Environmental effects of traffic

Peter Sonneveld

TU Delft, EWI (DIAM), Mekelweg 4, 2628 CD Delft, p.sonneveld@ewi.tudelft.nl

1. Problem formulation

Roads, highways etc cause noise, air pollution and other unwished contamination of the environment.
For a road system these effects can be modelled in terms of regions (buffer zones), defined for parts of the road called 'sections', in which the negative effects are considered to be unsustainable high for combination with human inhabitance (for example).
In the design process of (parts of) a road, these effects are dealt with by determining its environmental load to living areas. This quantity is identified with the surface area of the intersection of the buffer zones with the actual domain where people live. For the latter, postal code zones can be used.
The problem handled by the DEMIS group is

> Development of an algorithm, for determining the environmental load of each section of the 'road in design' in the design phase.

The following restrictions and simplifications are to be used in the algorithm:

(1) The algorithm must be implementable as interactive tool, in order to adapt the road design for decreasing the environmental load.
(2) Buffer zones are defined as polygons, and so are the postal code zones.
(3) Overlapping parts of road segments must be counted properly. In the present modelling this means 'counted only once'
(4) Therefore one has to deal with handling the overlap between the buffer zones of different sections. A (possible) complication: *Overlapping buffer zones do not necessarily belong to adjacent road segments.*

2. Participants of the work group

The following people have been working on this problem

- Poul Grashoff (*DEMIS, Initiator*)

– Ewa Matusiak (*Un of Vienna*)
– Etelvina Perez (*TU Delft*)
– Fahmi Naifar (*TU Delft*)
– Bob Planqué (*CWI*)
– Valeriu Savcenco (*CWI*)
– Peter Sonneveld (*TU Delft*)

3. Two approaches

After a first meeting of the complete group, two different approaches appeared to be promising.

1. Road segment centered approach. (*Bob, Valeriu & Fahmi*) This approach is based on the idea originated at DEMIS:
Make a description of the complete road in terms of non-overlapping polygons. The, from mathematical point of view rather simple, operations for doing this can be done using polygon manipulation tools that can be found in Matlab. (Allbeit only after some non-trivial search procedure)
At first, there was some disagreement on which parts of the intersections of more road-sections, should "belong to" which road section. In the case of a complicated road crossing situation, a cumulated overlap of many buffer zone polygons may require addition of environmental loads, at least for the air pollution component.
However, taking into account the simplicity of the environmental model, the group working on this idea decided to assign the mutual overlap to the first polygon in the system it is part from, assuming that the polygons are ordered some way. Proceeding like this, a purely additive construction can be chosen, which is of importance for the adaptivity of the algorithm.
The obvious question about the cumulative effects of more road sections on one postal code zone, could not be implemented, since also these effects are beyond the scope of the model.

2. Postal code zone centered approach. (*Eva & Etelvina*) In this alternative approach, the basis is the set of postal code zones. The positioning of the complete road is determined by the shortest path of which the buffer zone doesn't intersect any postal code zone. If the road must satisfy other types of restriction (which is always the case) then these can be met by penalty functions (for instance virtual postal code areas).
One advantage of this approach can be the possibility of actual minimization of the costs. One possible drawback can be that the resulting optimal road is far away from optimal in an other respect, like for instance an excessive length. This could be solved by modifying the concept "length."

The required software for this approach can be based on software that is standard in the world of optimization. The computational complexity can be reduced by using initially simple polygonal hulls for postal code zones with complicated shape. Fine tuning can be done by a mix of this approach and the road segment centered approach.
At the end of the *MFI*-week, this approach was only conceptually worked out.

4. First approach

Overview of the activities. After having chosen approach 1, the group decided quickly to use Matlab as programming platform. Some of the reasons:

(1) Every one was familiar with this environment.
(2) It is present in this university and on the laptop of one of the participants.
(3) For large multidimensional problems, it is an nearly optimal environment. So if someone wants to develop the required software from scratch in C or Fortran, he probably can't beat a well designed Matlab code, and certainly not in a finite time.
(4) If the software is to be used in a non-Matlab enviromnment, a stand-alone version of the software can be made with help of Matlab. Although none in our group has experience in that field, this seems to be not difficult.

Software for calculating intersections or unions of polygonal areas is basically simple. But practically not quite so simple. Luckily there exist free polygon clipping software for Matlab. In using this, the group met a first problem: The software was not fully grown up.
Apparently convex regions were expected, and some intersections couldn't be expressed at all, because they were non-connected, or multi-connected. As a matter of fact, the activities of the 'approach 2 - group' were considerably frustrated by these circumstances.
After some search on the web and in the personal network of one of the group members, a version of the software was found that seemed to be stable and reliable.
The mathematical set up for approach 1 can be formulated simply as follows. Denote the polygonal representation of the bufferzones by $B_1, \ldots, B_N$. Now define the sequences of domains $\{R_1, \ldots, R_N\}$ and $\{C_1, \ldots C_N\}$ by

$$R_1 = D_1, \quad C_k = D_k \setminus R_{k-1}, \quad R_k = R_{k-1} \cup C_k,\ k = 2, 3, \ldots, N$$

The sequence $\{C_k\}$ consists of disjoint regions, and can be used for calculating the environmental load.

At the presentation, the group could show some elementary results on the set up of the program, and some actual intersection examples based on actual data of the road system around Rotterdam.

Epilog. At first sight the zoning problem seems a rather simple straightforward problem. The difficulties are more in the modelling of the real problem, than in the actual mathematics of the chosen model. In fact the participants of the 'approach 1-group' had a strong feeling that it was a mere programming job. And if there hadn't been free polygon-clipping-software, I would have started right away writing it myself.
Now there *was* that type of software, so thinking stopped, and the work reduced to pushing buttons until the vehicle would be under control.
Some time after the workshop, I still hadn't received a fault-free version of the clippol package, nor I succeeded in downloading it myself. So I started to write my own. And discovered that this kind of manipulation isn't trivial at all! Using rectangle-like regions, building the required union leads to terrible kinds of non-connected and multi-connected regions. Not difficult of course, but it requires a suitable datastructure to handle that, and, more important, some thinking.
Apparently the other members of the group experienced similar things, and so the project felt asleep.
The question now is: did we achieve nothing at all? I think that 's not the case

Bad news: My own later experiments showed that a proper additive procedure is not possible. If a complete assemblage of bufferzones has been carried out, and bufferzone D_k is modified, in order to decrease the environmental load, the overlapping parts with bufferzones D_j, $j > k$ become active. But these themselves may consist partly of other overlapping zones. Keeping an administration of the overlaps is really a nasty job.
This we could have seen the first afternoon, if we had thought a little longer.

Good news: The complete calculation of the environmental load for a trial design of a road system can probably be done in a reasonably short time, as far as I may consider the scale of my experiments as representative for the real problem. In that case a completely new calculation could be used as a 'response' to any modification in an interactive process. This is of course a brute force solution, but what's wrong with that?

Final conclusion. In my opinion, this interesting problem deserved some more analysis time than we gave to it. This is due to the combination of the short period we had to show some results, the quick recognition of the fact it was a programming job, and the fact that we under-estimated the complications of precisely *that* part of the problem. If we had decided to start with making a mathematical analysis, as should be done *always*, there wouldn't have been a working program on a lab-scale, showing a piece of the requested work. But a reliable outline on how to handle the problem, and why it should work, could have been finished.

Finally, I think that usually an element of competition is present in work shops like this, and competition is not always improving the results of scientific activities, at least not in a contest of one week.