On Discriminating Geometric Objects and Its Relatives

Franck NIELSEN
SONY Computer Science Laboratory Inc.
E-mail: nielsen@csl.sony.co.jp

Abstract

Let $\mathcal{O} = \{O_1, ..., O_n\}$ be a family of n geometric objects. In this talk, we investigate the relationships between the following quantities:

0-Transversal number. Find the minimum number $\tau_0(\mathcal{O})$ such that there exists a point set $\mathcal{P}_0(\mathcal{O})$ of size $\tau_0(\mathcal{O})$ stabbing every object of $\mathcal{O}$, i.e. $O_i \cap \mathcal{P}_0(\mathcal{O}) \neq \emptyset$ for all $1 \leq i \leq n$.

Packing number. Find the maximum number $\phi(\mathcal{O})$ such that there exists a subset $\mathcal{I}(\mathcal{O}) \subseteq \mathcal{O}$ of pairwise non-intersecting objects ($|\mathcal{I}(\mathcal{O})| = \phi(\mathcal{O})$), i.e. $\forall O_i, O_j \in \mathcal{I}(\mathcal{O})$ with $j \neq i$ we have $O_i \cap O_j = \emptyset$.

Discriminating number. Find the minimum number $\delta(\mathcal{O})$ such that there exists a point set $\mathcal{D}(\mathcal{O})$ discriminating every object of $\mathcal{O}$ ($|\mathcal{D}(\mathcal{O})| = \delta(\mathcal{O})$), i.e. $O_i \cap \mathcal{D}(\mathcal{O}) \neq O_j \cap \mathcal{D}(\mathcal{O})$ for all $i \neq j$.

Since computing each of these numbers and their corresponding point sets are NP-complete, we focus on the relationships between these quantities and their approximation heuristics. We also compare with previous approaches based on set coding and show how geometry help in the design of new heuristics. We motivate our talk by giving three applications where it is desirable to get good approximation of these point sets.

Assembly tasks (*Discriminating*). In assembly tasks, one has to recognize parts arriving from a conveyor belt at some robot location so that the robot may grasp the object and handle it accordingly. Generally, the parts has already reduced the number of poses to a (small) finite set. We are looking for a minimum number of sensing points that allow to discriminate (i.e., recognize) every object. For example, we may embed point light detectors in the feeder tray and therefore seek for a minimum number of detectors and their appropriate locations.

Covering points (*Piercing*). Given a set of n points and a star-shaped polyhedra, we are looking for a minimum number of translated congruent copies of that pattern in order to cover the point set (each point belongs to at least one of the translated copy). This problem is dual to a piercing problem, where we are given n congruent copies of a dual pattern and try to find its transversal number and corresponding transversal point set.

Map labelling (*Packing*). Here, we want to automatically place labels on a map. A basic desirable property is that those labels do not overlap each other. The goal is to find a largest feasible placement. That is, to find an independent geometric set.

It has been shown that the corresponding abstract set problems (NP-complete) are hard to approximate unless $NP \subseteq DTIME[n^{\log \log n}]$. For example, the hitting set problem (or dually the set cover problem) and the discriminating set problem are $(1-\epsilon)\log n$-hard to approximate $(1 > \epsilon > 0)$, and the independent set problem is even worse with a $\Omega(n^{\frac{1}{4}})$ inapproximability bound. However, taking into account the geometric nature of objects, we are surprisingly able to get better precision-sensitive heuristics that avoid to compute the time costly incidence graphs.