Dependencies for the Meeting Point Protocol

1 The code

The implementation consists of 4 scripts, i.e., three scripts for the theree parties Alice, Bob, and Charlie, and an MPC module. The communication is assumed to happen over secure channel, the scripts assume that the communication over sockets is secured, the encryption methods for this are out of the scope of the present implementation, the choice of the encryption primitives are left to the implementer. Moreover, a secure group chat is assumed to be formed before the beginning of the protocol, that is also out of the implementation scope, standard methods are recommended, e.g, Signal.

1.1 The Parties' Implementation

For the implementation, we use the following standard Python libraries.

- 'socket': The 'socket' library in Python provides access to the socket interface, allowing communication over networks using sockets.
- 'pickle': Python's 'pickle' module is used for serializing and deserializing Python objects, enabling the conversion of complex data structures into a byte stream.
- 'base64': The 'base64' module provides functions to encode and decode data in Base64 format, commonly used for encoding binary data into ASCII characters.
- 'folium': 'folium' is a Python library used for visualizing geospatial data interactively, particularly for creating maps.
- 'requests': The 'requests' library is widely used for making HTTP requests in Python, simplifying the process of sending HTTP requests and handling responses. Used in the code for calling the LBS.
- 'random': The 'random' module provides functions for generating random numbers, sequences, and making choices randomly in Python.
- 'numpy': 'numpy' is a fundamental library for scientific computing in Python, offering support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions.
- 'pandas': The 'pandas' library is essential for data manipulation and analysis in Python, providing data structures like DataFrames for handling structured data efficiently. Used mainly to make and sort the list of the candidate locations.
- 'json': The 'json' module allows encoding and decoding JSON data in Python, facilitating the interchange of data between different systems.
- 'math': The 'math' module provides mathematical functions and constants in Python for performing various mathematical operations.
- 'time': The 'time' module in Python provides functions for working with time, including time measurement, conversions, and manipulation. Used mainly to measure the implementation running time.

  These libraries are open source standard Python libraries.

1.2 The MPC Module

Our implementation is compatible with any additive MPC protocol, for our demo, we use the usual additive secret sharing scheme, which only requires generating random numbers at the parties (no encryption). The module relies only on a random number generator. For this purpuse we use.

- "random": The random module in Python provides functions for generating random.

Please note that the security of such a module is based on the security of the random generator, thus, this module should be chosen carefully when adapting our implementation to other envirements.