Exact Mean Computation in Dynamic Time Warping Spaces

Markus Brill[1], Till Fluschnik[1], Vincent Froese[1], Brijnesh Jain[2], Rolf Niedermeier[1], David Schultz[2]

[1]Institut fur Softwaretechnik und Theoretische Informatik, TU Berlin, Berlin, Germany, {brill,till.fluschnik,vincent.froese,rolf.niedermeier}@tu-berlin.de

[2]Distributed Artificial Intelligence Laboratory,TU Berlin, Berlin, Germany, {Brijnesh-Johannes.Jain,David.Schultz}@dai-labor.de

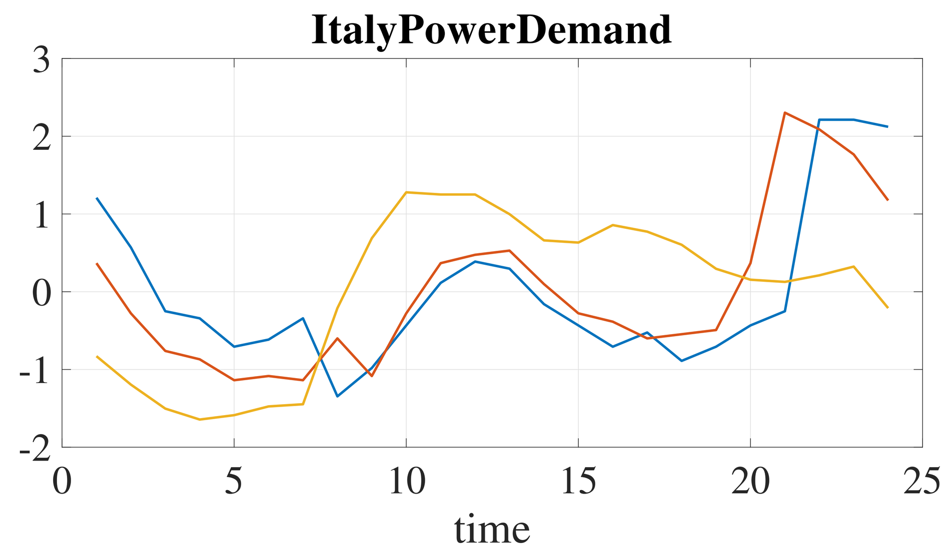
Abstract

Dynamic time warping constitutes a major tool for analyzing time series. In particular, computing a mean series of a given sample of series in dynamic time warping spaces (by minimizing the Fréchet function) is a challenging computational problem, so far solved by several heuristic, inexact strategies. We spot several inaccuracies in the literature on exact mean computation in dynamic time warping spaces. Our contributions comprise an exact dynamic program computing a mean (useful for benchmarking and evaluating known heuristics). Empirical evaluations reveal significant deficits of the state-of-the-art heuristics in terms of their output quality. Finally, we give an exact polynomial-time algorithm for the special case of binary time series.

Keywords: time series analysis, Fréchet function, exact dynamic programming, accuracy of heuristics

Time series

Ordered, finite sequence of rational numbers.

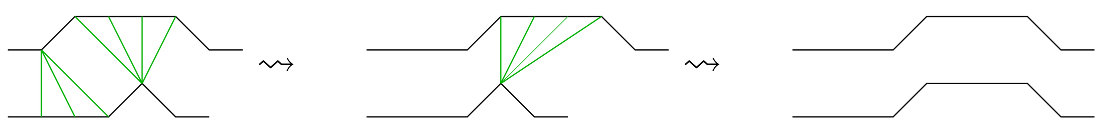


Measurements over time are ubiquitous, occurring in finance, multimedia, internet security, biology, climate research, medicine,...

Dynamic Time Warping (dtw)

Measures similarity of time series.

Idea: Stretch time series non-uniformly such that the resulting warped series align well in terms of some cost function.

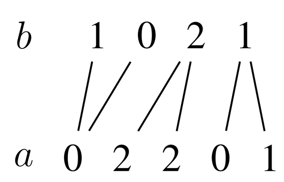


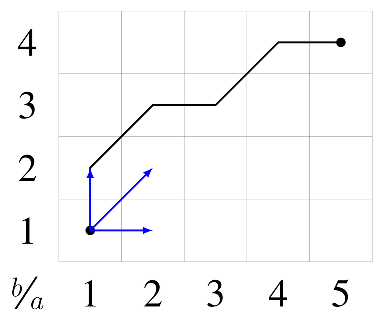
Definition

A *warping path of order* $m \times n$ is a sequence $p = (p_1, \dots, p_L)$ of index pairs $p_\ell = (i_\ell, j_\ell) \in [m] \times [n]$ such that
(i) $p_1 = (1, 1)$,
(ii) $p_L = (m, n)$, and
(iii) $p_{\ell+1} - p_\ell := (i_{\ell+1} - i_\ell, j_{\ell+1} - j_\ell) \in \{(1, 0), (0, 1), (1, 1)\}$ for all $1 \le \ell \le L - 1$.

Example: Time series $a = (0, 2, 2, 0, 1)$ and $b = (1, 0, 2, 1)$.

$p = ((1, 1), (1, 2), (2, 3), (3, 3), (4, 4), (5, 4))$

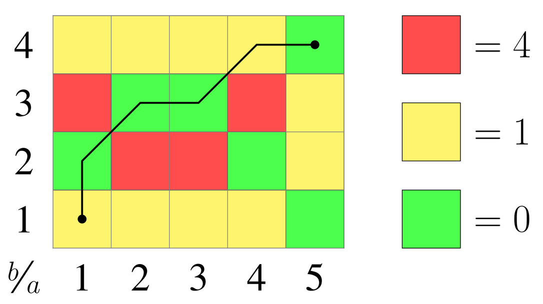




The *cost* $C_p(x, y)$ of warping path $p = (p_1, \dots, p_L)$ is defined as

$$C_p(x, y) := \sum_{\ell=1}^{L} (x_{i_\ell} - y_{j_\ell})^2.$$

$C_p(a, b) = 2$



The *dtw-distance* between $x = (x_1, \dots, x_m)$ and $y = (y_1, \dots, y_n)$ is

$$\mathrm{dtw}(x, y) := \min_{p \in \mathcal{P}_{m,n}} \left\{ \sqrt{C_p(x, y)} \right\}.$$

($\mathcal{P}_{m,n}$ is the set of all warping paths of order $m \times n$.)

Clustering tasks often require to compute a *mean*.

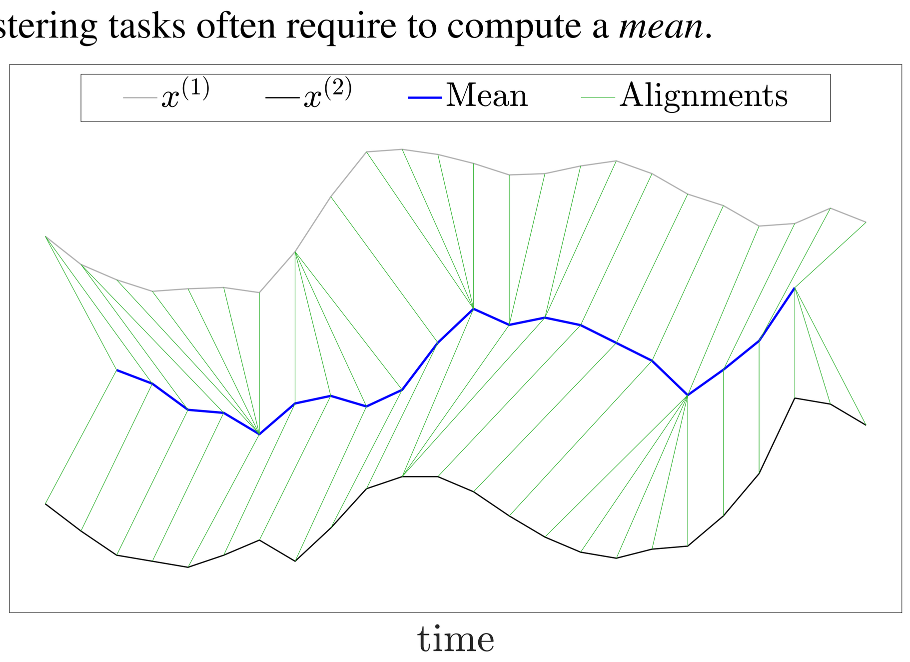


Figure 1: Mean of two time series from the ItalyPowerDemand dataset. The time series have been shifted along both axes for clearer presentation of the alignments.

DTW-MEAN
Input: Sample $\mathcal{X} = \left(x^{(1)}, \dots, x^{(k)}\right)$ of k univariate rational time series.
Task: Find a univariate rational time series z that minimizes the Fréchet function

$$F(z) = \frac{1}{k} \sum_{i=1}^{k} \left(\mathrm{dtw}(z, x^{(i)})\right)^2.$$

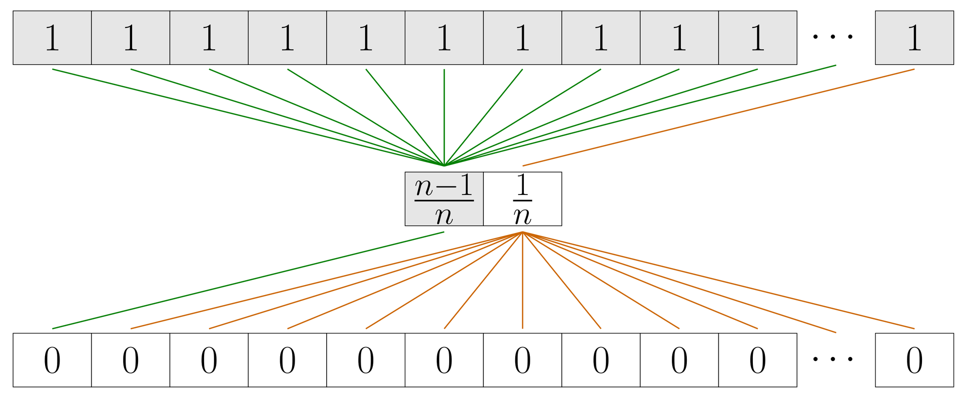


Figure 2: Example of a mean of length two of two time series of length n.

Our Contributions

- First *exact algorithm* running in $O(n^{2k+1} 2^k k)$ time
- *Benchmark* of existing heuristics
- $O(n^3 k)$-time algorithm computing a *binary mean* of binary time series

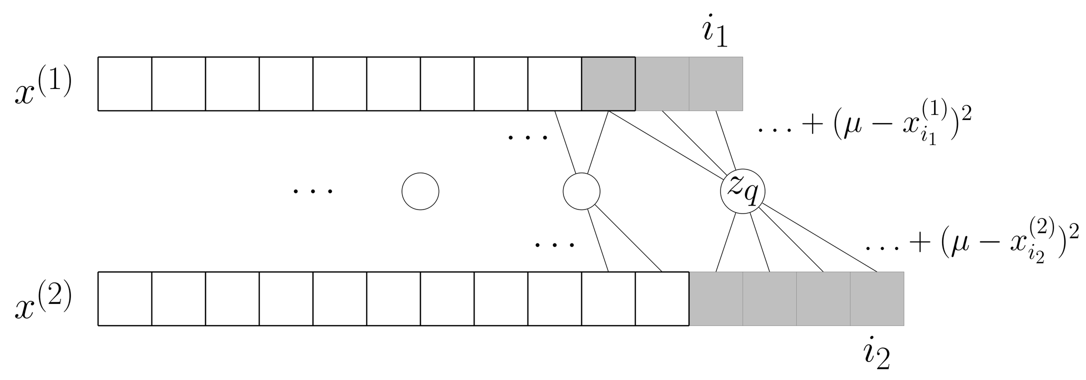
Theorem

A mean of k time series $x^{(1)}, \dots, x^{(k)}$ of maximum length n can be computed in $O(n^{2k+1} \cdot 2^k \cdot k)$ time.

Idea: Use dynamic programming.

Define k-dimensional table C of size n^k storing F-values of means for all subinstances.

Iteratively compute each table entry in $O(n^k \cdot 2^k \cdot nk)$ time.



Experiments

Goal: Assess performance of 5 state-of-the-art heuristics.

Algorithm	Acr.	Ref.
exact dynamic programming	EDP	
multiple alignment	MAL	[3, 4]
DTW barycenter averaging	DBA	[5]
soft-dtw	SDTW	[2]
batch subgradient	BSG	[2, 6]
stochastic subgradient	SSG	[6]

Table 1: List of algorithms compared in our experiments.

Data: 10 datasets from UCR Time Series Classification Archive.

UCR	#	L
ItalyPowerDemand	1096	24
SyntheticControl	600	60
SonyAIBORobotSurface1	621	70
ProximalPhalanxTW	605	80
ProximalPhalanxOutlineCorrect	891	80
MedicalImages	1141	99
TwoPatterns	5000	128
FaceAll	2250	131
ECG5000	5000	140
GunPoint	200	150

Table 2: Ten UCR time series data sets. (#: numbers of time series in the data set; L: lengths of the time series.)

Experiment 1

Compare algorithms on 1,000 pairs of time series of equal length varying from 24 to 150.

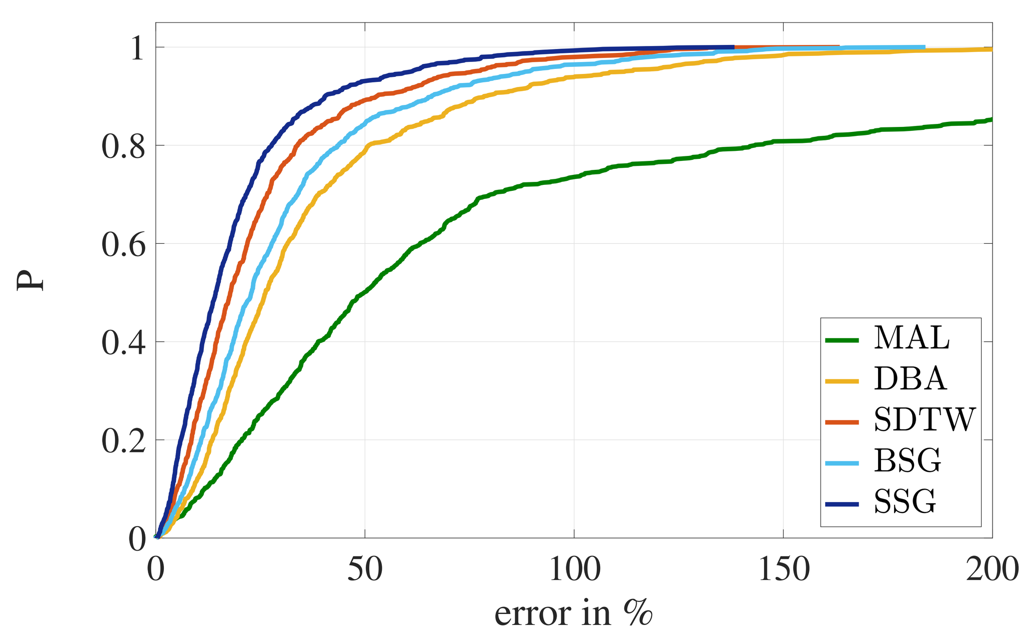


Figure 3: Results of experiment E1.

Error (%) of algorithm $A = 100 \cdot (F_A / F_{\text{opt}} - 1)$

F_A = F-value of solution found by A, F_{opt} = optimal F-value (DP)

	average error percentage	standard deviation	maximum error percentage	# of optimal solutions found
MAL	114.2	175.1	1093.9	3
DBA	37.1	34.7	274.3	0
SDTW	24.7	23.2	163.5	0
BSG	30.6	27.3	184.0	1
SSG	19.5	18.8	138.3	0

Table 3: Overview of the results from experiment E1.)

The heuristics perform rather poorly with an average error percentage of at least 19% and a maximum error percentage of at least 138%.

Experiment 2

Compare algorithms on 1,000 samples of k (downsampled) time series of length 8 for each $k = 2, \dots, 5$.

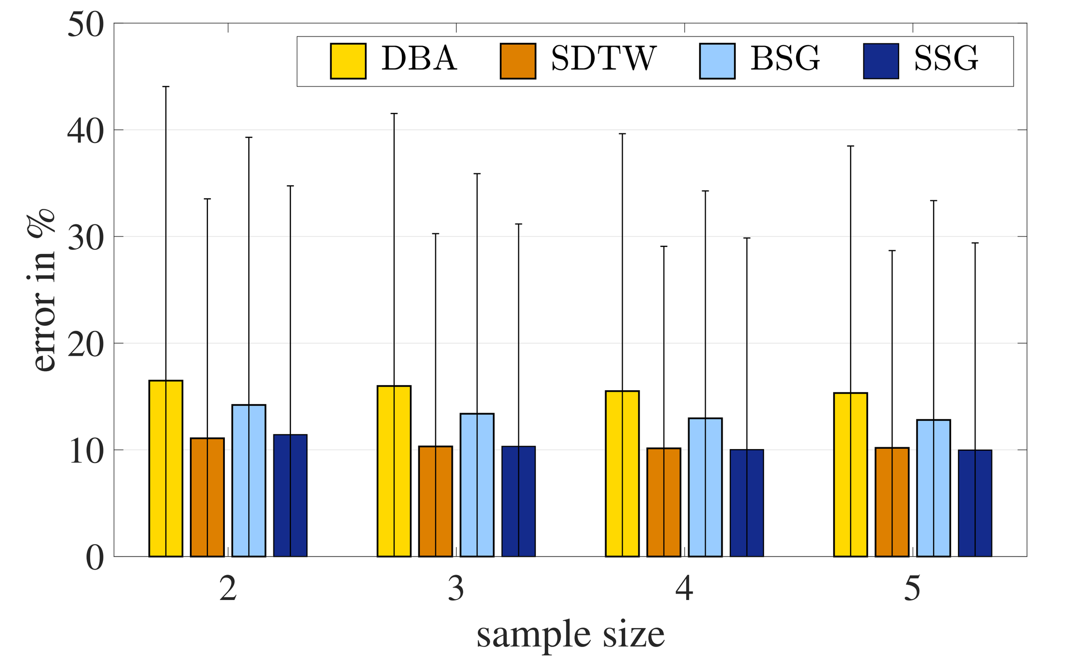


Figure 4: Results of experiment E2. Shown are the error percentages as a function of the sample size. Average error percentage and standard deviation.

The heuristics on average still deviate by at least 10% from the exact solution in all cases. All heuristics slightly improve on average with increasing sample size. A similar statement can be made for the standard deviation, which is still at a high level.

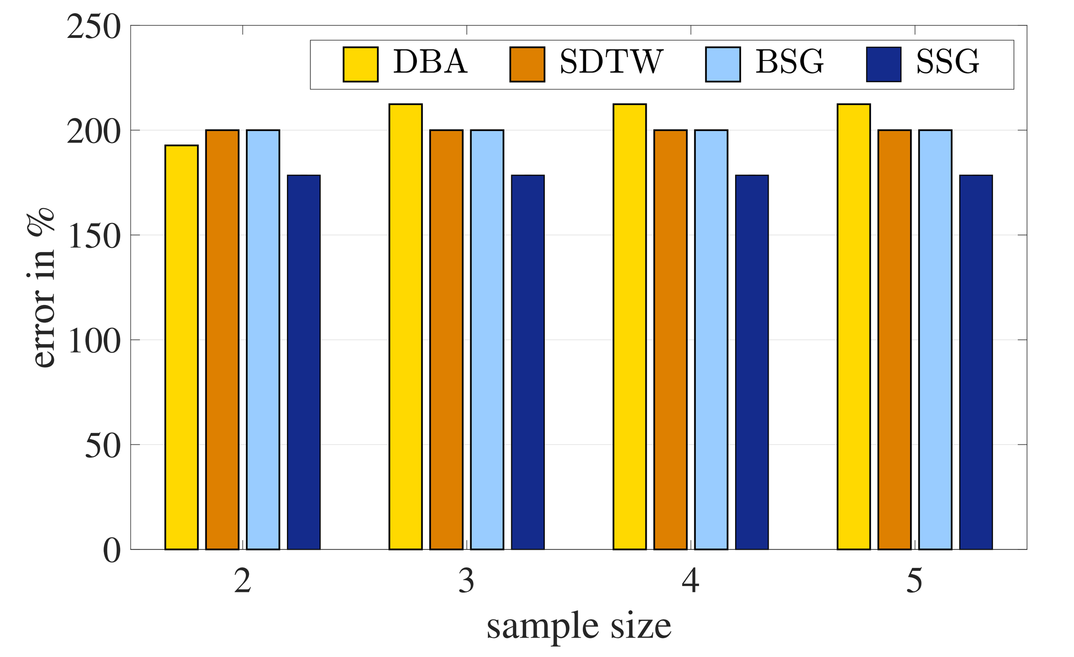


Figure 5: Results of experiment E2. Shown are the error percentages as a function of the sample size. Maximum error percentage.

In contrast, there is no visible trend with respect to the maximum error percentage.

Conclusion and Open Questions

- DTW-MEAN is a challenging (combinatorial) optimization problem solvable in $O(n^{2k+1} 2^k k)$ time (polynomial time for constant k)
- State-of-the-art heuristics perform rather poorly
- A binary mean of binary time series can be found in polynomial time

Open Questions:

Practical:

- Improve running time and find better heuristics
- Study characteristics (typical length, uniqueness)
- Performance of exact means in applications?
- Alternative mean definitions?

Theoretical:

- Polynomial-time solvable for fixed-length time series?
- Prove approximation guarantees

Remark: DTW-MEAN was recently proven to be NP-hard, W[1]-hard w.r.t k, and not $n^{o(k)}$-time solvable (assuming ETH) [1].

References

[1] Laurent Bulteau, Vincent Froese, and Rolf Niedermeier. Hardness of consensus problems for circular strings and time series averaging. *arXiv: 1804.02854*, 2018.
[2] Marco Cuturi and Mathieu Blondel. Soft-DTW: a differentiable loss function for time-series. In *Proc. of 34th ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 894–903. PMLR, 2017.
[3] Ville Hautamaki, Pekka Nykanen, and Pasi Franti. Time-series clustering by approximate prototypes. In *Proc. of 19th ICPR*, pages 1–4. IEEE, 2008.
[4] François Petitjean and Pierre Gançarski. Summarizing a set of time series by averaging: From Steiner sequence to compact multiple alignment. *Theor. Comput. Sci.*, 414(1):76–91, 2012.
[5] François Petitjean, Alain Ketterlin, and Pierre Gançarski. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, 44(3):678–693, 2011.
[6] David Schultz and Brijnesh Jain. Nonsmooth analysis and subgradient methods for averaging in dynamic time warping spaces. *Pattern Recognition*, 74(Supplement C):340–358, 2018.

Acknowledgements. This work is supported by the Deutsche Forschungsgemeinschaft under grants JA 2109/4-1 and NI 369/13-2, and by a Feodor Lynen return fellowship of the Alexander von Humboldt Foundation. The work on the theoretical part of this paper started at the research retreat of the Algorithmics and Computational Complexity group, TU Berlin, held at Boiensdorf, Baltic Sea, April 2017, with MB, TF, VF, and RN participating.