

CycleOurCity goes Mobile

Guião para as Experiências - Fase 1

Este guião tem como objectivos:

- Planear as micro-experiências a realizar de forma a quantificar os ganhos e custos de:
i) processos de mobile crowdsensing; ii) processos de offloading computational.
- Especificar quais as ferramentas a utilizar

O guião será assim dividido em duas secções:

1. Micro-Experiências para Adaptive Offloading
2. Micro-Experiências para Mobile-Crowdsensing

Micro-Experiências para Adaptive Offloading

O esquema de adaptive offloading foi implementado com o objetivo de reduzir os custos computacionais (e consequentes custos energéticos) associados aos processos de análise e processamento dos dados capturados pelos sensores. Através da redução destes custos é expectável que se verifique um decréscimo no consumo energético da aplicação e consequentemente um aumento no tempo de vida da bateria do dispositivo. Tendo isto em conta as experiências a serem especificadas deverão:

- Quantificar o consumo *baseline* da aplicação, i.e. sem qualquer offloading
- Quantificar os overheads incorridos (memória e performance) pela existência/presença do mecanismo de offloading
- Quantificar a performance/complexidade do processo de optimização dos pipelines
- Quantificar os ganhos/perdas incorridos pelo processo de offloading ao nível de: i) tempo de execução; ii) CPU; iii) “burndown” da bateria; iv) dados transmitidos; v) consumo energético

Configurações para Testes

Para fins de testes serão consideradas as seguintes configuração de pipelines:

Configuração A					
Tempo: 100ms Dados: 800B	Tempo: 200ms Dados: 700B	Tempo: 300ms Dados: 600B	Tempo: 400ms Dados: 400B	Tempo: 500ms Dados: 200B	Tempo: 600ms Dados: 50B
Configuração básica. Nesta configuração o tempo aumenta linearmente e os dados decrescem, à medida que o pipeline avança. O objectivo é permitir analisar como estes se relacionam e influenciam a regra escolhida.					

Configuração B

Tempo: 50ms Dados: 600B	Tempo: 100ms Dados: 600B	Tempo: 600ms Dados: 10B	Tempo: 150ms Dados: 50B	Tempo: 10ms Dados: 50B
----------------------------	-----------------------------	----------------------------	----------------------------	---------------------------

Configuração Pseudo-real. Esta configuração tem como objectivo simular uma configuração real, no sentido em que:

- Etapa 1 - Análise simples da validade dos dados
- Etapa 2 - Remoção de ruído, as amostras são corrigidas mas permanecem todas
- Etapa 3 - Merging e extracção de features, as amostras são convertidas num feature vector de tamanho reduzido.
- Etapa 3 - Classificação, a informação inferida é adicionada
- Etapa 4 - Lógicas finais

Configuração C

Tempo: 100ms Dados: 800B	Tempo: 200ms Dados: 700B	Tempo: 300ms Dados: 600B	Tempo: 400ms Dados: 400B	Tempo: 500ms Dados: 200B	Tempo: 600ms Dados: 50B
Tempo: 50ms Dados: 600B	Tempo: 100ms Dados: 600B	Tempo: 600ms Dados: 10B	Tempo: 150ms Dados: 50B	Tempo: 10ms Dados: 50B	

Combinação das duas configurações anteriores.

Configuração D

Tempo: 100ms Dados: 800B	Tempo: 200ms Dados: 700B	Tempo: 300ms Dados: 600B	Tempo: 400ms Dados: 400B	Tempo: 500ms Dados: 200B	Tempo: 600ms Dados: 50B				
Tempo: 50ms Dados: 600B	Tempo: 100ms Dados: 600B	Tempo: 600ms Dados: 10B	Tempo: 150ms Dados: 50B	Tempo: 10ms Dados: 50B					
Tempo: 50ms Dados: 600B	Tempo: 100ms Dados: 600B	Tempo: 600ms Dados: 10B	Tempo: 150ms Dados: 50B	Tempo: 10ms Dados: 100B	Tempo: 10ms Dados: 150B	Tempo: 10ms Dados: 200B	Tempo: 10ms Dados: 250B		
Tempo: 50ms Dados: 600B	Tempo: 100ms Dados: 600B	Tempo: 600ms Dados: 10B	Tempo: 150ms Dados: 50B						
Tempo: 50ms Dados: 600B	Tempo: 10ms Dados: 50B	Tempo: 600ms Dados: 10B	Tempo: 200ms Dados: 700B	Tempo: 10ms Dados: 50B					

Combinação das configuração anterior com dois pipelines adicionais. A ideia é analisar o impacto de um maior número de pipelines.

Métricas de Avaliação

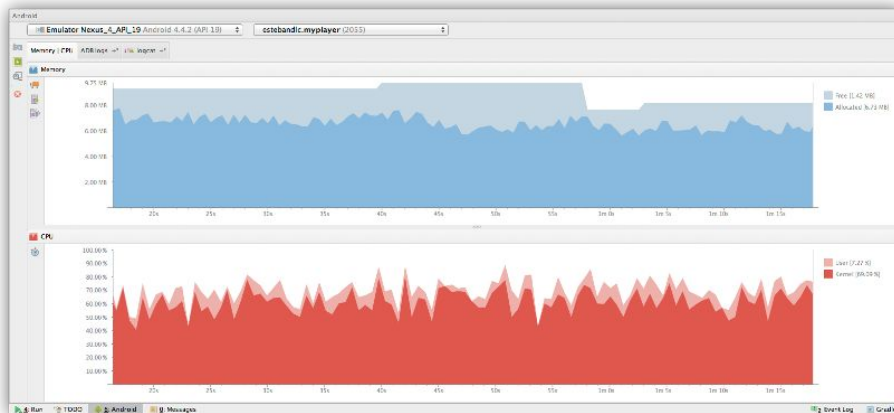
Estes testes têm como objetivo quantificar as seguintes métricas de avaliação:

A	Tempo médio de execução sequencial
B	Tamanho médio dos dados transmitidos
C	Overhead incorrido no CPU
D	Overhead incorrido ao nível da memória (Heap)
E	<i>Burdown-rate</i> da bateria (i.e velocidade de desgaste da bateria)
F	Consumo energético

Ferramentas

De forma a capturar informação que possibilite a quantificação das métricas acima descritas serão empregadas as seguintes técnicas e ferramentas:

A	<i>Tempo</i>	In-app , o tempo de execução é medido por Stage, usando o <i>clock</i> interno do sistema. O tempo de uma configuração será o igual à soma dos tempos das Stages que compõem cada pipeline.
B	<i>Dados Transmitidos</i>	In-app o tamanho dos dados transmitidos é medido por pipeline, calculando o tamanho dos dados produzidos pela última Stage de cada pipeline. Os dados transmitidos para cada configuração será o total dos dados transmitidos em cada pipeline.
C	<i>CPU</i>	Android performance tools , o Android studio permite gravar o “consumo” de CPU ao longo de uma sessão de testing. Estes dados são relativos a toda a aplicação, mas sabendo o <i>baseline</i> permitem ter uma noção dos overheads incorridos.
D	<i>Memória</i>	Android performance tools , o Android studio permite gravar o consumo de memória ao longo de uma sessão de testing. Estes dados são relativos a toda a aplicação mas permitem ter uma noção dos overheads incorridos.
E	<i>Battery Burndown</i>	Monitorização Manual , o battery burndown-rate pode ser inferido medindo quanto tempo é necessário para a bateria “descer” X%.
F	<i>Energia</i>	A determinar... (possivelmente o Trepn abaixo descrito)



Android Performance Tools (azul - memória; vermelho - cpu)

Outras Ferramentas

Trepp Profiler: Esta é uma aplicação Android, desenvolvida pela Qualcomm, que permite fazer profiling do dispositivo ou de uma aplicação em diferentes níveis, como energia, CPU ou dados transmitidos.

Plano de Testes

Nota	Para reduzir o impacto negativo de possíveis factores externos, cada experiência mencionada será repetida um total de 20 vezes. E em que cada experiência terá uma duração não inferior a 3 minutos.
	No caso do burndown da bateria cada experiência será apenas repetida 5 vezes, em que cada experiência terá duração indefinida, no sentido em que cada experiência terá a duração necessária para se verificar uma perda de 10% da bateria.
	Como o objectivo do offloading é reduzir os custos energéticos incorridos por processos computacionalmente intensivos e dado o facto das configurações propostas serem simuladas, durante as experiências propostas todos os sensores serão desligados.

Experiência 1 - Quantificação do baseline

Para cada uma das configuração mencionadas e sem a presença dos mecanismos offloading, serão quantificadas todas as métricas definidas. O objectivo desta experiência é criar um baseline que permita quantificar os ganhos ou custos incorridos pelo offloading.

Experiência 2 - Quantificação do overhead de offloading sem otimizações

Serão quantificadas todas as métricas definidas para cada uma das configuração na presença de offloading, sem no entanto executar a optimização dos pipelines. O objectivo desta experiência é quantificar os overheads incorridos (ao nível das diferentes métricas) causados pela presença de *profiling* do dispositivo (monitorização da bateria, rede e dados) e das Stages (tempo de execução e dados transmitidos).

É **expectável**:

- Aumento substancial na memória (heap) causado pelas estruturas de suporte
- Aumento negligenciável do overhead no CPU, causado pela actualização das estruturas de suporte.
- Inexistência de impacto ao nível do tempo de execução e dados transmitidos pelas diferentes configurações

Experiência 3 - Offloading estático (Bateria)

Para os níveis de bateria: 100%; 75%; 50%; 25% serão analisadas as versões optimizadas das configurações propostas, bem como a quantificação das métricas para cada uma das configurações optimizadas. Estas configurações optimizadas são geradas no início da sessão, não sendo alteradas ao longo da experiência, i.e são estáticas.

Experiência 4 - Offloading dinâmico (Bateria)

Nesta experiência a bateria é iniciada a 100% e a cada 3 minutos é reduzida 25%. O objectivo desta experiência é quantificar os custos associados ao offloading dinâmico durante o *runtime*.

É **expectável** que a cada redução:

- Tempo de processamento reduza
- O overhead do CPU reduza
- Verifiquem-se picos de CPU durante a optimização da configuração
- Aumento dos dados transmitidos (dependendo da configuração)

Experiência 5 - Offloading estático (Plano de dados e dados transmitidos)

Semelhante à experiência 3, mas para diferentes níveis do plano de dados (assume-se sempre ligação móvel). Adicionalmente o objectivo desta experiência passa por quantificar os ganhos ao nível da redução de dados a serem transmitidos.

Experiência 6 - Offloading Dinâmico (Plano de Dados e dados transmitidos)

Semelhante à experiência 4, mas para diferentes níveis do plano de dados (assume-se sempre ligação móvel). Adicionalmente o objectivo desta experiência passa por quantificar os ganhos ao nível da redução de dados a serem transmitidos.

Experiência 7 - Configuração Real

Assumindo a concretização dos dois pipelines propostos (declive e condições da estrada), serão replicadas as experiências anteriormente mencionadas (1-6) sob condições reais, *i.e.* com dados gerados pelos sensores e para os dois pipelines finais.

Ajuda
Avaliar as configurações propostas e se necessário propor correcções ou mesmo alternativas.

Micro-Experiências para Mobile Crowdsensing

O esquema de mobile crowdsensing tem dois objectivos primários:

- Classificar o declive de um segmento de um trajecto
- Classificar o tipo de pavimento e condições de um segmento de um trajecto

Nota	Assume-se que para o segundo objectivo já foi criado um classificador, usando a plataforma Weka, e que este funciona correctamente, <i>i.e.</i> com uma precisão pelo menos igual ou superior a 60%.
-------------	--

Para este segundo caso o **objectivo** das experiências é demonstrar que a aplicação é uma alternativa viável à classificação manual dos trajectos. Para tal:

- Serão escolhidos para cada tipo de pavimento e condições (más e boas) pelo 2 troços.
- Estes troços devem encontrar-se classificados no CycleOurCity, para que seja possível fazer uma comparação entre as classificações.

Fase 1 [Groundtruth] - Cada troço escolhido será analisado e classificado manualmente por uma fonte fidedigna. O declive em si será definido de acordo com as especificações da Rosa Félix.

Fase 2 - Cada troço será percorrido pelo menos 10 vezes, com a aplicação em execução, de forma a que esta possa inferir o tipo de pavimento e as suas condições.

Uma vez classificados pela aplicação cada um dos troços escolhidos, serão feitas comparações entre a *groundtruth*, as classificações no cycleourcity e as inferidas pela aplicação.

Adicionalmente serão aproveitadas estas experiências para analisar e/ou quantificar:

- Sucesso do mecanismos de remoção de outliers ao nível da geo-localização
- Quantificação de eventual ruído ao nível das amostras de pressão e acelerómetro

Ajuda
Escolher o melhor conjunto de troços para realizar ditos testes e estabelecer a <i>groundtruth</i> .