

# 위치인식 애플리케이션 전력 소비 분석을 위한 GPS 상태 모델링 기법

김덕기<sup>o</sup> 이정원

아주대학교 전자공학과

gcross07@ajou.ac.kr, jungwony@ajou.ac.kr

## GPS State Modeling for Power Consumption Estimation of Location-Aware Applications

Kim Deok Ki<sup>o</sup> Lee Jung Won

Department of Electrical and Computer Engineering, Ajou University

### 요 약

다양한 센서가 부착된 스마트 폰의 보급으로 상황인지 애플리케이션 시장의 규모가 발달하고 있다. 하지만, 한정된 전력으로 인해 원활한 서비스를 제공받기는 어렵다. 소프트웨어 관점에서의 저전력 상황인지 연구를 위해서는 애플리케이션의 센서 사용 패턴에 따른 전력 소비 분석이 중요하다. 따라서 본 논문에서는 전력 소모가 크고 위치정보를 수집하는 GPS 모듈의 동작을 API 호출을 기반으로 한 오토마타로 표현하고, 시중의 위치인식 애플리케이션들의 전력 소모를 측정하여, 애플리케이션 간의 전력 소비 패턴 및 소비 전력량의 차이를 분석해본다.

### 1. 서 론

대표적인 모바일 기기로 꼽을 수 있는 스마트 폰 및 태블릿 PC를 사용하는 인구 증가와 함께 모바일 앱 시장이 급속하게 성장하고 있다. 애플의 앱 스토어(App Store)의 경우 2013년 6월 기준 90만 개 이상의 앱을 제공하며, 구글 플레이(Google Play)는 약 70만 개의 앱을 제공하는 것으로 집계되고 있다. 모바일 앱 이용자 수도 2012년 약 12억 명에서 스마트 단말기의 확산에 따라 연 평균 29.7%씩 성장하여 2017년에는 약 44억 명에 달할 것으로 예상되고 있다. 스마트 폰은 GPS, Wi-fi, 가속도 센서, 조도 센서 등 다양한 모듈들을 이용하여 측정된 수치 정보를 운영체제에게 제공한다. 개발자는 센싱 정보들을 적절히 가공하여 컨텍스트(Context)를 생성하고, 이를 이용하여 새롭고 편리한 서비스를 제공할 수 있는 애플리케이션을 개발할 수 있다.[1]

그러나, 상황인지 애플리케이션의 과도한 센서나 네트워크 사용은 많은 전력 소모를 야기한다. 상황인지 애플리케이션의 소모 전력을 줄이는 노력이 필요하지만 기존의 상황인지 연구들은 대부분 보다 효율적으로 컨텍스트를 수집하는 방법을 제시하거나, 수집한 컨텍스트를 통해 지능화된 서비스를 제공해주는 방향으로 치중되어있으며, 소프트웨어의 관점에서 저전력 기법을 찾는 연구는 부족하다.

소프트웨어 관점에서 상황인지 저전력화에 관한 연구를 진행하려면 상황인지 애플리케이션이 컨텍스트 수집을 위하여 어떠한 방식으로 센서들을 사용하고, 그 과정에서 소비되는 전력이 얼마나 되는 지 분석해볼 필요가 있다. 기존의 소프트

웨어 관점에서의 모바일 전력 소비 분석 방법은 사전에 만든 센서 모듈에 대한 에너지 모델을 이용하는 방식[2]과, 단위 시간당 소비 전력에 센서 점유율을 곱하는 방식[3]이 있는데, 두 방법은 각각의 센서 모듈들이 소비하는 전력을 측정하는데 중점을 두고 있다.

하지만, 상황인지 애플리케이션의 관점에서 컨텍스트 수집을 위해 센서들을 활용하는 방법에 따라 전력 소비 양상이 다르게 나타날 수 있음을 분석하는 것은 애플리케이션들 간의 소비전력의 차이를 규명할 수 있는 기준이 된다. 그림 1은 똑같이 위치(Location) 컨텍스트를 수집하는 두 애플리케이션의 전력 소모 패턴에 관한 그림이다. 상황인지 애플리케이션이 센서 수집을 시작으로 서비스를 제공하기 까지 그림과 같은 계층을 가진다. 두 개의 애플리케이션 App A와 App B가 상황인지 서비스를 제공하기 위해 공통적으로 위치 컨텍스트를 필요로 하며, 컨텍스트를 수집하기 위해서 App A는 GPS와 Wifi를, App B는 GPS와 가속도 센서를 필요로 한다. 세 가지 센서들은 각각의 센서 사용 패턴을 가지며, 같은 종류의 센서라도 애플리케이션의 사용 방법에 따라 사용 패턴이 달라질 수 있다. 그리고 여러

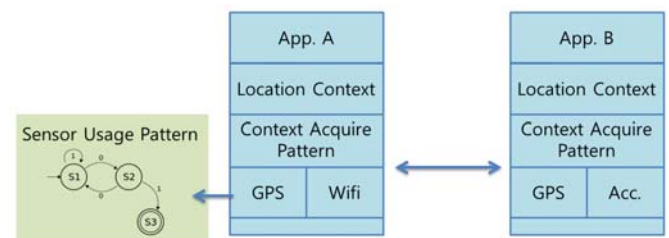


그림 1 위치 컨텍스트를 수집하는 두 애플리케이션의 컨텍스트 수집 패턴 비교 예시

\* 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 대학 ICT연구센터 육성지원사업의 연구결과로 수행되었음 (IITP-2015-H8501-15-1006)

센서 사용 패턴들의 조합으로 컨텍스트 수집 패턴을 표현할 수 있다. 따라서, 센서 사용 패턴을 전력 소모 상태에 따른 오토마타로 표현하여 애플리케이션이 다루는 센서의 전력 소모 분석이 가능하며, 결과적으로 애플리케이션이 컨텍스트를 수집하기 위한 전력 분석을 할 수 있다.

따라서, 본 논문에서는 저전력화된 컨텍스트 생성 기법 연구를 위한 모바일 상황인지 애플리케이션이 컨텍스트를 수집하는 과정에서 발생하는 센서의 동작을 알아보기 위하여 위치인식 애플리케이션의 GPS 동작 패턴을 오토마타로 표현하고, 시중의 두 애플리케이션의 전력을 측정하여 전력 소비 패턴을 비교해본다.

## 2. 관련연구

모바일 기기의 모듈 별 전력 소비 분석을 위한 연구는 어느 정도 진행되어왔다. 그 중 소프트웨어의 관점에서 전력 분석을 하는 연구로, Appscope[4]는 운영체제가 하드웨어 모듈에 접속하려는 요청을 감지하여, 모듈의 사용량과 상태를 파악하고 이를 사전에 만들어둔 에너지 모델에 대입하여 에너지 사용량을 얻는다. Appscope는 전력을 실측하지 않으며, 에너지 분석 툴이 PC 상에서 구동되기 때문에 사용하려면 스마트폰이 PC에 연결되어 있어야 한다. 때문에 이동하면서 전력 소비를 측정하기 어렵다. eprof[2]의 경우, 임의의 센서 모듈을 수행하면서 소모되는 전력을 측정하는 다음 소모 전력량 면에서 특징을 보이는 구간을 하나의 상태로 정의하고, 이를 기반으로 센서의 전력 소비량을 모델링하였다.

그 밖에 소프트웨어 관점에서 상황인지 애플리케이션의 저전력 기법으로, 여러 개의 상황인지 애플리케이션이 실행되고 있을 때, 컨텍스트 수집 방식을 조절하는 기법이 있다. [5]에서는 여러 애플리케이션이 컨텍스트 정보를 요구할 때, 각 애플리케이션의 요구사항을 저장하고 컨텍스트 수집을 중개하는 서비스를 두어 실제 센서의 동작을 최소화하는 방향으로 컨텍스트 정보를 전달한다. 이 경우, 센서의 동작을 최적화하지 않아, 단일 애플리케이션 실행 시 전력 절감의 효과가 없다.

본 연구에서는 위치인식을 위한 GPS 모듈의 동작을 세분화하고, 세분화한 동작을 최적화시키기 위해 애플리케이션 개발자의 관점에서 개발자가 사용하는 API 함수의 이름을 상태로 정의하여 센서의 동작을 추론하여

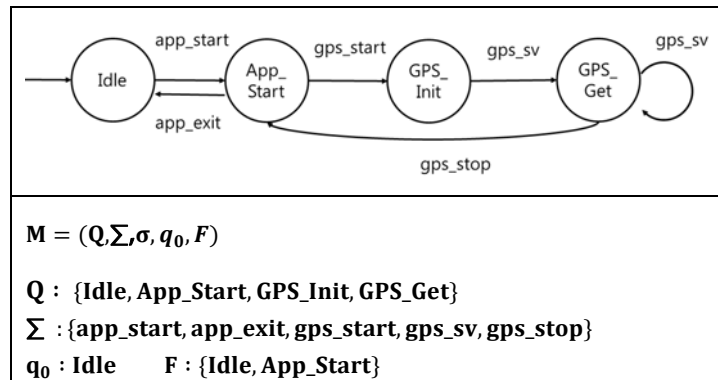


그림 2 오토마타 기반의 GPS 동작 모델링

모델링하는 방식을 취할 것이다.

## 3. 센서 모델링 기법

### 3.1 오토마타 기반 GPS 모델

그림 2는 추론할 수 있는 GPS 모듈의 활동을 네 가지 상태로 간소화하여 오토마타로 표현한 것이다.  $Q$ 는 위치인식 애플리케이션이 실행하면서 나타날 수 있는 모바일 기기의 상태를 나타낸 것으로, 애플리케이션이 실행되지 않은 상태(Idle), 애플리케이션이 실행되었으나, GPS 모듈은 동작하지 않는 상태(App\_Start), GPS 모듈을 초기화하는 상태(GPS\_Init), GPS 모듈을 이용하여 위치정보를 수집하는 상태(GPS\_Get)가 있다.  $\Sigma$ 은, 시스템 로그 기록을 통해 알 수 있는데, 3.2절에서 설명할 함수들을 통하여 얻을 수 있는 정보를 기반으로 한다.  $q_0$ 는 Idle상태이며,  $F$ 는 GPS가 사용 준비 단계인 Idle과 App\_Start 상태이다.

### 3.2 로그 분석용 소프트웨어 설계

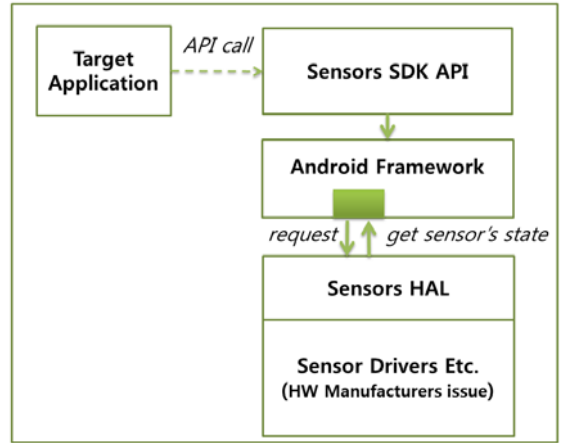


그림 3 로그 분석용 소프트웨어 개요

위 그림은 3.1의 오토마타에서  $\Sigma$ 를 얻기 위하여 안드로이드 시스템에서 어떠한 변화를 주었는지를 나타낸다. GPS 위치인식이 필요한 애플리케이션은 실행 시, 위치정보 수집을 위한 API를 사용한다. 해당하는 API를 호출할 때, 연관된 안드로이드 프레임워크에 로그 기록 클래스를 추가하여 프레임워크가 GPS와 통신할 때, 시스템 로그로 보고한다. 시스템 로그는 기 개발된 모바일 전력 측정 회로[6]와 시간적으로 동기화되어, 특정 상태에서의 전력 소비 패턴을 파악하게 한다.

애플리케이션을 실행과 종료(app\_start, app\_exit)는 ActivityManager가 남기는 로그를 통하여 알 수 있으며, GPS

표 1 GPS 모듈과 통신하는 API 함수 예시

함수명	추론 모듈 활동
onGpsStarted()	gps_start
onSvStatusChanged()	gps_sv
onLocationChanged()	
onGpsStopped()	gps_stop

모듈의 활동을 추론하기 위하여 로그를 분석하기 위해 사용하는 함수와 함수에 대응하는 이벤트( $\Sigma$ )는 표 1과 같다.

#### 4. 실험

##### 4.1 실험환경

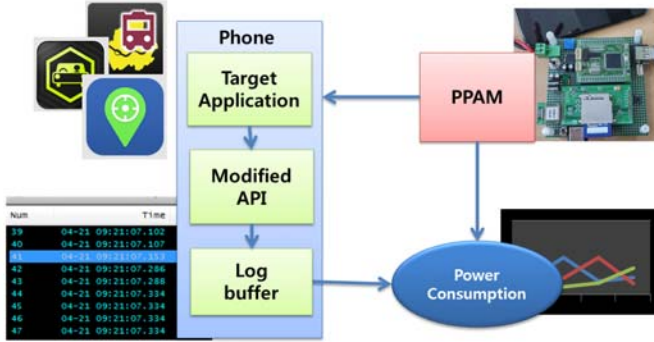


그림 4 소모 전력 측정 및 시스템 로그 기록 환경

실험기기는, Nexus 4를 이용하며, 로그 기록 소프트웨어는, 안드로이드 5.0.1 버전을 3.2절과 같이 변형하여 사용하였다. 소모 전력을 측정할 때 사용할 도구는 기 개발된 휴대용 전력 측정 회로(PPAM[6])를 이용한다. 이 전력 측정 회로는 0.2초마다 측정된 소모 전력을 파일로 생성하여 알려주는데, 이를 안드로이드 시스템 로그와 동기화하면 해당 시간의 소모 전력을 알 수 있다.

##### 4.2 결과

전력 소모를 측정한 애플리케이션은 최소한의 UI를 사용하는 두 개의 위치인식 애플리케이션을 선정하였다. 각각의 애플리케이션은 위치정보 수집명령을 내리면, 중단할 때까지 지정된 주기로 GPS 위치정보 수집을 수행하도록 설계되어있다. 다음 그래프는 선정한 애플리케이션(App B)의 시간-소모전력 그래프로, 그림 2 오토마타의 각 상태를 거치면서 측정된 소모 전력을 시간 순서대로 나타낸 것이다. 그래프의 경계선들은 오토마타의 상태가 변하는 순간을 나타낸다. 경계선으로 나뉘는 각 구간의 이름은 오토마타의 상태를 의미한다. 각 상태는 1~2분간 유지하도록 하였다.

App A의 경우, 실행 즉시 사용자의 위치를 실시간으로 확인하기 위하여 GPS 정보 수집을 시작하며, GPS

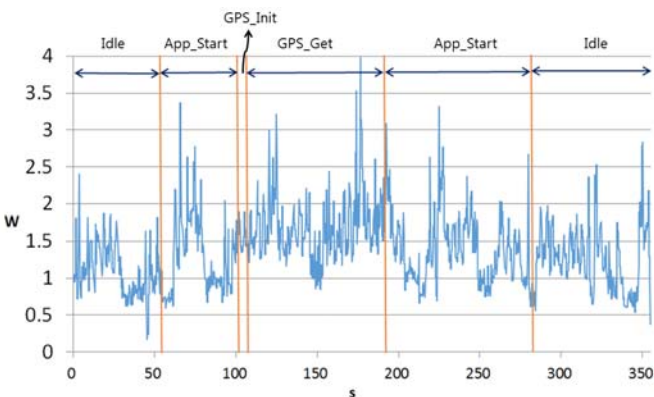


그림 5 App B의 시간(s)-소모전력(W) 그래프

표 2 위치인식 애플리케이션의 상태 별 소모 전력(mW)

상태	App A	App B
Idle	1053.39	1172.75
App Start	1253.09	1413.24
GPS_Init	1889.18	1836.14
GPS_Get	1956.15	1698.56

수집활동을 사용자가 제어할 수 없다. App A의 실행 시점에서 GPS\_Init 상태까지 수 초가 소요되고, GPS\_Get 상태로 전환한다. App B의 경우 근처 정류장을 검색하는 명령으로 GPS 위치정보 수집활동을 통제할 수 있다. GPS\_Get 상태에서 전력 소모의 차이를 보면, App A의 GPS 위치정보 수집 주기가 더 짧음을 예상할 수 있다.

표 2는 두 애플리케이션의 상태 별 평균 소모 전력을 나타낸 표이다. 그래프에서 구분된 면적에서 소비되는 전력의 평균값으로 구하였다. 표를 보면 두 애플리케이션의 상태 중 차이가 보이는 부분은 App\_Start와 GPS\_Get 상태이다. App\_Start 상태의 소모전력 차이는 애플리케이션에 따라 초기화하는 과정에 차이가 있기 때문인 것으로 생각되며, GPS\_Get 상태의 경우에는 GPS 수집 주기의 차이에 따른 것으로 보인다. 실제 시스템 로그를 조사한 결과, onLocationChanged() 함수를 호출하는 주기가 App A는 0.2초, App B는 0.25초로 App A의 주기가 더 짧았다.

#### 5. 결론 및 향후 연구

본 논문에서는 상황인지 애플리케이션이 컨텍스트 수집을 위하여 센서를 활용하는 패턴을 알아보기 위하여 위치인식 애플리케이션 개발 시 사용하는 API를 기반으로 한 GPS 동작 모델을 만들었다. 향후 더욱 미세한 분류 기준을 찾아서 세분화된 센서 모델을 만들고, 다양한 상황인지 애플리케이션의 센서 모델들을 비교하여 저전력화된 센서 모델을 고안하여 기존의 애플리케이션들에게 적용될 수 있도록 연구할 예정이다.

#### 6. 참고문헌

- [1] 김현진, 유정목, 박찬우, 김아영, 이준우. 모바일앱 분석 플랫폼 동향. 차세대 콘텐츠기술 특집. 29권 1호. 50-60. 2014. 02. 15.
- [2] Pathak, A., Hu, Y. C., & Zhang, M. (2012, April). Where is the energy spent inside my app?: fine grained energy accounting on smartphones with eprof. In Proceedings of the 7th ACM european conference on Computer Systems (pp. 29-42). ACM
- [3] Yang, Z. (2012). PowerTutor-A Power Monitor for Android-Based Mobile Platforms. EECS, University of Michigan, retrieved September, 2.
- [4] Yoon, C., Kim, D., Jung, W., Kang, C., & Cha, H. (2012, June). AppScope: Application Energy Metering Framework for Android Smartphone Using Kernel Activity Monitoring. In USENIX Annual Technical Conference (pp. 387-400).
- [5] 김문권, 이재유, 김수동. 에너지 효율적인 모바일 컨텍스트 모니터링 기법 설계. 제 16회 한국 소프트웨어공학 학술대회논문집. 제 16권 1호. 167-168. 2014.
- [6] 최기용, 이정원. 모바일 기기의 전력 측정을 위한 휴대용 시스템 한국정보과학회 정보과학회논문지 : 컴퓨팅의 실제 및 레터 제20권 제3호,131-142, 2014.3