



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

碩士學位論文

최대출력제어를 위한
풍속예측에 관한 연구

濟州大學校 大學院

風力工學部 風力電氣·制御시스템專攻

高 承 鎰

2015 年 8 月

**Study on Wind Estimation
for Maximum Power Control**

**Seung Youn Ko
(Supervised by professor Min Jae Kang)**

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING**

**Major of Wind Power Electricity and
Control System Engineering
Faculty of Wind Energy Engineering Graduate School
JEJU NATIONAL UNIVERSITY**

2015. 8.

목 차

LIST OF FIGURES	vi
LIST OF TABLES	vii
SUMMARY	viii
제 1 장 서 론	1
제 2 장 풍력터빈시스템	3
2.1 풍력에너지	3
2.2 최대출력제어	4
2.2.1 P&O제어	5
2.2.2 최적 TSR 제어	7
제 3 장 풍속예측을 이용한 최대출력제어	10
3.1 풍속예측	10
3.2 첫 번째 제안방법: 리그레션을 이용한 풍속예측	11
3.3 두 번째 제안 방법: 출력과 회전속도를 이용한 풍속예측	14
제 4 장 시뮬레이션 및 결과	16
4.1 시뮬레이션 모델	16
4.1.1 기존 MPC Block	17
4.1.2 제안된 풍속예측 MPC Block	17
4.2 시뮬레이션 결과	18
4.2.1 고정 풍속에서의 시뮬레이션 결과	18

4.2.2 가변 풍속에서의 시뮬레이션 결과	21
제 5 장 결 론	27
참 고 문 헌	29
Appendix A Matlab/Simulink 모델	31
Appendix B 사용된 프로그램 코드	41

LIST OF FIGURES

Fig. 1 Installed wind power capacity by region in the New Policies Scenario(IEA)	1
Fig. 2 Wind Turbine System Block	3
Fig. 3 Power Curve of Wind Turbine	5
Fig. 4 Control strategy of P&O	6
Fig. 5 MPC using P&O Method	7
Fig. 6 Power Coefficient(C_p) Versus TSR(λ)	8
Fig. 7 Power Curve using Optimal TSR method	9
Fig. 8 Plot for $h(x)$	12
Fig. 9 $h(x)$ and approximated equation $h_{reg}(x)$ by regression method	13
Fig. 10 (a) Plot for $h(x)$ and $h_{reg}(x)$, (b) Estimated wind speed by $h_{reg}(x)$ curves, (c) Expand (b) plot between 0~0.1 sec	13
Fig. 11 w_r-P curves depending on v	14
Fig. 12 Control strategy for proposed method	15
Fig. 13 PMSG Simulink Model	16
Fig. 14 MPC block(P&O)	17
Fig. 15 Wind Estimator(Proposed Method)	18
Fig. 16 Simulation result at constant wind speed(P&O Method) (a)Wind Speed (b)Rotate speed (c)Power	19
Fig. 17 Enlarged plot between 0~1.2[sec] of Fig. 16 (a)Wind Speed (b)Rotate speed (c)Power	19
Fig. 18 Simulation result at constant wind speed (Newton-Raphson Method) (a)Wind Speed (b)Rotate speed (c)Power	19
Fig. 19 Simulation result at constant wind speed (Newton-Raphson Method) (a)Wind Speed (b)Rotate speed (c)Power	19
Fig. 20 Simulation result at constant wind speed(Regression Method) (a)Wind Speed (b)Rotate speed (c)Power	20
Fig. 21 Enlarged plot between 0~0.5[sec] of Fig. 20 (a)Wind Speed (b)Rotate speed (c)Power	20
Fig. 22 Simulation result at constant wind speed(Proposed Method) (a)Wind Speed (b)Rotate speed (c)Power	20
Fig. 23 Enlarged plot between 0~0.25[sec] of Fig. 22 (a)Wind Speed (b)Rotate speed (c)Power	20

Fig. 24 Simulation result at dramatically varying wind speed (P&O Method) (a)Wind Speed (b)Rotate speed (c)Power	21
Fig. 25 Enlarged plot between 0~1.2[sec] of Fig. 24 (a)Wind Speed (b)Rotate speed (c)Power	21
Fig. 26 Simulation result at dramatically varying wind speed (Newton-Raphson Method) (a)Wind Speed (b)Rotate speed (c)Power	22
Fig. 27 Enlarged plot between 0~0.25[sec] of Fig. 26 (a)Wind Speed (b)Rotate speed (c)Power	22
Fig. 28 Simulation result at dramatically varying wind speed (Regression Method) (a)Wind Speed (b)Rotate speed (c)Power	22
Fig. 29 Enlarged plot between 0~0.25[sec] of Fig. 29 (a)Wind Speed (b)Rotate speed (c)Power	22
Fig. 30 Simulation result at dramatically varying wind speed (Proposed Method) (a)Wind Speed (b)Rotate speed (c)Power	23
Fig. 31 Enlarged plot between 0~0.25[sec] of Fig. 30 (a)Wind Speed (b)Rotate speed (c)Power	23
Fig. 32 Simulation result at gradually varying wind speed(P&O Method) (a)Wind Speed (b)Rotate speed (c)Power	24
Fig. 33 Enlarged plot between 0~1.2[sec] of Fig. 32 (a)Wind Speed (b)Rotate speed (c)Power	24
Fig. 34 Simulation result at gradually varying wind speed (Newton-Raphson Method) (a)Wind Speed (b)Rotate speed (c)Power	25
Fig. 35 Enlarged plot between 0~0.25[sec] of Fig. 34 (a) Wind Speed (b) Rotate speed (c)Power	25
Fig. 36 Simulation result at gradually varying wind speed (Regression Method) (a)Wind Speed (b)Rotate speed (c)Power	25
Fig. 37 Enlarged plot between 0~0.25[sec] of Fig. 36 (a)Wind Speed (b)Rotate speed (c)Power	25
Fig. 38 Simulation result at gradually varying wind speed (Proposed Method) (a)Wind Speed (b)Rotate speed (c)Power	26
Fig. 39 Enlarged plot between 0~0.25[sec] of Fig. 16 (a)Wind Speed (b)Rotate speed (c)Power	26
Fig. 40 Enlarged plot between 0~0.25[sec] of Fig. 16 (a)Wind Speed (b)Rotate speed (c)Power	26
Fig. 41 PMSG Simulink Model	31
Fig. 42 Wind Conversion model block	32

Fig. 43 Mechancial equation block	33
Fig. 44 PMSG Block	33
Fig. 45 dq Inverse Transform Block	34
Fig. 46 MPC block(P&O)	35
Fig. 47 FOC Block	35
Fig. 48 Converter block	36
Fig. 49 Modulator block	36
Fig. 50 Phase Locked Loop Block	37
Fig. 51. DC bus voltage block	37
Fig. 52 dq Inverse Transform block for Grid	38
Fig. 53 Inverter block	38
Fig. 54 DC bus voltage and Reactive power control block	39
Fig. 55 Active Power and Reactive power to grid Block	40
Fig. 56 R-L Filter block	40

LIST OF TABLES

Table 1 Wind turbine parameters	3
Table 2 value of constant(c)	4
Table 3 Parameters of the PMSG model	32

SUMMARY

The power control of the wind turbine system can be divided with two types depending on the wind speed. The two types are the maximum power control(MPC) and the limited power control. MPC is used for obtaining the maximum power below the rated wind speed. The limited power control is utilized to regulate the rated power above the rated wind speed. Perturbation and Observation(P&O) method and Optimal Tip Speed Ratio(TSR) control are most widely known methods for MPC. Perturbation and Observation(P&O) method is controlled by comparing the present and past values of the rotation speed. However, due to a number of processes, it has slow response time to track a maximum power. Optimal TSR control method requires a wind speed. However, measuring wind speed is not feasible since it is installed some distance away from wind turbines. For this reason, the estimated wind speed has been used for optimal TSR control. But the problem is that estimating the wind speed is related on inverse problems, thus, it is not so simple to solve. Because of these, neural networks or various numerical methods have been used to solve these problems.

In this paper, two new methods are introduced for MPC. The first method is to use a Regression method for estimating the wind speed through the power and rotational speed. And the second method uses the power curve which is a function of the rotation speed and wind speed for estimating the wind speed. Matlab/Simulink is used for simulations. To confirm the performance of proposed methods, 9.4kW PMSG model is used in the two difference wind case(at constant wind speed and at variable wind speed).

제 1 장 서 론

국제에너지기구(IEA) “World Energy Outlook 2013“ 에 따르면 2035년경의 세계 에너지 수요는 30%정도 증가할 것으로 예측하고 있으며 신재생에너지는 전 세계 총 발전량의 1/3 이상을 담당하게 될 것으로 전망하고 있다. 2012년에 세계 풍력 에너지 발전량은 44GW가 증가하여 282GW가 되었고 세계 전력 생산량의 2.3%를 생산하였다. 이것은 2005년에 대비 약 5배 증가한 수치이다. 2012년에는 주로 중국이 12GW 그리고 미국과 EU 두 국가를 합치면 12GW의 풍력 에너지 발전용량이 증가되었다. 또한, 풍력에너지는 New Policies Scenario에 따르면 2011년부터 2035년 까지 연 평균 6%의 성장률을 거둘 것이며 이로 인해 2035년에 2,800TWh의 시간당 생산량을 거둘 것이고 세계 전력 공급의 7.5%에 해당하는 1,130GW를 풍력에너지가 담당하게 된다. Fig. 1은 국가별 풍력 에너지의 설치 용량을 나타낸다[1].

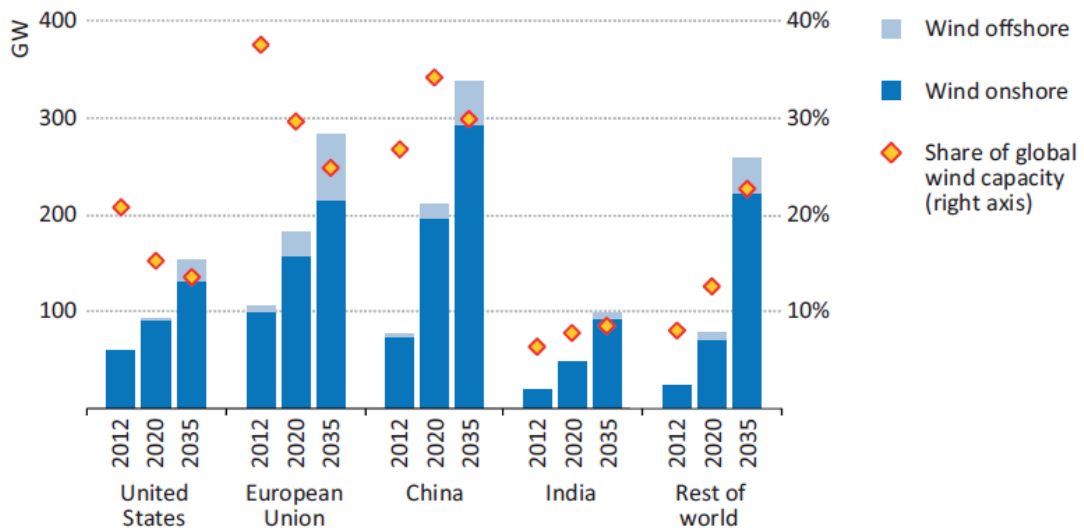


Fig. 1 Installed wind power capacity by region in the New Policies Scenario(IEA)[1]

국내의 경우 2020년 까지 신재생에너지는 총 전력 발전량의 5.2%를 2035년 까지 11%를 생산한다는 계획이며, 이중 풍력에너지는 2020년에 11.3%를 생산하고 2035년에는 6.9%가 상승한 18.2%를 생산한다는 계획이다. 이것은 신재생에너지 중에서 2번째로 많은 비중을 차지하는 것이다[2][3].

또한, 현재 제주도 근처에 위치한 가파도와 전라남도 해안 근처에 위치한 가사도에서는 풍력에너지와 태양광에너지를 이용한 마이크로그리드를 구축할 계획이다[3]. 이처럼 내륙과 떨어져 있는 지역의 전력 수급에서 신재생에너지의 역할이 커지고 있으며 그중 풍력에너지의 역할이 크다는 것을 알 수 있다.

위에서 알 수 있듯이 풍력에너지에 대한 역할이 점차 커지고 있는 상황에서 풍력터빈의 정확한 출력제어가 필요할 것이다. 풍력터빈의 출력제어는 정격풍속 미만에서와 정격풍속 이상에서의 제어로 나뉜다. 우선 정격풍속 미만일 때 최대 출력을 내기 위한 제어가 필요하다. 또한, 정격풍속 이상에서의 제어는 정격 출력 이상의 출력을 내는 것을 방지하기 위하여 출력 제한 제어가 이루어진다. 이 제어는 주로 블레이드 피치각을 제어함으로써 제어가 이루어진다[9]. 최대 출력 방법에는 회전속도와 출력의 과거와 현재 값을 비교하여 최대출력제어를 하는 P&O(Perturbation and Observation)방법이 있다. 하지만 이 방법을 이용하여 최대출력제어를 하기 위해서는 여러 프로세스를 거쳐야 하기 때문에 반응 속도가 느리다. 다른 방법으로 가장 널리 알려진 방법으로는 최적 TSR(Tip Speed Ratio) 제어 방법이 있다. 하지만 이 방식은 주로 측정된 풍속을 이용하여 제어를 하는데 풍속을 측정하기 위한 풍속계는 보통 풍력터빈과 떨어진 거리에 설치되어 있어 풍력터빈에서 정확한 풍속 측정이 불가능하며 이러한 풍속계를 사용함으로써 비용이 증가하고 전체 시스템의 안정성이 떨어질 우려가 있다[4]. 이러한 이유로 최근에는 풍속을 예측하여 최적 TSR 제어 방법에 적용하는 방법이 연구되고 있다. 예를 들어 Bhowmik은 출력계수의 비선형 다항식을 이용하여 수치해석의 반복적인 방법으로 풍속을 예측하였다[5][6]. 그러나 비선형다항식의 근을 구하는 것은 복잡하고 시간을 소비 할 수 있어 시스템의 효율을 떨어뜨릴 수 있다. 또한 Li는 다층 신경망(Multi Layer Neural Network)을 이용하여 풍속을 예측하는 방법을 제안하였다[7]. 다층 신경망은 실시간 비선형 입출력 매핑시스템을 잘 구성하는 것으로 알려져 있다. 그러나 신경망은 학습이 중요한데, 학습은 데이터의 종류에 따라 원하는 목표치만큼 학습이 어려운 경우들도 생긴다.

본 논문에서는 정격 풍속 미만에서의 최대출력제어(MPC)에 대하여 논한다. 측정된 풍속을 이용한 방법이 아닌 예측된 풍속을 이용한 최적 TSR 제어를 위하여 새롭게 풍속 예측하는 방법을 제안하였으며 제안된 방법과 기존의 풍속예측 방법을 이용한 최대출력제어를 시뮬레이션 하여 비교해 보았다.

제 2 장 풍력터빈시스템

2.1 풍력에너지

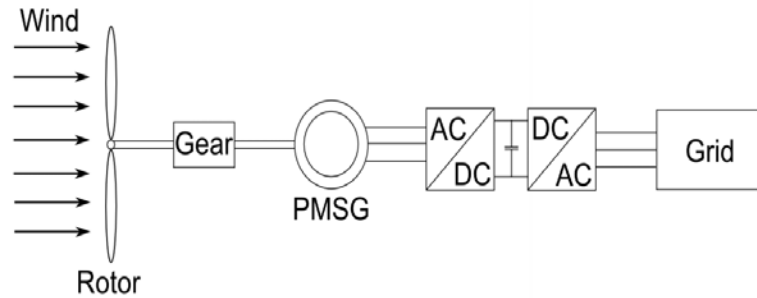


Fig. 2 Wind Turbine System Block

Fig. 2는 풍력터빈 시스템 블록도를 나타낸 그림이다. 로터와 기어박스가 연결되어 있고 기어박스에서 본 논문에서 사용되는 발전기인 PMSG(Permanent Magnet Synchronous Generator)로 축이 연결되어 있다. PMSG는 Back-to-Back 컨버터를 통해 계통으로 연결되어 있다. 풍력터빈의 특성은 [12]을 참조하였으며 풍력 터빈의 특성은 Table 1과 같고, 바람에 의해 얻을 수 있는 에너지(P_w)는 식(2-1)와 같이 나타낼 수 있다[9]-[12].

Table 1 Wind turbine parameters

Parameter	Value	Unit
Rotor radius	2.4	[m]
Gearing	5	[-]
Rated wind Speed	12	[m/s]
Maximum power coefficients	0.411	[-]

$$P_w = \frac{1}{2} \rho A v^3 C_p(\lambda, \beta) \quad (2-1)$$

여기서 ρ 는 공기밀도를 나타내며 A 는 로터의 면적, v 는 풍속, $C_p(\lambda, \beta)$ 는 풍력 터빈의 출력계수를 나타낸다. 풍력터빈의 출력계수는 Betz의 한계에 따르면

59.3% 이상을 넘을 수 없다[8]. 이 출력계수는, 또한, 날개끝단 속도비(λ)와 블레이드 피치각(Blade pitch angel, β)에 대한 식으로 나타낼 수 있으며 그 식은 식(2-2)에 나타내었다[10].

$$C_p(\lambda, \beta) = c_1 \left(\frac{c_2}{\lambda_i} - c_3\beta - c_4\beta^5 - c_6 \right) e^{-\frac{c_7}{\lambda_i}} \quad (2-2)$$

여기서, $\lambda_i = \left[\frac{1}{\lambda + c_8\beta} - \frac{c_9}{\beta^3 + 1} \right]^{-1}$ 이다.

또한, λ 는 블레이드 끝단의 회전속도와 풍속의 비(Tip Speed Ratio, TSR)로써 식(2-3)처럼 나타낸다.

$$\lambda = \frac{R\omega}{v} \quad (2-3)$$

식(2-2)에서 $c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9$ 는 상수로써 [10]의 값들을 참조하였고 그 값은 Table 2 에 나타내었다.

Table 2 value of constant(c)

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9
Heier(1998)	0.5	116	0.4	0	-	5	21	0.08	0.035

식(2-2)과 식(2-3)을 통해 출력계수(C_p)가 블레이드 피치각(β)과 TSR(λ)에 영향을 받는 것을 확인하였다. 풍속을 제어할 수는 없지만 블레이드 피치와 회전속도를 이용하여 C_p 를 제어할 수 있을 것이다. 제어된 C_p 를 이용하여 풍속에 따른 식(2-1)의 최대출력을 얻을 수 있을 것이다.

2.2 최대출력제어

Fig. 3은 풍속에 따른 풍력터빈의 출력 곡선을 나타낸다. 그림에서 Region I는 시동풍속 (v_{cutin})부터 정격풍속(v_{rated})까지의 구간으로써 최대출력을 내도록 제어를 한다. Region II는 정격풍속(v_{rated})부터 차단풍속(v_{cutout})까지를 나타내며 발전

기에 무리가 가지 않게 출력을 제한하는 제어가 이루어진다[4][9][10].

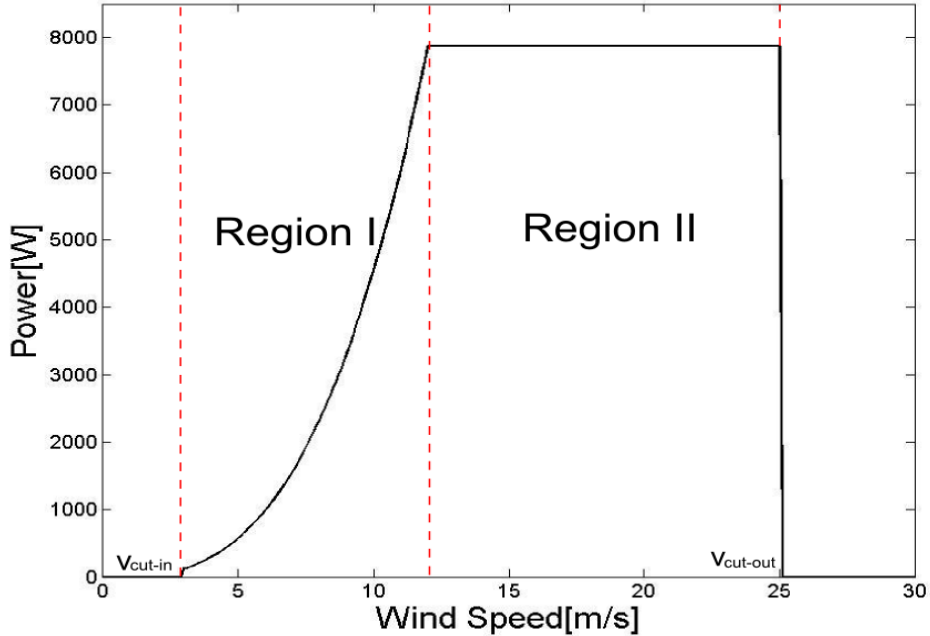


Fig. 3 Power Curve of Wind Turbine

본 논문에서는 Region I에서 최대출력제어를 하기 위하여 사용되는 방법 중 P&O 방법과 최적 TSR 제어방법을 소개한다. P&O 방법은 [4][11][12]에서 소개된 방법으로 출력과 회전속도의 과거와 현재 값을 비교하여 최대출력제어를 하는 방법이다. 하지만 복잡한 프로세스를 반복적으로 수행하기 때문에 반응속도가 느리다는 단점이 있다. 다른 방법으로는 최적 TSR 제어 방법이 있는데, 이 방법은 측정 또는 예측한 풍속을 이용하여 최적 TSR 값으로 유지시켜주기 위하여 회전속도를 조절해주는 제어 방법이다. 풍속을 측정하여 최적 TSR 제어하는 방법이 가장 널리 알려져 있다[4]. 그리고 풍속을 예측하여 최적 TSR 제어 하는 방법은 [5]-[9]에 소개되어있다.

2.2.1 P&O 제어

Fig. 3에서 Region I은 정격풍속 미만이기 때문에 풍력터빈의 정격출력을 얻을

수가 없다. 따라서 이 구간에서는 최대한 많은 출력을 끌어내기 위한 최대출력제어(Maximum Power Control, MPC)가 이루어져야 한다. P&O 제어는 풍속 측정 없이 단순히 출력과 회전속도 측정만을 통하여 MPC 제어를 할 수 있는 방법이다. 때문에, 실제 발전기 제어에 사용되고 있는 추가적인 센서가 필요 없으며, 시스템 파라미터 및 모델 오차에 의한 정상상태 오차가 없어 항상 정확한 MPC 제어를 할 수 있다는 장점을 갖는다. 하지만 정확한 측정을 위하여 많은 시간이 요구된다는 단점을 가지고 있다[4][11][12].

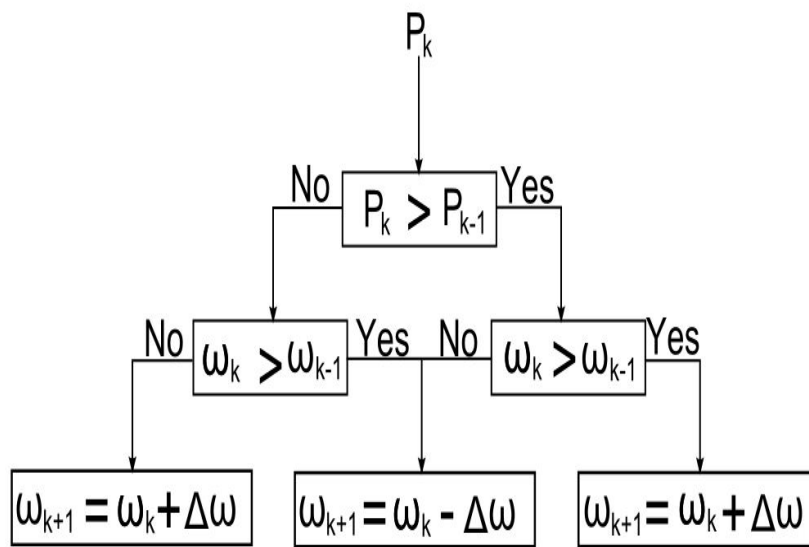


Fig. 4 Control strategy of P&O

P&O의 알고리즘 순서는 다음과 같다.

- i. 현재 출력(P_k)과 현재 회전속도(ω_k)를 측정한다.
- ii. 현재 출력(P_k)과 이전 출력(P_{k-1})과 비교한다.
- iii. 현재 회전속도(ω_k)와 이전 회전속도(ω_{k-1})를 비교한다.
- iv. P_k 와 ω_k 둘다 P_{k-1} 과 ω_{k-1} 보다 크거나 작다면 $\omega_{ref} = \omega_k + \Delta\omega$ 를 한다.
- v. P_k 와 ω_k 중 하나가 과거 값인 P_{k-1} 과 ω_{k-1} 보다 작다면 $\omega_{ref} = \omega_k - \Delta\omega$ 을 한다.
- vi. i부터 반복한다.

위에서 설명한 P&O 제어 전략을 Fig. 4에 나타내었다. 현재 출력(P_k)가 이전 출력(P_{k-1}) 보다 큰 값을 갖는지 비교 하고 다시 현재 회전 속도(ω_k)와 이전 회전 속도(ω_{k-1}),의 크기가 비교된다. 이 결과에 따라 ω_k 에 설정된 회전속도($\Delta\omega$)를 더하거나 빼서 다음 회전속도 ω_{k+1} ,를 제어해준다. 제어된 속도를 통해 출력 값이 변하게 되고 변한 출력 값은 다시 위와 같은 프로세스를 반복하여 최대 출력 값을 찾게 된다.

Fig. 5는 P&O 방법을 사용했을 때 최대출력제어가 되는 과정을 그림으로 나타낸 것이다. 현재 출력과 이전 출력이 비교되고 이전 출력보다 크고 회전속도 역시 이전 회전속도보다 클 경우 회전속도는 설정된 크기만큼 증가하고 이러한 과정은 P_{k-1} 과 P_k 가 같아질 때까지 이 단계를 계속 반복하는 것을 알 수 있다.

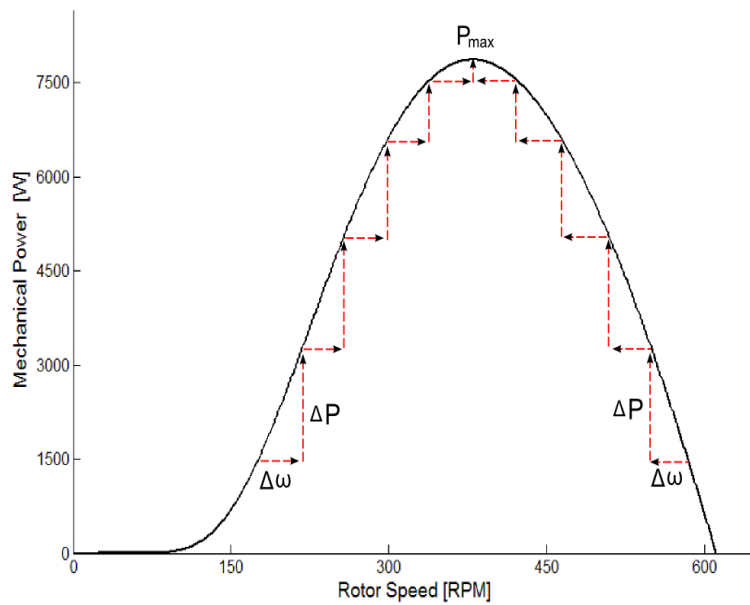


Fig. 5 MPC using P&O method

2.2.2 최적 TSR제어

최적 TSR 제어 방법은 풍력 터빈 제어에서 가장 널리 사용되고 있는 방법이다. Fig. 6은 식(2-2)에서 β 를 0으로 한 후 TSR(λ)의 변화에 따른 출력계수(C_p)를 나타낸 것이다. 이 그림을 보면 λ 에 따라 C_p 가 변하며 λ 가 8일 때 C_p 가 4.11

로 최대값($C_{p_{max}}$)을 갖는 것을 알 수 있다. $C_{p_{max}}$ 일때의 λ 를 λ_{opt} 라고 하고 λ_{opt} 을 유지 시킨다면 최대 출력제어가 가능할 것이다. 만약 λ_{opt} 가 일정하다면 이 값을 유지한다면 풍력터빈에서 최대출력을 낼 수 있을 것이다. 이와 같이 최적 TSR (λ_{opt})을 유지하기 위해 풍속을 측정하여 레퍼런스 회전속도(ω_{ref})를 계산하는 방법을 최적 TSR 방법이라고 한다. λ_{opt} 를 내기 위한 레퍼런스 회전속도(ω_{ref})에 대한 식은 식(2-4)에 나타내었다[4].

$$\omega_{ref} = \frac{\lambda_{opt}}{R}v \quad (2-4)$$

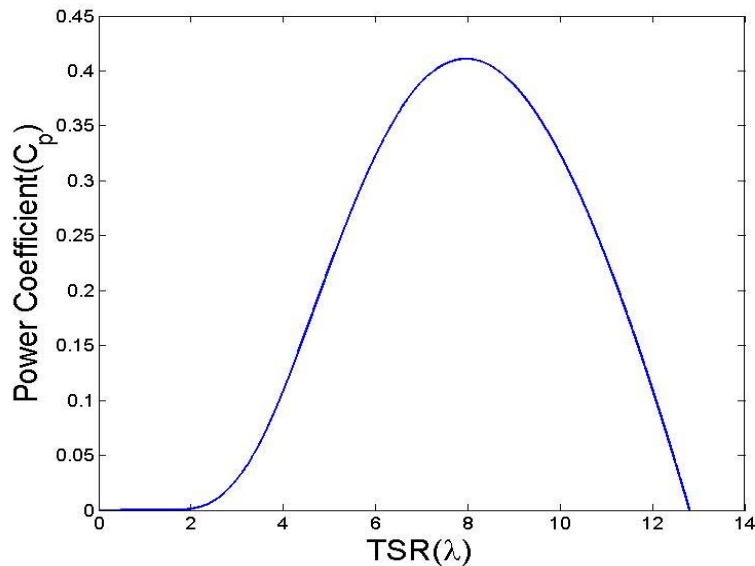


Fig. 6 Power Coefficient(C_p) Versus TSR(λ)

Fig. 7에서 실선은 풍속과 회전속도에 따른 출력을 그래프로 나타낸 것이다. 그리고 점선은 λ 를 최적 값(λ_{opt})으로 유지했을 때의 출력 곡선을 나타낸다. 그림에서 알 수 있듯이 λ_{opt} 일 때의 출력인 점선이 풍속에 따른 출력인 점선의 최대 값을 지나는 것을 확인할 수 있다.

측정된 풍속을 이용한 최적 TSR 제어 방법은 간단해 보이지만 정확한 풍속을 측정하는 것이 힘들며 풍력발전 시스템의 비용이 증가된다는 단점이 있다[4]. 하지만 풍속 예측을 이용하여 예측된 풍속을 최적 TSR 제어 방법에 적용한다면 이러한 단점을 보완할 수 있어 이러한 방법들이 연구되고 있다.

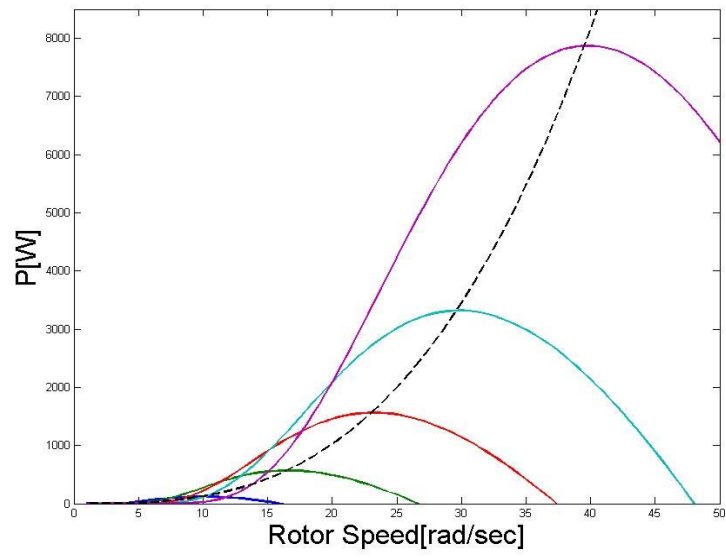


Fig. 7 Power Curve using Optimal TSR method

제 3 장 풍속 예측을 이용한 최대출력제어

앞 장에서 풍력터빈시스템과 최대출력제어 방법에 대해서 알아보았다. 회전속도와 출력을 이용하여 최대출력제어를 하는 방법인 P&O방법을 알아보았다. 하지만 P&O 방법은 여러 단계의 프로세스를 거쳐야하기 때문에 반응속도가 느리다는 단점이 있다. 그리고 가장 널리 사용되는 방법인 최적 TSR 제어 방법을 알아 보았다. 이 방법은 풍속을 알아야 최대출력제어가 가능한 방법이기 때문에 측정된 풍속 또는 예측된 풍속을 가지고 최적 TSR 제어가 가능할 것이다. 본 장에서는 최적 TSR 제어를 하기 위한 풍속 예측 방법에 대해서 소개한다.

3.1 풍속예측

식(3-1)은 식(2-3)을 예측된 풍속(v_{est})에 대한 식으로 나타낸 것이다. 이 식을 통해서 회전속도(ω)와 TSR(λ)을 알면 풍속을 예측할 수 있다.

$$v_{est} = \frac{R \cdot \omega}{\lambda} \quad (3-1)$$

C_p 와 λ 에 대한 식인 식(2-2)와 식(2-3)를 출력 식인 식(2-1)에 대입하면 식(3-2)처럼 나타낼 수 있다. 이 식을 보면 출력(P)과 회전속도(ω), TSR(λ)만이 남게 되며 P 와 ω 를 측정하여 λ 를 구할 수 있다.

$$P = 0.52 \cdot \rho \cdot \pi \cdot R^5 \cdot \omega^3 \cdot \frac{1}{\lambda^3} \cdot \left(\frac{116}{\lambda} - 9.06 \right) \cdot e^{-\frac{21}{\lambda}} \quad (3-2)$$

하지만, 식(3-1)을 보면 식이 난해하여 쉽게 λ 의 근을 찾을 수 없다. 따라서 [5]와 [6]에서는 뉴턴-랩슨 방법이나 바이섹션 방법 같은 반복 연산을 통해 λ 를 찾아 풍속을 예측하는 방법을 소개하였다.

이와 같이 수치해석을 이용한 방법 외에 다른 방법으로는 신경망을 이용한 풍속예측 방법이 있다. 인공신경망은 인간의 두뇌의 구조와 동작방식을 모델로 하여 만들어진 기법으로 다양한 신경망 구조 방법이 있으며 [7]에서는 이중 일반적

으로 많이 사용되고 있는 오류 역전파(Back-Propagation) 알고리즘을 사용한 풍속 예측방법을 소개하고 있다. 이외에도, [8][9]에서 신경회로망을 이용한 풍속예측 방법과 이를 이용한 최대출력 제어가 소개되어 있다.

3.2 첫 번째 제안방법: 리그레션을 이용한 풍속예측

본 논문에서 첫 번째 제안 방법은 식(3-1)를 간단한 방정식으로 근사화하여 λ 를 쉽고 빠르게 구하는 방법이다. 간단한 방정식으로 근사화하기 위하여 식(3-1)에서 λ 를 $\frac{1}{x}$ 로 치환하고 λ 이외의 상수들을 P 로 이동시켜 식(3-3)처럼 x 의 함수로 표현하였다.

$$x^3 \cdot (116x - 9.06) \cdot e^{-21x} - \frac{P}{0.52 \cdot \rho \cdot \pi \cdot R^5 \cdot \omega^3} = 0 \quad (3-3)$$

식(3-3)에서 x 에 대한 부분을 $h(x)$ 로 하고 출력과 회전속도에 대한 부분을 $C(P, \omega)$ 로 나누면 다음과 같이 표현할 수 있다.

$$h(x) + C(P, \omega) = 0 \quad (3-4)$$

$$h(x) = x^3 \cdot (116 \cdot x - 9.06)e^{-21 \cdot x} \quad (3-5)$$

$$C(P, \omega) = -\frac{P}{0.52 \cdot \rho \cdot \pi \cdot R^5 \cdot \omega^3} \quad (3-6)$$

식(3-5)는 x 에 대한 식이며 Fig. 8에 x 에 따른 $h(x)$ 를 그래프로 나타내었다. 식(3-6)는 출력 값 P 와 회전속도 ω 를 알 수 있으면 결정되는 상수 값이다. 출력 계수(C_p)와 $TSR(\lambda)$ 에 대한 그래프인 Fig. 4에서 알 수 있듯이 λ 가 2~13 영역 사이에서 출력이 발생한다. 따라서 Fig. 8에서 중요한 x 의 범위는 대략 0.08~0.45라 할 수 있다. 즉, 이 구간을 2차 또는 3차식으로 간단히 나타낼 수 있다면 풍속을 예측하여 최대출력제어를 할 수 있을 것이다. 본 논문에서는 식을 근사화시키기 위하여 리그레션 방법을 사용하였다. 식(3-7)은 리그레션 방법을 이용한 2차방정식을 나타낸다.

$$ax^2 + bx + c + C(P, \omega_r) = 0 \quad (3-7)$$

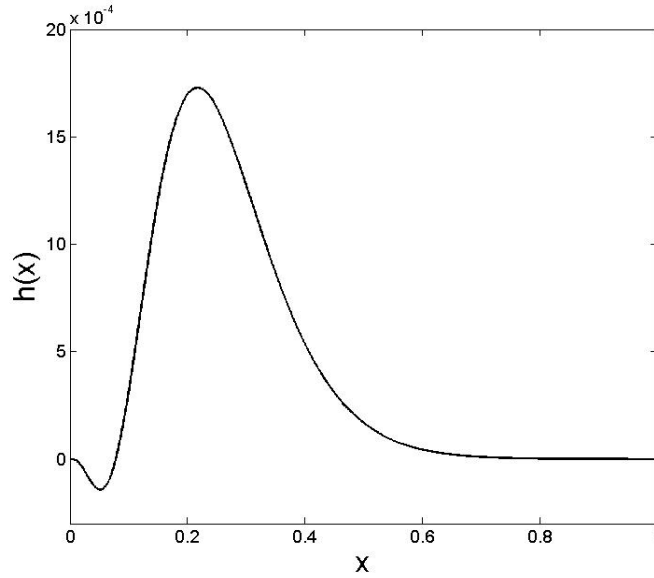


Fig. 8 Plot for $h(x)$

여기서 a, b, c 는 $h(x)$ 를 정해진 구간에서 2차 방정식으로 리그레션하여 결정되는 값이다. 즉, 식(3-6)의 $C(P, w_r)$ 만 결정되면 바로 2차방정식을 풀어서 x 의 값을 구할 수 있다. 방정식의 근은 2개 얻어지는 데, Fig. 8에서 알 수 있듯이 근은 모두 양수이며 유효한 근은 둘 중 작은 것임을 알 수 있다.

리그레션은 효율적인 2차곡선을 찾기 위하여 다양한 구간에서 실험하였다. 실험결과 x 의 범위를 0.08~0.2구간에서 $h(x)$ 를 리그레션한 것이 가장 효율적임을 알았다. 그 이유는 시뮬레이션 그림들에서 알 수 있듯이 시스템이 가동되고 바로 λ 가 5이상을 유지하기 때문이다. 고로 λ 는 5~13구간에서 거의 작동하므로 x 의 범위를 0.08~0.2구간으로 하는 것이 타당하다고 여겨진다. Fig. 9은 이 구간에서 리그레션한 2차곡선을 보여주고 근사식은 다음과 같다.

$$h_{reg}(x) = -0.0848x^2 + 0.0382x - 0.0025 \quad (3-8)$$

Fig. 10(a)는 $h(x)$ 곡선과 오차가 거의 없고 x 가 0.2 이후에 모습이 서로 다른 4개의 2차 곡선을 $h(x)$ 곡선과 비교한 그림이다. Fig. 10(b)는 서로 다른 4개의 곡선들을 이용하여 풍속 예측 결과인 데 거의 결과가 같음을 알 수 있다. 따라서 x 축의 중요한 구간은 0.08~0.2인 것을 알 수 있으며 이는 λ 구간이 5~13임을 의미한다.

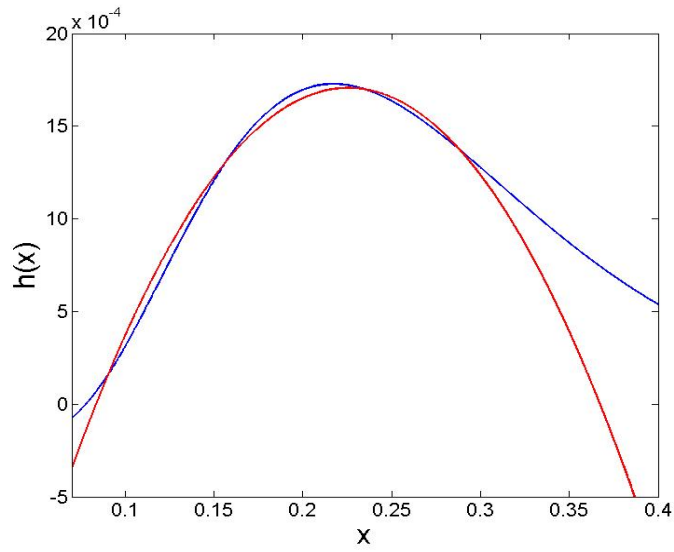


Fig. 9 $h(x)$ and approximated equation $h_{reg}(x)$ by regression method

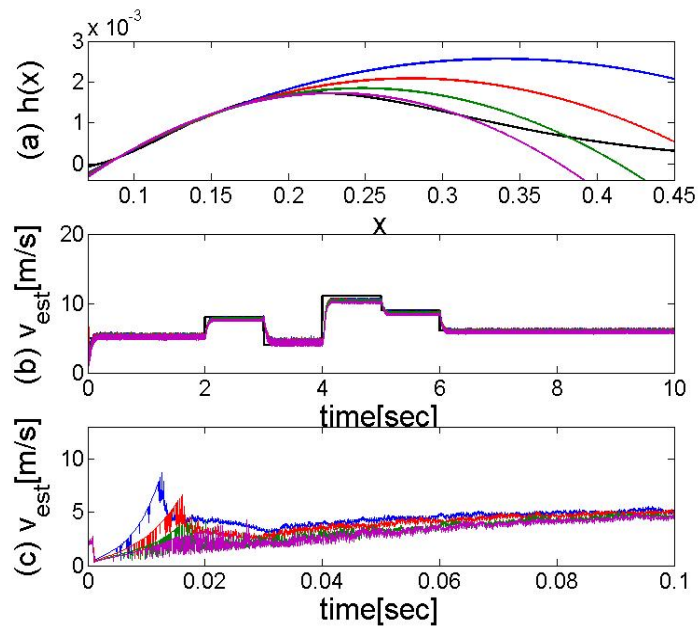


Fig. 10 (a) Plot for $h(x)$ and $h_{reg}(x)$, (b) Estimated wind speed by $h_{reg}(x)$ curves, (c) Expand (b) plot between 0~0.1 sec

Fig. 10(c)는 Fig. 10(b)에서 0초~0.1초 구간을 확대한 그림이다. Fig. 10(c)를

보면 0초~0.1초 구간에서 오버슈트가 발생하며 x 축 0.2 이후 구간에서 $h(x)$ 곡선과 오차가 클수록 오버슈트가 크다는 것을 알 수 있다. 하지만 정상 상태로 돌아오는 시간이 약 0.01초로 매우 짧기 때문에 x 축 0.2 구간 이후의 영역은 큰 영향을 주지 않은 것을 알 수 있다.

3.3 두 번째 제안방법: 출력과 회전속도를 이용한 풍속예측

두 번째로 제안하는 방법은 출력과 회전속도만을 이용하지만 측정된 출력과 측정된 회전속도를 이용하여 계산한 출력을 비교하여 풍속을 예측하는 방법이다. 식(2-1), 식(2-2), 식(2-3)을 이용하여 출력(P)과 회전속도(ω_r)와 풍속의 관계를 이용하여 풍속(v)을 예측하는 방법이다.

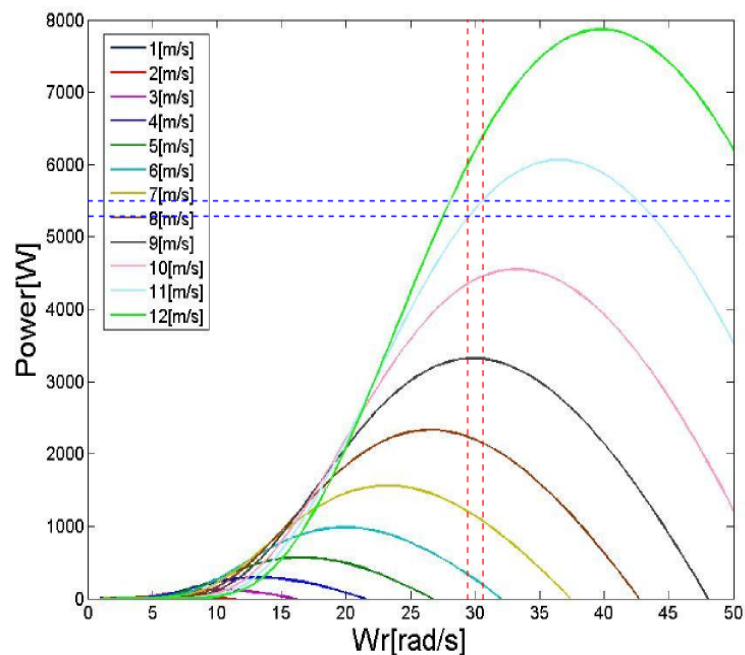


Fig. 11 w_r - P curves depending on v

Fig. 11은 식(2-1)을 이용하여 w_r 과 v 에 따른 P 의 변화를 그래프로 나타낸 것이다. 그림을 보면 출력 값과 회전속도를 알면 풍속을 예측할 수 있다는 것을 알 수 있다. 예를 들어 측정된 출력과 회전속도가 각각 5500[W], 30[rad/s]이라면, 출력이 5500[W]일 때 이와 연관된 풍속은 11[m/s]와 12[m/s]이지만 회전속도가 30[rad/s]이기 때문에 풍속이 11[m/s]

라는 것을 예측할 수 있다. 이러한 특성을 이용하여 새로운 알고리즘을 제안하였다.

측정된 출력과 회전속도를 이용한 제안된 알고리즘은 다음과 같다.

- i). 측정된 출력(P_m)과 회전속도(ω_m)를 입력한다.
- ii). 초기치 풍속을 시동풍속(v_{ini})으로 설정한다.
- iii). 측정된 회전속도와 설정된 풍속을 식(2-3)에 대입하여 λ 을 구한다.
- iv). 피치각 β 를 0으로 하고, 계산된 λ 와 풍속으로 식(2-1)을 사용하여 출력을 구한다.
- v). iv)에서 구한 출력(P_c)과 측정된 출력(P_m)을 비교하여 측정된 출력이 크면 풍속을 증가시켜($v = v + \Delta v$) iii)과정부터 다시 반복한다.
- vi). iv)에서 구한 출력(P_c)이 측정된 출력(P_m)보다 클 경우 예측 풍속(v_{est})을 $v_{est} = v_{k-1} + \frac{\Delta P_m}{\Delta P_c} \Delta v$ 를 통해 구한다. 여기서, $v_{k-1} = v - \Delta v$ 이고 ΔP_m 과 ΔP_c 는 현재 측정된 출력과 현재 계산된 출력에서 이전 계산된 출력 값을 뺀 값이다.
- vii). 레퍼런스 회전속도(ω_{ref})를 $\omega_{ref} = \frac{8}{R} v$ 이용하여 구한다.

Fig. 12은 위에서 설명한 제어 전략을 그림으로 나타낸 것이다.

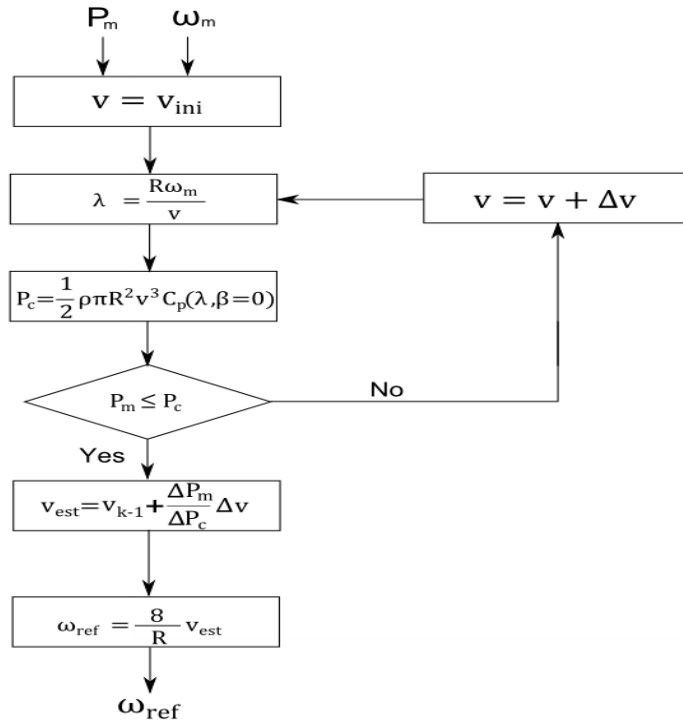


Fig. 12 Control strategy for proposed method

제 4 장 시뮬레이션 및 결과

시뮬레이션에서 사용된 발전기 모델은 정격 9.4kW의 소형 PMSG 모델을 사용하였다. 시뮬레이션은 Matlab/Simulink를 사용하였으며 두 개의 바람 모델을 가지고 풍속 측정 없이 출력제어가 가능한 기존의 방법인 P&O와 뉴턴-랩슨 방정식을 이용하여 풍속예측을 하여 최대출력제어를 하는 방법과 본 논문에서 제안한 리그레션과 새로운 알고리즘을 이용하여 풍속을 예측하고 이를 갖고 최대출력 제어 시뮬레이션 결과를 비교해보았다.

4.1 Matlab/Simulink 모델

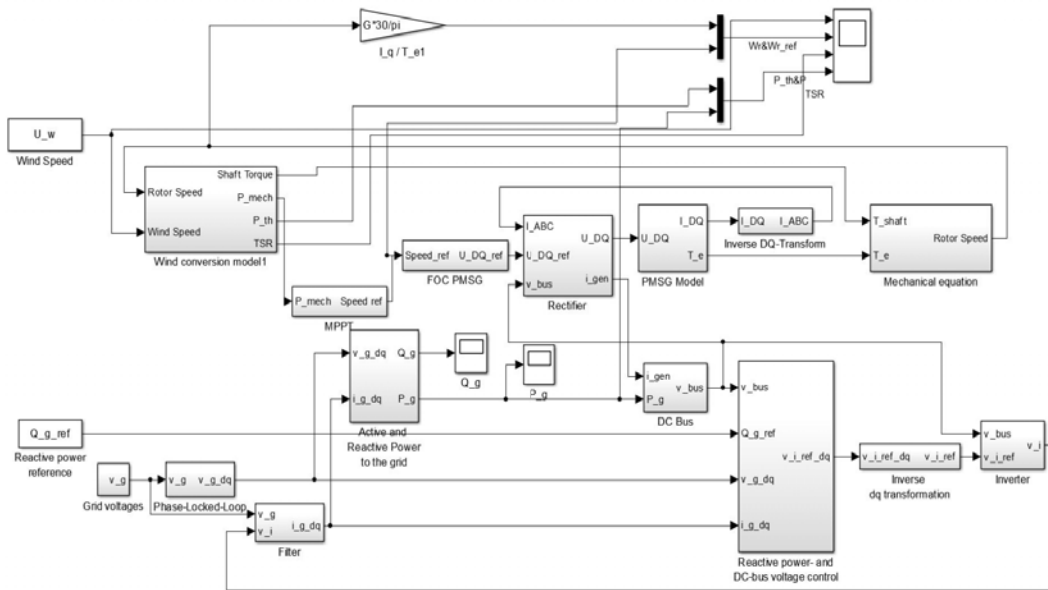


Fig. 13 PMSG Simulink Model

Fig. 13은 [12]에서 모델링한 PMSG의 Matlab/Simulink 모델이다. 본 논문에서의 시뮬레이션 모델은 기계적 모델과 전기적 모델을 Simulink를 이용하여 설계한 것이다. 이 모델을 이용하여 풍력 터빈 축의 최대출력 제어와 계통 축 제어인 DC 부스 전압과 무효 전력 제어를 시뮬레이션 할 수 있다. 블록 내부의 자세한

사항은 Appendix A에 기술하였다.

4.1.1 기존 MPC Block

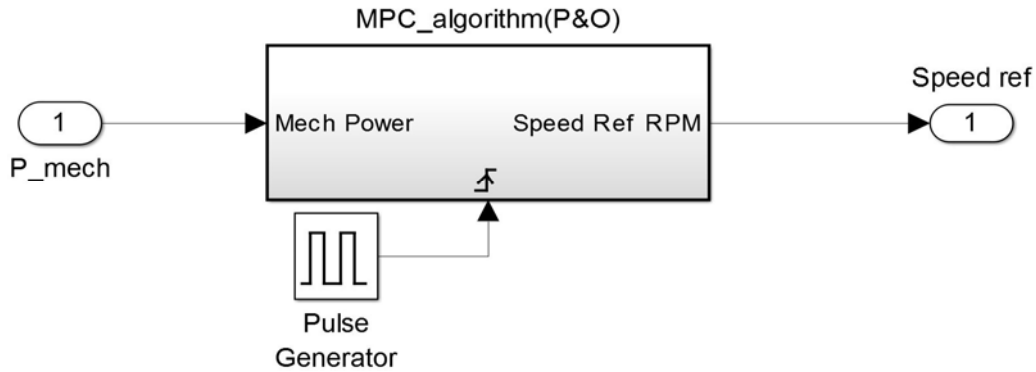


Fig. 14 MPC block(P&O)

본 논문에서 제안한 방법을 시뮬레이션 하기 위해 수정된 부분은 MPC 블록이다. Fig. 14은 기존의 MPC 블록 내부를 나타낸다. 이 블록은 P&O 방법을 이용한 것으로 기계적 출력(P_{mech})를 입력으로 받아 설정된 시간마다 P&O 알고리즘에 따라 설정된 값 만큼 회전 속도를 증가 또는 감소시킨 레퍼런스 회전속도(ω_{ref})를 출력으로 내보낸다.

4.1.2 제안된 풍속예측 MPC Block

풍속 예측을 위하여 Fig. 15와 같은 모델을 설계하여 Fig. 14의 MPC 블록을 대체하였다. 출력과 회전속도를 이용하여 Matlab Function 블록을 통해 TSR을 구하였고 식(3-2)을 이용해 풍속을 예측하였다. Matlab Function 블록은 Matlab 코드를 작성하는 블록이다. 풍속 예측을 위해 뉴턴-랩슨, 리그레션, 제안된 알고리즘에 대한 코드가 이 Matlab Function 블록에 코딩되었다. 마지막으로 예측한 풍속을 최적 TSR 제어 방법에 이용해 ω_{ref} 를 계산한다.

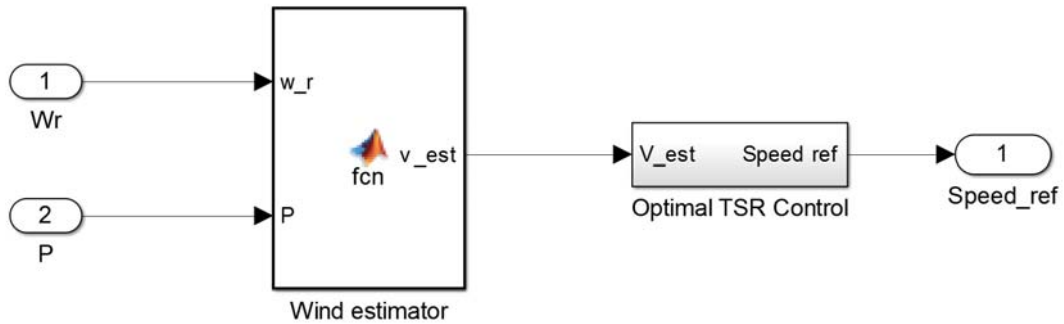


Fig. 15 Wind Estimator(Proposed Method)

4.2 시뮬레이션 결과

앞 절에서 설명된 시뮬레이션 모델을 이용하여 P&O 방법과 뉴턴랩슨 방법 그리고 제안된 방법인 리그레션 방법과 제안된 알고리즘을 이용한 방법을 시뮬레이션 하여 결과를 비교하였다.

4.2.1 고정 풍속에서의 시뮬레이션 결과

Fig. 16-Fig. 23은 풍속이 일정할 때 최대 출력제어를 시뮬레이션한 결과이다. 그림에서 (a), (b), (c)는 풍속, 회전속도, 출력을 각각 나타낸다. 또한 (b)에서 파란선은 회전속도를 붉은선은 레퍼런스 회전속도를 나타낸다. Fig. 16은 P&O제어의 시뮬레이션 결과를 나타내며 Fig. 17은 그 결과의 과도영역을 확대한 그림이다. 그림을 보면 최대 출력을 내기까지 약 1초가 걸리는 것을 확인할 수 있다. 또한 Fig. 16(b)를 보면 최대 출력에 도달했지만 프로세스가 수행되어 레퍼런스 회전속도가 바뀌고 있는 것을 확인할 수 있다. Fig. 17(b)를 보면 과도영역에서 레퍼런스 회전속도가 계단 형식으로 서서히 증가하고 있으며 오버슈트가 거의 없음을 알 수 있다.

Fig. 18는 뉴턴랩슨 방법을 이용한 제어 방법의 시뮬레이션 결과이고 Fig. 19는 과도영역을 확대한 그림이다. 반응속도가 매우 빠른 것을 그림을 통해 확인할 수 있지만 과도상태에서의 오버슈트가 상당히 높다는 것을 확인할 수 있다.

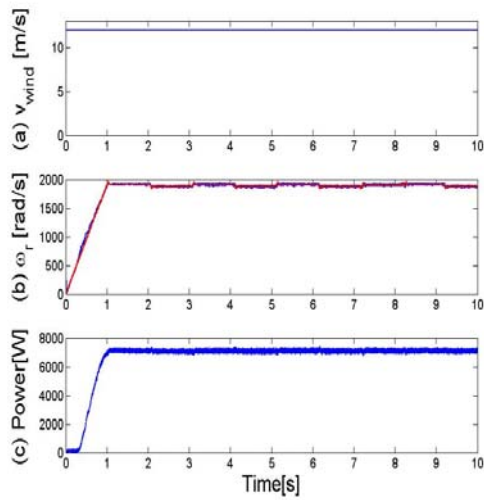


Fig. 16 Simulation result at constant wind speed(P&O Method)

(a) Wind Speed (b) Rotate speed
(c)Power

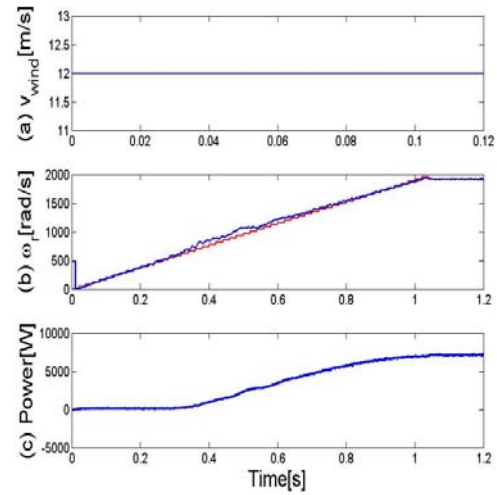


Fig. 17 Enlarged plot between 0~1.2[sec] of Fig. 16

(a) Wind Speed (b) Rotate speed
(c)Power

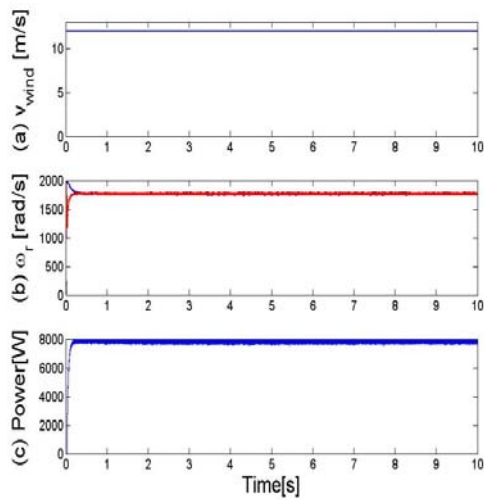


Fig. 18 Simulation result at constant wind speed(Newton-Raphson Method)

(a) Wind Speed (b) Rotate speed
(c)Power

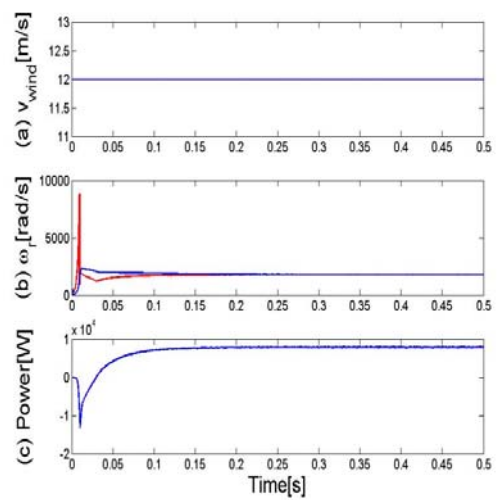


Fig. 19 Simulation result at constant wind speed(Newton-Raphson Method)

(a) Wind Speed (b) Rotate speed
(c)Power

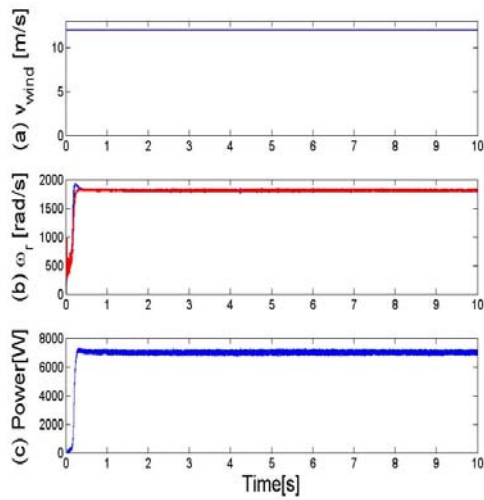


Fig. 20 Simulation result at constant wind speed(Regression Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

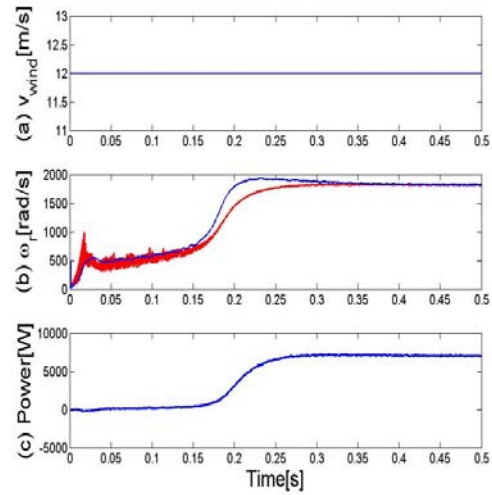


Fig. 21 Enlarged plot between 0~0.5[sec] of Fig. 20
 (a) Wind Speed (b) Rotate speed
 (c)Power

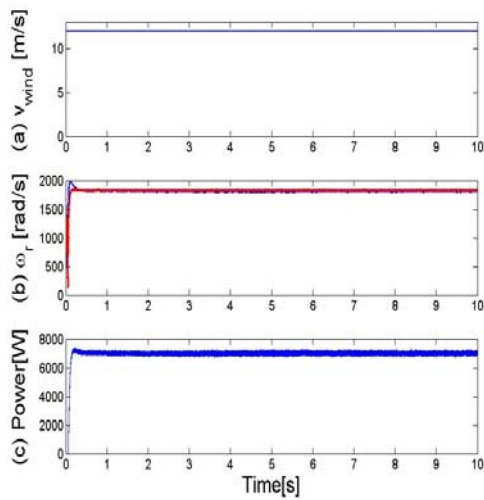


Fig. 22 Simulation result at constant wind speed(Proposed Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

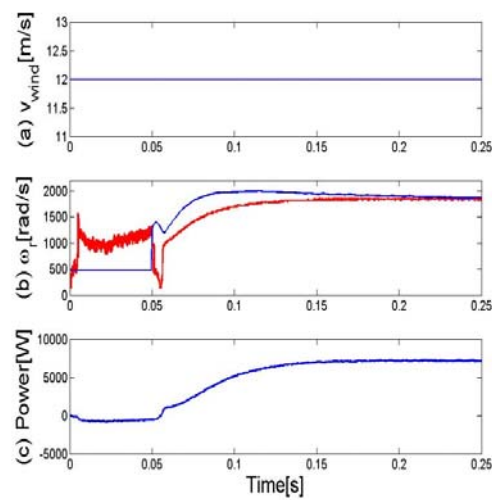


Fig. 23 Enlarged plot between 0~0.25[sec] of Fig. 22
 (a) Wind Speed (b) Rotate speed
 (c)Power

Fig. 20과 Fig. 21은 리그레션 방법을 이용한 최대출력제어 시뮬레이션 결과와 과도영역을 확대한 모습을 각각 나타낸다. 뉴턴랩슨 방법에 비해 최대출력까지 도달하는 시간이 늦지만 과도상태에서의 오버슈트가 작다는 것을 확인할 수 있다.

Fig. 22과 Fig. 23는 제안된 알고리즘을 이용한 최대출력제어 시뮬레이션 결과이다. 반응속도가 빠르지만 0.1초 이전의 상태에서 레퍼런스 회전속도가 순간 하락한 후 다시 정상상태로 가는 것을 확인할 수 있다.

4.2.2 가변 풍속에서의 시뮬레이션 결과

(1) 급격히 변하는 바람에서의 시뮬레이션 결과

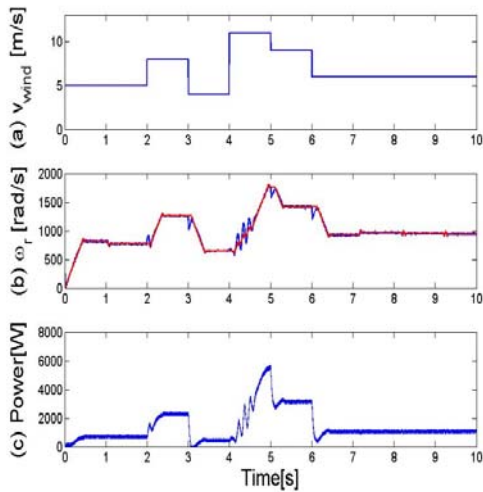


Fig. 24 Simulation result at dramatically varying wind speed (P&O Method)
(a) Wind Speed (b) Rotate speed
(c)Power

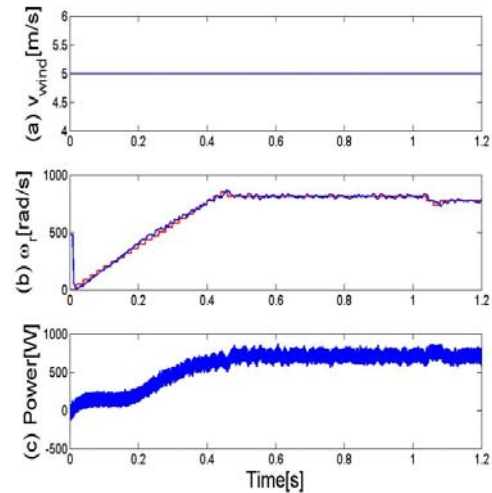


Fig. 25 Enlarged plot between 0~1.2[sec] of Fig. 24
(a) Wind Speed (b) Rotate speed
(c)Power

급격히 바람이 변할 때를 가정하여 시뮬레이션을 해보았다. Fig. 24과 Fig 25은 P&O 방법을 이용한 시뮬레이션 결과와 과도영역을 확대한 그림을 각각 나타낸다. P&O 방법은 풍속의 변화에 즉각적인 대응을 못하는 것을 확인할 수 있다. Fig. 24(b)에서 붉은 선은 레퍼런스 회전속도를 파란 선은 회전속도를 나타내는데 풍속의 변화에 레퍼런스 회전속도가 즉각적인 반응을 못하여 Fig. 24(c)의 출력이 제대로 제어가 안 되는 것을 확인할 수 있다.

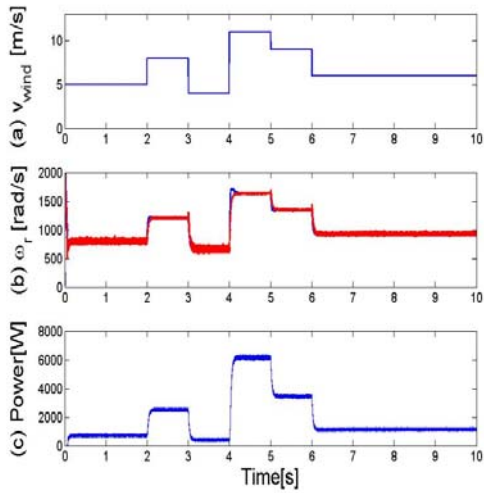


Fig. 26 Simulation result at dramatically varying wind speed(Newton-Raphson Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

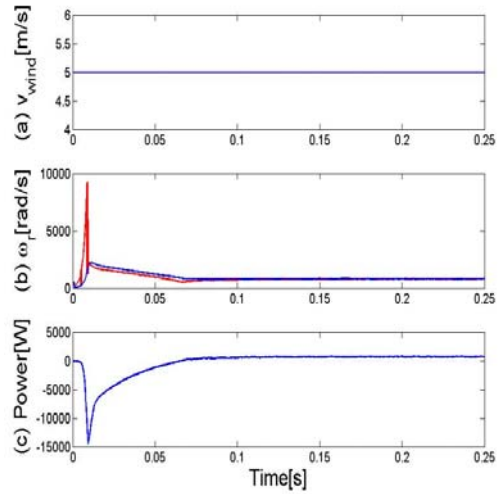


Fig. 27 Enlarged plot between 0~0.25[sec] of Fig. 26
 (a) Wind Speed (b) Rotate speed
 (c)Power

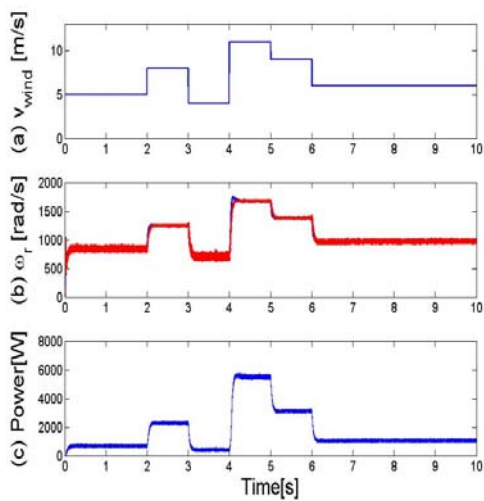


Fig. 28 Simulation result at dramatically varying wind speed(Regression Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

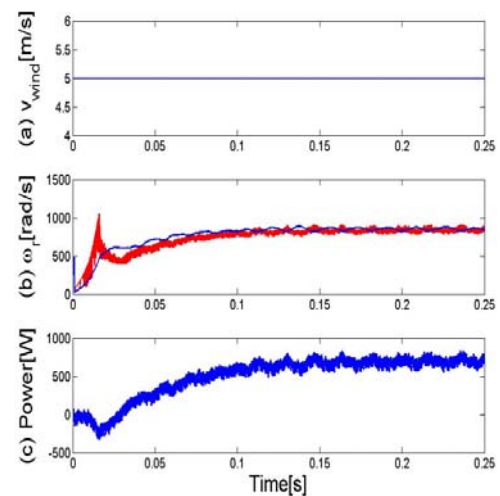


Fig. 29 Enlarged plot between 0~0.25[sec] of Fig. 28
 (a) Wind Speed (b) Rotate speed
 (c)Power

Fig. 26와 Fig. 27은 뉴턴-랩슨 방법을 이용한 시뮬레이션 결과와 과도영역을 나타낸다. 풍속의 변화에 즉각적으로 반응을 하지만 초기 과도상태일 때 큰 오버슈트가 발생하는 것을 확인할 수 있다. 또한, 풍속이 급격히 변할 때에는 오버슈트가 발생하지만 큰 크지 않다는 것을 알 수 있다. Fig. 28와 Fig. 29는 리그레션 방법이 시뮬레이션 결과와 과도영역을 그리고 Fig. 30과 Fig. 31은 제안된 알고리즘을 이용한 방법의 시뮬레이션 결과와 과도영역을 각각 나타낸다. 그림을 보면 풍속의 변화에 즉각적으로 반응을 하며 과도상태에서도 뉴턴-랩슨 방법에 비해 훨씬 안정적인 것을 확인할 수 있다. 급격히 풍속이 변하는 상황에서는 뉴턴-랩슨 방법과 비슷한 수준의 오버슈트가 발생하고 있다.

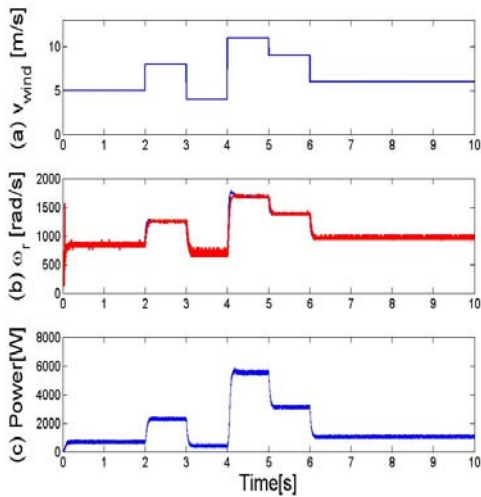


Fig. 30 Simulation result at dramatically varying wind speed(Proposed Method)

(a) Wind Speed (b) Rotate speed
(c)Power

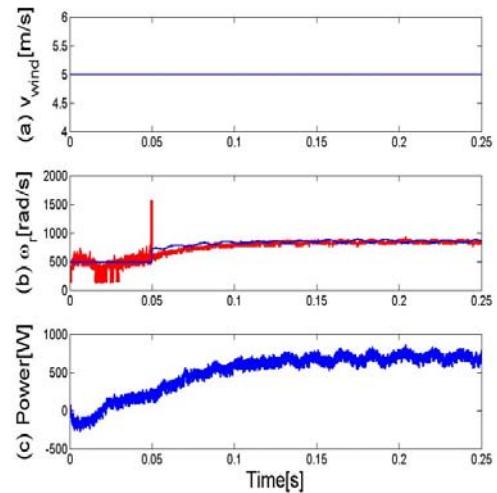


Fig. 31 Enlarged plot between 0~0.25[sec] of Fig. 30

(a) Wind Speed (b) Rotate speed
(c)Power

(2) 서서히 변하는 바람에서의 시뮬레이션 결과

마지막으로 바람이 서서히 변할 때를 가정하여 시뮬레이션을 하였다. Fig. 32과 Fig. 33는 P&O방법을 이용한 시뮬레이션 결과와 과도영역을 각각 나타낸다. 그림에서 볼 수 있듯이 풍속의 변화에 즉각적인 대응을 못하고 있다. Fig. 34과 Fig.

35는 뉴턴-랩슨 방법을 이용한 시뮬레이션 결과와 그 과도영역을 나타낸다. 풍속의 변화에 즉각적으로 반응을 하지만 이 결과 역시 초기 과도상태일 때 큰 오버슈트가 발생하는 것을 확인할 수 있다. 하지만, 급격히 변하는 풍속에서와는 달리 서서히 변하는 풍속에서는 오버슈트가 나타나지 않았다. Fig. 36과 Fig. 37은 리그레션 방법의 시뮬레이션 결과와 과도영역을 그리고 Fig. 38와 Fig. 39는 제안된 알고리즘을 이용한 방법의 시뮬레이션 결과와 과도영역을 각각 나타낸다. 이 결과 역시 풍속의 변화에 즉각적으로 반응을 하며 과도상태에서도 뉴턴-랩슨 방법에 비해 안정적이며 이후의 영역에서 비슷하게 제어가 되고 있음을 확인할 수 있다.

3가지 유형의 시뮬레이션 결과를 통해서 제안된 방법이 P&O 방법보다 반응속도가 빠르다는 것을 확인할 수 있었다. 하지만 근소하게나마 뉴턴-랩슨 방법보다 반응속도가 느린 것을 고정풍속에서의 시뮬레이션 결과를 통해 알 수 있었다.

그러나 과도영역을 비교하였을 때 뉴턴-랩슨 방법의 시뮬레이션 결과 오버슈트가 상당히 높은 반면 제안된 방법은 낮은 것을 확인할 수 있었는데 이는 뉴턴-랩슨 방법에서 처음에 풍속을 예측하지 못하고 프로세스만 반복을 하기 때문이다.

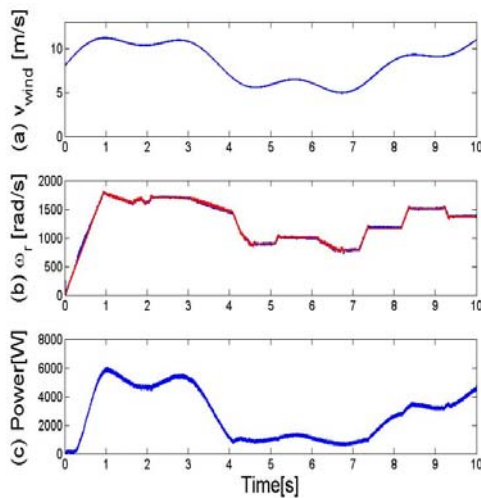


Fig. 32 Simulation result at gradually varying wind speed(P&O Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

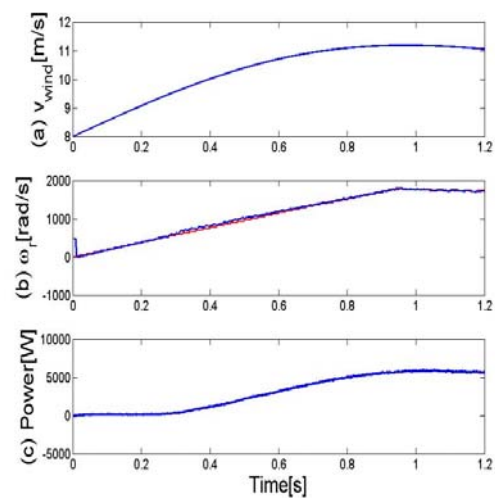


Fig. 33 Enlarged plot between 0~1.2[sec] of Fig. 32
 (a) Wind Speed (b) Rotate speed
 (c)Power

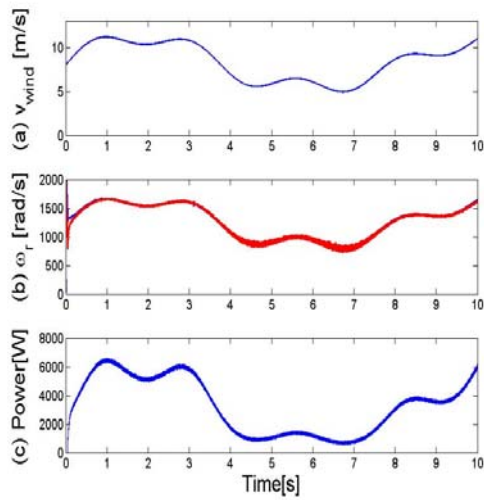


Fig. 34 Simulation result at gradually varying wind speed(Newton-Raphson Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

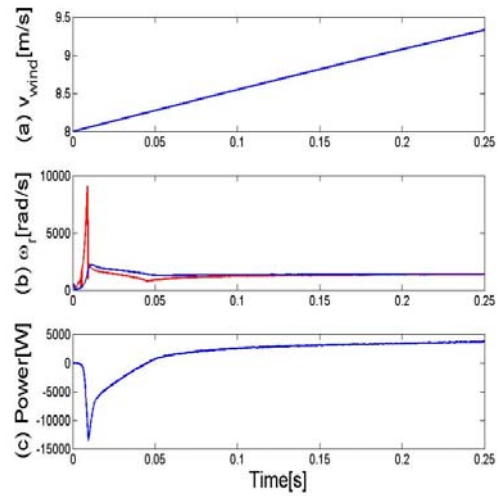


Fig. 35 Enlarged plot between 0~0.25[sec] of Fig. 34
 (a) Wind Speed (b) Rotate speed
 (c)Power

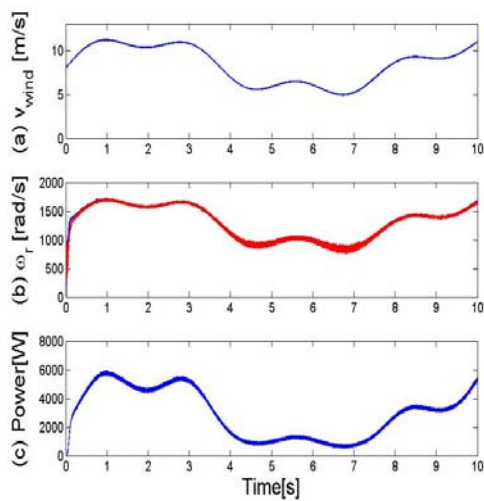


Fig. 36 Simulation result at gradually varying wind speed(Regression Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

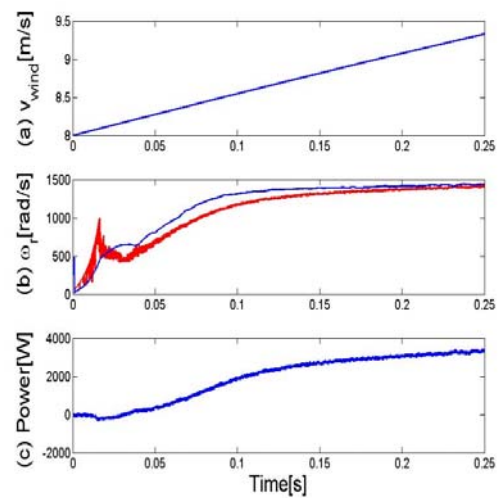


Fig. 37 Enlarged plot between 0~0.25[sec] of Fig. 36
 (a) Wind Speed (b) Rotate speed
 (c)Power

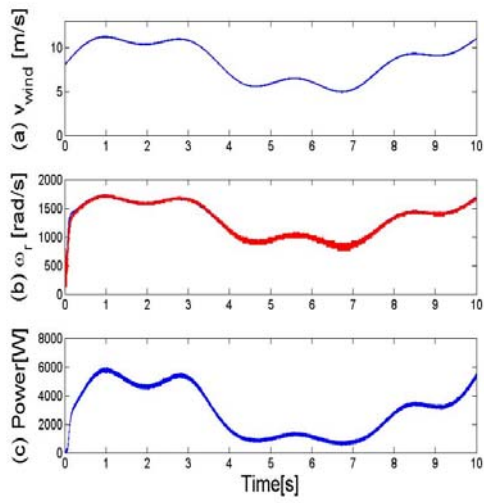


Fig. 38 Simulation result at gradually varying wind speed(Proposed Method)
 (a) Wind Speed (b) Rotate speed
 (c)Power

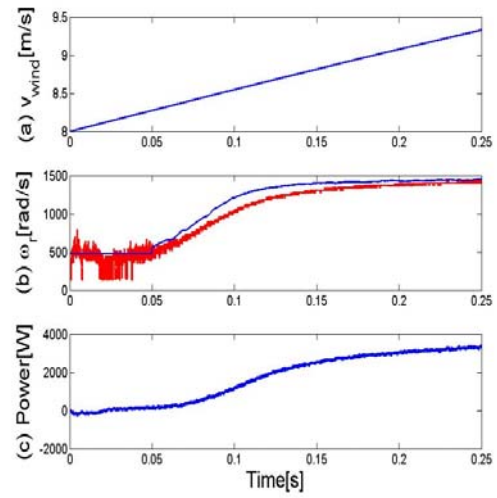


Fig. 39 Enlarged plot between 0~0.25[sec] of Fig. 38
 (a) Wind Speed (b) Rotate speed
 (c)Power

제 5 장 결 론

신재생에너지의 생산량이 전 세계적으로 증가하고 있으며 이중 풍력에너지의 중요도가 높아지고 있다. 따라서 풍력발전의 효율을 높이기 위하여 터빈 제어에 관한 연구가 활발히 진행되고 있다. 풍력터빈의 출력제어는 정격풍속 미만에서와 정격풍속 이상에서의 제어로 나뉜다. 우선 정격풍속 미만일 때 최대출력제어를 하며 정격풍속 이상에서는 출력제한제어가 이루어진다. 이중 본 논문에서는 정격풍속 미만일 때 최대출력제어를 위한 풍속예측에 대한 연구를 하였다.

최대출력제어 방법으로 P&O 방법과 최적 TSR제어 방법이 사용되고 있다. P&O 방법은 출력과 회전속도의 현재 값과 이전 값을 비교하며 회전속도를 증가시켜 최대출력제어를 하는 방법인데 정밀한 최대출력제어가 가능했지만 반응속도가 느려 바람의 변화에 즉각적인 대응을 하지 못한다. 측정된 풍속을 이용한 최적 TSR제어 방법은 그 방법이 간단해 보이지만 실제로 풍속계는 풍력터빈과 거리가 있는 곳에 설치되어 있어서 정확한 측정이 힘들어 정밀한 최대출력이 불가능하다. 때문에 예측된 풍속을 최적 TSR 제어방법에 이용하는 방법이 연구되고 있다. 기존에는 뉴턴-랩슨이나 바이섹션 같이 반복 연산을 이용해 풍속의 근사치를 찾는 방법([5],[6])과 신경회로망을 이용해 풍속을 예측하는 방법([7],[8])이 소개되어있다. 하지만 풍속을 예측하려면 복잡한 식을 풀어야 하기 때문에 여러 프로세스를 거쳐야 한다. 본 논문에서는 복잡한 식을 근사화하거나 알고리즘을 간단히 하여 풍속을 예측하는 방법을 제안하였다.

첫 번째로 제안한 방법은 풍속을 예측하기 위해 리그레션 방법을 이용하여 복잡한 식을 근사화시켜 풍속을 예측하는 방법이다. 시뮬레이션 결과를 통해 P&O 제어 방법과 비교하여 즉각적인 변화에 반응이 빠른 것을 확인하였고, 뉴턴-랩슨을 이용한 방법과 비슷한 결과를 얻는 것을 확인하였다. 뉴턴-랩슨 방법을 이용하면 과도 상태에서 근사해를 찾지 못하여 오랫동안 반복되는 것을 확인할 수 있다. 하지만 리그레션을 이용한 방법은 과도상태에서도 즉각적으로 풍속을 예측하고 있다는 것을 확인할 수 있었다.

두 번째로 제안된 방법은 풍속과 회전속도에 따른 출력곡선을 이용하여 풍속

을 예측하는 알고리즘이다. 이 방법 역시 시뮬레이션 결과 P&O제어 방법에 비해 즉각적인 풍속의 변화에 빠르게 반응하며 과도상태에서도 즉각적으로 풍속을 예측하고 있다는 것을 확인할 수 있었다.

제안된 방법들의 성능을 확인하기 위하여 Matlab/Simulink를 사용하여 시뮬레이션 하였으며, 시뮬레이션에 사용된 모델은 9.4kW PMSG 풍력터빈이다.

참 고 문 헌

- [1] International Energy Agency, World Energy Outlook, 2013. OECD/IEA, 2013.
- [2] SuHyeon Han and Hyun Woo Shin, "Policy Trends of Renewable Energy in Korea", Renewable Energy Research and Applications, 2014
- [3] Chul Soo Seo, Sang Ho Park, Jun Shin Lee, and Seung Tae Cha, "Offshore Wind Power Planning in Korea", Power Electronics and Applications, 2013
- [4] Abdullah M.A., Yatim A.H.M., Tan C.W., Saidur R. "A review of maximum power point tracking algorithms for wind energy systems" Renewable and Sustainable Energy Reviews, 2012
- [5] Shibashis Bhowmik, Rene Spee and Johan H.R. Enslin, "Performance Optimization for Doubly Fed Wind Power Generation Systems", IEEE Transactions on Industry Applications, Vol. 35, No. 4, 1999
- [6] Shibashis Bhowmik and Rene Spee, "Wind Speed Estimation Based Variable Speed Wind Power Generation", Industrial Electronics Society, 1998
- [7] 문대선, 김성호, "풍력발전기를 위한 신경망 기반의 풍속 추정 및 MPPT 기법에 관한 연구", 한국지능시스템학회, Vol. 20, No. 6, pp.852-857, 2010
- [8] Hui Li, K.L. Shi, P.G. McLaren "Neural-Network-Based Sensorless maximum Wind Energy Capture With Compensated Power Coefficient" IEEE Transactions on Industry Applications, 2005
- [9] Wei Qiao, Wei Zhou, Jose M. Aller and Ronald G. Harley "Wind Speed Estimation Based Sensorless Output Maximization Control for a Wind Turbine Driving a DFIG" IEEE Transactions on Power Electronics, Vol. 23, No. 3, 2008
- [9] 남윤수, "풍력터빈 시스템 제어", 인터비전, 2013

- [10] Thomas Ackermann "Wind Power in Power Systems", John Wiley&Sons, Ltd, 2005
- [11] 정해광, 이교범 "풍력발전 시스템의 전력전자기술 - 설계 및 제어" 한티 미디어, 2014
- [12] Chris Skovgaard Hansen, Kristian Didriksen Lund, Nicklas Christensen and SeungYoun Ko "Operation of PMSG for Small Grid Connected Wind Turbine System with Maximum Power Point Tracking", Bachelor Project, Aalborg University, 2014
- [13] Wei Qiao, Wei Zhou, Jose M. Aller, and Ronald G. Harley "Wind Speed Estimation Based Sensorless Output Maximization Control for a Wind Turbine Driving a DFIG" IEEE Trans. Power Electronics, vol, 23, no. 3, pp. 1156-1169 May 2008

Appendix. A Matlab/Simulink 모델

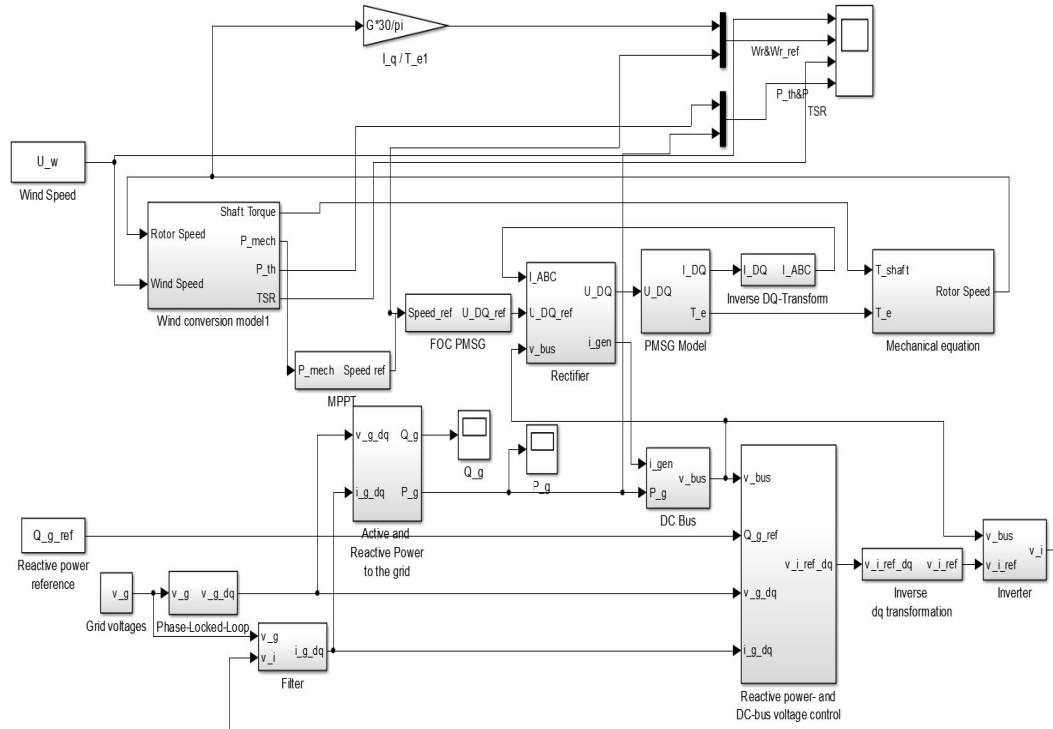


Fig. 40 PMSG Simulink Model

Fig. 40은 [12]에서 모델링한 PMSG의 Matlab/Simulink 모델이다. 본 논문에서의 시뮬레이션 모델은 기계적 모델과 전기적 모델을 Simulink를 이용하여 설계한 것이다. 이 모델을 이용하여 풍력 터빈 축의 최대출력 제어와 계통 측 제어인 DC 부스 전압과 무효 전력 제어를 시뮬레이션 할 수 있다. PMSG 특성은 Table 3에 나타내었으며 모델링에 사용된 식들은 [12]을 참조하였다.

A.1 Wind Conversion Model Block

Fig. 41은 Wind conversion model 블록 내부 모습을 나타낸다. 입력은 회전속도와 풍속이며 이를 통해 샤프트 토크(T_{shaft}), 기계적 출력(P_{mech}), 이론상의 최대 출력(P_{th}) 그리고 TSR 을 계산한다.

Table 3 Parameters of the PMSG model

Parameter	Symbol	Value	Unit
Rated Power	P_{PM}	9.4	[kW]
Rated Torque	T_{PM}	20	[Nm]
Rated Current	i_{PM}	24.5	[A]
Rated frequency	f_{PM}	300	[Hz]
Rated Speed	n_{PM}	4500	[rpm]
Number of pole pairs	$n_{pp_{pm}}$	4	[-]
Stator phase resistance	R_s	0.18	[Ω]
Direct axis inductance	L_d	2	[mH]
Quadrature axis inductance	L_q	2	[mH]
Moment of inertia	J	0.0048	[kg · m ²]
Permanent magnet flux	λ_m	0.123	[Wb]

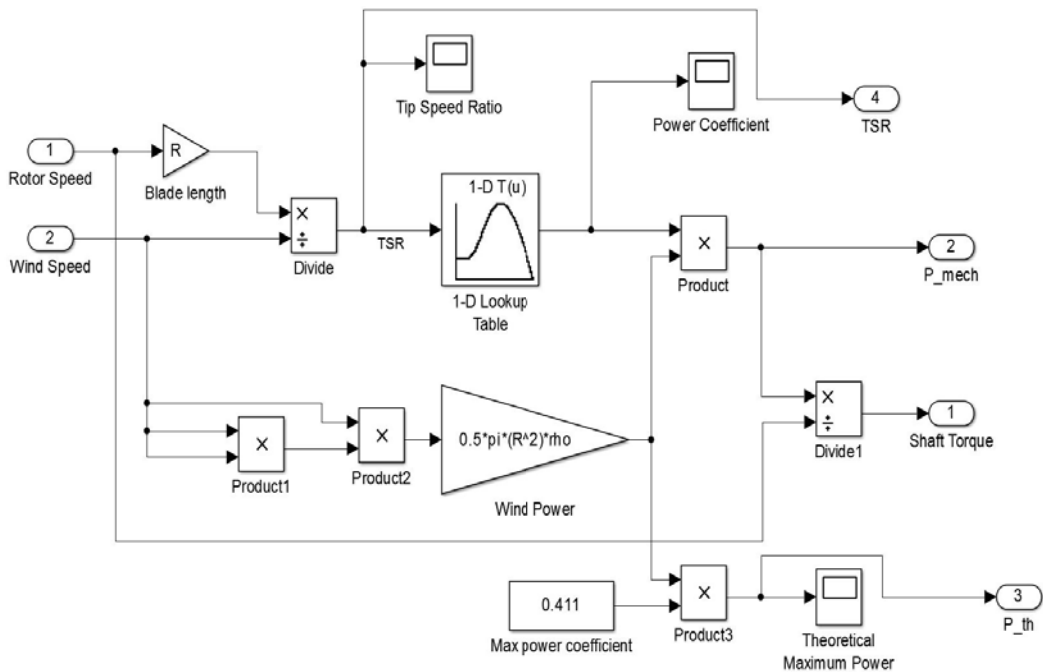


Fig. 41 Wind Conversion model block

A.2 Mechanical Equation Block

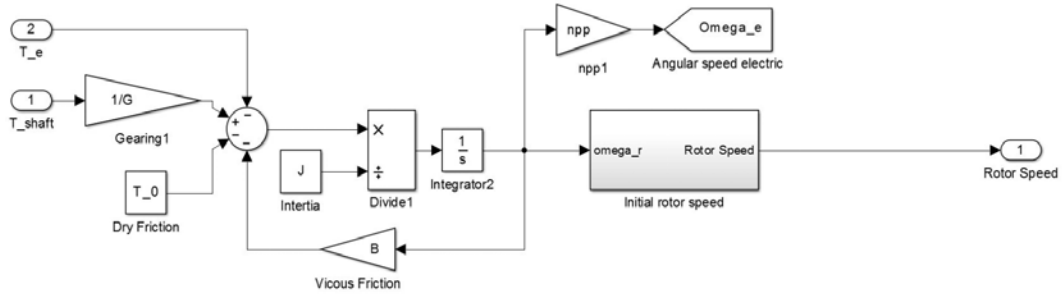


Fig. 42 Mechanical equation block

Fig. 42는 Mechanical equation 블록 내부를 나타낸 것이다. 모델링에 사용된 식은 [12]에서 참고하였다. 이 블록에서 입력 토크를 통해 전기적 회전속도(ω_e)와 로터의 회전속도(ω_r)를 계산한다.

A.3 PMSG Block

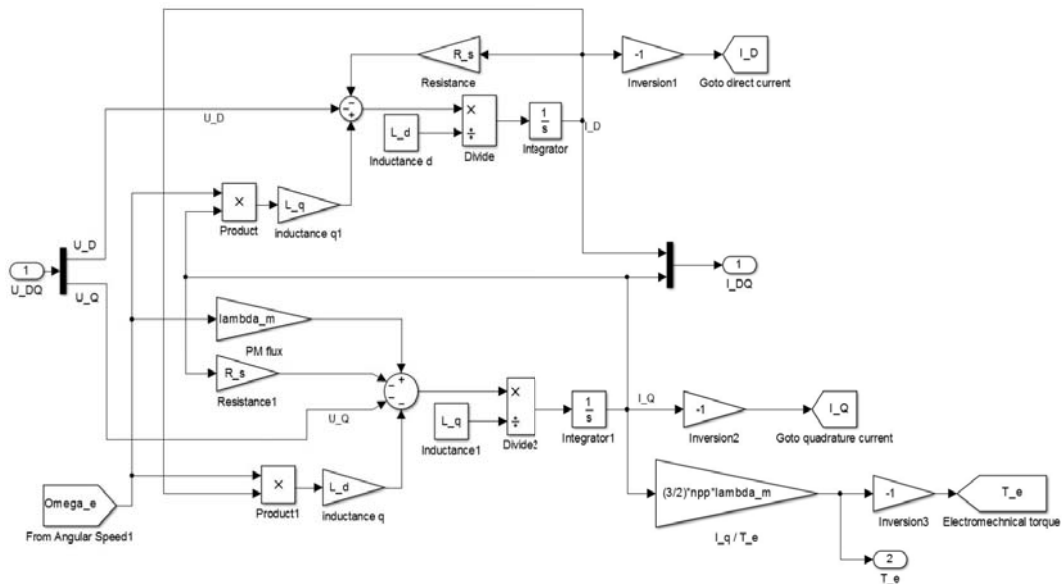


Fig. 43 PMSG Block

Fig. 43는 PMSG 모델의 내부를 나타낸 것이다. PMSG 모델은[12]을 참조하여

모델링 한 것이다. 여기서, V_d , V_q , I_d , I_q , L_d , L_q 는 dq변환한 전압, 전류, 인덕턴스의 값을 나타낸다. R_s 는 고정자의 저항을, ω_e 는 전자 회전속도를 그리고 λ_m 은 영구자석의 쇠교자속을 나타낸다. PMSG의 자세한 내용은 [12]을 참조하였다. dq 변환된 전압(V_{dq})가 입력이 되어 dq변환된 전류(i_{dq})와 전자기적 토크(T_e)가 계산된다.

A.4 dq Inverse Transform Block

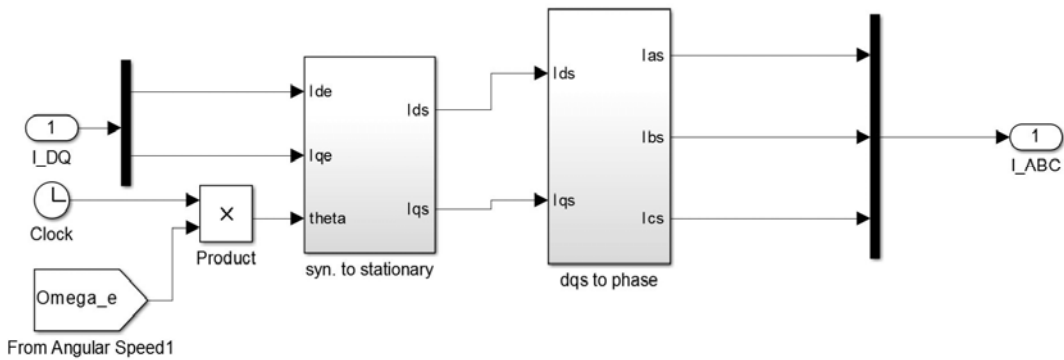


Fig. 44 dq Inverse Transform Block

Fig. 44은 dq 역변환 블록 내부의 모습이다. PMSG 블록에서 계산된 I_{dq} 를 역 변환 하여 3상 교류 전류로 계산하여 출력한다. 변환은 회전좌표계($d-q$ 축)에서 정지좌표계($\alpha-\beta$ 축)로 변환하고 정지좌표계($\alpha-\beta$ 축)에서 3상으로 변환한다. dq 변환에 대한 자세한 내용은 [12]에 나와있다.

A.5 MPC Block(P&O)

Fig. 45은 MPC 블록 내부를 나타낸다. 이 블록은 P&O 방법을 이용한 것으로 기계적 출력(P_{mech})를 입력으로 받아 설정된 시간마다 P&O 알고리즘에 따라 설정된 값 만큼 회전 속도를 증가 또는 감소시킨 레퍼런스 회전속도(ω_{ref})를 출력으로 내보낸다.

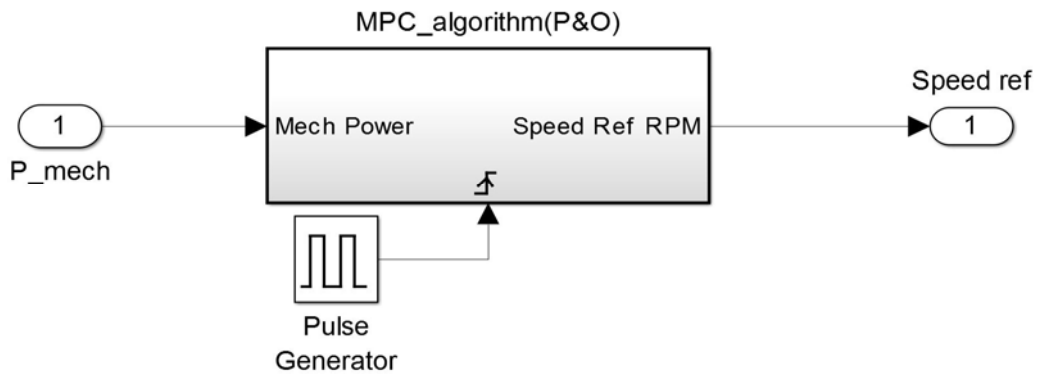


Fig. 45 MPC block(P&O)

A.6 FOC Block

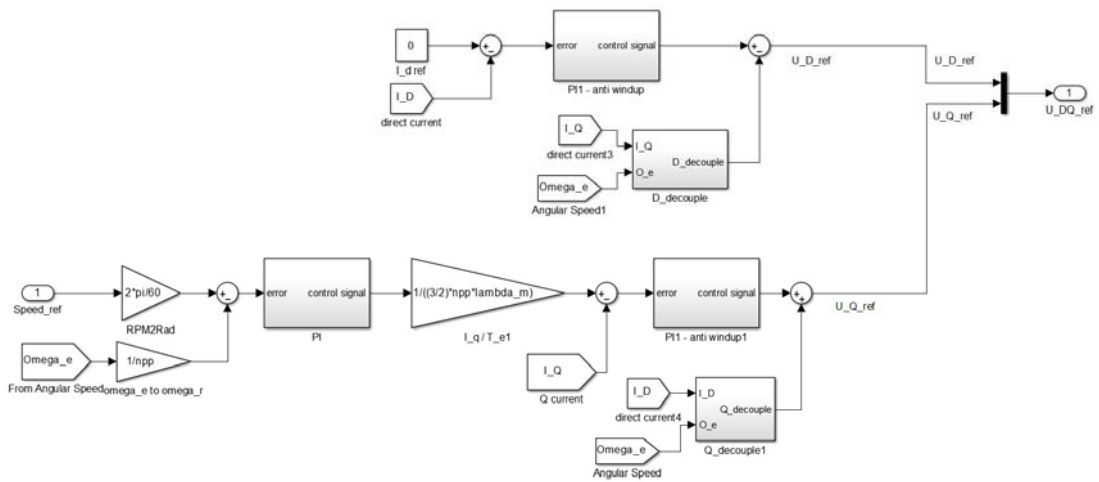


Fig. 46 FOC Block

Fig. 46은 FOC(Field Oriented Control) 블록 내부의 모습이다. FOC 방법은 로터의 회전속도와 전류를 측정하여 PMSG 동작을 제어하는 방법이다. 즉, 측정된 회전속도를 Fig. 45에서 얻은 레퍼런스 회전속도(ω_{ref})와 PMSG 블록에서 계산된 I_{dq} 를 이용하여 회전속도를 제어하기 위하여 dq변환된 레퍼런스 전압(V_{dqref})을 계산한다. FOC 방법에 대한 자세한 내용은 [12]를 참조하였다.

A.7 Converter Block

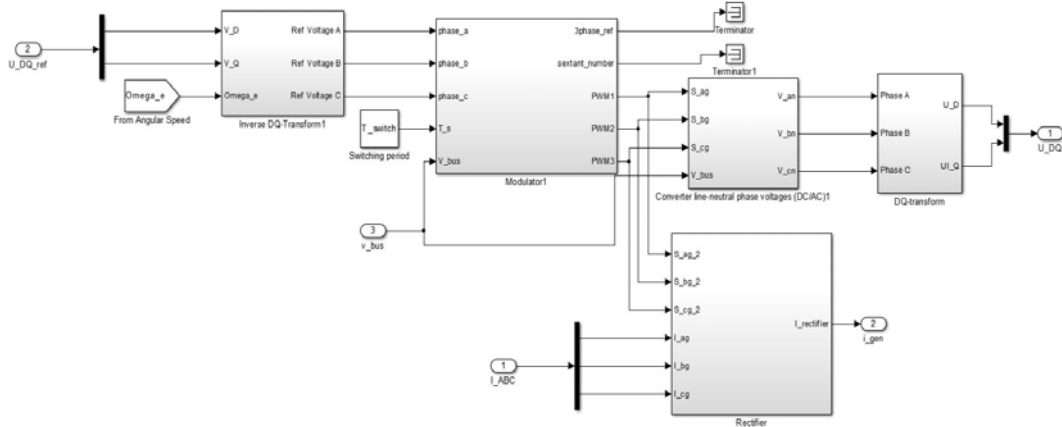


Fig. 47 Converter block

Fig. 47은 컨버터 블록 내부를 나타낸다. Fig. 46에서 얻는 레퍼런스 전압으로 직류전압의 크기를 변환시키기 위한 전압 조정기(Modulator) 블록 내부에 포함되어 있다. 전압 조정기의 내부는 Fig. 48에 나타내었다. V_{dqref} 에서 역변환된 레퍼런스 전압($V_{abc,ref}$)이 전압조정기 블록에서 PWM 신호로 변형되어 컨버터 블록에서 전압(V_{abc})와 컨버터 전류(i_{gen})가 계산된다. 컨버터는 식스 스위치 컨버터로 디자인 되었으며 자세한 내용은 [12]에서 참조하였다.

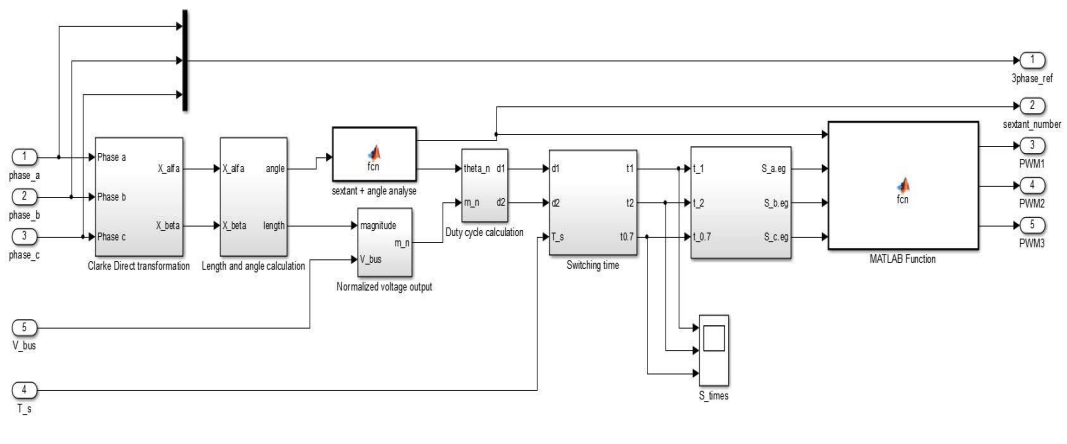


Fig. 48 Modulator block

A.8 Phase Locked Loop Block

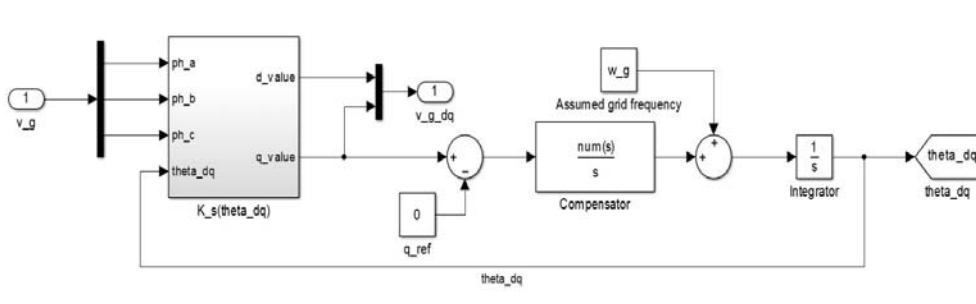


Fig. 49 Phase Locked Loop Block

Fig. 49은 phase locked loop 블록의 내부를 나타낸다. 이 블록은 계통 전압의 평형을 유지하기 위하여 설계된 블록이다. 계통의 전압을 입력으로 받고 θ_{dq} 가 일정하게 유지되도록 피드백 제어를 수행한다. 자세한 내용은 [12]를 참조하였다.

A.9 DC Bus Voltage Block

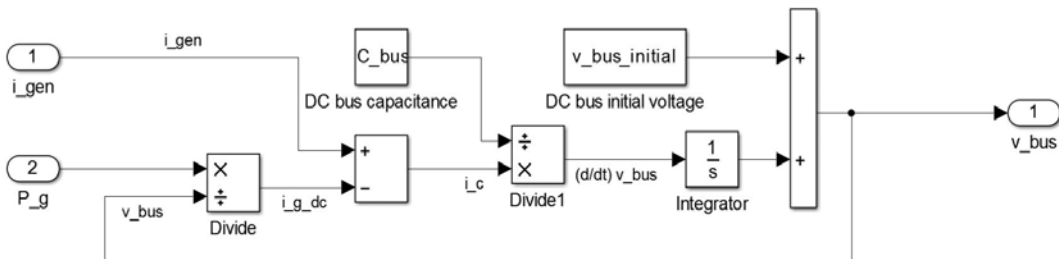


Fig 50. DC bus voltage block

Fig. 50는 DC bus 전압 블록이다. 계통에서의 전력(P_g)과 인버터에서 계산된 전류(i_{gen})을 통해 DC 부스 전압이 계산된다.

A.10 dq Inverse Transform Block for Grid

계통측 컨버터에 대한 dq 역변환 블록을 나타낸다. Fig. 52에서 얻은 인버터

dq 레퍼런스 전압($V_{dq_i_ref}$)과 phase locked loop 블록에서 고정시킨 θ_{dq} 를 입력으로 하여 dq 역변환 식을 이용하여 $V_{abc_i_ref}$ 를 구한다.

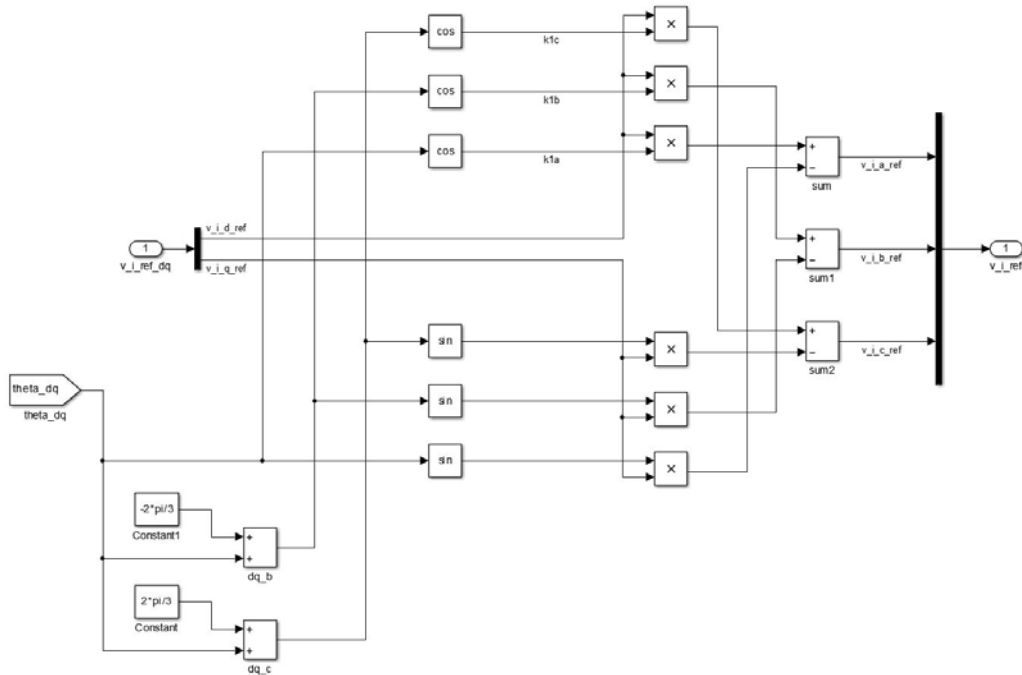


Fig. 51 dq Inverse Transform block for Grid

A.11 Inverter Block

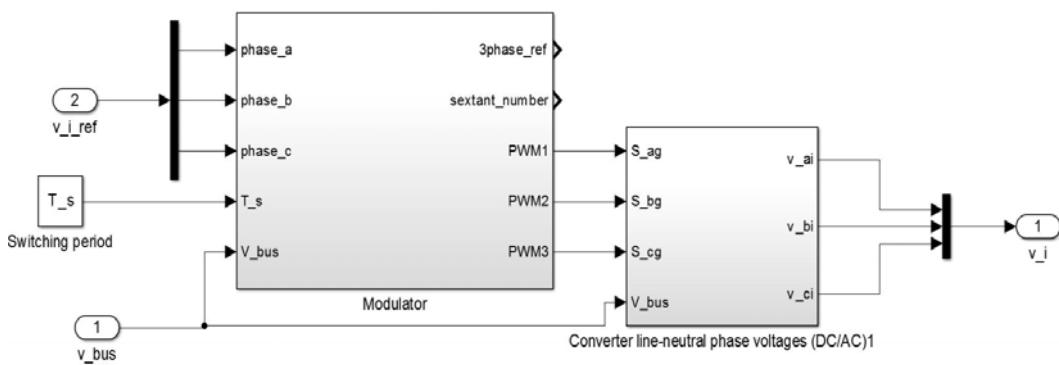


Fig. 52 Inverter block

Fig. 52는 인버터 블록의 내부를 나타낸다. Fig. 51에서 계산된 $V_{abc_i_ref}$ 를 전

압조정기에서 PWM 신호로 변형후 컨버터에서 PWM 신호화 V_{bus} 를 이용하여 인버터 전압(V_{abc_i})를 계산한다. 전압 조정기 블록의 내부는 Fig. 48과 같다.

A.12 DC Bus Voltage and Reactive Power Control Block

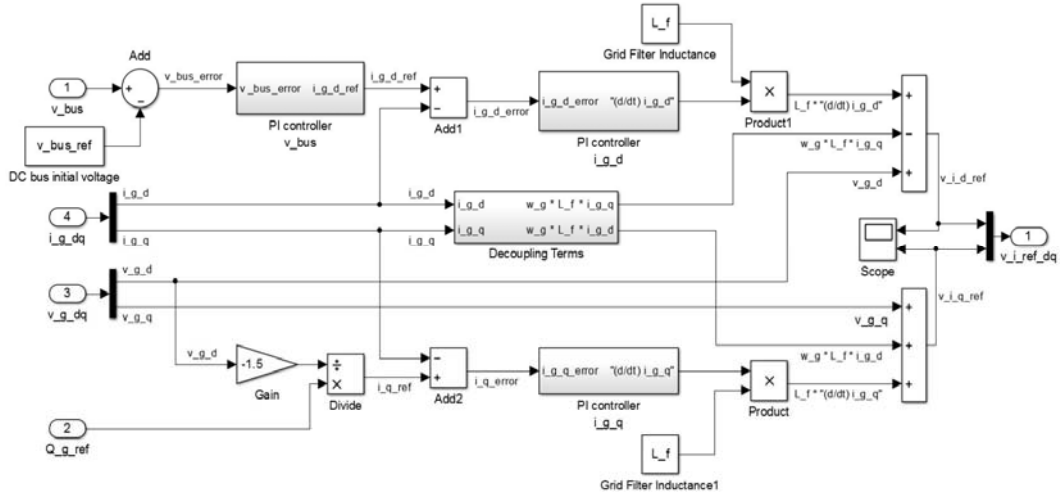


Fig. 53 DC bus voltage and Reactive power control block

Fig. 53은 DC 부스 전압과 무효 전력 제어 블록의 내부를 나타낸다. DC 부스 전압(V_{bus}), dq 변환된 계통 전압(V_{g_dq}), dq 변환된 계통 전류(I_{g_dq}), 무효 전력(Q_g)을 입력으로 하여 V_{bus} 와 Q_g 를 제어하기 위하여 dq변환된 레퍼런스 전압(V_{ref_dq})를 구한다. 이를 구하기 위하여 PI제어기가 이용되었으며 자세한 내용은 [12]를 참조하였다.

A.13 Active Power and Reactive Power to Grid Block

Fig. 54는 계통으로 전달되는 유효 전력과 무효 전력을 나타낸다. Fig. 20에서 DC 부스 전압과 무효 전력을 제어하고 제어된 dq변환된 전압과 전류가 계산되어 무효 전력과 유효 전력으로 계통에 전달된다.

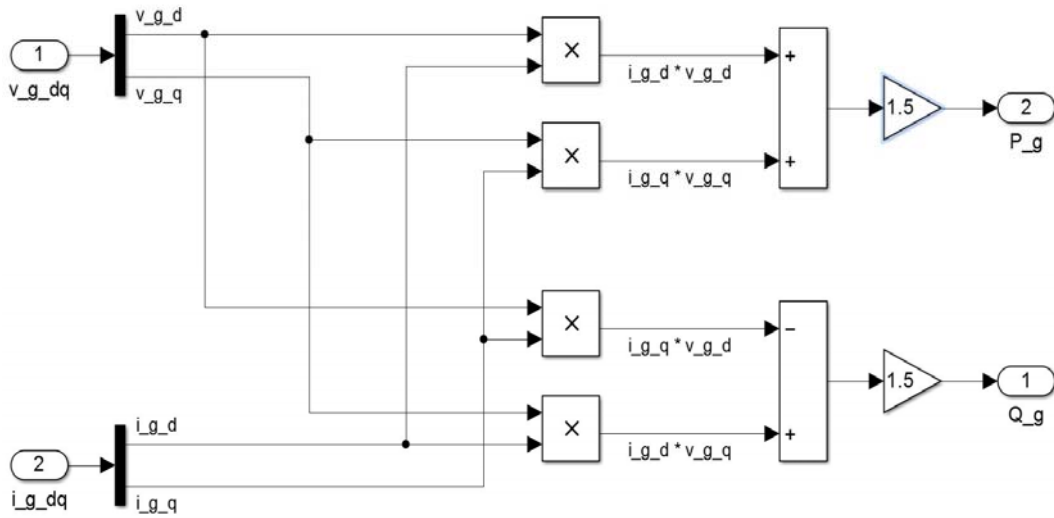


Fig. 54 Active Power and Reactive power to grid Block

A.14 R-L Filter block

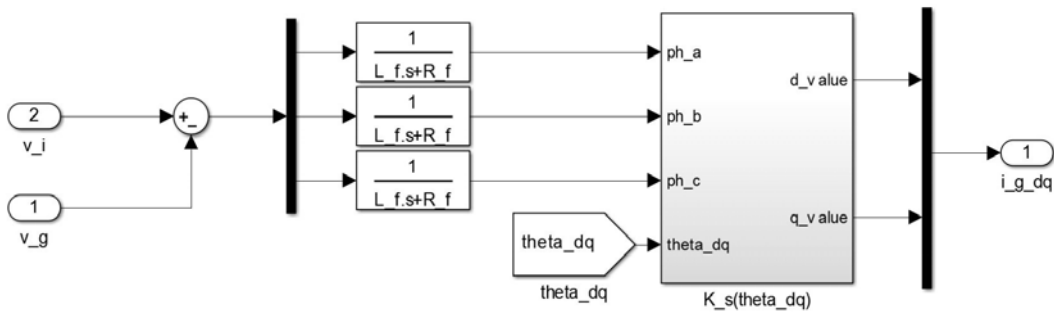


Fig. 55 R-L Filter block

인버터에서 변환된 전압의 파형을 개선시키기 위해서 필터가 필요하다. 이를 위해 본 시뮬레이션 모델에서는 R-L필터를 설계하였는데, Fig. 55에 나타내었다.

Appendix. B 사용된 프로그램 코드

B.1 풍력터빈시스템

%% The following script contains parameters for modelling of the WES with a PMSG:

```
Ts = 0.00005;           %Simulation Time
T_switch = 0.0005;     %Rectifier switching Time
T_mppt = 0.02;         %MPPT calculation time
L_d = 2E-3;            %d-axis inductance of PMSG
L_q = 2E-3;            %q-axis inductance of PMSG
R_s = 0.18;            %Winding resistance - [Ohm]
npp = 4;               %number of pole pair number. - [-]
J = 0.0048;           %Moment of inertia - [kg/m^2]
lambda_m = 0.123;     %Permanent magnet flux linkage -[Wb] (Taken
from "Encoderless Vector Control of PMSG for
                        %Wind Turbine Applications" - a master thesis
from 2010)
T_0 = 0.2;            %Dry friction
B = 3E-3;             %Viscous Friction

%% Controllers:
K_P_T = 2.7;          %Current controller proportional gain
K_I_T = 1000;         %Current controller integration gain
K_P_S = 0.55;         %Speed controller proportional gain
K_I_S = 12;           %Speed controller integration gain

%% Reactive power reference
Q_g_ref = 0;
```

```

%% Wind speed model
U_w = 5;           %Wind Speed
rho = 1.225;      %Air density
R = 2.4;         %Rotor radius
G = 5;           %Gearing Ratio

%% Calculation of the Power coefficient for look up table
v_w = 12;        %A wind velocity
omega_m = 0.1:0.1:64; %A vector containing different rotor angular
velocities
C_p(length(v_w),length(omega_m)) = 0;
lambda(length(v_w),length(omega_m)) = 0;

for i = 1:length(v_w)
    for j = 1:length(omega_m)
        lambda(i,j) = (omega_m(j)*R)/(v_w(i)); %The tip speed ratio
        lambda_i = lambda(i,j)/(1-0.035*lambda(i,j));
        C_p(i,j) = 0.5*(116*(1/lambda_i)-5)*exp(-(21/lambda_i));
    end
end

%% Grid values
V_m = 230;
w_g = 100*pi;
%% DC bus values
C_bus = 0.1;
v_bus_initial = 1000;
v_bus_ref = 1000;

```

```

%% Filter
L_f = 0.050;
R_f = 1.0;
%% Converter
T_s = 0.00025;

```

B.2 Matlab Function : P&O 제어

```

mean_1 = (speed_ref + speed_ref_b + speed_ref_bb + speed_ref_bbb)/4;
mean_2 = (speed_ref_b + speed_ref_bb + speed_ref_bbb + speed_ref_bbbb)/4;

if mean_1 == mean_2 ;
    speed_ref = mean_2;
end

if speed_ref_b == speed_ref_bb ;
    speed_ref = speed_ref_b;
end
end

```

B.3 Matlab Function : 뉴턴-랩슨

```

function [v_est] = fcn(P, Wr)
rho=1.225; %공기 밀도
R=2.6; %블레이드 반경
G=5; %기어비
C=P/(0.52125*rho*pi*R^5*wr^3); %출력식에서 TSR을 제외한 상수값

iter1 = 1;

```

```

iter2 = 1;
xr=0.1;
while(1)
    xrold=xr;
    f_x_1= xr^3*(116*xr-9.06)*exp(-21*xr)-C;
    f_x_dev_1= -xr^2*(2436*xr^2-654.26*xr+27.18)*exp(-21*xr);
    x=xr-f_x_1/f_x_dev_1;    %뉴턴 랩슨 공식
    err=abs((x-xr)/x)*100;    % 오차
    lambda_i=(x-0.035)^(-1);
    Cp_es=0.5*(116/lambda_i-5)*exp(-21/lambda_i);
    xr=x;
    if xr<=0.035 || Cp_es >0.411
        xr=0.035+0.1*iter2;
        iter2=iter2+1;
        if iter2 >= 10
            iter2=1;
        end
    elseif Cp_es<0 || xr>1
        xr=xr/xr-0.1*iter2;
        iter2=iter2+1;
        if iter2 >= 10
            iter2=1;
        end
    end
    iter=1;
end
if err <=0.000001, break, end
if iter1 == 1500, break, end
iter1=iter1+1;
end

```

```
v_est=R*wr/TSR;
```

B.4 Matlab Function : 리그레션

```
function v_est = fcn(P,Wr)
rho = 1.225;           %Air density
R = 2.4;              %Rotor radius
G = 5;                %Gearing Ratio
a1 = -0.0021;
a2 = 0.0299;
a3 = -0.0534;
c_reg = a1-P/(0.52*rho*pi*R^5*Wr^3); %근의 공식에서 c에 해당함
x = sqrt(complex(a2^2-4*a3*c_reg));
TSR = inv(real((-a2+x)/(2*a3)));
v_est=R*wr/TSR;
```

B.5 Matlab Function : 제안된 알고리즘

```
function [v_est] = fcn(P,Wr)
rho = 1.225;           %Air density
R = 2.4;              %Rotor radius
G = 5;                %Gearing Ratio
lambda_opt=8;
v_ref=1;
v_step=0.2;
TSR=abs(R*w_r/v_ref);
Cp=(abs(0.5*(116*1/TSR-9.06)*exp(-21*(1/TSR-0.035))));
if Cp <= 0, Cp=0; end
P_cal=(abs(0.5*rho*pi*R^2*v_ref^3*Cp));
```

```

while(1)
    if P > P_cal
        wr=w_r;
        v_ref=v_ref+v_step;
        TSR=(abs(R*w_r/v_ref)) ;
        Cp=(abs(0.5*(116*1/TSR-9.06)*exp(-21*(1/TSR-0.035))));
        if Cp <= 0, Cp=0; end
        P_cal=(abs(0.5*rho*pi*R^2*v_ref^3*Cp));
    end
    if P <= P_cal
        break
    end

    if v_ref > 12; break, end
end
v_est=abs(v_ref-abs(P/P_cal)*v_step);

```


감사의 글

대학원에 입학한지 어느덧 2년 반이라는 시간이 지났습니다. 많은 분들의 노력과 열정으로 풍력에 대하여 좋은 환경에서 많은 것을 배울 수 있었습니다. 긴 시간동안 모자란 저에게 끊임없이 지도와 편달을 아끼지 않아주신 강민제 교수님께 가장 먼저 존경과 감사의 말씀을 전하고 싶습니다. 그리고 바쁘신 와중에도 본 논문을 자상하게 심사해주신 김경연 교수님과 김호찬 교수님께 깊이 감사드립니다. 또한, 학사시절 많은 가르침을 주신 전자공학과와 고성택 교수님, 도양희 교수님, 이광만 교수님, 고석준 교수님 그리고 지금은 은퇴하신 김경식 교수님께도 깊이 감사드립니다.

대학원을 다니는 동안 아끼지 않으며 풍력공학부 학생들과 학생들의 학업을 위해 오늘도 고생하시는 허종철 교수님과 고경남 교수님 그리고 김호민 교수님께 늘 감사드리며 행정실 선생님들에게도 정말 감사하다는 말씀을 드리고 싶습니다.

함께 연구실 생활을 하는 상우, 나현, 지호, 지현 그리고 지금은 졸업한 민호에게도 고맙다는 말을 전하고 2년이라는 시간을 함께 대학원 생활을 한 종호, 민호, 승언이에게도 감사의 말을 전합니다. 또한, 제게 도움을 준 풍력대학원 동기와 선후배 분들께 감사의 마음을 전합니다.

마지막으로 저를 나아주시고 길러주신 그리고 언제나 든든한 서포터가 되어주신 아버지 어머니 그리고 형에게 감사의 마음을 전합니다.