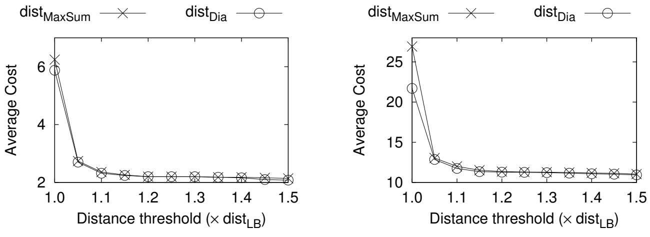
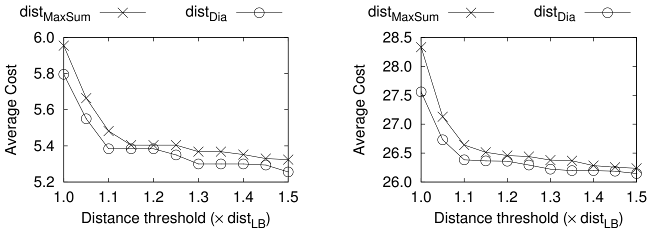
Cost-Aware and Distance-Constrained Collective Spatial Keyword Query (Appendix)



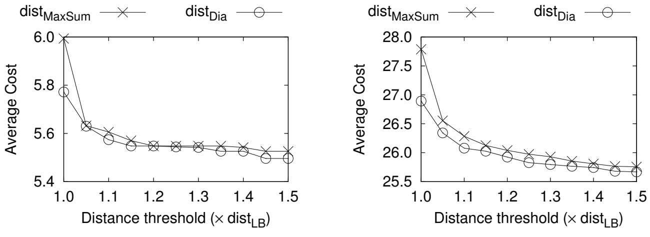
(a) $cost_{Max}$ (b) $cost_{Sum}$

Fig. 10. Effect of average cost on B (Yelp)



(a) $cost_{Max}$ (b) $cost_{Sum}$

Fig. 11. Effect of average cost on B (Hotel)



(a) $cost_{Max}$ (b) $cost_{Sum}$

Fig. 12. Effect of average cost on B (GN)

Appendix A
Proof of the NP-hardness of Approximation

Proof: We prove this theorem by a reduction from the collective spatial keyword query (CoSKQ) problem [21]. Given a query q with a location $q.\lambda$ and a set of keywords $q.\psi$, the CoSKQ problem is to find a set of objects G such that (1) they cover all the query keywords and (2) the distance of G, $dist(G)$, is minimized. The decision problem of the CoSKQ problem is that given a problem instance of CoSKQ and a value C, it checks whether there exists a set of objects G such that G covers $q.\psi$ and $dist(G) < C$. It has been shown in [21] that the CoSKQ problem with both the $dist_{Dia}(G)$ function and the $dist_{MaxSum}(G)$ function is NP-hard.

We prove by contradiction. Suppose that we have a polynomial-time c-approximation algorithm $\mathcal{A}$ for the CD-CoSKQ problem with $c \geq 1$. In other words, in the case that the problem instance of CD-CoSKQ has feasible solutions, $\mathcal{A}$ would return a feasible solution with its cost at most c times the cost of the optimal solution; and it returns an empty set otherwise. It follows that this algorithm could be used solve the decision problem of the CoSKQ problem as follows.

Given the decision problem of a CoSKQ instance, we run $\mathcal{A}$ with the query location and query keywords the same as those of the CoSKQ problem and the distance threshold B as C. Then, if $\mathcal{A}$ returns a non-empty solution, we conclude that the answer to the decision problem is yes; and otherwise, no. Thus, this leads to a contradiction which finishes the proof. □

Appendix B
Setting the default distance threshold B

The results for the dataset Yelp are shown in Figure 10. According to the results, the average costs of the solutions decrease when the distance threshold B increases. Still, the rate of decrease is very small when $n > 1.1$. Thus, we set the default value of B to 1.1 times $dist_{LB}$.

The results for the dataset Hotel and GN with different distance threshold give similar clues and are shown in Figure 11 and Figure 12, respectively.

Appendix C
Effect of query size

Dataset Hotel. The results with $cost_{Max}$ and $dist_{MaxSum}$ are presented in Figure 13. According to Figure 13(a), the running times of the algorithm increase when $|q.\psi|$ increase, and our CD-Exact runs faster than Combi-Exact. According to Figure 13(b), our CD-Appro runs faster than Cao-Appro and Long-Appro, and it can always achieve cost ratio α smaller than 1. Besides, the distance ratio β of CD-Appro is slightly larger than Cao-Appro and Long-Appro, but is close to 1.

The results with $cost_{Sum}$ and $dist_{MaxSum}$ are presented in Figure 14. According to Figure 14(a), our CD-Exact runs faster than Combi-Exact, and their difference increases with the query size. According to Figure 14(b), the approximation algorithms have similar running times. CD-Appro achieve better cost ratios than Cao-Appro and Long-Appro consistently, while CD-Appro has the distance ratios close to 1.

The results with $cost_{Max}$ and $dist_{Dia}$ for dataset Hotel are similar and are presented in Figure 15. The results with $cost_{Sum}$ and $dist_{Dia}$ for dataset Hotel are presented in Figure 16.

Dataset GN. The results with $cost_{Sum}$ and $dist_{MaxSum}$ are presented in Figure 18. According to Figure 18(a), our CD-Exact runs faster than Combi-Exact, especially when query size is large. According to Figure 18(b), CD-Appro and Cao-Appro have similar running times, while Long-Appro is much slower. Besides, CD-Appro always achieve the cost ratio close to 1, and outperform Cao-Appro and

Long-Appro. All of them have distance ratios close to 0.9 and smaller than 1.

The results with $cost_{Max}$ and $dist_{Dia}$ for dataset GN are presented in Figure 19. The results with $cost_{Sum}$ and $dist_{Dia}$ for dataset GN are presented in Figure 20.

Appendix D
Effect of B

We set the distance threshold $B = dist_{LB} \times n$, where $dist_{LB}$ is the distance cost of the solution found by the approximation algorithm [21] for the CoSKQ problem. We vary n from $\{1.0, 1.05, 1.1, 1.15, 1.20\}$. The default value of $|q.\psi|$ = 6.

The results with $cost_{Max}$ and $dist_{MaxSum}$ on the dataset Yelp are shown in Figure 21. According to Figure 21(a), the running times of both CD-Exact and Combi-Exact do not change much when B increases, and CD-Exact is much faster than Combi-Exact. It is probably because when B increases, the number of relevant objects increases, but at the same time it would be easier to find the feasible set with minimum cost in an iteration since the budget is relaxed.

According to Figure 21(b), the running times of the approximation algorithms do not change much when B increases, and both CD-Appro runs much faster than Cao-Appro and Long-Appro. For CD-Appro, it is probably because when B increases, the number of objects processed increases with the number of key objects. But on the other hand, a better solution could possibly be found within each iteration, and reducing the total number of iterations needed. Thus, the overall running times remain similar. Besides, the cost ratios α of the approximation algorithms increase when B increases, while that of CD-Appro remains close to 1. The reason is that a larger B could allow a smaller cost in the optimal solution, but Cao-Appro and Long-Appro cannot fully utilize this advantage, while our CD-Appro is able to explore possible better solutions. Besides, the distance ratio β of the approximation algorithms decrease when B increases. This is simply because B is the denominator in calculating the distance ratios.

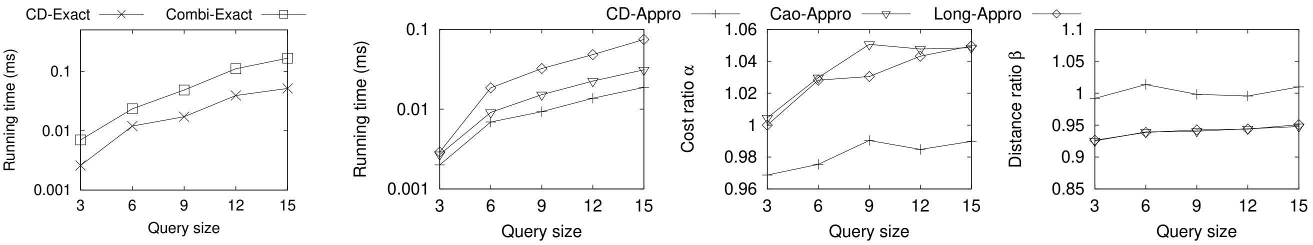
The results with $cost_{Sum}$ and $dist_{MaxSum}$ on the dataset Yelp are shown in Figure 22. According to Figure 22(a), the running times of both CD-Exact and Combi-Exact increase when B increases, and CD-Exact is much faster than Combi-Exact. According to Figure 22(b), CD-Appro runs faster than Cao-Appro and Long-Appro, and only increase slightly when B increases.

The results on $dist_{Dia}$ provide similar clues and are presented in Figure 23 and Figure 24.

Appendix E
Scalability Test

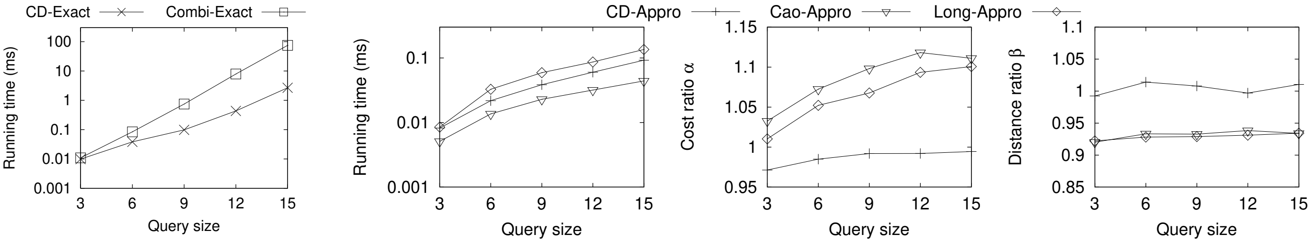
The scalability test results with $cost_{Sum}$ and $dist_{MaxSum}$ are presented in Figure 25. According to Figure 25(a), our CD-Exact is scalable wrt to the number of objects in the datasets, e.g., it ran within 10s on a dataset with 10M objects. Besides, according to Figure 25(b), our CD-Appro is scalable to large datasets, e.g., they ran within 2s on a dataset with 10M objects. The cost ratio α of CD-Appro is always smaller than 1, while its distance ratio β is slightly larger than 1.

The results with $cost_{Max}$ with $dist_{Dia}$ are similar and are presented in Figure 26. The results with $cost_{Sum}$ with $dist_{Dia}$ are presented in Figure 27.



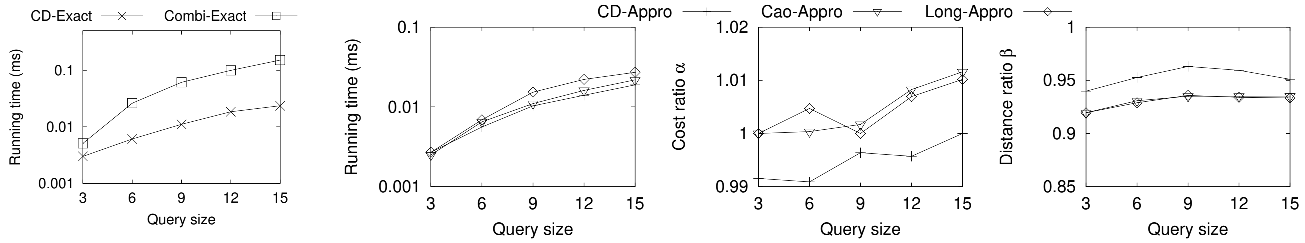
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 13. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Max}$, $dist_{MaxSum}$, Hotel)



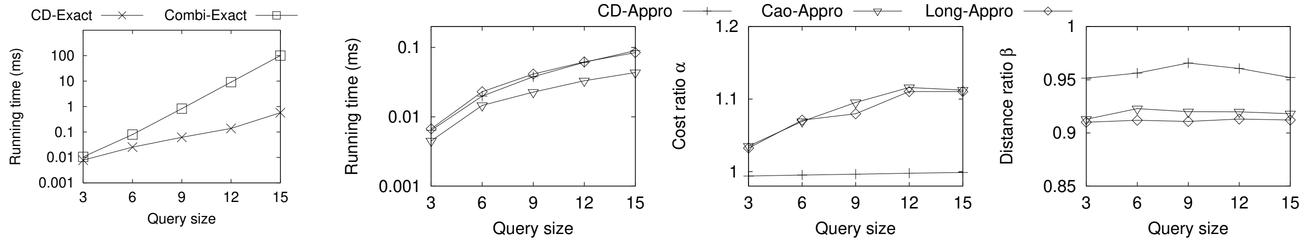
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 14. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Sum}$, $dist_{MaxSum}$, Hotel)



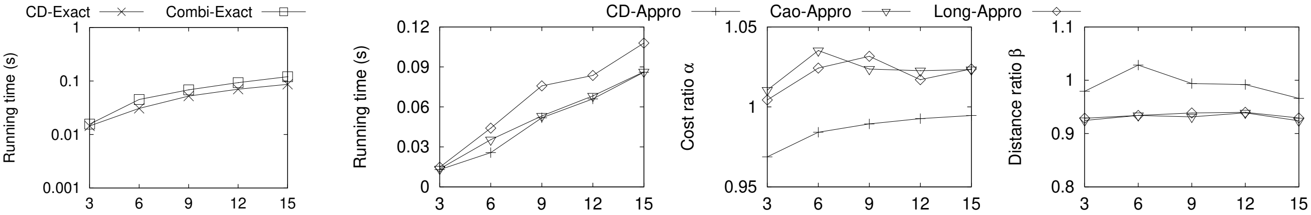
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 15. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Max}$, $dist_{Dia}$, Hotel)



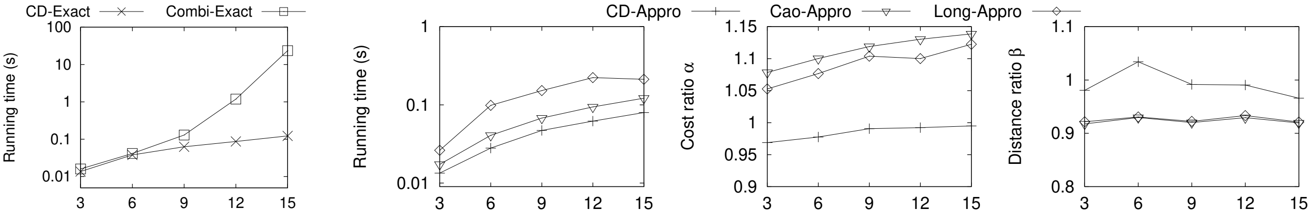
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 16. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Sum}$, $dist_{Dia}$, Hotel)



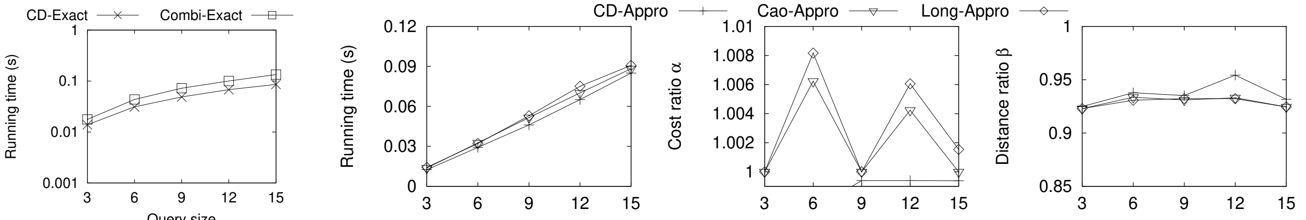
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 17. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Max}$, $dist_{MaxSum}$, GN)



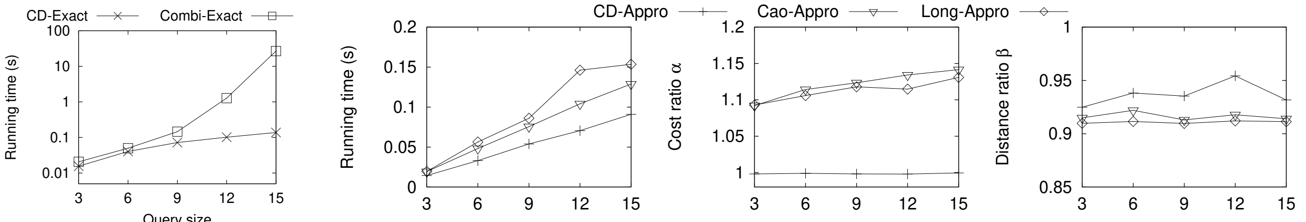
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 18. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Sum}$, $dist_{MaxSum}$, GN)



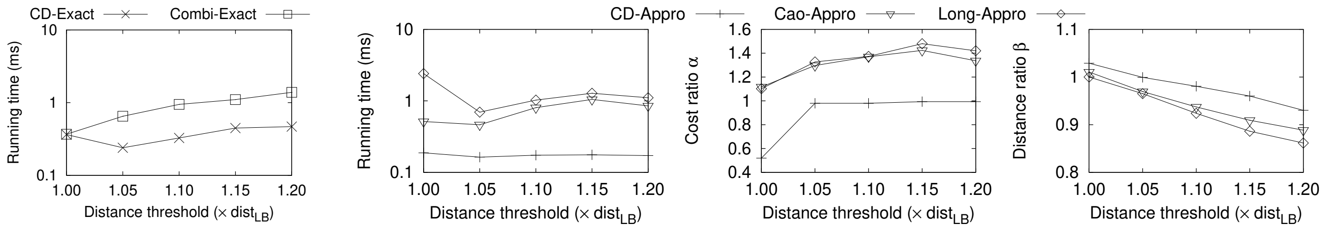
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 19. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Max}$, $dist_{Dia}$, GN)



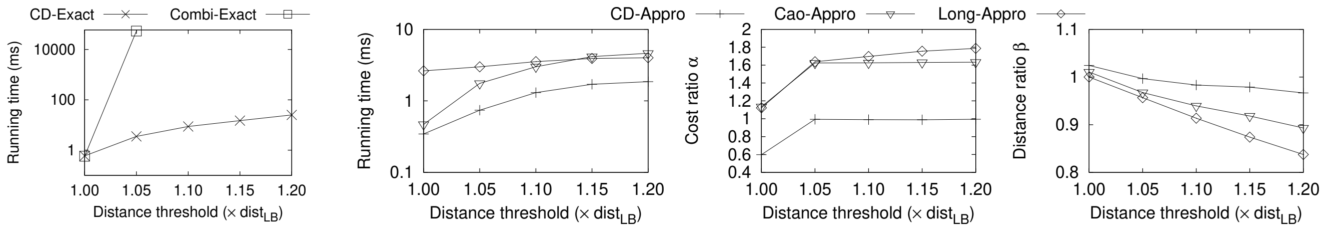
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 20. Effect of Query Size (i.e., $|q.\psi|$) ($cost_{Sum}$, $dist_{Dia}$, GN)



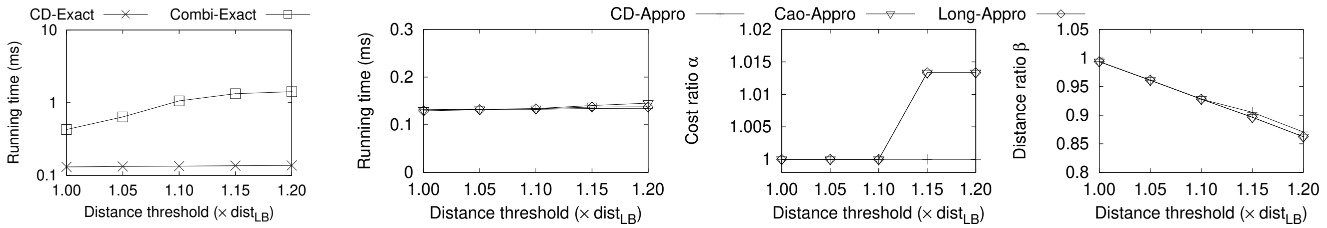
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 21. Effect of B ($cost_{Max}$, $dist_{MaxSum}$, Yelp)



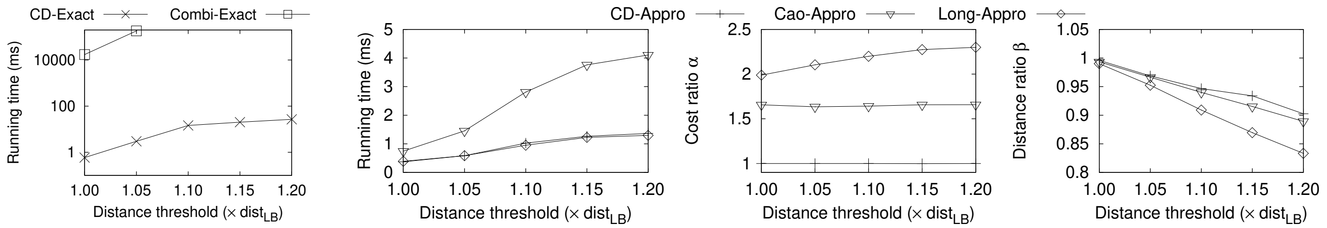
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 22. Effect of B ($cost_{Sum}$, $dist_{MaxSum}$, Yelp)



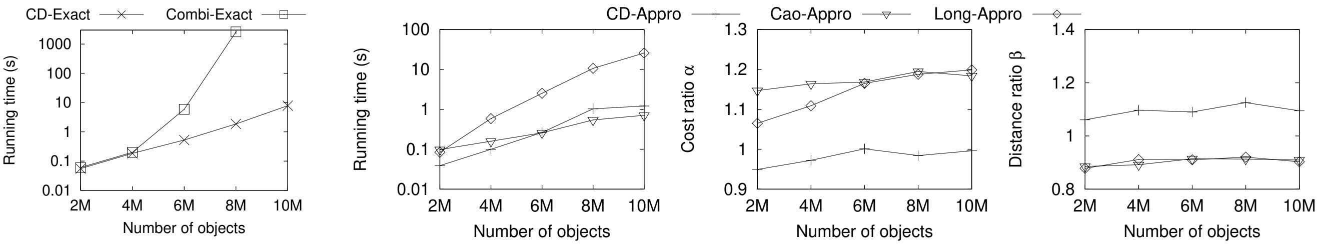
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 23. Effect of B ($cost_{Max}$, $dist_{Dia}$, Yelp)



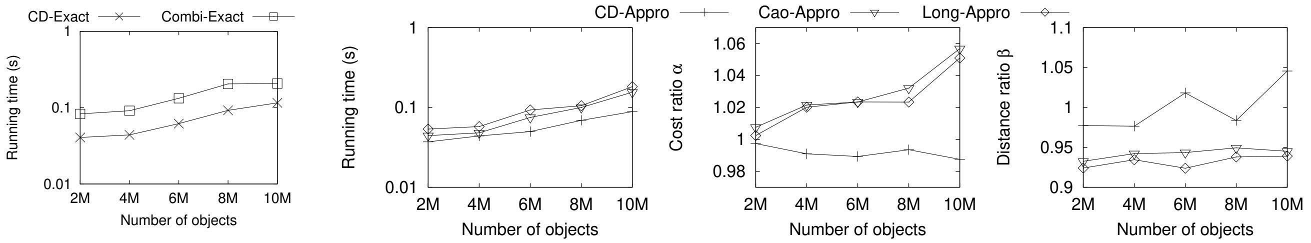
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 24. Effect of B ($cost_{Sum}$, $dist_{Dia}$, Yelp)



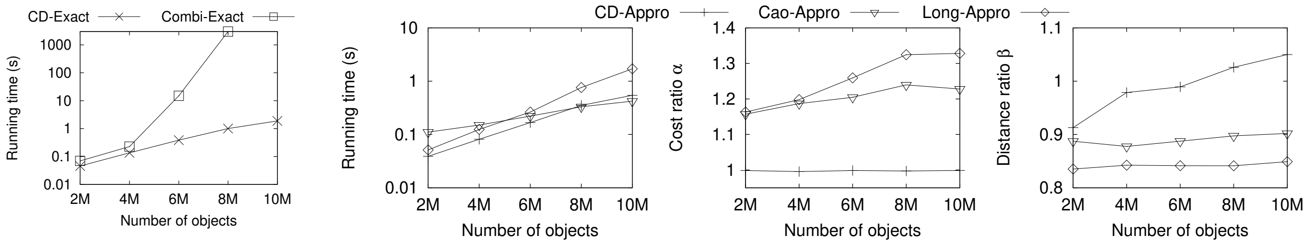
(a) Exact Algorithms (b) Approximation Algorithms

Fig. 25. Scalability Test ($cost_{Sum}$, $dist_{MaxSum}$)



(a) Exact Algorithms (b) Approximation Algorithms

Fig. 26. Scalability Test ($cost_{Max}$, $dist_{Dia}$)



(a) Exact Algorithms (b) Approximation Algorithms

Fig. 27. Scalability Test ($cost_{Sum}$, $dist_{Dia}$)