. The quality is then defined as the cumulative waiting time between the need of service and the completion of service over all locations. The evaluation budget is determined by the driving time between locations. Only when a vehicle becomes available, the new situation has to be considered, so the time in between can be seen as the evaluation budget.

The interesting aspect arising here is that previous choices indirectly influence the future service demand, as the time between failures is relative to the previous service time. Moreover, the evaluation budget for the algorithm is also indirectly influenced by the past choices, depending on the driving times of all vehicles.

This scenario was selected as an artificial problem with real data to examine the propensities of FLA in the context of dynamic optimization problems. In particular, we were interested in knowing a-priori a relative difficulty of the scenarios and wanted to see whether this is reflected in the measurements and how it changes over time.

3.2 Experimental Setup

To obtain realistic data we retrieved power grid facilities from Open Street Map¹ using the Overpass API [11] for querying public data. We created several different scenarios with different distributions: once in Upper Austria (query shown in Listing 1) and once in and around Vienna. We limited the locations to the 200 largest facilities (by area) in the respective regions. We used a local instance of the Openrouteservice² to calculate the distance matrix using a snapshot of Open Street Maps provided by geofabrik.de³. The simulation was implemented inside HeuristicLab [21] and is described in more detail in a second publication in this series.

```
area[name="Oberösterreich"]->.a;
(way(area.a)[power=station];
 way(area.a)[power=substation];
 way(area.a)[power=generator];
 way(area.a)[power=plant]; );
```

Listing 1: Overpass Query for Power Grid Facilities in Upper Austria

After each change in the dynamic problem, a “snapshot” of the current state was created with the help of persistent data structures [12] to limit the required memory. These snapshots were then subjected to a whole array of fitness landscape analysis methods. The obtained measurements were then further analyzed and visualized using Jupyter and Pandas [8]. Here, an additional “smoothing” step was introduced, in particular to facilitate later clustering as the individual time series had noticeable variation due to different evaluation points in time. Therefore, each time series has been smoothed using neighbor regression with a fixed radius.

¹ <https://openstreetmap.org>.

² <https://openrouteservice.org> by HeiGIT.

³ <https://geofabrik.de>.

4 Results and Analysis

While the snapshots of the runs with different difficulties occur at different times, we have plotted them on a homogeneous time scale which coincidentally also represents the current evaluation budget. Figure 3 shows the results of some very basic FLA measures. The horizontal axis contains the various difficulties while the vertical axis shows time and evaluation budget. Points in between the actual snapshots are simply filled in with the value of the previous snapshot. Furthermore, as stated earlier, the values have been smoothed along the vertical axis using a fixed-radius neighbor regression to account for small variations. It is important to note that this smoothing can also be applied during an active run as the required information is limited to data available in a single execution. As can be seen in Fig. 3, however, the classic auto correlation and its derived correlation length [22] show hardly any systematic difference over time and neither over varying difficulty levels.

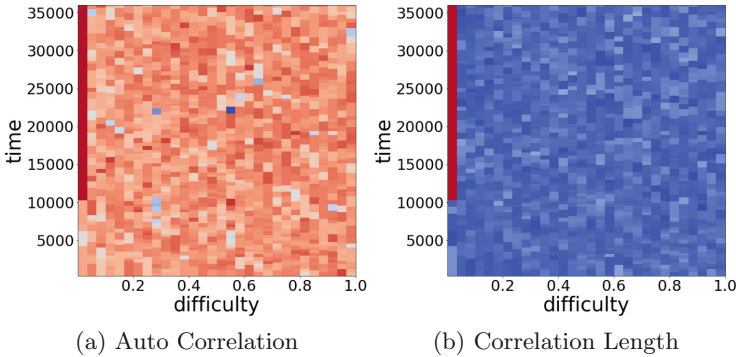


Fig. 3. Basic FLA values

When turning to the analysis of neutrality and theoretic analysis the picture changes. Figure 4 shows several promising insights into the fitness landscape that could be obtained. Figure 4a shows the neutral neighbor fractions, i.e. the number of neighbors at any sample point that have the same fitness. This clearly shows the areas where optimization does hardly make any difference, which in this case indicates areas of little interest because they are “too easy” or not worth optimizing. On the other hand, Fig. 4b shows the quality delta at which the information content is maximized, or in other words at which scale “interesting” changes occur. Here interesting changes can only be observed at very large scale which can indicate that the problem becomes hard. Finally, Fig. 4c show the information content of random walks at minimum scale and, therefore, the “interestingness” for an optimization algorithm. Which coincidentally is also the area where an optimization makes most sense and has most potential. The area to the left is too easy to make a difference, while the area to the upper right is

too difficult, so new locations need servicing while all vehicles are still occupied and the number of requests keeps increasing.

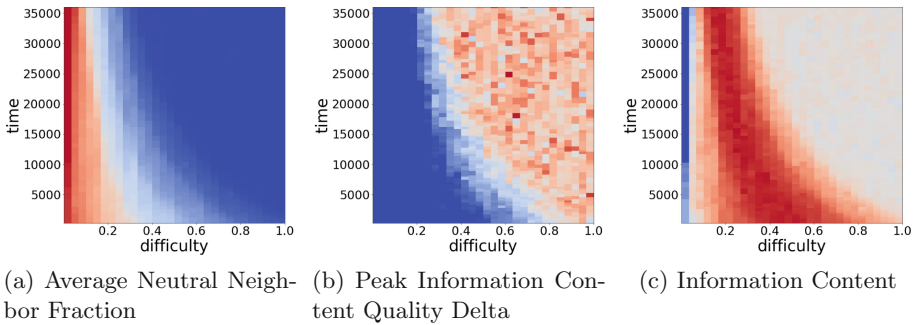


Fig. 4. FLA values characterizing optimization potential

5 Conclusions and Future Work

While the level of these experiments is still very basic, knowing what to measure and being able to observe the development over time seem to be possible given the right FLA techniques. It should be noted, however, that for different problems, other measures could be necessary as the problem characteristics change. We could observe that with simple FLA measures we were able to see boundaries between situations where optimization is appropriate. This seems a promising perspective for further algorithm introspection which is important for dynamic problems. It could enable an algorithm to make more informed decisions about evaluation budget, or an outcry for help if it is able to detect that the problem is becoming increasingly more difficult and optimization alone will not be able to handle the situation. In the future we will focus on more complex dynamic effects such as varying failure probabilities over time and determining whether this effect remains observable by FLA. Moreover, a more complex interplay between failures will be studied, e.g. one power outage could lead to another. Finally, more advanced FLA measures can be applied such as measuring isotropy or distribution of FLA values over the landscape.

References

1. Branke, J.: *Evolutionary Optimization in Dynamic Environments*. Springer, New York (2002). <https://doi.org/10.1007/978-1-4615-0911-0>
2. Chicano, F., Whitley, L.D., Alba, E.: A methodology to find the elementary landscape decomposition of combinatorial optimization problems. *Evol. Comput.* **19**(4), 597–637 (2011)
3. Herrmann, S., Ochoa, G., Rothlauf, F.: PageRank centrality for performance prediction: the impact of the local optima network model. *J. Heuristics* **24**, 243–264 (2018)

4. Jin, Y., Branke, J.: Evolutionary optimization in uncertain environments - a survey. *IEEE Trans. Evol. Comput.* **9**, 303–317 (2005)
5. Jones, T.: Evolutionary algorithms, fitness landscapes and search. Ph.D. thesis, University of New Mexico, Albuquerque, New Mexico (1995)
6. Khoudjia, M.R., Sarasola, B., Alba, E., Talbi, E.G., Jourdan, L.: Metaheuristics for dynamic vehicle routing. In: Alba, E., Nakib, A., Siarry, P. (eds.) *Metaheuristics for Dynamic Optimization*, vol. 433, pp. 265–289. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-30665-5_12
7. Liu, R., Jiang, Z.: A hybrid large-neighborhood search algorithm for the cumulative capacitated vehicle routing problem with time-window constraints. *Appl. Soft Comput.* **80**, 18–30 (2019)
8. McKinney, W.: Data structures for statistical computing in Python. In: van der Walt, S., Millman, J. (eds.) *Proceedings of the 9th Python in Science Conference*, pp. 56–61 (2010)
9. Morrison, R.: *Designing Evolutionary Algorithms for Dynamic Environments*. Springer, Heidelberg (2004). <https://doi.org/10.1007/978-3-662-06560-0>
10. Nguyen, T., Yang, S., Branke, J.: Evolutionary dynamic optimization: a survey of the state of the art. *Swarm Evol. Comput.* **6**, 1–24 (2012)
11. Olbricht, R.M.: Data retrieval for small spatial regions in OpenStreetMap. In: Jokar Arsanjani, J., Zipf, A., Mooney, P., Helbich, M. (eds.) *OpenStreetMap in GIScience*. LNCS, pp. 101–122. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-14280-7_6
12. Pitzer, E., Affenzeller, M.: Facilitating evolutionary algorithm analysis with persistent data structures. In: Moreno-Díaz, R., Pichler, F., Quesada-Arencibia, A. (eds.) *EUROCAST 2017*. LNCS, vol. 10671, pp. 416–423. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-74718-7_50
13. Pitzer, E., Affenzeller, M., Beham, A.: A closer look down the basins of attraction. In: *UK Conference on Computational Intelligence* (2010, in press)
14. Pitzer, E., Affenzeller, M., Beham, A., Wagner, S.: Comprehensive and automatic fitness landscape analysis using heuristiclab. In: Moreno-Díaz, R., Pichler, F., Quesada-Arencibia, A. (eds.) *EUROCAST 2011*. LNCS, vol. 6927, pp. 424–431. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-27549-4_54
15. Pitzer, E., Beham, A., Affenzeller, M.: Generic hardness estimation using fitness and parameter landscapes applied to the robust taboo search and the quadratic assignment problem. In: *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2012)*. ACM (2012)
16. Pitzer, E., Beham, A., Affenzeller, M.: Automatic algorithm selection for the quadratic assignment problem using fitness landscape analysis. In: Middendorf, M., Blum, C. (eds.) *EvoCOP 2013*. LNCS, vol. 7832, pp. 109–120. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37198-1_10
17. Richter, H.: Dynamic fitness landscape analysis. In: Yang, S., Yao, X. (eds.) *Evolutionary Computation for Dynamic Optimization Problems*. *Studies in Computational Intelligence*, vol. 490, pp. 269–297. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38416-5_11
18. Stadler, P., Wagner, G.: The algebraic theory of recombination spaces. *Evol. Comput.* **5**, 241–275 (1998)
19. Toth, P., Vigo, D.: *The Vehicle Routing Problem*. SIAM, Philadelphia (2002)
20. Vassilev, V.K., Fogarty, T.C., Miller, J.F.: Information characteristics and the structure of landscapes. *Evol. Comput.* **8**(1), 31–60 (2000)

21. Wagner, S., et al.: Architecture and design of the heuristiclab optimization environment. In: Klempous, R., Nikodem, J., Jacak, W., Chaczko, Z. (eds.) *Advanced Methods and Applications in Computational Intelligence. Topics in Intelligent Engineering and Informatics*, vol. 6, pp. 197–261. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-319-01436-4_10
22. Weinberger, E.: Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biol. Cybern.* **63**(5), 325–336 (1990)