



A Benchmarking Framework for Easy and Reliable Wireless Experimentations

Michael Mehari , Wei Liu , Stefan Bouckaert , Ingrid Moerman,
Stratos Keranidis , Thanasis Korakis, Iordanis Koutsopoulos,
Van Wesemael Peter , and Pollin Sofie



The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n°258301 (CREW project).

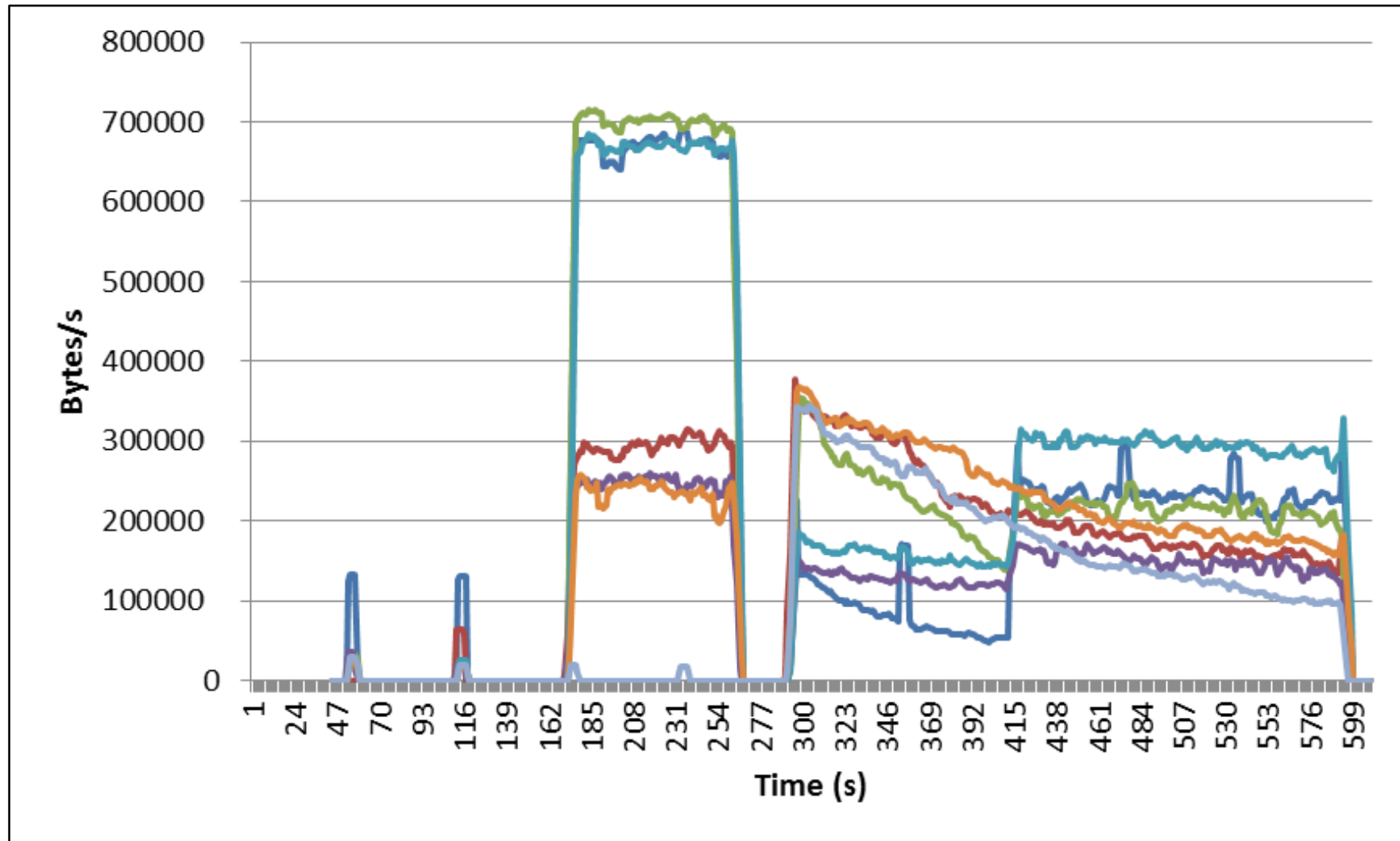


OUTLINE



- **The need for benchmarking framework**
- **Proposed solution**
- **Proof of concept**
- **Conclusion and future work**

Repeatability of experiments



SUT refers to a system which is continuously checked for performance improvement.

Ex. a network of wireless nodes sending temperature data based on a routing algorithm

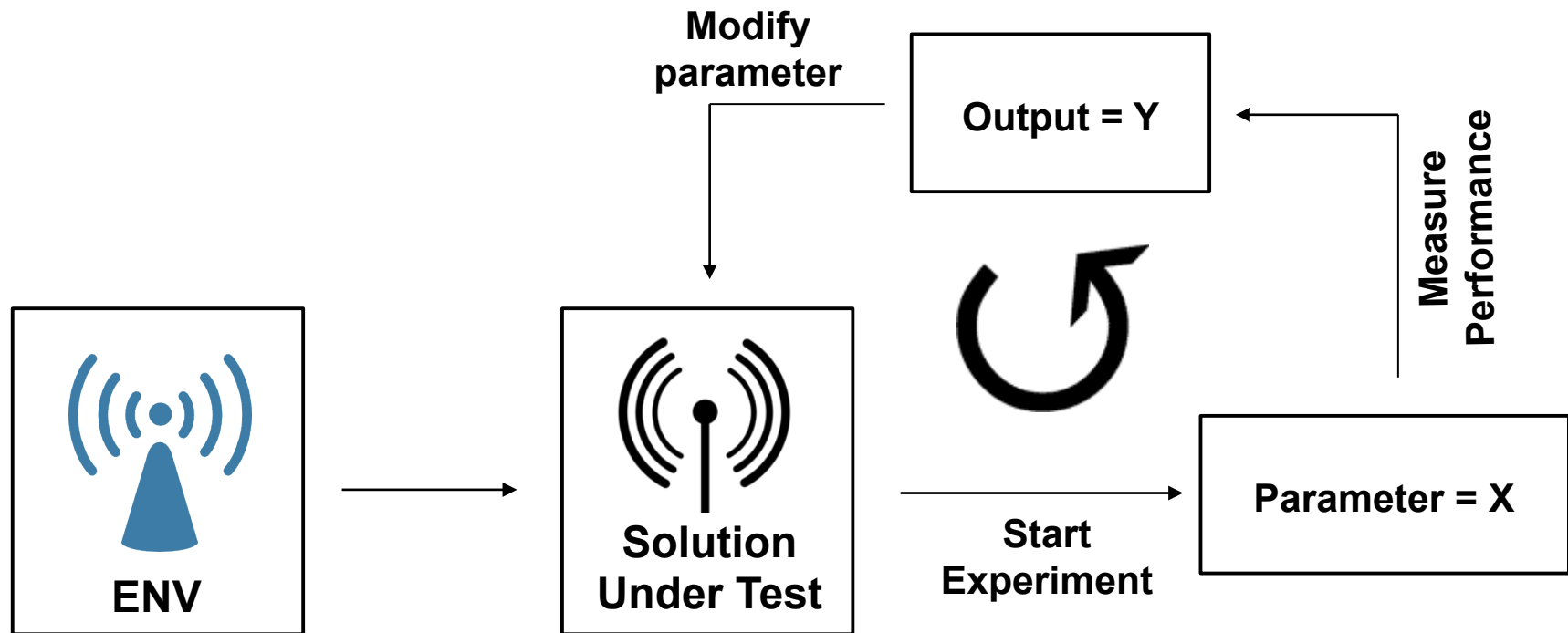


This is also a system which emulates a real world scenario.

Ex. interference pattern of a nearby WI-FI Hotspot.



Each time modifying parameters and restarting experiments is tedious and error prone.





Documentation and reuse of experiments



The wireless experiment is conducted in an environment using two nodes one acting as a sender and the other acting as a receiver. This way, performance is improved by a factor of 4.5 for this configuration.



The Question is, will a different experimenter make use of such a result to his/her work?

Experimenters usually focus on results and thus making it difficult to re-construct and benchmark their experiments with other experiments



Needs for Benchmarking Framework



- **Make repeatability of experiments possible**
- **Put aside time consuming and error prone experiment configuration of the environment**
- **Leave the burden from experimenters and help them focus on their Solution Under Test**
- **Making parameter space optimization experiments possible**
- **Reuse and built further from others experiment result**



OUTLINE



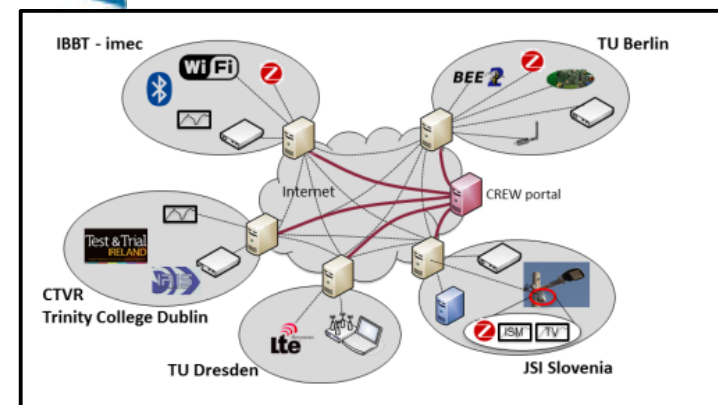
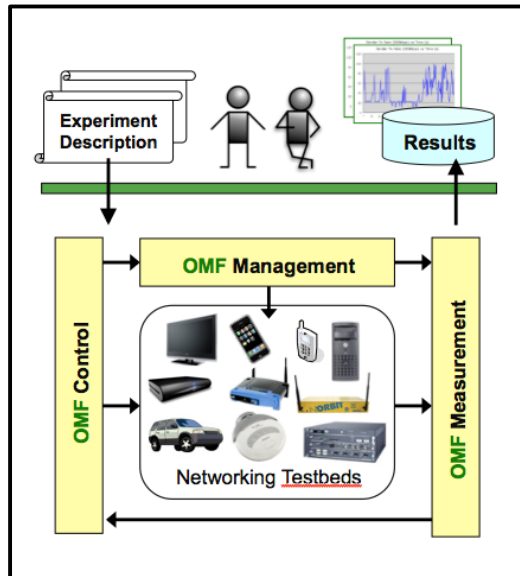
- **The need for benchmarking framework**
- **Proposed solution**
- **Proof of concept**
- **Conclusion and future work**



CREW Benchmarking Framework



What is CREW BM Framework?





CREW Benchmarking Framework



Front View

Experimentation Section

Start New Configuration
Load/Configure Experiment
<input type="button" value="Choose File"/> No file chosen <input type="button" value="Load"/>
► Load/Execute Experiment

Result Analysis

Solution One		
Config	<input type="button" value="Choose File"/>	No file chosen
Database	<input type="button" value="Choose File"/>	No file chosen
<input type="button" value="Load"/>		



Solution Two		
Config	<input type="button" value="Choose File"/>	No file chosen
Database	<input type="button" value="Choose File"/>	No file chosen
<input type="button" value="Load"/>		



Inside CREW Benchmarking Framework



Experiment configuration

Experiment Abstract

Title :

Author :

Contact Information :

Experiment summary :

Three pair of nodes create repeatable interference patterns. UDP streams of 30Mbps are used for the experiment.

Experiment duration (sec) :

Applications included in experiment

Platform

Application


Version

Description





➤ **Input Format**

➤ **Output Format**

Nodes included in experiment

Select nodes  **Zotac22** Image

Interface	Mode	Channel	ESSID	IP	Action
<input type="text" value="wlan0"/>	<input type="text" value="managed"/>	<input type="text" value="1"/>	<input type="text" value="INT1"/>	<input type="text" value="192.168.0.22"/>	<input type="button" value="X"/>

Application	Inst ID	Report	Parameter	TimeLine	Action
<input type="text" value="iwconfig"/>	<input type="text" value="ClientA_iwconfig"/>	<input type="checkbox"/>			<input type="button" value="X"/>
<input type="text" value="iperf"/>	<input type="text" value="ClientA_iperf"/>	<input type="checkbox"/>			<input type="button" value="X"/>



Inside CREW Benchmarking Framework



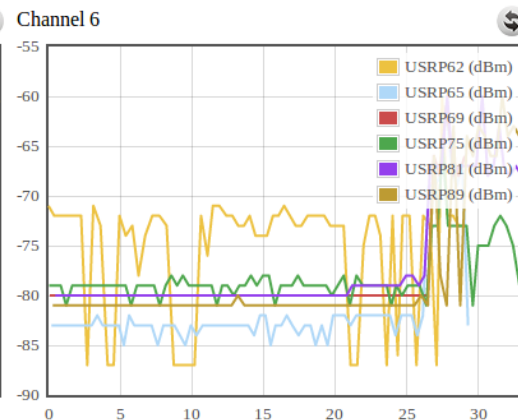
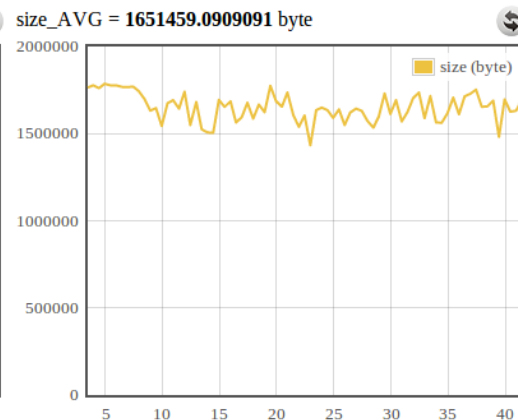
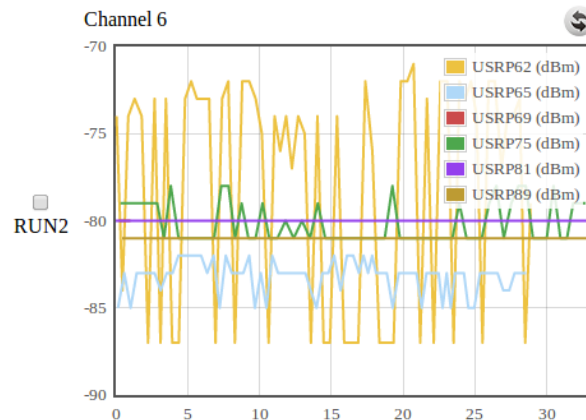
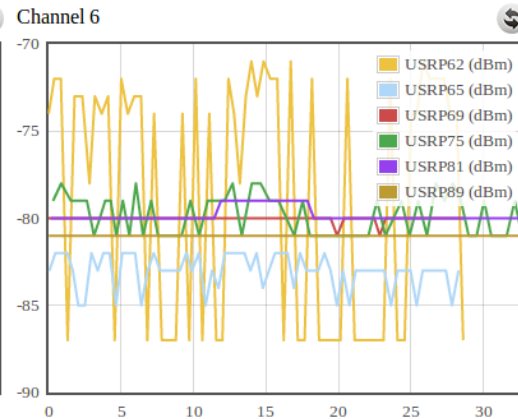
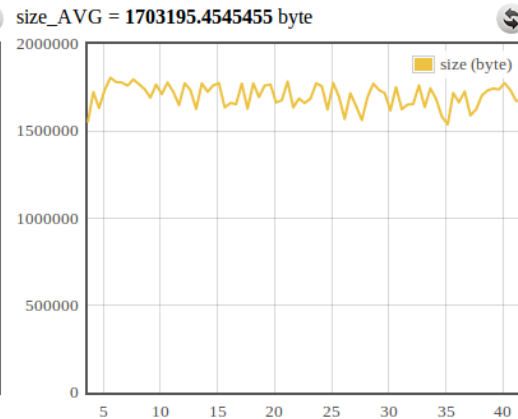
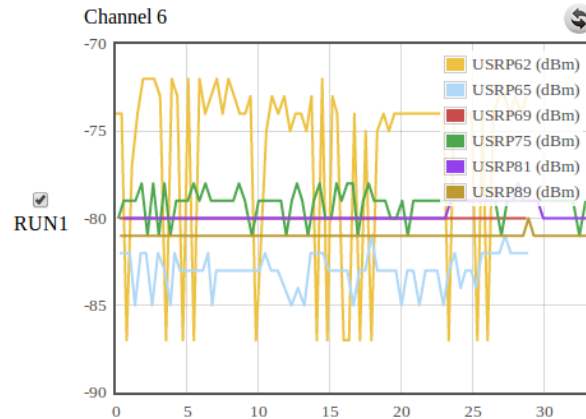
Experiment execution



Pre Experiment

During Experiment

Post Experiment



CORRELATION MATRIX

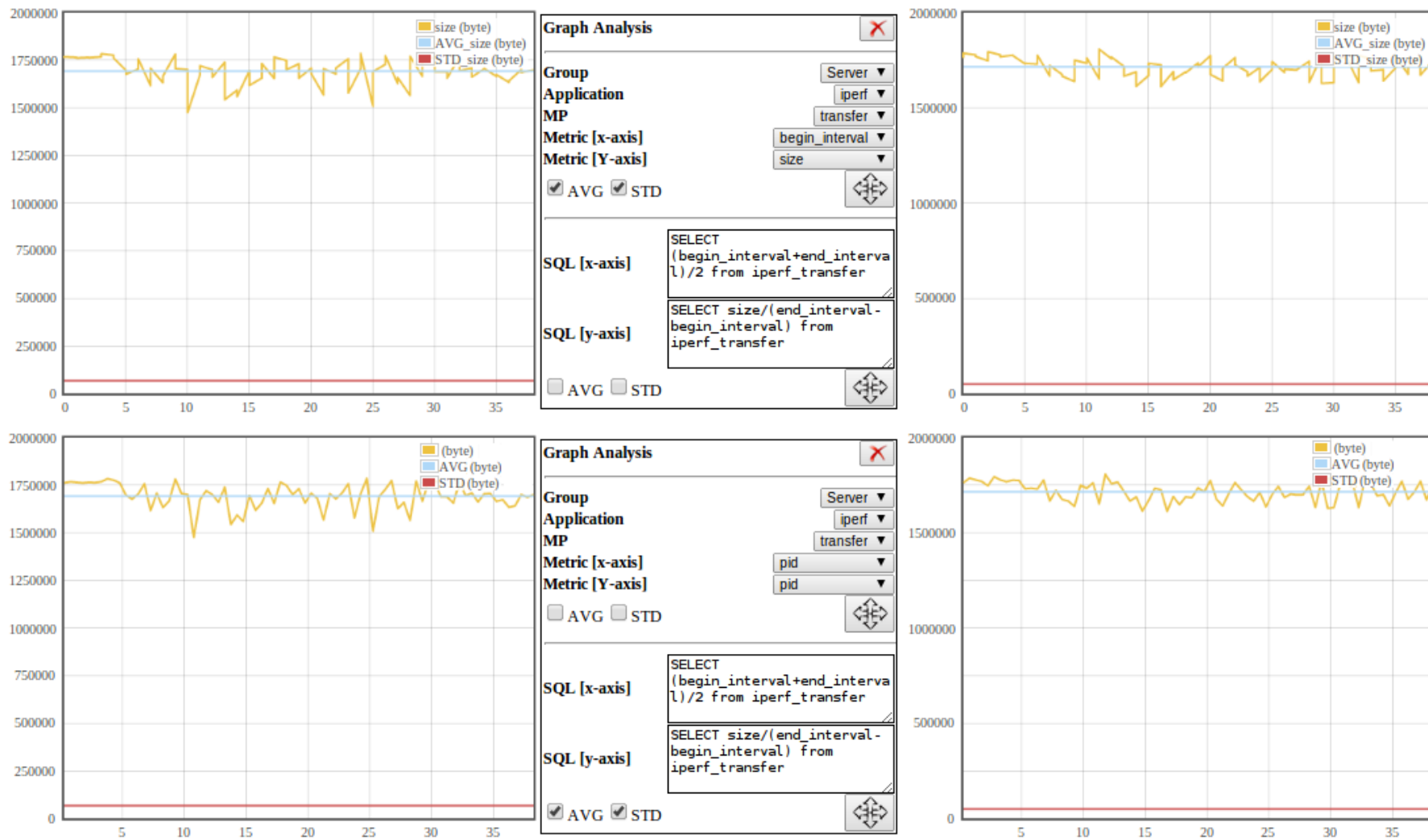
ID	R1	R2
R1	1	0.1688270
R2	0.1688270	1



Inside CREW Benchmarking Framework



Experiment benchmarking: result analysis





Inside CREW Benchmarking Framework



Experiment benchmarking: score calculation

Score Calculation

Metric Weight

```
SELECT AVG(size/(end_interval-begin_interval))/3570000 FROM iperf_transfer
```

Metric Weight

```
SELECT 10/AVG(PC) FROM iperf_transfer
```

Score = $TPH_VALUE * TPH_WEIGHT + PC_VALUE * PC_WEIGHT$

Score = $0.948449197860963 * 7 + 0.539070555807175 * 3 = 8.256356011$

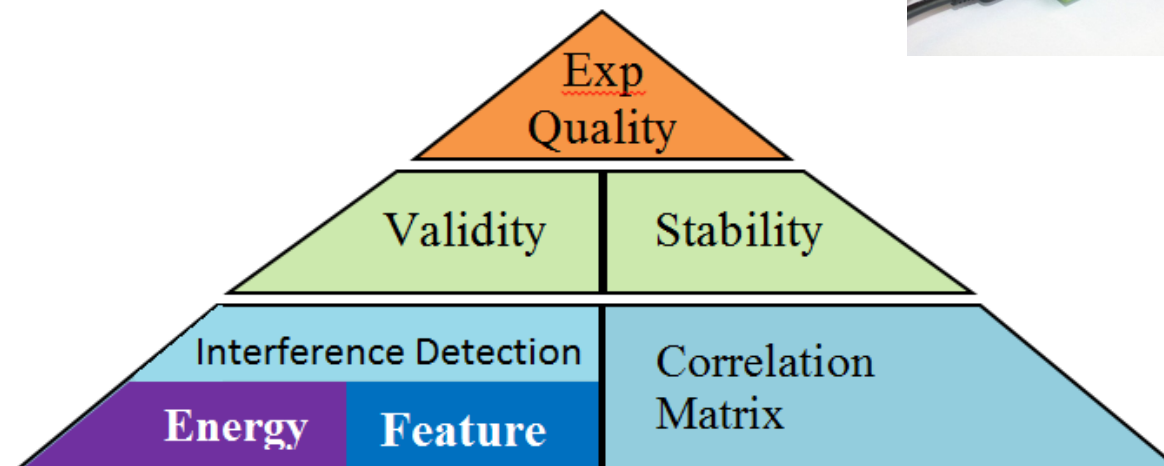
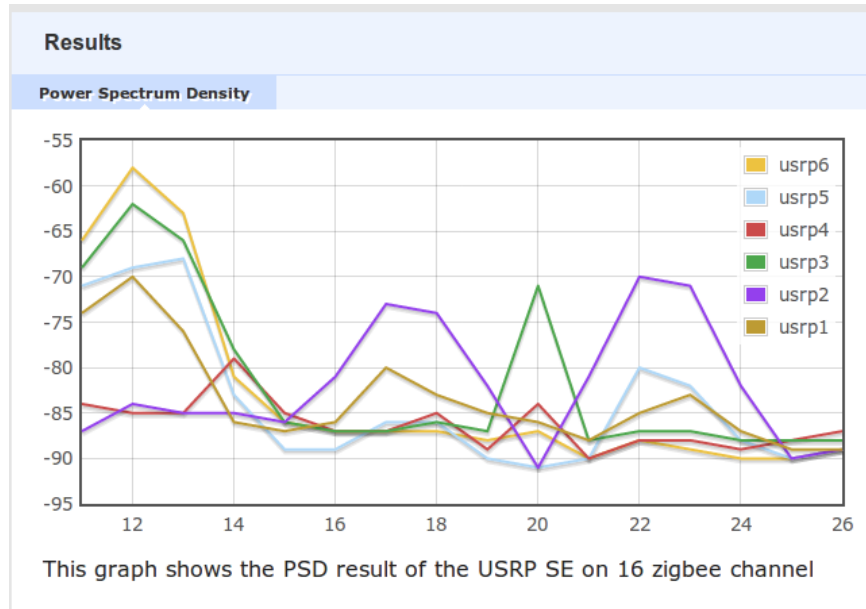
Throughput factor calculation

Power consumption factor calculation

Joint score calculation

Score out of 10 (7+3)

Experiment Validity and Stability

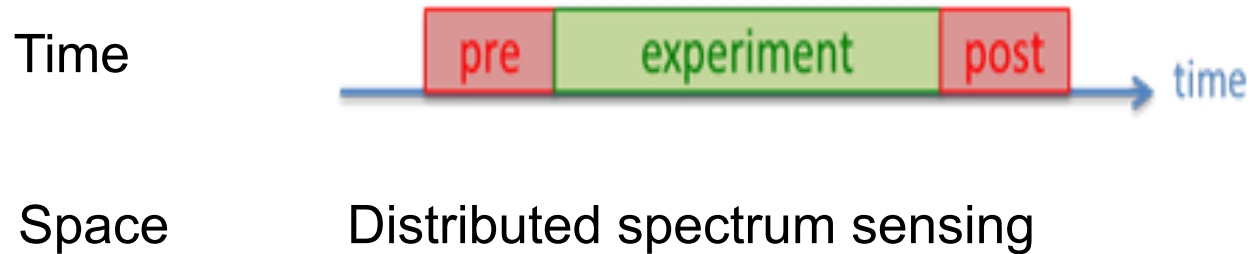




Inside CREW Benchmarking Framework



Experiment Validity :- Interference Detection





Experiment Stability

- Repeat experiment
- Cross correlation
- Outliers identification

Round	R1	R2	R3	R4
R1	1	0.2	0.78	0.97
R2	0.2	1	0.84	0.43
R3	0.78	0.84	1	0.93
R4	0.97	0.43	0.93	1



OUTLINE



- **The need for benchmarking framework**
- **Proposed solution**
- **Proof of concept**
- **Conclusion and future work**



TCP window size optimization



Objective

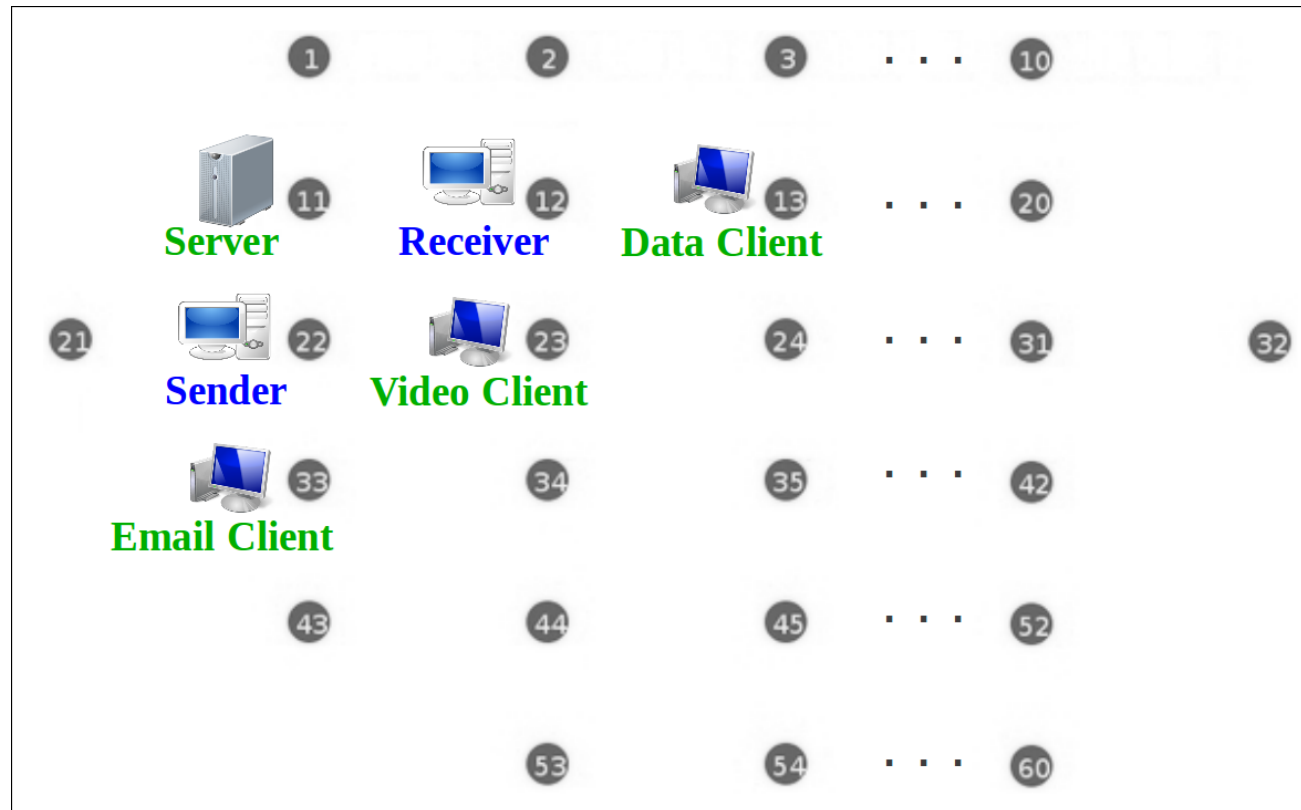
- Show how optimal parameter searching is conducted.
- Show the efficiency of using advanced searching methods like incremental search method within CREW benchmarking framework.



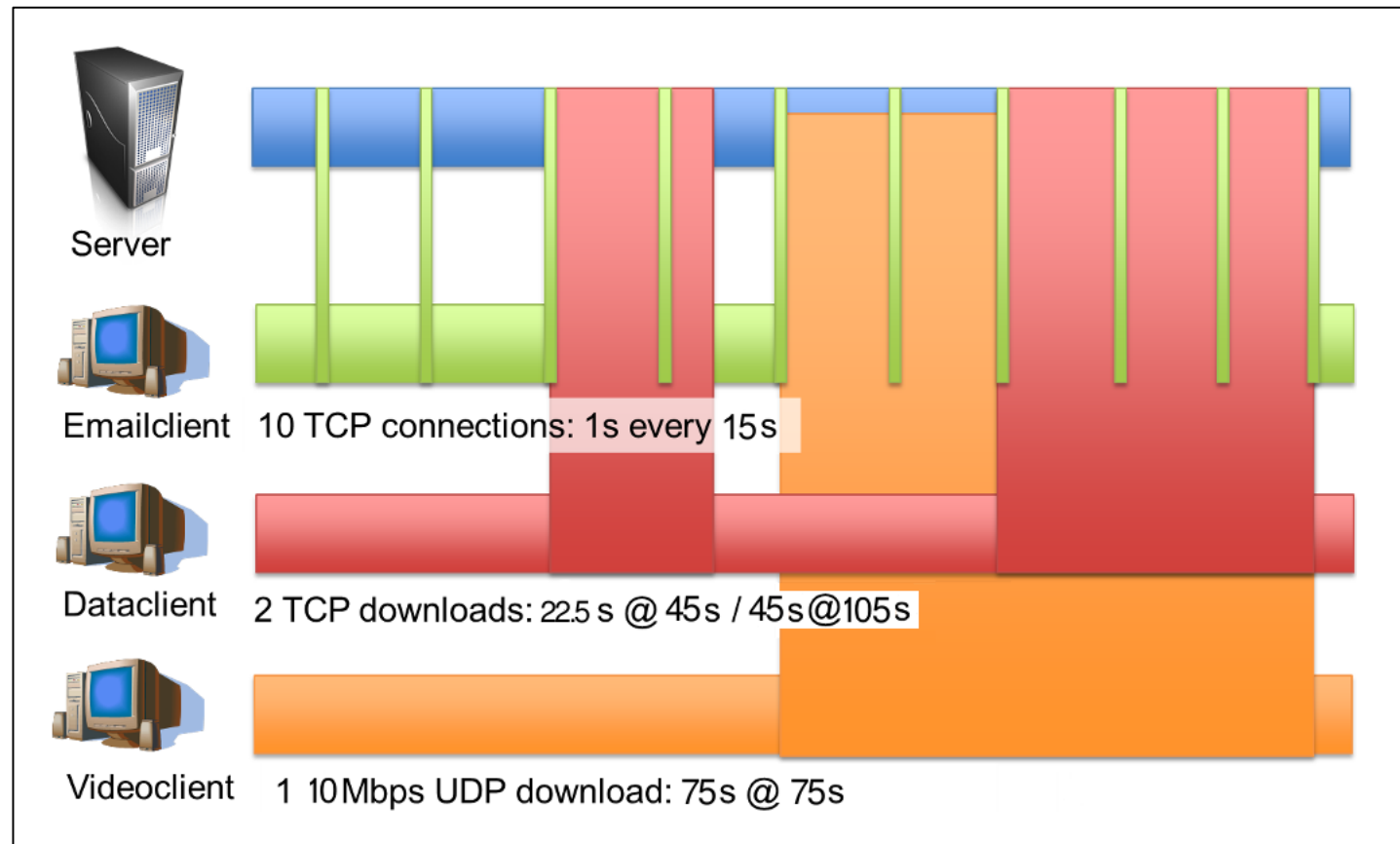
TCP window size optimization



Experiment layout



Experiment scenario

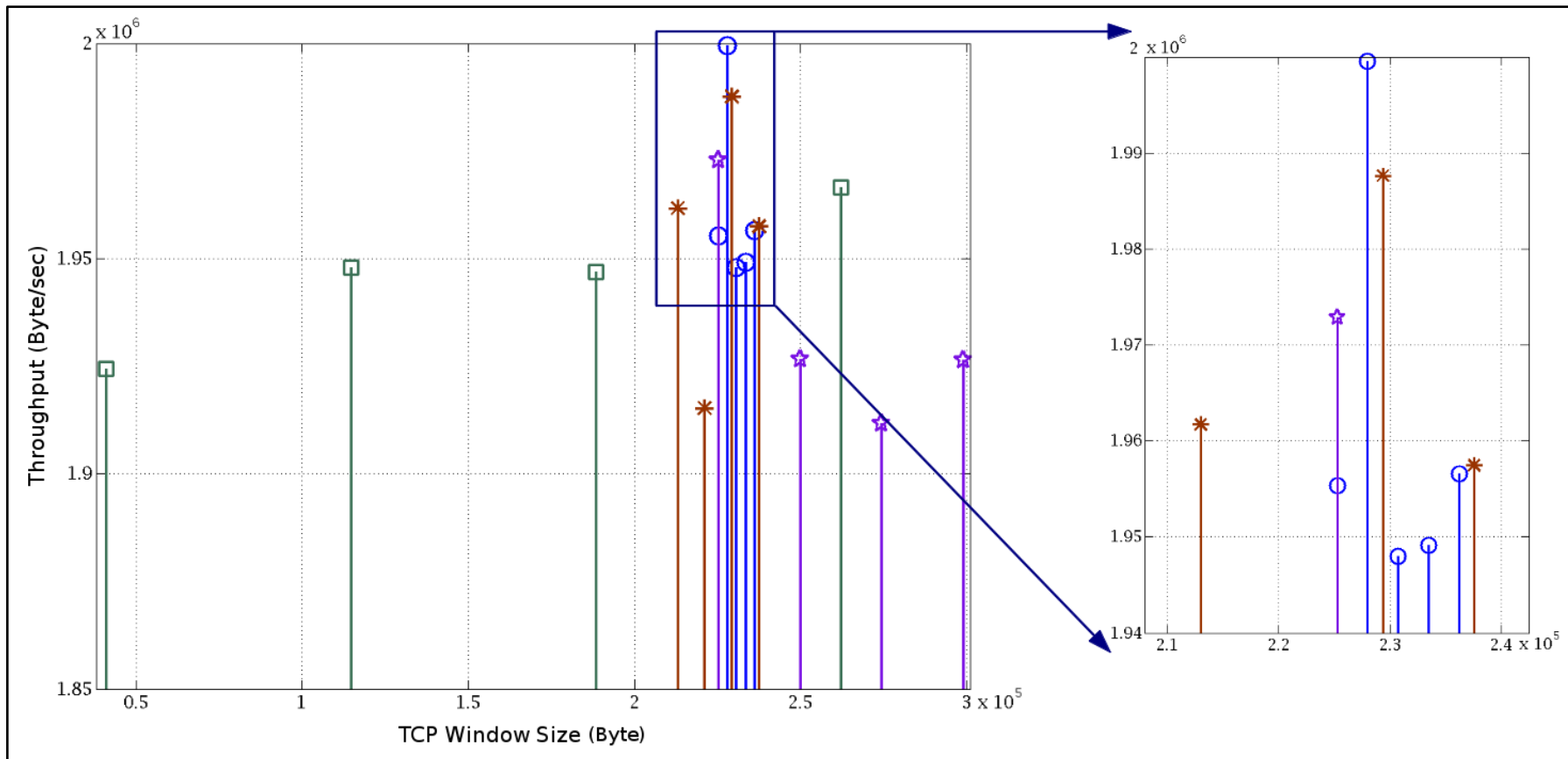




TCP window size optimization



Experiment result





OUTLINE



- **The need for benchmarking framework**
- **Proposed solution**
- **Proof of concept**
- **Conclusion and future work**



Conclusion



- **Why benchmarking is needed**
- **Our contribution**
 - Easy setup
 - Intelligent schedule
 - Parameter searching
 - Reliability
- **Focus of the near future**
 - Extend parameter searching algorithm
 - Portability and flexibility



Questions?

