Appendix A
Supplementary Materials

A. Obstacles on the map

The following equations are defined for the Lake g_{LA} and Checkerboard g_{CH} obstacles. These equations are mapped to the grids with the grid cell positions of (x, y). With these functions, cells of the grid can be identified where obstacles will be positioned, hence cells with $v_{max} = 0$. The two provided obstacle functions can be used as a constraining function when running an optimisation algorithm. They take the cell's coordinate as an input and output a *True* or *False* value determining if the specified cell is an obstacle. For all obstacle functions holds: $\{x \in \mathbb{N} | 1 \le x \le x_{max}\}$ and $\{y \in \mathbb{N} | 1 \le y \le y_{max}\}$.

$$g_{CH}(x, y) = \text{sign}\left(\sin\left(\frac{\pi}{2} + \pi x\right)\right) + \text{sign}\left(\sin\left(\frac{\pi}{2} + \pi y\right)\right)$$
$$- 2\,\Pi\,(x - x_{max})\,\Pi\,(y - y_{max}) = 2 \tag{8a}$$

$$\Pi(x) = H\left(x + \frac{1}{2}\right) - H\left(x - \frac{1}{2}\right) \tag{8b}$$

where $H(x)$ is the so-called Heaviside step function.

$$g_{LA}(x, y) =$$
$$\left(x - 1 - \frac{x_{max}}{2}\right)^2 + \left(y - 1 - \frac{y_{max}}{2}\right)^2 - (r\,x_{max})^2 < 0 \tag{9}$$

where r denotes the radius ratio.

B. Velocity functions

To determine the velocity v_{max} of each cell, except obstacle cells with $v_{max} = 0$, we have used the following equation, representing three street types, i.e. highways, country roads and city streets, derived from the usual speed-limits in Germany. The function takes the cell's coordinates as an input and outputs the respecting v_{max} for that cell. The provided values can be exchanged or extended to represent other road networks. For all the velocity function holds: $\{x \in \mathbb{N} | 1 \le x \le x_{max}\}$ and $\{y \in \mathbb{N} | 1 \le y \le y_{max}\}$.

$$v_{max}(x, y) = \begin{cases} 130, & \text{if } w(x, y) > 0.9 \\ 50, & \text{if } w(x, y) < -0.4 \\ 100, & \text{else} \end{cases} \tag{10}$$

where $w(x, y) = \max\left(sin(x-1), cos(y-1)\right)$.

Derived from this property, also the expected delay per path segment is defined.

$$delay(n_i, n_{i+1}) =$$
$$\begin{cases} 2 & \text{if } v_{max}(n_i) \ne v_{max}(n_{i+1}) \\ 3 & \text{if } v_{max}(n_i) = v_{max}(n_{i+1}) = 50 \\ 1 & \text{if } v_{max}(n_i) = v_{max}(n_{i+1}) = 100 \\ \frac{1}{5} & \text{otherwise} \end{cases} \tag{11}$$

(a) Elevation profile for $nh = M$ (b) Elevation profile for $nh = 1$

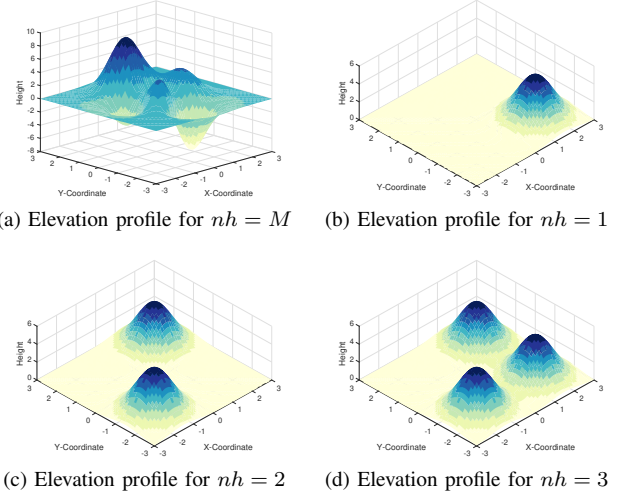(c) Elevation profile for $nh = 2$ (d) Elevation profile for $nh = 3$

Fig. 8. Different elevation profiles of the proposed benchmark

TABLE III
INTEGER SEQUENCES OF POSSIBLE NUMBER OF PATHS FROM THE
NORTH-WESTERN CORNER TO THE SOUTH-EASTERN ONE WITH NO
OBSTACLES ON OEIS.ORG FOR THE SIZE OF NXN
ASLETISMAC_NO_Xn_Yn_P_K(2,3)_B(F,T)

Benchmark type	Integer sequence
K3,BF	A001850
K3,BT	A140518
K2,BF	A000984
K2,BT	A007764

C. Height functions

In addition to the presented height functions, we show in Figure 8 a visual representation of the different available options.

D. Number of possible paths

The number of possible paths are represented by specific integer sequences (visible at oeis.org) and is shown in Table III:

E. Data sets and Code

To enable researchers to use the proposed benchmark, we also publish the code to generate different benchmark instances and the obtained true Pareto-fronts and sets. Everything can be downloaded here: https://ci.ovgu.de/Publications/TEVC_WM_2020-p-910.html. We used Java and the jMetal framework in version 6 [50]. However, the code enables researchers to create different grids and export them as a *csv-file* to import it in other software or to use other programming languages. The codes also contain a readme file.

F. Real-World Data

OpenStreetMap provides the GPS-coordinates for a grid representation which can be easily used to measure the path length for the first objective. As for the second objective

concerning the delay (number of accidents), we used the publicly available accident statistic data from 2018[3] and mapped them to the imported network. Since the coordinates of the accidents are mostly different from the available nodes in the network, we defined an R-tree index [41] on the network and performed a nearest node search for each accident. In this way, we aligned each accident to a node in the network. The third objective was measured using the Google Maps Elevation API[4]. The elevation is obtained in meters over the sea level and written to the node's properties. For the smoothness, we simplified the network to straight connections between nodes. Therefore, it is obtained in the same way as in the proposed benchmark. From the OpenStreetMap network, we could also obtain the information about speed limits per street segment. We calculated the time needed per segment as the ratio of distance and speed. Summing up the values of each segment results in the total traveling time (Objective 5). For the experiments, we take the same parameter settings as above with only one-point cross-over.

G. Results

Figure 9 illustrates the obtained IGD^+ values with respect to the different types of the problem instances. Figures 10 to 14 show true Pareto-fronts, sets and results from the algorithms for five different instances. For the smoothness objective, values are given in degrees.

---

[3]https://web.archive.org/web/20200704125405/https://unfallatlas.
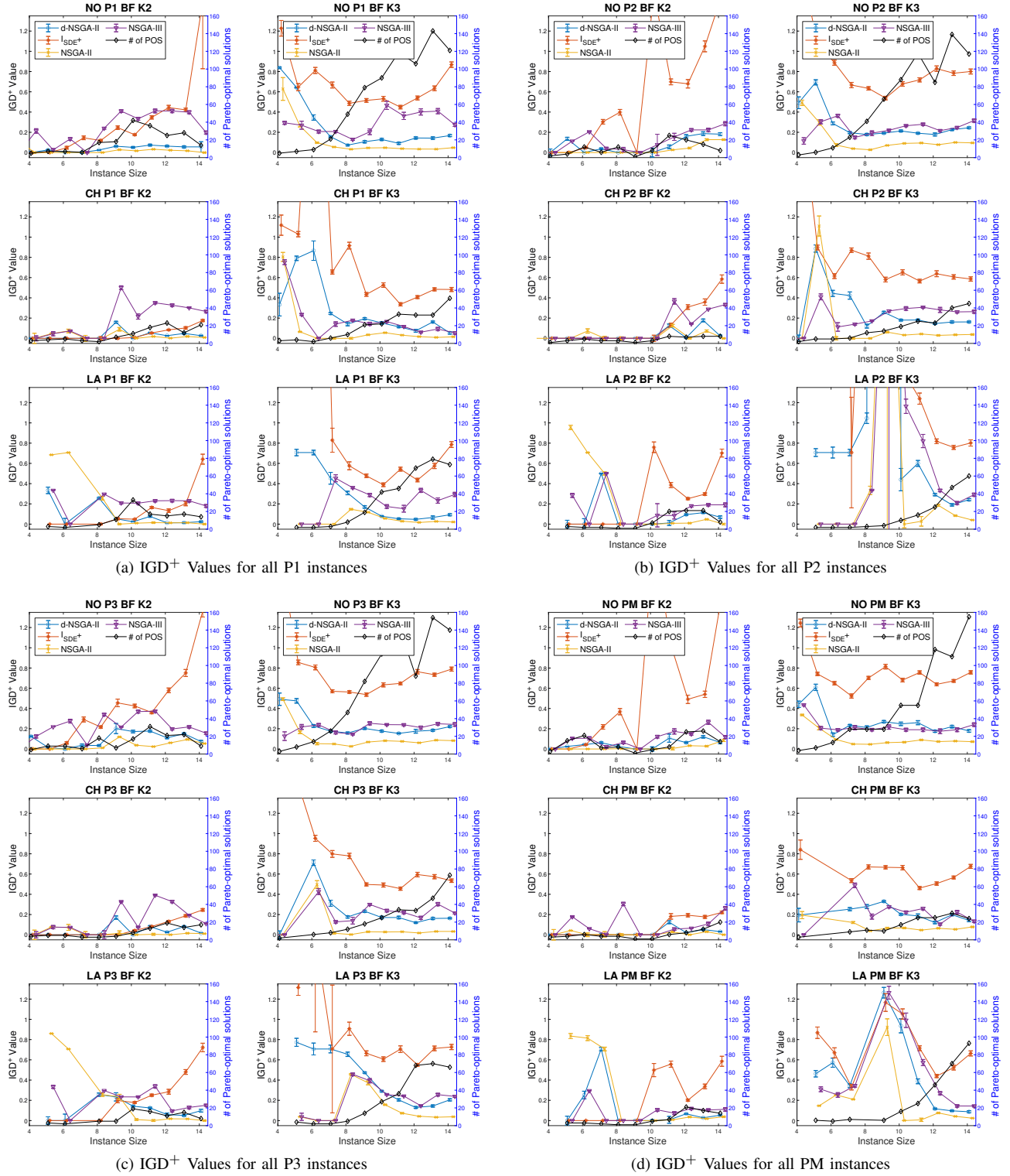statistikportal.de/_opendata2019.html
[4]https://developers.google.com/maps/documentation/elevation/start

(a) IGD$^+$ Values for all P1 instances

(b) IGD$^+$ Values for all P2 instances

(c) IGD$^+$ Values for all P3 instances
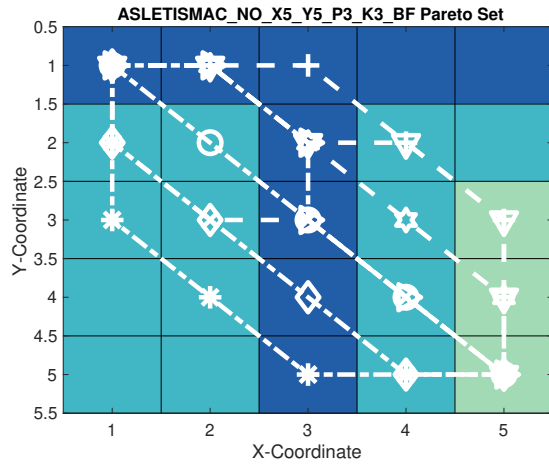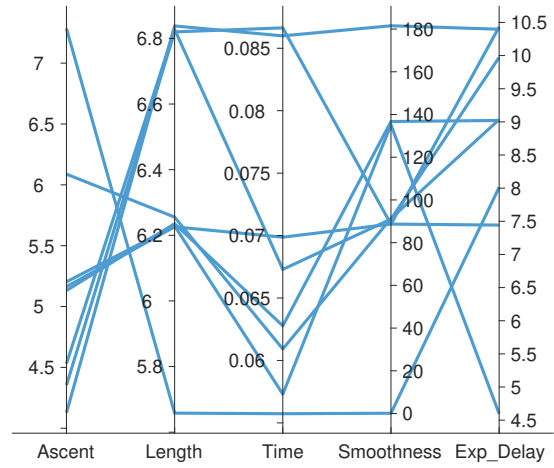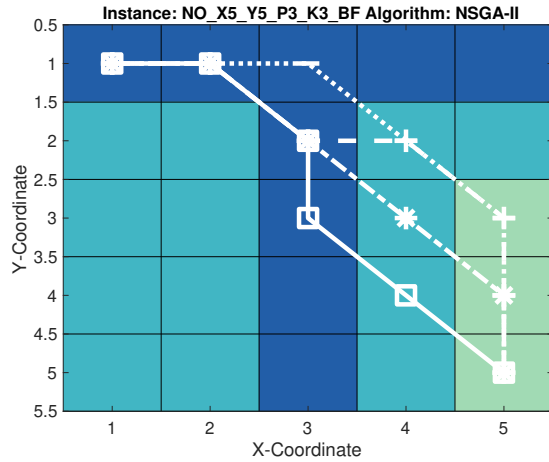
(d) IGD$^+$ Values for all PM instances

Fig. 9. The obtained IGD$^+$ values with respect to the different type, ordered by instance size
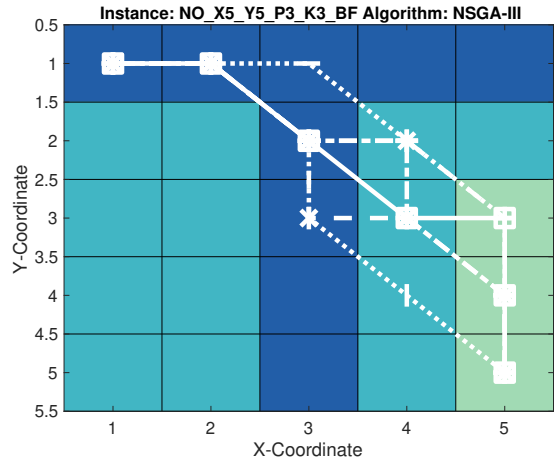
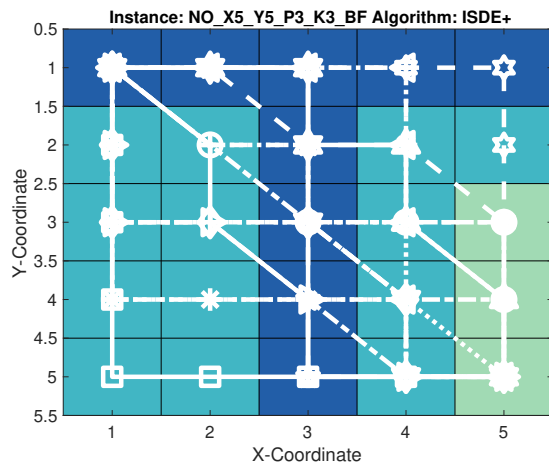(a) Pareto-Set of instance ASLETISMAC_NO_X5_Y5_P3_K3_BF

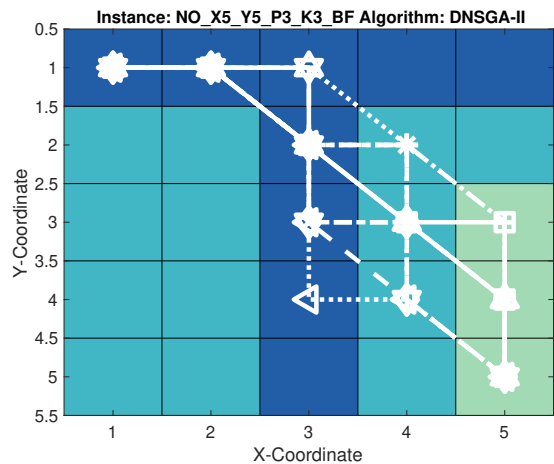(b) Pareto-Front of instance ASLETISMAC_NO_X5_Y5_P3_K3_BF

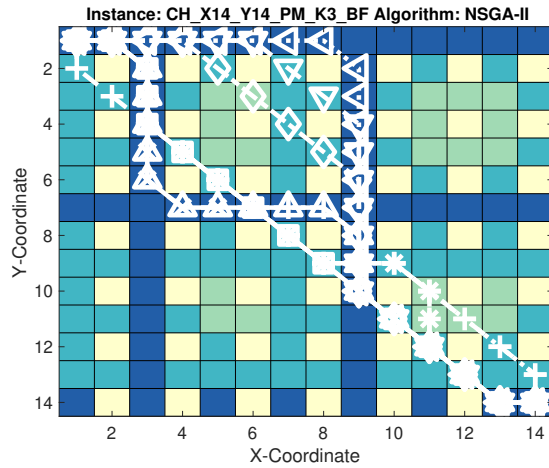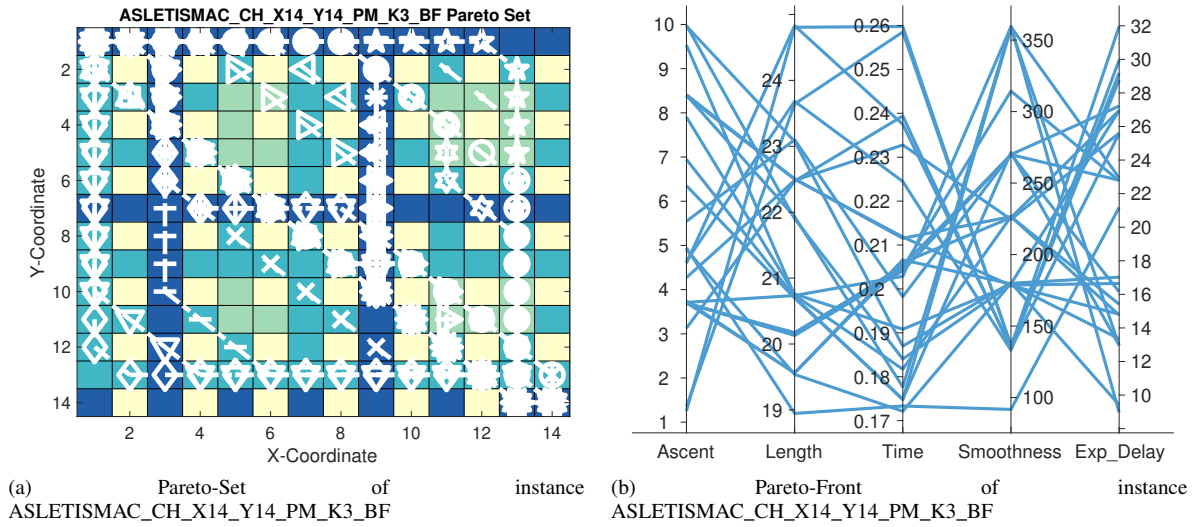(c) Result set of algorithm NSGA-II

(d) Result set of algorithm NSGA-III
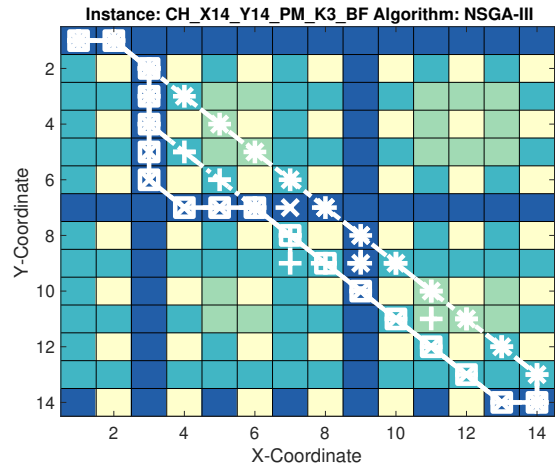
(e) Result set of algorithm $I_{SDE}+$

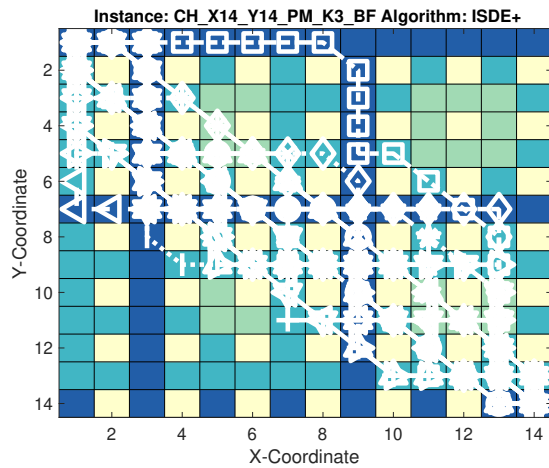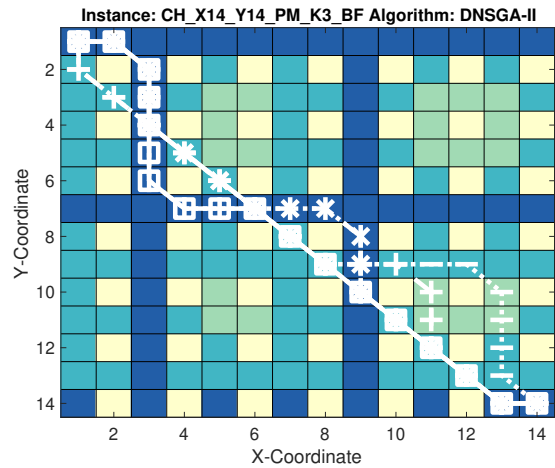(f) Result set of algorithm D-NSGA-II

Fig. 10. Pareto-Set and Front of instance ASLETISMAC_NO_X5_Y5_P3_K3_BF and result sets of all algorithm (median run with respect to IGD$^+$ value)

(a) Pareto-Set of instance ASLETISMAC_CH_X14_Y14_PM_K3_BF

(b) Pareto-Front of instance ASLETISMAC_CH_X14_Y14_PM_K3_BF

(c) Result set of algorithm NSGA-II

(d) Result set of algorithm NSGA-III

(e) Result set of algorithm I$_{SDE}+$

(f) Result set of algorithm D-NSGA-II

Fig. 11. Pareto-Set and Front of instance ASLETISMAC_CH_X14_Y14_PM_K3_BF and result sets of all algorithm (median run with respect to IGD$^+$ value)
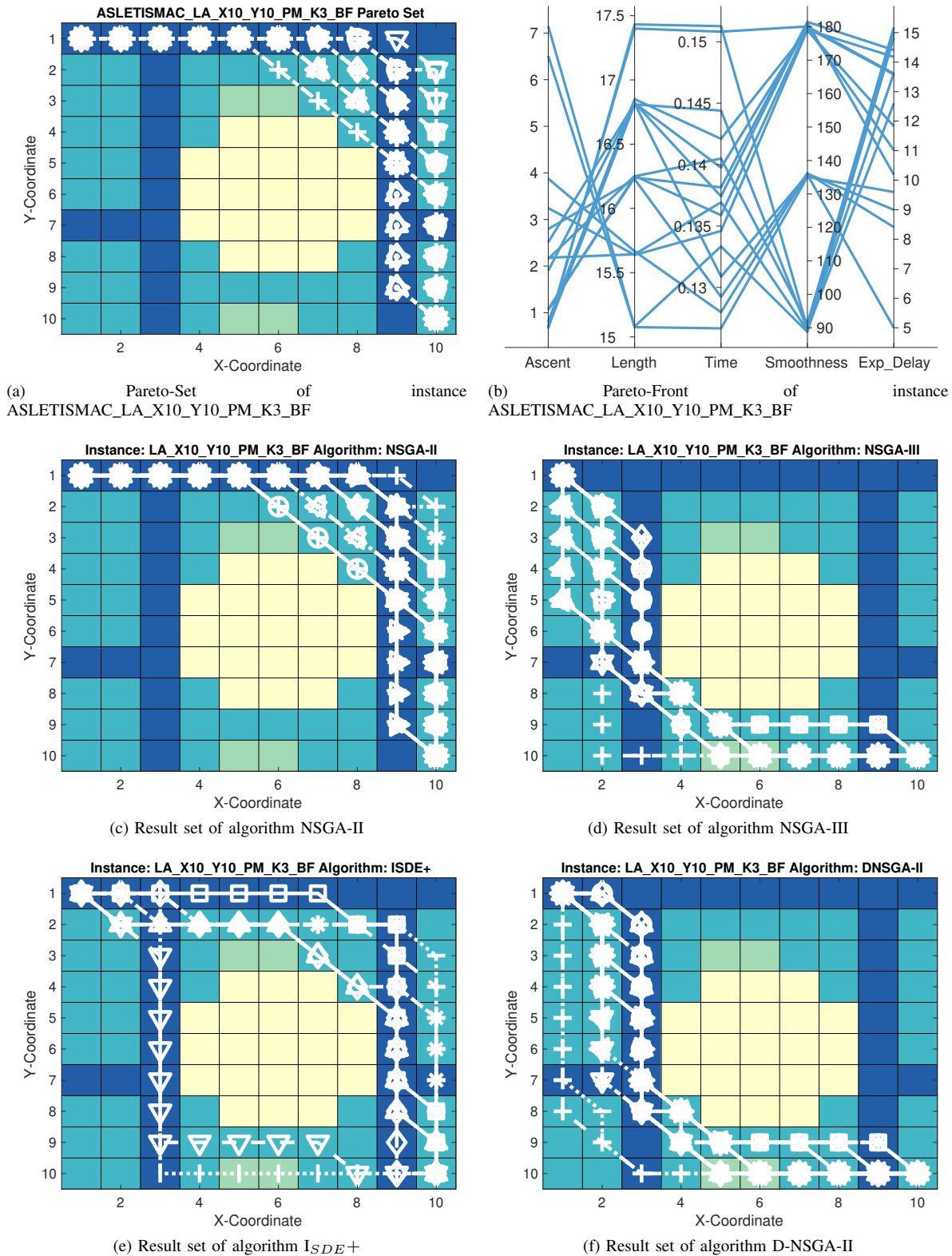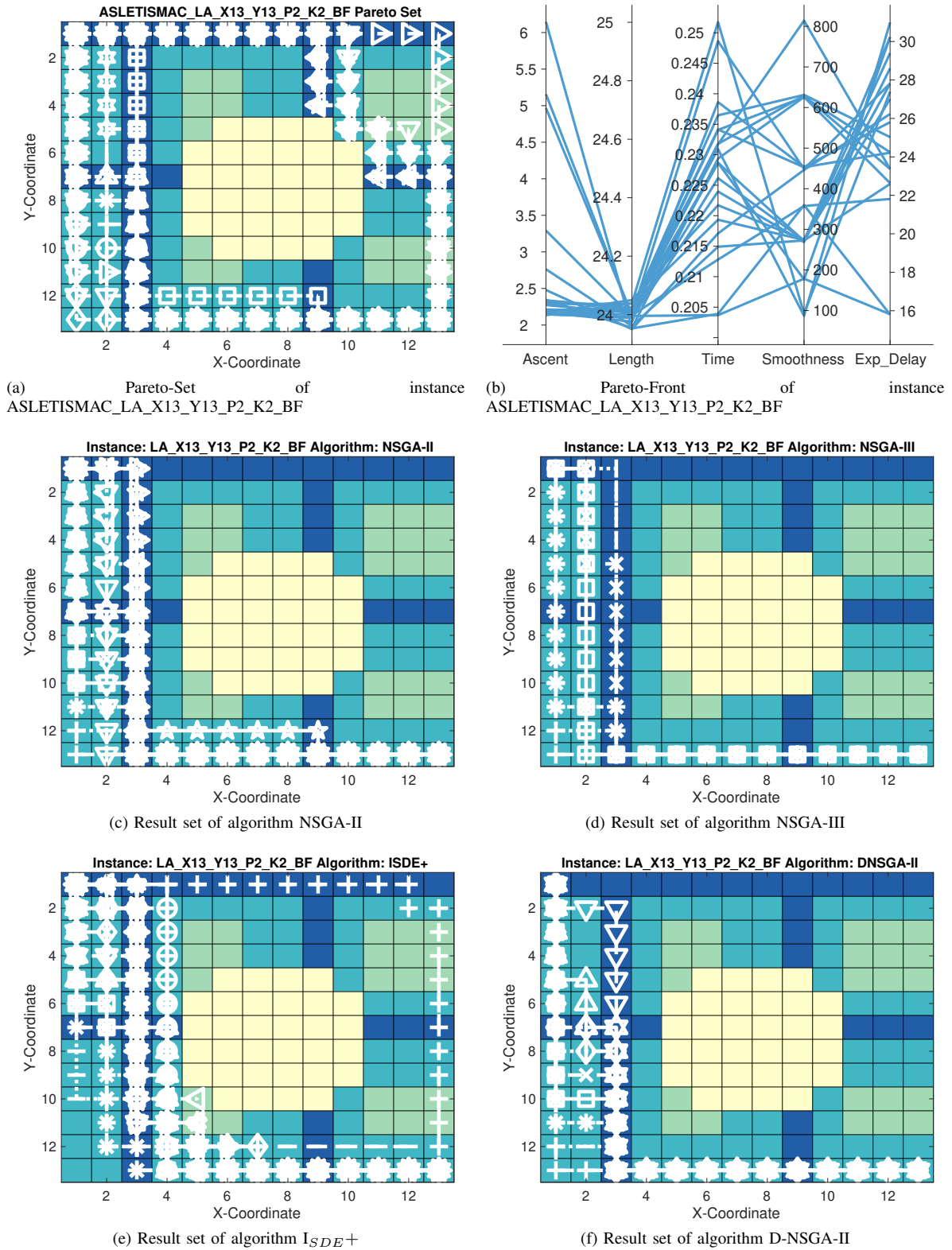
(a) Pareto-Set of instance ASLETISMAC_LA_X10_Y10_PM_K3_BF

(b) Pareto-Front of instance ASLETISMAC_LA_X10_Y10_PM_K3_BF

(c) Result set of algorithm NSGA-II

(d) Result set of algorithm NSGA-III

(e) Result set of algorithm I$_{SDE}+$

(f) Result set of algorithm D-NSGA-II

Fig. 12. Pareto-Set and Front of instance ASLETISMAC_LA_X10_Y10_PM_K3_BF and result sets of all algorithm (median run with respect to IGD$^+$ value)
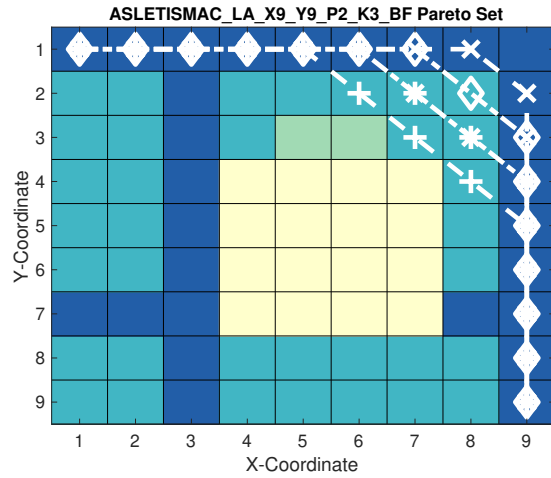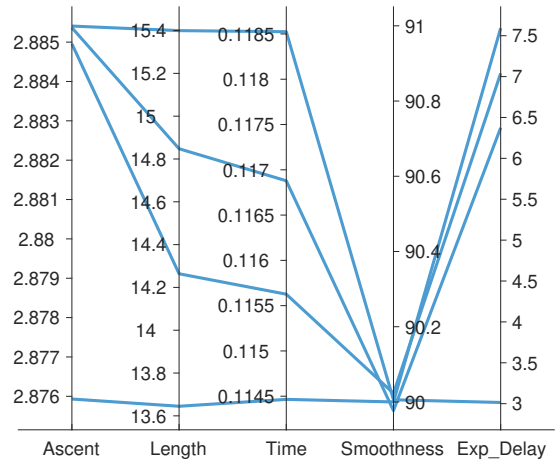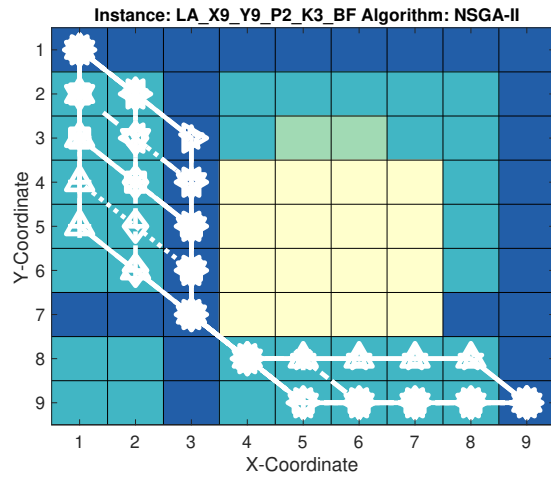
(a) Pareto-Set of instance ASLETISMAC_LA_X13_Y13_P2_K2_BF

(b) Pareto-Front of instance ASLETISMAC_LA_X13_Y13_P2_K2_BF

(c) Result set of algorithm NSGA-II

(d) Result set of algorithm NSGA-III

(e) Result set of algorithm $I_{SDE}+$

(f) Result set of algorithm D-NSGA-II

Fig. 13. Pareto-Set and Front of instance ASLETISMAC_LA_X13_Y13_P2_K2_BF and result sets of all algorithm (median run with respect to IGD^+ value)
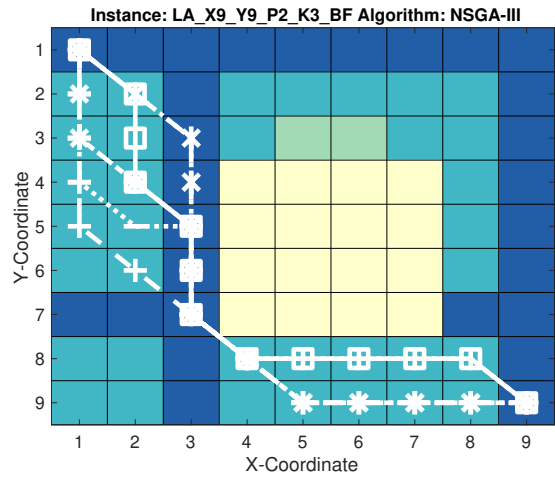
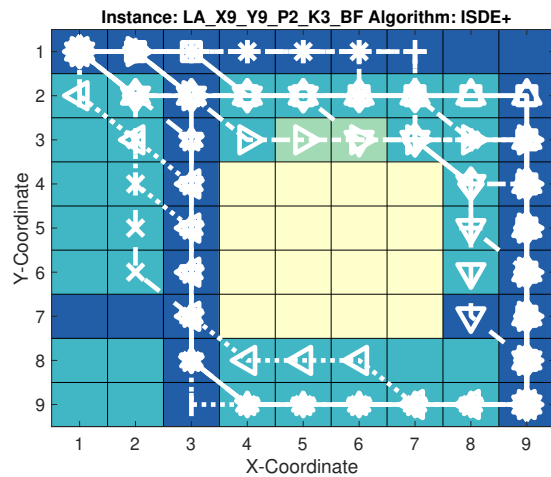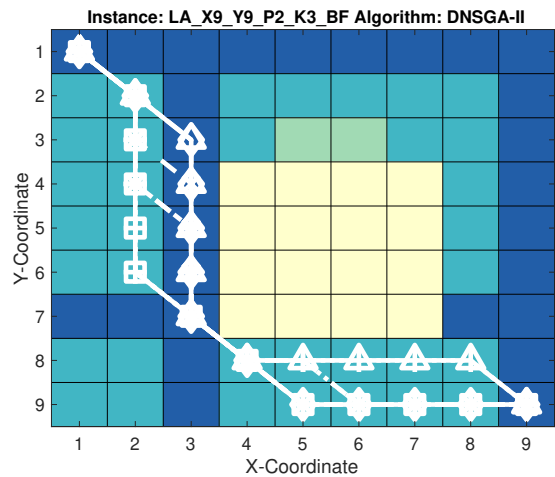(a) Pareto-Set of instance ASLETISMAC_LA_X9_Y9_P2_K3_BF

(b) Pareto-Front of instance ASLETISMAC_LA_X9_Y9_P2_K3_BF

(c) Result set of algorithm NSGA-II

(d) Result set of algorithm NSGA-III

(e) Result set of algorithm $I_{SDE}+$

(f) Result set of algorithm D-NSGA-II

Fig. 14. Pareto-Set and Front of instance ASLETISMAC_LA_X9_Y9_P2_K3_BF and result sets of all algorithm (median run with respect to IGD$^+$ value)