Implementation of a Security-Dependability Adaptive Voting Scheme

Ryan Quint - quintr@vt.edu
Noah Badayos - nbadayos@vt.edu
David Mazur - dmazur@vt.edu

Feb 29, 2012
Data Mining Theory to Application

Adaptive Relaying: The ability to modify, update, or change the settings of a protection scheme.
Application of Voting Scheme

- What is system state?
  - Safe
  - Stressed
- “Stressed”? 
  - Increase Security
  - Relays Vote
- “Safe”? 
  - Increase Dependability
  - Normal tripping of CB
Implementations Explored

Phasor Measurement Network

SVP Configurator
IEC-61131 Languages

SEL-3378

C37.118

Computer

openPDC

Grid Protection Alliance

SQL Server 2008

Open Source
C# Programming

Programmable Automation Controller

Hard-wired

R1
R2
R3

Slave 1
Slave 2
Master

SEL Fast Operate

Hard-wired or SEL Mirrored Bits

Breaker

52TC
Connection of openPDC and SQL

openPDC Custom Action Adapter

CPU

Frame

PublishFrame

Decision Tree 2

SQL

30x per sec

vote

"openPDC"

Vote

0
Connection of SQL and PAC

PAC

ODBC
3000x per sec

CPU

SQL
“openPDC”

<table>
<thead>
<tr>
<th>Vote</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
</tbody>
</table>
Automation Controller Voting

- Asynchronous
- Ladder Logic programming
- Software updates to relay logic, i.e. voting
- Centralized stamping of data
  - Relay trips
  - Output

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breaker_Full.Utc</td>
<td>ET#1969-12-31-19:00:00.000000 (GMT-05:00)</td>
</tr>
<tr>
<td>R1_Full.Utc</td>
<td>ET#1969-12-31-19:00:00.000000 (GMT-05:00)</td>
</tr>
<tr>
<td>R2_Full.Utc</td>
<td>ET#1969-12-31-19:00:00.000000 (GMT-05:00)</td>
</tr>
<tr>
<td>R3_Full.Utc</td>
<td>ET#1969-12-31-19:00:00.000000 (GMT-05:00)</td>
</tr>
</tbody>
</table>
PAC Performance

Histogram

<table>
<thead>
<tr>
<th>PAC Delay Non-Voting [ms]</th>
<th>PAC Delay Voting [ms]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>1.187324</td>
</tr>
<tr>
<td>Median</td>
<td>1.101</td>
</tr>
<tr>
<td>Mode</td>
<td>1</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.358331</td>
</tr>
<tr>
<td>Minimum</td>
<td>0.32</td>
</tr>
<tr>
<td>Maximum</td>
<td>2.199</td>
</tr>
<tr>
<td>Mean</td>
<td>1.9561</td>
</tr>
<tr>
<td>Median</td>
<td>2.01925</td>
</tr>
<tr>
<td>Mode</td>
<td>2.2</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.457734</td>
</tr>
<tr>
<td>Minimum</td>
<td>0.875</td>
</tr>
<tr>
<td>Maximum</td>
<td>3.1755</td>
</tr>
</tbody>
</table>
Conclusions

• Decision Trees implemented in PDC
  – Hardware: SEL-3378
  – Software: openPDC/SQL Server

• Voting Logic implemented
  – Master-Slave relay configuration
    • No added delay, relay config changed
  – Automation controller voting device
    • $\leq 3.18$ ms PAC delay, no relay changes
References


