Quick thought on Distributed vs Central solvers

Tom Kealy

November 4, 2014

1 Background

2 Sensing Model

We are considering a radio environment with a single primary user (PU) and a network of J =50 nodes collaboratively trying to sense and reconstruct the PU signal, either in a fully distributed manner (by local message passing), or by transmitting measurements to a fusion centre which then solves the linear system.

2.1 Setup

We are trying to sense and reconstruct a wideband signal, divided into L channels. We have a (connected) network of M ($= 50$) nodes placed uniformly at random within the square $[0, 1] \times [0, 1]$. They individually take measurements in the following way: by mixing the incoming analogue signal $x(t)$ with a mixing function $p_i(t)$ aliasing the spectrum. $x(t)$ is assumed to be bandlimited and composed of up to k uncorrelated transmissions over the L possible narrowband channels - i.e. the signal is k-sparse.

We can represent the sensing problem as a linear system by concatenating the measurements of each node into a single system i.e. $b = Ax + sigma$. Where, each row of the matrix represents the measurements of a single node. I.e we are trying to solve the following problem:

$$\min ||x||_1 \text{ subject to } ||Ax - b||_2^2 \leq sigma \tag{2.1.1}$$

where x represents the signal, A is the sensing matrix and b the set of (noisy) measurements we take.

The problem is coupled at each node by the variable x. To ease this issue, at each node create a local copy, denoted x_m (so there will be M copies of this variable).

Then, for consistency, we constrain the problem, so that each copy is identical: $x_1 = x_2 = \dots x_M$. Equivalently, given the graphical structure of the problem, we require that $x_i = x_j$ for each edge of the network.

So now we are solving the problem

$$\min \frac{1}{J} \sum_J ||x_j||_1 \text{ subject to } A_p x = b_p \text{ and } x_i = x_j \{i, j\} \in E \tag{2.1.2}$$

The algorithm we use follows from writing the augmented Lagrangian and using the alternating direction method of multipliers (ADMM). Each node solves the following problem locally, iteratively where t denotes the iteration index:

$$x^m(t) = \min\left\{ ||y - Ax - \sigma||_2^2 + \frac{\lambda}{M}||x||_1 + \sum_{m' \in N_m} \mu_{mm'}\left(x^m - x^{m'}\right) + \sum_{m' \in N_m} \frac{c}{2}||x^m - x^{m'}||_2^2 \right\} \tag{2.1.3}$$

and updates its own multipliers using a stochastic gradient iteration:

$$\mu_{mm'}(t) = \mu_{mm'}(t-1) + c\left(x^m(t) - x^{m'}(t)\right) \tag{2.1.4}$$

The $x + m$ and $\mu_{mm'}$ are then communicated with their one hop neighbours.
The terms $\left(x^m(t) - x^{m'}(t)\right)$ and $\sum_{m' \in N_m} \frac{c}{2}||x^m - x^{m'}||_2^2$ can be thought of as the distributed algorithm being sensitive to differences between the estimation at each node whilst seeking an energy minimising solution in each clique.