Proposal Outlines

1. **Project background**

What is urban planning?

Why urban planning is important?

Topological characterization and neighborhood optimization

Which metrics can be used to evaluate the optimization effect? What’s the motivation of their designs?

Generative models!

1. **Research motivation**

Instead of using the analytical methods as Christa et.al introduced [1, 2] for block topology optimization, the generative AI methods are desired for solving this problem in order to obtain diverse redesign possibilities which may be suitable for various scenarios.

**General idea**: Basically, we need to learn a mapping function that could transfer the input “bad” block into “good” block. The generative model is to approximate this mapping. For this purpose, we need a block geometry dataset with both “good” blocks and “bad” blocks. Then use the “bad” block as the input and the “good” block as the output to train the generative model. In principle, we want the well-trained generative model could tell us a bunch of reblocking design candidates by inputting the original “bad” block, which have various performance improvements and implementation costs for flexible choice.

1. **Algorithms**
   1. Metrics for block “goodness”or “badness”

It’s important to to define which topological structure the block has can be treated as good; otherwise, bad.

Block complexity: k\_complex

Average Distance in the blocks

Block area, country id

Travel cost, road construction, facilities

* 1. Collection of good / bad block data
     1. Using simulations

Considering the analytical method for block optimization, the single block can be optimized to derive multiple “good” block structures. We can treat this method as a simulation tool to obtain “good” data by input “bad” data. That means we may only need to collect the bad block data (such data collection and annotation should be really complicated and time-consuming). With this tool, we can get a dataset with paris of real “bad” data and simulated “good” data, where the overall structure between the good and bad blocks should be closely similar.

**Pros:** the good and bad cases have point-to-point mapping; the generative model directly learns an optimization plan for a particular block. Large amounts of data can be available.

**Cons:** the simulated “good” block may not be true “good”.

* + 1. Using real data

Generally, we suppose there are two types of real “good”/”bad” block data. 1) the block which was historically “bad” but is currently “good”; e.g. Tokyo (1950 vs 2010); 2) the good blocks and bad blocks in different regions at the same time. E.g. Cape Town (2010) vs New York (2010). For the former one, it can use the previous simulation method, except for the truth that only very limited number of “good” structures are available. For the later case, the “good” block is not directly optimized from the bad one, which means the overall structures before and after optimization may be seriously different.

**Pros:** the optimization can be practical and meaningful.

**Cons**: the data size can be limited. The good/bad data may not be point-to-point mapping. That means the using traditional generative model could only approximate the overall good block distribution, rather than the particular one. So the optimized block may change a lot in the structure.

* 1. Methods

cVAE

GAN??

cycleGAN for disentangled block pairs

* 1. Data preparation

The block data can be represented as graph data. So the generative process is a graph generation process

* 1. Performance metrics

Block complexity reduction

Structure change constraints

…