

Wireless Networked Control System에서 Actuator 기반 Direct Actuation Update 방법

경연웅¹, 김태국², 김영준^{3*}

¹공주대학교 정보통신공학과 교수, ²부경대학교 컴퓨터공학부 교수, ³경남대학교 컴퓨터공학부 교수

Direct Actuation Update Scheme based on Actuator in Wireless Networked Control System

Yeunwoong Kyung¹, Tae-Kook Kim², Youngjun Kim^{3*}

¹Professor, Division of Information & Communication Engineering, Kongju National University

²Professor, School of Computer Engineering, Pukyung University

³Professor, School of Computer Science and Engineering, Kyungnam University

요약 최근 Internet of Things (IoT) 기반 Wireless Networked Control System (WNCS)에서 Sensor의 Status Update 및 Actuator로의 Actuation Update 분석을 위해 정보의 신선도를 측정하는 지표인 Age of Information (AoI)가 고려되고 있다. 또한 WNCS에 Edge Computing (EC)이 도입되면서 기존의 Cloud Computing 기반 아키텍처보다 낮은 AoI를 보장할 수 있다. 하지만 Controller가 관리하는 Sensor의 수가 증가하면서 Controller에 부하가 증가하여 AoI 요구사항을 만족시키지 못하는 문제점이 발생하게 되었다. 본 연구에서는 이러한 문제를 해결하기 위해 Actuator의 컴퓨팅 능력을 활용하여 Sensor의 Status Update를 해당 지역의 Actuator가 가용할 때 직접적으로 전송하여 Actuator가 직접 Actuation Update를 수행함으로써 AoI 요구사항을 만족시키고자 한다. 이를 위해 본 연구에서는 AoI 분석을 위한 분석 모델을 제시하였고 시뮬레이션을 통해 제안하는 방법이 기존 방법 대비 AoI를 줄일 수 있음을 보였다.

주제어 : 무선 네트워크 기반 제어 시스템, 정보의 신선도, 엣지 컴퓨팅, 상태 업데이트, 부하 분배

Abstract Age of Information (AoI) has been introduced in wireless networked control systems (WNCSs) to guarantee timely status updates. In addition, as the edge computing (EC) architecture has been deployed in NCS, EC close to sensors can be exploited to collect status updates from sensors and provide control decisions to actuators. However, when lots of sensors simultaneously deliver status updates, EC can be overloaded, which cannot satisfy the AoI requirement. To mitigate this problem, this paper uses actuators with computing capability that can directly receive the status updates from sensors and determine the control decision without the help of EC. To analyze the AoI of the actuation update via EC or directly using actuators, this paper developed an analytic model based on timing diagrams. Extensive simulation results are included to verify the analytic model and to show the AoI with various settings.

Key Words : Wireless Networked Control System, Age of Information, Edge Computing, Status Update, Load Distribution

본 과제(결과물)는 2022년도 교육부의 재원으로 한국연구재단의 지원을 받아 수행된 지자체-대학 협력기반 지역혁신 사업의 결과입니다. (2021RIS-003)

*교신저자 : 김영준(youngjun@kyungnam.ac.kr)

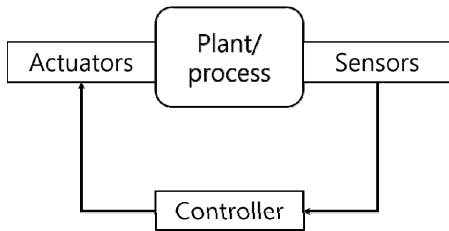
접수일 2022년 12월 13일

수정일 2023년 1월 11일

심사완료일 2023년 1월 15일

1. 서론

4차 산업혁명의 등장으로 인해 스마트 제조 및 지능형 건설 등의 환경에서 센싱, 통신, 그리고 제어를 수행하는 Industrial Internet of Things (IIoT)에 대한 관심이 높아졌다[1-4]. 이에 따라 실시간 및 신뢰성 있는 IIoT 제어를 위한 Wireless Networked Control Systems (WNCSs)의 역할이 커지고 있다[5].



[Fig. 1] A wireless networked control system

일반적으로 WNCS는 Fig. 1과 같이 Plant 내에서 담당할 영역의 현재 상황을 측정하는 Sensor들과 Sensor들에서 측정된 데이터를 수집하고 컴퓨팅 능력을 기반으로 Actuator의 Actuation을 결정하는 Controller, 그리고 실제 Plant의 물리적인 process를 수행하는 Actuator로 구성된다. 또한 최근 mobile robot과 automated guided vehicle (AGV)와 같은 이동성이 있는 actuator들이 도입되면서 유선이 아닌 무선으로 제어되는 것이 보편화되었다. 이러한 WNCS에서는 크게 Sensor에서 측정된 데이터를 Controller로 전송하는 Status Update와 Controller에서 수집된 데이터 기반으로 Actuator의 행동을 결정하여 전송하는 Actuation Update의 2가지 update가 발생한다. 이런 Update가 지연될 경우에는 비효율적인 공정처리 및 인명 피해와 같은 문제가 발생할 수 있기 때문에 Sensor의 제일 최신 신선한 정보가 시스템에 반영될 수 있는 Age of Information (AoI)에 대한 필요성이 대두되었다.

AoI는 정보 업데이트의 신선도를 측정하는 지표로 도입되었다[6]. AoI는 정보가 생성된 이후 목적지에 도착하기까지의 elapsed 시간으로 정의된다. 대부분의 AoI 기반 연구에서는 Sensor에서 수집된 정보가 Controller에 수집되는 Status Update에 초점이 맞추어져 있지만[6-8], 본 연구에서는 WNCS에서 Sensor에서 수집된 제일 최신 정보가 Actuator까지 반영되는 시간을 고려하고자 한다.

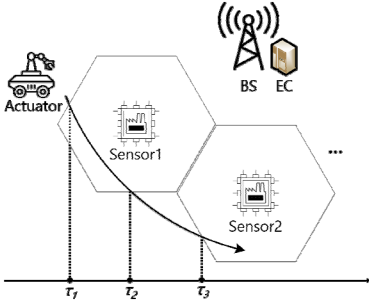
한편, 컴퓨팅 자원을 사용자 가까이 배치하는 edge computing (EC)에 대한 개념이 WNCS에 도입되기 시

작하였고[9] 이에 따라 기존 클라우드 기반의 Controller보다 더 낮은 AoI를 제공할 수 있게 되었다. 하지만 Controller에 할당된 Sensor의 수가 증가하고, Status Update 수가 증가함에 따라 Controller에 과부하가 발생할 수 있다[10,11]. 이 문제를 해결하고자 본 연구에서는 컴퓨팅 능력을 가진 Actuator가 Sensor로부터 직접적으로 Status Update를 수신하여 자신이 담당하고 있는 영역의 Actuation Update를 직접 수행하여 EC의 부하를 기회적으로 줄일 수 있는 방법을 제안하였다. 이는 최근 컴퓨팅 기능이 탑재된 Actuator가 도입이 되어 자신이 담당하고 있는 영역에서의 Actuation History 및 Local View를 갖고 있는 연구들이 제시되고 있는데 [12,13] WNCS에서도 이를 활용할 수 있는 방법이다. 제안하는 방법에서 Actuator는 mobile robot 또는 AGV처럼 이동하는 기기이기 때문에 해당 기기가 이동을 하다가 자신의 관리 영역 내에 Status Update가 존재하는 Sensor를 만났을 때, Sensor로부터 직접적으로 Status Update를 수신한다. 그리고 자신이 담당하고 있는 영역에서의 Actuation History 및 Local View를 기반으로 자신의 Actuation Update를 수행하여 물리적인 process를 수행하게 된다. 만약 Sensor가 Status Update를 수행하고자 할 때 Actuator가 근처에 존재하지 않는다면, 기존 방법과 마찬가지로 EC에게 Status Update를 수행하고, EC가 해당 Actuator에게 Actuation Update를 수행하게 된다.

2. 제안하는 방법

2.1 시스템 모델

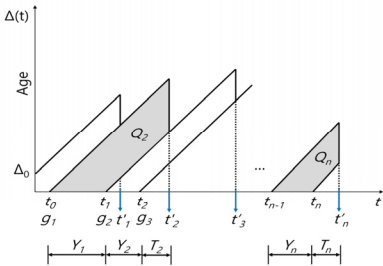
제안하는 방법의 시스템 모델은 Fig. 2와 같다. 본 연구에서는 EC가 특정 네트워크 (예: 공장 네트워크 등)의 통신을 관장하는 기지국(Base Station; BS)에 연결되어 있는 것을 가정하였고, Sensor는 Status Update를 해당 EC로 항상 전송할 수 있는 상황이다. 또한 Actuator는 정해진 경로를 통해 이동하며 그림에서 τ_1 에서 τ_2 와 같이 자신이 관리하는 영역 내에 Sensor와 컨택할 수 있으며 이를 Actuator 컨택 시간이라고 정의한다. 본 연구에서는 해당 컨택 시간 분포가 평균 $1/\lambda$ 를 갖는 exponential distribution을 따른다고 가정하였다[14]. 해당 컨택 시간 내에서 Sensor는 Actuator로 Status Update를 수행할 수 있고, 컨택 시간 이후에는 EC를 통해 Actuator로 Update를 수행한다.



[Fig. 2] System model

2.2 AoI 분석

제안하는 방법의 AoI 분석을 위해 Actuator 관점에서의 age 변화는 Fig. 3과 같다. 본 연구에서는 이전 데이터의 컴퓨팅 또는 전송이 끝나면 새로운 데이터가 발생하는 Zero-Wait Message Generation 정책을 가정하였다[6-8]. Fig. 3에서의 g_i 는 Sensor에서 새로운 i 번째 측정 데이터가 발생한 시간을 의미하고, $g_i - g_{i-1}$ 인 Y_i 는 데이터 발생 간격을 의미한다. 이 때, t_i 에 대한 해석이 컨택 시간 내외에서 다르게 수행되어야 한다. 먼저 컨택 시간 내에서 t_i 는 Sensor의 Status Update가 Actuator에 도착한 시간을 의미하고, t'_i 은 이를 기반으로 Actuation Update가 완료되는 시간을 의미한다. 그러므로 Actuator 입장에서 $t_i - t'_i$ 인 T_i 는 Status Update 수신부터 Actuation Update까지의 시간을 의미하고 Actuation Update가 되는 t'_i 에 AoI가 reset된다. 반면, 컨택 시간이 아닌 경우에 t_i 는 Sensor의 Status Update가 Controller에 도착한 시간을 의미하고, t'_i 은 이를 Controller에서 컴퓨팅하여 Actuation Update를 생성하고, 이를 Actuator가 수신한 시간을 의미한다. 그러므로 Controller 입장에서 $t_i - t'_i$ 인 T_i 는 Status Update 수신부터 Actuation Update를 생성하고 이를 Actuator가 수신하기까지의 시간을 의미하고 Actuator 입장에서 Actuation Update가 되는 t'_i 에 AoI가 reset된다.



[Fig. 3] Example change in age at the actuator

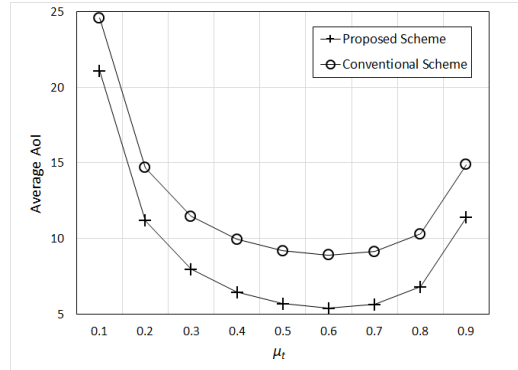
먼저, Fig. 3의 Q_i 는 다음과 같이 계산될 수 있다.

$$Q_i = T_i Y_{i-1} + Y_i Y_{i-1} + \frac{1}{2} Y_i^2 \quad (1)$$

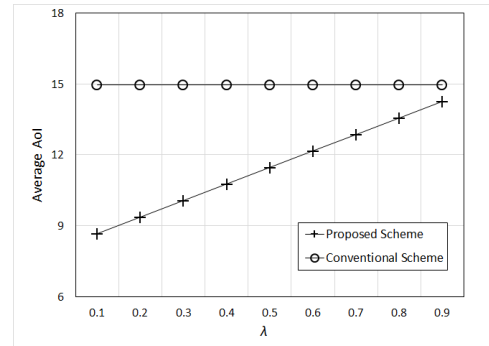
Q_i 를 기반으로 전체 면적을 구해서 시간으로 나누어 평균 AoI를 계산할 수 있는데, 이 때 EC와 Actuator의 컴퓨팅 시간이 exponential 분포를 따른다고 가정한다면 컨택 시간 내에 Actuator가 직접 Status Update를 수신하고 Actuation Update를 수행하는 경우 평균 AoI는 다음과 같이 계산될 수 있다[15].

$$AoI_{avg} = \frac{1}{\mu_s} \left(\frac{2\mu_t^3 - \mu_t^2 \mu_s + \mu_t \mu_s^2}{\mu_s (\mu_s + \mu_t) (\mu_s - \mu_t)} + \frac{2\mu_s}{\mu_t} + 1 \right) \quad (2)$$

위의 식에서 μ 는 Sensor의 transmission rate를 의미하고 μ_s 는 Actuator의 computing rate를 의미한다 [12]. 또한 컨택 시간 외에서는 식(2)에서 Controller에서 계산된 Actuation Update가 Actuator로 전송되는 시간을 더해서 평균 AoI를 계산할 수 있다.



[Fig. 4] Average AoI according to μ_t



[Fig. 5] Average AoI according to λ

3. 성능 분석

제안하는 방법의 성능을 분석하기 위해 항상 EC로 Status Update를 수행하는 기존 방법과 제안하는 방법을 비교하였다. 시뮬레이션을 위한 디플트 값으로 λ 는 0.5로 가정하였고, μ_s 와 μ_t 는 각각 1과 0.3으로 가정하였다.

Fig. 4는 μ_t 가 변할 때 제안하는 방법과 기존 방법의 Average AoI를 보여준다. Fig. 4에서 알 수 있듯이 μ_s 가 1로 설정되어 있기 때문에 μ_t/μ_s 가 작을 때에는 전송율이 떨어지기 때문에 AoI가 증가하게 되고, μ_t/μ_s 이 클 때에는 높은 전송율로 인해 컴퓨팅 큐에 쌓일 수 있게 되어서 AoI가 증가하게 되는 것을 볼 수 있다. 또한 제안하는 방법이 기존 방법에 비해 Actuator 자체적인 컴퓨팅을 활용하기 때문에 낮은 AoI를 갖는 것을 확인할 수 있다.

Fig. 5는 λ 가 변할 때 제안하는 방법과 기존 방법의 Average AoI를 보여준다. Fig. 5를 통해 Sensor가 Actuator를 만날 확률이 높을수록 직접적인 Update를 통해 AoI가 감소하는 것을 확인할 수 있다. 해당 결과를 통해 컴퓨팅 능력을 갖고 있는 Actuator를 활용하는 방법의 필요성을 확인할 수 있었다.

4. 결론 및 향후 연구 방안

본 논문은 Wireless Networked Control System (WNCS)에서 Sensor의 Status Update를 컴퓨팅 능력을 갖고 있는 Actuator에서 직접 처리하여 Controller의 부하를 줄이면서 낮은 Age of Information(AoI)를 제공할 수 있는 방안을 제안하였다. 성능 분석을 통해 킨택 시간이 길어지거나 Actuator의 computing rate가 증가할수록 제안하는 방법의 AoI가 적어짐을 보였다. 향후에는 multi-source AoI 분석을 진행할 예정이며 다양한 큐잉 모델을 활용하여 EC의 부하가 감소하는 성능을 함께 입증하고자 한다.

REFERENCES

- [1] R.K.Manasano, R.J.Rodrigues, E.P.Godoy, and D.Colon, "A New Adaptive Controller in Wireless Networked Control Systems," IEEE Industry Applications Magazine, Vol.25, No.2, pp.12-22, 2019.
- [2] S.Park and J.Bae, "A Design on the Zone Master Platform based on IIoT Communications for Smart Factory Digital Twin," Journal of The Korea Internet of Things Society, Vol.6, No.4, pp.81-87, 2020.
- [3] M.Lee "Performance Evaluation of Smoothing Algorithm for Efficient Use of Network Resources in IoT environments," Journal of The Korea Internet of Things Society, Vol.7, No.2, pp.47-53, 2021.
- [4] M.Lee "Smoothing Algorithm Considering Server Bandwidth and Network Traffic in IoT Environments," Journal of The Korea Internet of Things Society, Vol.8, No.1, pp.53-58, 2022.
- [5] W.Liu, P.Popovski, Y.Li, and B.Vucetic, "Wireless Networked Control Systems With Coding-Free Data Transmission for Industrial IoT," IEEE Internet of Things Journal, Vol.7, No.3, pp.1788-1801, 2020.
- [6] Y.Sun, E.U.Biyikoglu, R.D.Yates, C.E.Koksal, and N.B.Shroff, "Update or Wait: How to Keep Your Data Fresh," IEEE Trans. on Information Theory, Vol.63, No.11, pp.7492-7508, 2017.
- [7] X.Wang, C.Chen, J.He, S.Zhu, and X.Guan, "AoI-Aware Control and Communication Co-Design for Industrial IoT Systems," IEEE Internet of Things Journal, Vol.8, No.10, pp.8464-8473, 2021.
- [8] J.P.Champati, H.A.Zubaidy, and J.Gross, "Statistical Guarantee Optimization for AoI in Single-Hop and Two-Hop FCFS Systems With Periodic Arrivals," IEEE Trans. on Communications, Vol.69, No.1, pp.365-381, 2021.
- [9] W.Dai, H.Nishi, V.Vyatkin, V.Huang, Y.Shi, and X.Guan, "Industrial Edge Computing: Enabling Embedded Intelligence," IEEE Industrial Electronics Magazine, Vol.13, No.4, pp.48-56, 2019.
- [10] D.W.Lee, K.Cho, and S.H.Lee, "Analysis on Smart Factory in IoT Environment," Journal of The Korea Internet of Things Society, Vol.5, No.2, pp.1-5, 2019.
- [11] Y.Kyung, "Delayed offloading scheme for IoT tasksconsidering opportunistic fog computing environment," Journal of The Korea Internet of Things Society, Vol.6, No.4, pp.89-92, 2020.
- [12] L.Ravaglia, M.Rusci, D.Nadalini, A.Capotondi, F.Conti, and L.Benini, "A TinyML Platform for On-Device Continual Learning With Quantized Latent Replays," IEEE Journal on Emerging and Selected Topics in Circuits and Systems, Vol.11, No.4, pp.789-802, 2021.
- [13] R.S.Iborra and A.F.Skarmeta, "TinyML-Enabled Frugal Smart Objects: Challenges and Opportunities," IEEE Circuits and Systems Magazine, Vol.20, No.3, pp.4-18, 2020.
- [14] H.Ko and Y.Kyung, "Performance Analysis and Optimization of Delayed Offloading System With Opportunistic Fog Node," IEEE Trans. on Vehicular Technology, Vol.71, No.9, pp.10203-10208, 2022.
- [15] Q.Kuang, J.Gong, X.Chen, and X.Ma, "Analysis on Computation-Intensive Status Update in Mobile Edge Computing," IEEE Trans. on Vehicular Technology, Vol.69, No.4, pp.4353-4366, 2020.

경 연 웅(Yeunwoong Kyung) [종신회원]



- 2011년 2월 : 고려대학교 전기전
자전파공학부(공학사)
- 2016년 8월 : 고려대학교 전기전
자전파공학부(공학박사)
- 2016년 9월 ~ 2020년 3월 : 삼
성전자 무선사업부 책임연구원

- 2020년 3월 ~ 2022년 8월 : 한신대학교 컴퓨터공학부
교수
- 2022년 9월 ~ 현재 : 국립공주대학교 정보통신공학과
교수

<관심분야>

사물인터넷(IoT), SDN, 5G/6G, 이동성, 모바일 서비스

김 태 국(Tae-Kook Kim) [종신회원]



- 2004년 8월 : 고려대학교 전기전
자전파공학부(공학사)
- 2006년 8월 : 고려대학교 메카트
로닉스학과(공학석사)
- 2014년 8월 : 고려대학교 모바일
솔루션학과(공학박사)

- 2016년 3월 ~ 2022년 2월 : 동명대학교 AI학부 교수
- 2022년 3월 ~ 현재 : 국립부경대학교 컴퓨터공학부 교수

<관심분야>

사물인터넷(IoT), 콘텐츠 전송 네트워크(CDN), 이동성, 인
공지능(AI), 빅데이터, 모바일 서비스

김 영 준(Youngjun Kim) [정회원]



- 2010년 2월 : 고려대학교 전기전
자전파공학부(공학사)
- 2022년 8월 : 고려대학교 전기전
자전파공학부(공학박사)
- 2022년 9월 ~ 현재 : 경남대학교
컴퓨터공학부 교수

<관심분야>

사물인터넷(IoT), WiFi, SDN, 모바일 서비스