Chapter 7

Conclusions and Future Work

This work describes a stream data grid where open-world queries arrive from users and applications external to the system. The long running, continuous nature of the queries require that the queries are optimized to reduce network usage and ensure even load distribution on the grid nodes. Query plans were classified as either composable and non-composable query plans based on the number of data sources being used to compute the query plan. The computational complexity for determining the optimal query plan leading to minimal network usage for composable and non-composable query plans were evaluated. Even though polynomial time non-composable query plans exist project and selection operators, the frequent arrival and revocation of queries would mean that such query plans need to be computed after every query arrival and revocation thereby rendering such globally optimal query plan computations infeasible.

To optimize in an environment where queries arrive and get revoked frequently,
the notion of emergent optimization was proposed where each grid node autonomously takes decisions based on some local self-interest objective and the global query plan emerges as a consequence of these local decisions. The design of such a system is described in Chapter 4 and various source selection strategies which determine the source nodes a grid node selects to answer a query are discussed. The analytical modeling of strategies show that some strategies are better suited to reducing network usage while others are better at distributing load evenly. The experiments performed in Chapter 5 show that emergent query optimization can not only optimize network usage and distribute load evenly but a grid node can also automatically change its source selection strategy based on the query loads to ensure the global optimization objective is met.

While the work presented in this thesis shows that emergent optimization can be used to address the challenges in open-world query processing, there are some modifications which can be made to the system to address specific issues and is part of the planned future work. The next section describes the current challenges in the system which are yet to be addressed.

7.1 Future Work

As mentioned in Chapter 1, the physical sensor at which the required data may be available changes over time because queries may request streams based on their contents without explicitly specifying the sensor(s) from which the data need to be fetched. This adds a second level of openness to the system and is not addressed
in this work. The two key challenges in generating query results in such a system are managing the continuous mapping of data to a physical sensor and ensuring continuity (zero stoppage) of query results.

In the current emergent optimization technique, the strategy selection is based on payoffs for a particular strategy and if the number of grid nodes adopting a particular strategy drops to zero, the grid will never consider that strategy. This might lead to the grid being unable to adjust the varying conditions of the grid and query characteristics. To mitigate this issue, grid nodes perform a strategy reset after some specified time random interval and select strategies at random to evaluate the possibility of an alternate strategy being better than the one in use currently. However as seen from the experimental results, the strategy resets occur only at specific pre-defined intervals resulting in two problems, 1) the grid is unable to switch to a strategy which is not being currently used except after resets which may lead to sub-optimality if the grid conditions change long before a strategy reset and 2) since a reset results in all strategies being selected uniformly, the resulting query plans are sub-optimal for next few generations before the grid converges to the correct strategy. If such resets can be avoided and the grid conditions monitored continuously to select the correct source selection strategy, the overall optimization results can be bettered.

The fitness of any given strategy is evaluated based on the payoffs a particular strategy receives during a generation over the entire grid. Since the payoff is summed over the entire grid, the fitness function can only select a strategy for the entire grid. While this is not a problem if varying query patterns affect the entire
grid uniformly, this approach of evaluating strategies globally will not be suitable for handling localized query loads. To ensure that the grid is able to address locality of query loads, strategy evaluation needs to be addressed locally as well.

The source selection strategy as mentioned earlier in Chapter 4 is a critical element of emergent optimization. In this work only simple strategies like distance ordering, random ordering and load ordering were evaluated. The greater the amount of information available, the source selection strategy can be more sophisticated resulting in greater levels optimization and leads to a strategy hierarchy as shown in Figure 4.2. It is expected that the usage of sophisticated strategies would lead to lesser network usage and even load distribution. However, increased information would also mean increased communication costs and hence there would be an inflexion point after which the cost of communication would outweigh any performance improvement considerations.

On a more philosophical note, open world problems are challenging given that there is a lack of complete information about all aspects of the system. In the system described in this work, queries arrive and are revoked frequently thereby making it infeasible to optimize for optimality and is a representation of a much bigger real life problem of performing tasks in open-world systems. Most real life systems do not have complete understanding of the environment (which changes dynamically) and yet has to perform tasks in it.

The solution steps for completing a task itself in such a system is shown in Figure 7.1 and can be either,

1. completely and explicitly defined as part of the system
2. the knowledge about the solution embedded in the system without being explicitly specified

There are two possible approaches to ensure that a completely defined task (1) is carried out in an open-world environment as expected, (a) by enforcing collaboration using normative constraints and rule based frameworks as described earlier in hidden adversary systems and (b) by using a multi-agent approach where individual nodes collaborate with each other as presented in this work.

To determine the solution steps from the innate knowledge in the system as described in (2) is a very difficult challenge and is one of the key differentiating aspects between man and machines. Intelligence is the ability generate solutions for a task from the innate knowledge about the system in a given environment. Several disciples from medical (mapping human brain activity) to computer science (artificial intelligence) have tried to understand, reason and implement human like problem solving skills to machines without fantastical results. Once a given problem can be dissected into more manageable and understandable tasks, either a normative framework or a multi-agent collaborative approach needs to be
taken to solve the task.