Cryptographic Protocols
Spring 2018
Part 8

Ideal Broadcast

Definition: Broadcast
Definition (Input x1, Output y1, . . . , yn)

• Consistency: Every (correct) player receives the same output y.
• Validity: Sender correct ⇒ every player receives output y_i = x_1.
• Termination: Every player eventually receives output.

Sender correct:
Always:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
</tbody>
</table>

Definition: Consensus
Definition (Inputs x_1, . . . , x_n, Outputs y_1, . . . , y_n)

• Consistency: Every (correct) player receives the same output y.
• Persistency: All correct players have input x ⇒ y_i = x.
• Termination: Every player eventually receives output.

Pre-agreement: Always:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
</tbody>
</table>

Known Results (Broadcast/Consensus)

<table>
<thead>
<tr>
<th>Setting</th>
<th>Condition</th>
<th>Literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o setup</td>
<td>information-theoretic t &lt; n/3</td>
<td>[PSL80, BGP89]</td>
</tr>
<tr>
<td></td>
<td>cryptographic BC: t &lt; n</td>
<td>[DS82]</td>
</tr>
<tr>
<td></td>
<td>i.t. Cons: t &lt; n/2</td>
<td>[PW92]</td>
</tr>
</tbody>
</table>

Standard Model

Players
• Player set \( P \) = \{ P_1, \ldots, P_n \}

Network
• Complete
• Synchronous
• Authenticated

Adversary
• Threshold \( t < n/3 \)
• Active (Byzantine)
• Unlimited (unconditional security)
Broadcast vs Consensus

Broadcast: \((x, \perp, \ldots, \perp) \rightarrow (y_1, \ldots, y_n)\)

Consensus: \((x_1, \ldots, x_n) \rightarrow (y_1, \ldots, y_n)\)

Broadcast from Consensus

1. \(P_i\): send \(x_i\) to every \(P_j\).
2. \((y_1, \ldots, y_n) = \text{Consensus}(x_1, \ldots, x_n)\)
3. \(\forall P_j\): return \(y_j\)

Consensus from Broadcast

1. \(\forall P_i\): Broadcast \((x_i)\)
2. \(\forall P_j\): return \(y_j = \text{majority of received } x_i's\)

Definition: Weak Consensus

Definition (Inputs \(x_1, \ldots, x_n\), Outputs \(y_1, \ldots, y_n\))
- **Weak Consistency:** \(\exists y \in \{0, 1\}\) such that \(\forall P_i: y_i \in \{0, 1, \perp\}\).
- **Persistency:** All correct players have input \(x \Rightarrow y_i = x\).
- **Termination:** Every player eventually receives output.

<table>
<thead>
<tr>
<th>Pre-agreement</th>
<th>Always</th>
</tr>
</thead>
<tbody>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(\perp)</td>
<td>(\perp)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(\perp)</td>
<td>(\perp)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(\perp)</td>
<td>(\perp)</td>
</tr>
</tbody>
</table>

Protocol Weak Consensus

\(\text{WeakConsensus}(x_1, \ldots, x_n) \rightarrow (y_1, \ldots, y_n)\)

1. \(\forall P_i\): send \(x_i\) to every \(P_j\)
2. \(\forall P_j\): \(y_j = \begin{cases} 
0 & \text{if } \#\text{Zeros} \geq n - t \\
1 & \text{if } \#\text{Ones} \geq n - t \\
\perp & \text{else}
\end{cases}\)
3. \(\forall P_j\): return \(y_j\)

Definition: Graded Consensus

Definition (Inputs \(x_1, \ldots, x_n\), Outputs \((y_1, g_1), \ldots, (y_n, g_n)\))
- **Graded Consistency:** Correct \(P_i\) has \(g_i = 1 \Rightarrow \forall \text{corr. } P_j: y_j = y_i\).
- **Graded Persistency:** All corr. players have input \(x \Rightarrow (y_i, g_i) = (x, 1)\).
- **Termination:** Every player eventually receives output.

<table>
<thead>
<tr>
<th>Pre-agreement:</th>
<th>Always:</th>
</tr>
</thead>
<tbody>
<tr>
<td>(b)</td>
<td>(b, 1)</td>
</tr>
<tr>
<td>(\perp)</td>
<td>(\perp)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b, 1)</td>
</tr>
<tr>
<td>(\perp)</td>
<td>(\perp)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b, 1)</td>
</tr>
<tr>
<td>(\perp)</td>
<td>(\perp)</td>
</tr>
</tbody>
</table>

Protocol Graded Consensus

\(\text{GradedConsensus}(x_1, \ldots, x_n) \rightarrow ((y_1, g_1), \ldots, (y_n, g_n))\)

1. \((z_1, \ldots, z_n) = \text{WeakConsensus}(x_1, \ldots, x_n)\)
2. \(\forall P_i\): send \(z_i\) to every \(P_j\).
3. \(\forall P_j\): \(y_j = \begin{cases} 
0 & \text{if } \#\text{Zeros} \geq \#\text{Ones} \\
1 & \text{if } \#\text{Zeros} < \#\text{Ones} \\
\perp & \text{else}
\end{cases}\)
4. \(\forall P_j\): return \((y_j, g_j)\)
Definition: King Consensus

Definition (Inputs $x_1, \ldots, x_n$, Outputs $y_1, \ldots, y_n$)

- **King Consistency**: King is correct $\implies \exists y : \forall$ correct $p_i : y_i = y$.
- **Persistency**: All correct players have input $x \implies y_i = x$.
- **Termination**: Every player eventually receives output.

### Protocols King Consensus (King $P_k$) and Consensus

**KingConsensus$_k(x_1, \ldots, x_n) \rightarrow (y_1, \ldots, y_n)$**

1. $((z_1, g_1), \ldots, (z_n, g_n)) = \text{GradedConsensus}(x_1, \ldots, x_n)$
2. $P_k$: send $z_k$ to every $P_j$.
3. $\forall P_j$: $y_j = \begin{cases} z_j & \text{if } g_j = 1 \\ z_k & \text{else} \end{cases}$
4. $\forall P_j$: return $y_j$

**Consensus$(x_1, \ldots, x_n) \rightarrow (y_1, \ldots, y_n)$**

1. for $k = 1$ to $t + 1$:
   
   $$(x_1, \ldots, x_n) = \text{KingConsensus}_k(x_1, \ldots, x_n)$$
2. $\forall P_j$: return $x_j$